

Aurélio Campilho
Mohamed Kamel (Eds.)

LNCS 4141

Image Analysis and Recognition

Third International Conference, ICIAR 2006
Póvoa de Varzim, Portugal, September 2006
Proceedings, Part I

1
Part I

 Springer

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

University of Dortmund, Germany

Madhu Sudan

Massachusetts Institute of Technology, MA, USA

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Moshe Y. Vardi

Rice University, Houston, TX, USA

Gerhard Weikum

Max-Planck Institute of Computer Science, Saarbruecken, Germany

Aurélio Campilho Mohamed Kamel (Eds.)

Image Analysis and Recognition

Third International Conference, ICIAR 2006
Póvoa de Varzim, Portugal, September 18-20, 2006
Proceedings, Part I

Volume Editors

Aurélio Campilho

University of Porto, Faculty of Engineering, Institute of Biomedical Engineering

Rua Dr. Roberto Frias, 4200-465, Porto, Portugal

E-mail: campilho@fe.up.pt

Mohamed Kamel

University of Waterloo, Department of Electrical and Computer Engineering

200 University Avenue West, Waterloo, Ontario N2L 3G1, Canada

E-mail: mkamel@uwaterloo.ca

Library of Congress Control Number: 2006932295

CR Subject Classification (1998): I.4, I.5, I.3.5, I.2.10, I.2.6, F.2.2

LNCS Sublibrary: SL 6 – Image Processing, Computer Vision, Pattern Recognition, and Graphics

ISSN 0302-9743

ISBN-10 3-540-44891-8 Springer Berlin Heidelberg New York

ISBN-13 978-3-540-44891-4 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

springer.com

© Springer-Verlag Berlin Heidelberg 2006

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper SPIN: 11867586 06/3142 5 4 3 2 1 0

Preface

ICIAR 2006, the International Conference on Image Analysis and Recognition, was the third ICIAR conference, and was held in Póvoa de Varzim, Portugal. ICIAR is organized annually, and alternates between Europe and North America. ICIAR 2004 was held in Porto, Portugal and ICIAR 2005 in Toronto, Canada. The idea of offering these conferences came as a result of discussion between researchers in Portugal and Canada to encourage collaboration and exchange, mainly between these two countries, but also with the open participation of other countries, addressing recent advances in theory, methodology and applications.

The response to the call for papers for ICIAR 2006 was higher than the two previous editions. From 389 full papers submitted, 163 were finally accepted (71 oral presentations, and 92 posters). The review process was carried out by the Program Committee members and other reviewers; all are experts in various image analysis and recognition areas. Each paper was reviewed by at least two reviewers, and also checked by the conference Co-chairs. The high quality of the papers in these proceedings is attributed first to the authors, and second to the quality of the reviews provided by the experts. We would like to thank the authors for responding to our call, and we wholeheartedly thank the reviewers for their excellent work and for their timely response. It is this collective effort that resulted in the strong conference program and high-quality proceedings in your hands.

We were very pleased to be able to include in the conference program keynote talks by three world-renowned experts: Ling Guan, Electrical and Computer Engineering at Ryerson University, Canada; Mubarak Shah, Agere Chair of Computer Science, University of Central Florida, USA, and John Oommen, School of Computer Science at Carleton University in Ottawa, Canada. We would like to express our sincere gratitude to each of them for accepting our invitations.

We would like to thank Khaled Hammouda, the webmaster of the conference, for maintaining the Web pages, interacting with the authors and preparing the proceedings. We would like to thank the conference secretariat for administrative assistance. We would also like to thank members of the Local Organization Committee for their advice and help. We also appreciate the help of the Springer editorial staff, for supporting this publication in the LNCS series.

Finally, we were very pleased to welcome all the participants to this conference. For those who did not attend, we hope this publication provides a good view into the research presented at the conference, and we look forward to meeting you at the next ICIAR conference.

ICIAR 2006 – International Conference on Image Analysis and Recognition

General Chair

Aurélio Campilho
University of Porto, Portugal
campilho@fe.up.pt

General Co-chair

Mohamed Kamel
University of Waterloo, Canada
mkamel@uwaterloo.ca

Local Organizing Committee

Ana Maria Mendonça
University of Porto, Portugal
amendon@fe.up.pt

Jorge Alves Silva
University of Porto, Portugal
jsilva@fe.up.pt

António Miguel Monteiro
University of Porto, Portugal
apm@fe.up.pt

Luís Corte-Real
University of Porto, Portugal
lreal@fe.up.pt

Armando Jorge Padilha
University of Porto, Portugal
padilha@fe.up.pt

Conference Secretariat

Gabriela Afonso
Biomedical Engineering Institute, Portugal
iciar06@fe.up.pt

Webmaster

Khaled Hammouda
University of Waterloo, Canada
hammouda@pami.uwaterloo.ca

Supported by:



Universidade do Porto

FEUP Faculdade de Engenharia

Department of Electrical and Computer Engineering, Faculty of Engineering, University of Porto, Portugal



INEB – Biomedical Engineering Institute, Portugal



PAMI – Pattern Analysis and Machine Intelligence Group, University of Waterloo, Canada

Advisory Committee

| | |
|----------------------|---|
| M. Ahmadi | University of Windsor, Canada |
| P. Bhattacharya | Concordia University, Canada |
| T.D. Bui | Concordia University, Canada |
| M. Cheriet | University of Quebec, Canada |
| Z. Duric | George Mason University, USA |
| M. Ejiri | Japan |
| G. Granlund | Linköping University, Sweden |
| L. Guan | Ryerson University, Canada |
| M. Haindl | Institute of Information Theory and Automation, Czech Republic |
| E. Hancock | The University of York, UK |
| J. Kovacevic | Carnegie Mellon University, USA |
| M. Kunt | Swiss Federal Institute of Technology (EPFL), Switzerland |
| K.N. Plataniotis | University of Toronto, Canada |
| A. Sanfeliu | Technical University of Catalonia, Spain |
| M. Shah | University of Central Florida, USA |
| M. Sid-Ahmed | University of Windsor, Canada |
| C.Y. Suen | Concordia University, Canada |
| A.N. Venetsanopoulos | University of Toronto, Canada |
| M. Viergever | University of Utrecht, Netherlands |
| B. Vijayakumar | Carnegie Mellon University, USA |
| J. Villanueva | Autonomous University of Barcelona, Spain |
| R. Ward | University of British Columbia, Canada |
| D. Zhang | The Hong Kong Polytechnic University, Hong Kong |

Program Committee

| | |
|-----------------|--|
| S. Abdallah | American University of Beirut, Lebanon |
| W. Abd-Almageed | University of Maryland, USA |
| R. Abugharbieh | University of British Columbia, Canada |
| S. Abdallah | American University of Beirut, Lebanon |
| W. Abd-Almageed | University of Maryland, USA |
| R. Abugharbieh | University of British Columbia, Canada |
| P. Aguiar | Institute for Systems and Robotics, Portugal |
| M. Ahmed | Wilfrid Laurier University, Canada |
| J. Alirezaie | Ryerson University, Canada |
| D. Androutsos | Ryerson University, Canada |
| H. Araujo | University of Coimbra, Portugal |
| N. Arica | Turkish Naval Academy, Turkey |
| J. Barron | University of Western Ontario, Canada |
| O. Basir | University of Waterloo, Canada |
| J. Batista | University of Coimbra, Portugal |
| C. Bauckhage | York University, Canada |
| A. Bernardino | Technical University of Lisbon, Portugal |
| P. Bhattacharya | Concordia University, Canada |
| J. Bioucas | Technical University of Lisbon, Portugal |
| B. Boufama | University of Windsor, Canada |
| T.D. Bui | Concordia University, Canada |
| A. Campilho | University of Porto, Portugal |
| E. Cernadas | University of Vigo, Spain |
| B. Chalmond | University of Cergy-Pontoise, France |
| M. Cheriet | University of Quebec, Canada |
| F. Cheriet | École Polytechnique de Montréal, Canada |
| D. Chiu | University of Guelph, Canada |
| D. Clausi | University of Waterloo, Canada |
| M. Correia | University of Porto, Portugal |
| L. Corte-Real | University of Porto, Portugal |
| J. Costeira | Technical University of Lisbon, Portugal |
| D. Cuesta-Frau | Polytechnic University of Valencia, Spain |
| J. Cunha | University of Aveiro, Portugal |
| V. Di Gesù | Università degli Studi di Palermo, Italy |
| J. Dias | University of Coimbra, Portugal |
| E. Dubois | University of Ottawa, Canada |
| Z. Duric | George Mason University, USA |
| A. Elmoataz | Université de Caen, France |
| M. El-Sakka | University of Western Ontario, Canada |
| R. Fazel | University of Manitoba, Canada |
| M. Ferretti | University of Pavia, Italy |
| P. Fieguth | University of Waterloo, Canada |

| | |
|-----------------------|---|
| M. Figueiredo | Technical University of Lisbon, Portugal |
| A. Fred | Technical University of Lisbon, Portugal |
| G. Freeman | University of Waterloo, Canada |
| V. Grau | University of Oxford, UK |
| M. Greenspan | Queen's University, Canada |
| L. Guan | Ryerson University, Canada |
| M. Haindl | Institute of Information Theory and Automation, Czech Republic |
| E. Hancock | University of York, UK |
| B. Huet | Institut Eurecom, France |
| A. Hunter | Lincoln University, UK |
| J. Jiang | University of Bradford, UK |
| J. Jorge | INESC-ID, Portugal |
| J. Kamarainen | Lappeenranta University of Technology, Finland |
| M. Kamel | University of Waterloo, Canada |
| M. Kechadi | University College Dublin, Ireland |
| G. Khan | Ryerson University, Canada |
| S. Krishnan | Ryerson University, Canada |
| A. Krzyzak | Concordia University, Canada |
| R. Laganière | University of Ottawa, Canada |
| X. Li | University of London, UK |
| R. Lins | Universidade Federal de Pernambuco, Brazil |
| J. Lorenzo-Ginori | Universidad Central "Marta Abreu" de Las Villas, Cuba |
| S. Lu | Memorial University of Newfoundland, Canada |
| R. Lukac | University of Toronto, Canada |
| A. Marcal | University of Porto, Portugal |
| J. Marques | Technical University of Lisbon, Portugal |
| M. Melkemi | Université de Haute Alsace, France |
| A. Mendonça | University of Porto, Portugal |
| M. Mignotte | University of Montreal, Canada |
| A. Monteiro | University of Porto, Portugal |
| I. Nyström | Uppsala Universitet, Sweden |
| J. Orchard | University of Waterloo, Canada |
| A. Ouda | University of Western Ontario, Canada |
| A. Padilha | University of Porto, Portugal |
| P. Payeur | University of Ottawa, Canada |
| F. Perales | University of the Balearic Islands, Spain |
| F. Pereira | Technical University of Lisbon, Portugal |
| N. Peres de la Blanca | University of Granada, Spain |
| E. Petrakis | Technical University of Crete, Greece |
| P. Pina | Technical University of Lisbon, Portugal |
| A. Pinho | University of Aveiro, Portugal |
| J. Pinto | Technical University of Lisbon, Portugal |
| F. Pla | Universitat Jaume I, Spain |
| K. Plataniotis | University of Toronto, Canada |

| | |
|-----------------|---|
| T. Rabie | University of Toronto, Canada |
| P. Radeva | Autonomous University of Barcelona, Spain |
| E. Ribeiro | Florida Institute of Technology, USA |
| L. Rueda | University of Windsor, Canada |
| M. Queluz | Technical University of Lisbon, Portugal |
| F. Samavati | University of Calgary, Canada |
| J. Sánchez | University of Las Palmas de Gran Canaria, Spain |
| B. Santos | University of Aveiro, Portugal |
| M. Savvides | Carnegie Mellon University, USA |
| G. Schaefer | Nottingham Trent University, UK |
| P. Scheunders | University of Antwerp, Belgium |
| J. Sequeira | Ecole Supérieure d'Ingénieurs de Luminy, France |
| M. Shah | University of Central Florida, USA |
| J. Silva | University of Porto, Portugal |
| W. Skarbek | Warsaw University of Technology, Poland |
| B. Smolka | Silesian University of Technology, Poland |
| J. Sousa | Technical University of Lisbon, Portugal |
| H. Suesse | Friedrich-Schiller University Jena, Germany |
| S. Sural | Indian Institute of Technology, India |
| G. Thomas | University of Waterloo, Canada |
| H. Tizhoosh | University of Waterloo, Canada |
| S. Torres | Universidad de Concepción, Chile |
| B. van Ginneken | Image Sciences Institute, Netherlands |
| D. Vandermeulen | Catholic University of Leuven, Belgium |
| M. Vento | University of Salerno, Italy |
| B. Vijayakumar | Carnegie Mellon University, USA |
| J. Vitria | Computer Vision Center, Spain |
| Y. Voisin | Université de Bourgogne, France |
| E. Vrscay | University of Waterloo, Canada |
| M. Wirth | University of Guelph, Canada |
| J. Wu | University of Windsor, Canada |
| F. Yarman-Vural | Middle East Technical University, Turkey |
| J. Yeow | University of Waterloo, Canada |
| J. Zelek | University of Waterloo, Canada |
| G. Zheng | University of Bern, Switzerland |
| D. Ziou | University of Sherbrooke, Canada |

Reviewers

| | |
|-------------|--|
| N. Alajlan | University of Waterloo, Canada |
| J. Awad | University of Waterloo, Canada |
| T. Barata | Technical University of Lisbon, Portugal |
| J. Barbosa | University of Porto, Portugal |
| S. Berretti | University of Florence, Italy |

| | |
|-------------------|---|
| A. Bevilacqua | ARCES-DEIS University of Bologna, Italy |
| J. Boyd | University of Calgary, Canada |
| J. Cardoso | INESC Porto, Portugal |
| M. Coimbra | IEETA-University of Aveiro, Portugal |
| A. Dawoud | University of South Alabama, USA |
| P. Dias | University of Aveiro, Portugal |
| I. El Rube' | University of Waterloo, Canada |
| E. Galmar | Institut Eurecom, France |
| P. García-Sevilla | Universitat Jaume I, Spain |
| J. Glasa | Slovak Academy of Science, Slovakia |
| J. Grim | Institute of Information Theory and Automation, Czech Republic |
| C. Hong | Hong Kong Polytechnic, Hong Kong |
| A. Kong | University of Waterloo, Canada |
| S. Mohamed | University of Waterloo, Canada |
| A. Mohebi | University of Waterloo, Canada |
| F. Monteiro | Biomedical Engineering Institute, Portugal |
| M. Nappi | University of Salerno, Italy |
| E. Papageorgiou | University of Patras, Greece |
| J. Pérez | Departamento de Informática Escuela Politécnica, Spain |
| A. Picariello | University of Naples, Italy |
| A. Puga | University of Porto, Portugal |
| A. Quddus | University of Waterloo, Canada |
| S. Rahnamayan | University of Waterloo, Canada |
| R. Rocha | Biomedical Engineering Institute, Portugal |
| F. Sahba | University of Waterloo, Canada |
| J. Sanches | Technical University of Lisbon, Portugal |
| J. Silva | University of Coimbra, Portugal |
| J. Silva | Universidade Federal de Pernambuco, Brazil |
| A. Silva | Universidade Federal de Pernambuco, Brazil |
| J. Sotoca | Universitat Jaume I, Spain |
| L.F. Teixeira | INESC Porto, Portugal |
| L.M. Teixeira | INESC Porto, Portugal |
| J. Traver | Universitat Jaume I, Spain |
| M. Vega-Rodríguez | University of Extremadura, Spain |
| C. Vinhais | Biomedical Engineering Institute, Portugal |
| A. Zaim | University of Texas, USA |

Table of Contents – Part I

Invited Session

| | |
|---|----|
| Self-Organizing Trees and Forests: A Powerful Tool in Pattern Clustering and Recognition..... | 1 |
| <i>Ling Guan</i> | |
| On Optimizing Dissimilarity-Based Classification Using Prototype Reduction Schemes..... | 15 |
| <i>Sang-Woon Kim, B. John Oommen</i> | |

Image Restoration and Enhancement

| | |
|---|-----|
| General Adaptive Neighborhood Image Restoration, Enhancement and Segmentation..... | 29 |
| <i>Johan Debayle, Yann Gavet, Jean-Charles Pinoli</i> | |
| Frozen-State Hierarchical Annealing..... | 41 |
| <i>Wesley R. Campaigne, Paul Fieguth, Simon K. Alexander</i> | |
| Fast and Robust Filtering-Based Image Magnification..... | 53 |
| <i>Wenze Shao, Zhihui Wei</i> | |
| An Efficient Post-processing Using DCT Domain Projections Onto Convex Sets..... | 63 |
| <i>Changhoon Yim</i> | |
| Rank-Ordered Differences Statistic Based Switching Vector Filter..... | 74 |
| <i>Guillermo Peris-Fajarnés, Bernardino Roig, Anna Vidal</i> | |
| Mathematical Analysis of “Phase Ramping” for Super-Resolution Magnetic Resonance Imaging..... | 82 |
| <i>Gregory S. Mayer, Edward R. Vrscay</i> | |
| Blind Blur Estimation Using Low Rank Approximation of Cepstrum ... | 94 |
| <i>Adeel A. Bhutta, Hassan Foroosh</i> | |
| An Image Interpolation Scheme for Repetitive Structures..... | 104 |
| <i>Hiệp Luong, Alessandro Ledda, Wilfried Philips</i> | |
| A Discontinuous Finite Element Method for Image Denoising..... | 116 |
| <i>Zhaozhong Wang, Feihu Qi, Fugen Zhou</i> | |
| An Edge-Preserving Multigrid-Like Technique for Image Denoising..... | 126 |
| <i>Carolina Toledo Ferraz, Luis Gustavo Nonato, José Alberto Cuminato</i> | |

| | |
|---|-----|
| Fuzzy Bilateral Filtering for Color Images | 138 |
| <i>Samuel Morillas, Valentín Gregori, Almanzor Sapena</i> | |
| MPEG Postprocessing System Using Edge Signal Variable Filter | 146 |
| <i>Chan-Ho Han, Suk-Hwan Lee, Seong-Geun Kwon, Ki-Ryong Kwon, Dong Kyue Kim</i> | |
| Computational Framework for Family of Order Statistic Filters for Tensor Valued Data | 156 |
| <i>Bogusław Cyganek</i> | |
| Gaussian Noise Removal by Color Morphology and Polar Color Models | 163 |
| <i>Francisco Ortiz</i> | |
| Image Segmentation | |
| A Shape-Based Approach to Robust Image Segmentation | 173 |
| <i>Samuel Dambreville, Yogesh Rathi, Allen Tannenbaum</i> | |
| Novel Statistical Approaches to the Quantitative Combination of Multiple Edge Detectors | 184 |
| <i>Stamatia Giannarou, Tania Stathaki</i> | |
| Bio-inspired Motion-Based Object Segmentation | 196 |
| <i>Sonia Mota, Eduardo Ros, Javier Díaz, Rodrigo Agis, Francisco de Toro</i> | |
| An Effective and Fast Scene Change Detection Algorithm for MPEG Compressed Videos | 206 |
| <i>Z. Li, J. Jiang, G. Xiao, H. Fang</i> | |
| Automatic Segmentation Based on AdaBoost Learning and Graph-Cuts | 215 |
| <i>Dongfeng Han, Wenhui Li, Xiaosuo Lu, Tianzhu Wang, Yi Wang</i> | |
| Accurate Contour Detection Based on Snakes for Objects with Boundary Concavities | 226 |
| <i>Shin-Hyoung Kim, Ashraf Alattar, Jong Whan Jang</i> | |
| Graph-Based Spatio-temporal Region Extraction | 236 |
| <i>Eric Galmar, Benoit Huet</i> | |
| Performance Evaluation of Image Segmentation | 248 |
| <i>Fernando C. Monteiro, Aurélio C. Campilho</i> | |
| Improvement of Image Transform Calculation Based on a Weighted Primitive | 260 |
| <i>María Teresa Signes Pont, Juan Manuel García Chamizo, Higinio Mora Mora, Gregorio de Miguel Casado</i> | |

| | |
|--|-----|
| Topological Active Nets Optimization Using Genetic Algorithms | 272 |
| <i>O. Ibáñez, N. Barreira, J. Santos, M.G. Penedo</i> | |
| An Evaluation Measure of Image Segmentation Based on Object Centres | 283 |
| <i>J.J. Charles, L.I. Kuncheva, B. Wells, I.S. Lim</i> | |
| Active Shape Model Based Segmentation and Tracking of Facial Regions in Color Images | 295 |
| <i>Bogdan Kwolek</i> | |
| Image and Video Processing and Analysis | |
| On the Adaptive Impulsive Noise Attenuation in Color Images | 307 |
| <i>Bogdan Smolka</i> | |
| A Simple Method for Designing 2D $ M $ -Channel Near-PR Filter Banks with Linear Phase Property | 318 |
| <i>Guangming Shi, Liang Song, Xuemei Xie, Danhua Liu</i> | |
| Fast Hermite Projection Method | 329 |
| <i>Andrey Krylov, Danil Korchagin</i> | |
| Posterior Sampling of Scientific Images | 339 |
| <i>Azadeh Mohebi, Paul Fieguth</i> | |
| Image Denoising Using the Lyapunov Equation from Non-uniform Samples | 351 |
| <i>João M. Sanches, Jorge S. Marques</i> | |
| New Method for Fast Detection and Removal of Impulsive Noise Using Fuzzy Metrics | 359 |
| <i>Joan-Gerard Camarena, Valentín Gregori, Samuel Morillas, Guillermo Peris-Fajarnés</i> | |
| Background Subtraction Framework Based on Local Spatial Distributions | 370 |
| <i>Pierre-Marc Jodoin, Max Mignotte, Janusz Konrad</i> | |
| Morphological Image Interpolation to Magnify Images with Sharp Edges | 381 |
| <i>Valérie De Witte, Stefan Schulte, Etienne E. Kerre, Alessandro Ledda, Wilfried Philips</i> | |
| Adaptive Kernel Based Tracking Using Mean-Shift | 394 |
| <i>Jie-Xin Pu, Ning-Song Peng</i> | |
| Real Time Sobel Square Edge Detector for Night Vision Analysis | 404 |
| <i>Ching Wei Wang</i> | |

A New Video Images Text Localization Approach Based on a Fast Hough Transform 414
Bassem Bouaziz, Walid Mahdi, Abdelmajid Ben Hamadou

Video Sequence Matching Using Singular Value Decomposition 426
Kwang-Min Jeong, Joon-Jae Lee, Yeong-Ho Ha

The Papoulis-Gerchberg Algorithm with Unknown Signal Bandwidth 436
Manuel Marques, Alexandre Neves, Jorge S. Marques, João Sanches

Image and Video Coding and Encryption

Continuous Evolution of Fractal Transforms and Nonlocal PDE Imaging 446
Edward R. Vrscay

A Fast Algorithm for Macroblock Mode Selection in H.264 Video Coding 458
Donghyung Kim, Jongho Kim, Seungjong Kim, Jechang Jeong

Visual Secret Sharing Scheme: Improving the Contrast of a Recovered Image Via Different Pixel Expansions 468
Ching-Nung Yang, Tse-Shih Chen

Fast and Efficient Basis Selection Methods for Embedded Wavelet Packet Image Coding 480
Yongming Yang, Chao Xu

Fractal Image Coding as Projections Onto Convex Sets 493
Mehran Ebrahimi, Edward R. Vrscay

DCT-Domain Predictive Coding Method for Video Compression 507
Kook-yeol Yoo

Simple Detection Method and Compensation Filter to Remove Corner Outlier Artifacts 516
Jongho Kim, Donghyung Kim, Jechang Jeong

Rate Control Algorithm for High Quality Compression of Static Test Image in Digital TV System 526
Gwang-Soon Lee, Soo-Wook Jang, Chan-Ho Han, Eun-Su Kim, Kyu-Ik Sohng

JPEG2000-Based Resolution- and Rate-Constrained Layered Image Coding 535
Jing-Fung Chen, Wen-Jyi Hwang, Mei-Hwa Liu

| | |
|--|-----|
| A Permutation-Based Correlation-Preserving Encryption Method for Digital Videos | 547 |
| <i>Daniel Socek, Hari Kalva, Spyros S. Magliveras, Oge Marques, Dubravko Čulibrk, Borko Furht</i> | |
| A Fast Scheme for Converting DCT Coefficients to H.264/AVC Integer Transform Coefficients | 559 |
| <i>Gao Chen, Shouxiun Lin, Yongdong Zhang</i> | |
| A Fast Full Search Algorithm for Motion Estimation Using Priority of Matching Scan | 571 |
| <i>Taekyung Ryu, Gwang Jung, Jong-Nam Kim</i> | |
| Using Space-Time Coding for Watermarking Color Images | 580 |
| <i>Mohsen Ashourian</i> | |
| Blind PSNR Estimation of Video Sequences, Through Non-uniform Quantization Watermarking | 587 |
| <i>Tomás Brandão, Paula Queluz</i> | |
| Pattern Recognition Using Neighborhood Coding | 600 |
| <i>Ing Ren Tsang, Ing Jyh Tsang</i> | |
| Image Retrieval and Indexing | |
| Naming of Image Regions for User-Friendly Image Retrieval | 612 |
| <i>Andrea Kutics, Akihiko Nakagawa</i> | |
| Shape Retrieval Using Shape Contexts and Cyclic Dynamic Time Warping | 624 |
| <i>Vicente Palazón, Andrés Marzal</i> | |
| Topological Active Nets for Object-Based Image Retrieval | 636 |
| <i>David García-Pérez, Stefano Berretti, Antonio Mosquera, Alberto Del Bimbo</i> | |
| Iterative 3-D Pose Correction and Content-Based Image Retrieval for Dorsal Fin Recognition | 648 |
| <i>John Stewman, Kelly Debure, Scott Hale, Adam Russell</i> | |
| Feature Selection for Retrieval Purposes | 661 |
| <i>Marco Reisert, Hans Burkhardt</i> | |
| DCT-Domain Image Retrieval Via Block-Edge-Patterns | 673 |
| <i>K.J. Qiu, J. Jiang, G. Xiao, S.Y. Irianto</i> | |
| Visual Aspect: A Unified Content-Based Collaborative Filtering Model for Visual Document Recommendation | 685 |
| <i>Sabri Boutemedjet, Djemel Ziou</i> | |

Intellisearch: Intelligent Search for Images and Text on the Web 697
Epimenides Voutsakis, Euripides G.M. Petrakis, Evangelos Milios

Automatic Shot-Change Detection Algorithm Based on Visual
 Rhythm Extraction 709
*Kwang-Deok Seo, Seong Jun Park, Jin-Soo Kim,
 Samuel Moon-Ho Song*

Motion Analysis

Global Motion Estimation: Feature-Based, Featureless, or Both ?! 721
Rui F.C. Guerreiro, Pedro M.Q. Aguiar

Background Updating with the Use of Intrinsic Curves 731
Joaquín Salas, Pedro Martínez, Jordi González

Towards a New Paradigm for Motion Extraction 743
Luc Florack, Bart Janssen, Frans Kanters, Remco Duits

Fast Motion Estimation Using Spatio Temporal Filtering 755
V. Bruni, D. De Canditiis, D. Vitulano

Optic Flow from Multi-scale Dynamic Anchor Point Attributes 767
B.J. Janssen, L.M.J. Florack, R. Duits, B.M. ter Haar Romeny

The Effect of Presmoothing Image Sequences on the Computation
 of Optical Flow 780
J.V. Condell, B.W. Scotney, P.J. Morrow

Optical Flow Based Frame Interpolation of Ultrasound Images 792
Tae-Jin Nam, Rae-Hong Park, Jae-Ho Yun

An Iterative Multiresolution Scheme for SFM 804
*Carme Julià, Angel Sappa, Felipe Lumbreras, Joan Serrat,
 Antonio López*

Occlusion-Based Accurate Silhouettes from Video Streams 816
Pedro M.Q. Aguiar, António R. Miranda, Nuno de Castro

Tracking

Towards Constrained Optimal 3D Tracking 827
Huiying Chen, Youfu Li

A Generic Approach to Object Matching and Tracking 839
Xiaokun Li, Chiman Kwan, Gang Mei, , Baoxin Li

Image Based Visual Servoing: A New Method for the Estimation
 of the Image Jacobian in Dynamic Environments 850
L. Pari, J.M. Sebastián, C. González, L. Angel

| | |
|---|-----|
| QP_TR Trust Region Blob Tracking Through Scale-Space with Automatic Selection of Features | 862 |
| <i>Jingping Jia, Qing Wang, Yanmei Chai, Rongchun Zhao</i> | |
| Human Posture Analysis Under Partial Self-occlusion | 874 |
| <i>Ruixuan Wang, Wee Kheng Leow</i> | |
| Particle Filtering with Dynamic Shape Priors | 886 |
| <i>Yogesh Rathi, Samuel Dambreville, Allen Tannenbaum</i> | |
| The OBSERVER: An Intelligent and Automated Video Surveillance System | 898 |
| <i>Duarte Duque, Henrique Santos, Paulo Cortez</i> | |
| A Novel Spatio-temporal Approach to Handle Occlusions in Vehicle Tracking | 910 |
| <i>Alessandro Bevilacqua, Stefano Vaccari</i> | |
| A Robust Particle Filter-Based Face Tracker Using Combination of Color and Geometric Information | 922 |
| <i>Bogdan Raducanu, Y. Jordi Vitrià</i> | |
| Author Index | 935 |

Table of Contents – Part II

Pattern Recognition for Image Analysis

| | |
|--|-----|
| Using Local Integral Invariants for Object Recognition in Complex Scenes | 1 |
| <i>Alaa Halawani, Hashem Tamimi, Hans Burkhardt, Andreas Zell</i> | |
| Sharing Visual Features for Animal Categorization: An Empirical Study | 13 |
| <i>Manuel J. Marín-Jiménez, Nicolás Pérez de la Blanca</i> | |
| Object Categorization Using Kernels Combining Graphs and Histograms of Gradients | 23 |
| <i>F. Suard, A. Rakotomamonjy, A. Benschrair</i> | |
| Alternative Approaches and Algorithms for Classification | 35 |
| <i>Askin Demirkol, Zafer Demir, Erol Emre</i> | |
| A Pool of Classifiers by SLP: A Multi-class Case | 47 |
| <i>Sarunas Raudys, Vitalij Denisov, Antanas Andrius Bielskis</i> | |
| A Graph Spectral Approach to Consistent Labelling | 57 |
| <i>Hongfang Wang, Edwin R. Hancock</i> | |
| Gesture Recognition Using a Marionette Model and Dynamic Bayesian Networks (DBNs) | 69 |
| <i>Jörg Rett, Jorge Dias</i> | |
| On Subspace Distance | 81 |
| <i>Xichen Sun, Qiansheng Cheng</i> | |
| Ant Based Fuzzy Modeling Applied to Marble Classification | 90 |
| <i>Susana M. Vieira, João M.C. Sousa, João R. Caldas Pinto</i> | |
| A General Weighted Fuzzy Clustering Algorithm | 102 |
| <i>Zhiqiang Bao, Bing Han, Shunjun Wu</i> | |
| Integration of Expert Knowledge and Image Analysis Techniques for Medical Diagnosis | 110 |
| <i>P. Spyridonos, E.I. Papageorgiou, P.P. Groumpos, G.N. Nikiforidis</i> | |

Computer Vision

| | |
|--|-----|
| Depth Recovery from Motion and Defocus Blur | 122 |
| <i>Huei-Yung Lin, Chia-Hong Chang</i> | |
| Using Cartesian Models of Faces with a Data-Driven and Integrable Fitting Framework | 134 |
| <i>Mario Castelán, Edwin R. Hancock</i> | |
| A Novel Omnidirectional Stereo Vision System with a Single Camera . . . | 146 |
| <i>Sooyeong Yi, Narendra Ahuja</i> | |
| A Neural Network for Simultaneously Reconstructing Transparent and Opaque Surfaces | 157 |
| <i>Mohamad Ivan Fanany, Itsuo Kumazawa</i> | |
| A Light Scattering Model for Layered Rough Surfaces | 169 |
| <i>Hossein Ragheb, Edwin R. Hancock</i> | |
| Model Based Selection and Classification of Local Features for Recognition Using Gabor Filters | 181 |
| <i>Plinio Moreno, Alexandre Bernardino, José Santos-Victor</i> | |
| Inferring Stochastic Regular Grammar with Nearness Information for Human Action Recognition | 193 |
| <i>Kyungeun Cho, Hyungje Cho, Kyhyun Um</i> | |
| Real Time Vehicle Pose Using On-Board Stereo Vision System | 205 |
| <i>Angel D. Sappa, David Gerónimo, Fadi Dornaika, Antonio López</i> | |
| Content-Based 3D Retrieval by Krawtchouk Moments | 217 |
| <i>Pan Xiang, Chen Qihua, Liu Zhi</i> | |
| Uncalibrated Visual Servoing in 3D Workspace | 225 |
| <i>Paulo J. Sequeira Gonçalves, A. Paris, C. Christo, J.M.C. Sousa, J.R. Caldas Pinto</i> | |
| A Real-Time 3D Modeling System Using Multiple Stereo Cameras for Free-Viewpoint Video Generation | 237 |
| <i>Hansung Kim, Itaru Kitahara, Kiyoshi Kogure, Kwanghoon Sohn</i> | |
| CAD Model Visual Registration from Closed-Contour Neighborhood Descriptors | 250 |
| <i>Steve Bourgeois, Sylvie Naudet-Collette, Michel Dhome</i> | |

Biometrics

| | |
|--|-----|
| Is Enough Enough? What Is Sufficiency in Biometric Data? | 262 |
| <i>Galina V. Veres, Mark S. Nixon, John N. Carter</i> | |
| Improving Minutiae Detection in Fingerprints Using Multiresolution Contrast Enhancement | 274 |
| <i>Angelo Chianese, Vincenzo Moscato, Antonio Penta, Antonio Picariello</i> | |
| A Combined Radial Basis Function Model for Fingerprint Distortion | 286 |
| <i>Xuefeng Liang, Tetsuo Asano, Hui Zhang</i> | |
| Face and Ear: A Bimodal Identification System | 297 |
| <i>Andrea F. Abate, Michele Nappi, Daniel Riccio</i> | |
| Comparison of Novel Dimension Reduction Methods in Face Verification | 305 |
| <i>Licesio J. Rodríguez-Aragón, Cristina Conde, Enrique Cabello</i> | |
| Automatic 3D Face Feature Points Extraction with Spin Images | 317 |
| <i>Cristina Conde, Licesio J. Rodríguez-Aragón, Enrique Cabello</i> | |
| Face Recognition by Cortical Multi-scale Line and Edge Representations | 329 |
| <i>João Rodrigues, J.M. Hans du Buf</i> | |
| Generic Facial Encoding for Shape Alignment with Active Models | 341 |
| <i>William Ivaldi, Maurice Milgram, Stéphane Gentric</i> | |
| Ultra Fast GPU Assisted Face Recognition Based on 3D Geometry and Texture Data | 353 |
| <i>Andrea Francesco Abate, Michele Nappi, Stefano Ricciardi, Gabriele Sabatino</i> | |
| Face Recognition from Spatially-Morphed Video Sequences | 365 |
| <i>R. Sebastião, Jorge A. Silva, A.J. Padilha</i> | |

Shape and Matching

| | |
|--|-----|
| Spanning Trees from the Commute Times of Random Walks on Graphs | 375 |
| <i>Huaijun Qiu, Edwin R. Hancock</i> | |

On a Polynomial Vector Field Model for Shape Representation 386
Mickael Chekroun, Jérôme Darbon, Igor Ciril

A Fast Algorithm for Template Matching 398
Afsaneh Kohandani, Otman Basir, Mohamed Kamel

Shape Recognition Via an a Contrario Model for Size Functions. 410
Andrea Cerri, Daniela Giorgi, Pablo Musé, Frédéric Sur, Federico Tomassini

A Stable Marriages Algorithm to Optimize Satisfaction and Equity. 422
Nikom Suvonvorn, Bertrand Zavidovique

A Novel Approach for Affine Point Pattern Matching 434
Herbert Suesse, Wolfgang Ortmann, Klaus Voss

Geometric Invariant Curve and Surface Normalization 445
Sait Sener, Mustafa Unel

Estimating 3D Facial Shape and Motion from Stereo Image Using
 Active Appearance Models with Stereo Constraints. 457
Jaewon Sung, Daijin Kim

Approximation of a Polyline with a Sequence of Geometric
 Primitives. 468
Eugene Bodansky, Alexander Gribov

Biomedical Image Analysis

Real-Time Denoising of Medical X-Ray Image Sequences: Three
 Entirely Different Approaches 479
Marc Hensel, Thomas Pralow, Rolf-Rainer Grigat

Analysis of Fuzzy Clustering Algorithms for the Segmentation of
 Burn Wounds Photographs 491
A. Castro, C. Bóveda, B. Arcay

New Characteristics for the Classification of Burns: Experimental
 Study 502
Irene Fondón, Begoña Acha, Carmen Serrano, Manuel Sosa

The Class Imbalance Problem in TLC Image Classification 513
António Verejão Sousa, Ana Maria Mendonça, Aurélio Campilho

| | |
|--|-----|
| Faster, More Accurate Diffusion Filtering for Fetal Ultrasound Volumes | 524 |
| <i>Min-Jeong Kim, Hyun-Joo Yun, Myoung-Hee Kim</i> | |
| Fully Automatic Determination of Morphological Parameters of Proximal Femur from Calibrated Fluoroscopic Images Through Particle Filtering | 535 |
| <i>Xiao Dong, Guoyan Zheng</i> | |
| Analysis System of Endoscopic Image of Early Gastric Cancer | 547 |
| <i>Kwang-Baek Kim, Sungshin Kim, Gwang-Ha Kim</i> | |
| Transmission Tomography Reconstruction Using Compound Gauss-Markov Random Fields and Ordered Subsets | 559 |
| <i>A. López, J.M. Martín, R. Molina, A.K. Katsaggelos</i> | |
| Semivariogram Applied for Classification of Benign and Malignant Tissues in Mammography | 570 |
| <i>Valdeci Ribeiro da Silva Jr., Anselmo Cardoso de Paiva, Aristófanés Corrêa Silva, Alexandre Cesar Muniz de Oliveira</i> | |
| A Method for Interpreting Pixel Grey Levels in Digital Mammography | 580 |
| <i>Dieter Roller, Constanza Lampasona</i> | |
| Prostate Tissue Characterization Using TRUS Image Spectral Features | 589 |
| <i>S.S. Mohamed, A.M. Youssef, E.F. El-Saadany, M.M.A. Salama</i> | |
| Multi-dimensional Visualization and Analysis of Cardiac MR Images During Long-Term Follow-Up | 602 |
| <i>Min-Jeong Kim, Soo-Mi Choi, Yoo-Joo Choi, Myoung-Hee Kim</i> | |
| A Multiclassifier Approach for Lung Nodule Classification | 612 |
| <i>Carlos S. Pereira, Luís A. Alexandre, Ana Maria Mendonça, Aurélio Campilho</i> | |
| Lung Parenchyma Segmentation from CT Images Based on Material Decomposition | 624 |
| <i>Carlos Vinhais, Aurélio Campilho</i> | |
| Digitisation and 3D Reconstruction of 30 Year Old Microscopic Sections of Human Embryo, Foetus and Orbit | 636 |
| <i>Joris E. van Zwieten, Charl P. Botha, Ben Willekens, Sander Schutte, Frits H. Post, Huib J. Simonsz</i> | |

Skin Lesion Diagnosis Using Fluorescence Images 648
*Suhail M. Odeh, Eduardo Ros, Ignacio Rojas,
 Jose M. Palomares*

Special Session: Brain Imaging

3D Method of Using Spatial-Varying Gaussian Mixture and Local
 Information to Segment MR Brain Volumes 660
Zhigang Peng, Xiang Cai, William Wee, Jing-Huei Lee

Robust Ordering of Independent Spatial Components of fMRI Data
 Using Canonical Correlation Analysis 672
Wang Shijie, Luo Limin, Zhou Weiping

EpiGauss: Spatio-temporal Characterization of Epileptogenic Activity
 Applied to Hypothalamic Hamartomas 680
José Maria Fernandes, Alberto Leal, João Paulo Silva Cunha

Special Session: Remote Sensing Image Processing

Identification of Martian Polygonal Patterns Using the Dynamics
 of Watershed Contours 691
Pedro Pina, José Saraiva, Lourenço Bandeira, Teresa Barata

Fast Sparse Multinomial Regression Applied to Hyperspectral
 Data 700
Janete S. Borges, José M. Bioucas-Dias, André R.S. Marçal

A Bayesian Approach for Building Detection in Densely Build-Up
 High Resolution Satellite Image 710
Zongying Song, Chunhong Pan, Q. Yang

Striping Noise Removal of Satellite Images by Nonlinear
 Mapping 722
Euncheol Choi, Moon Gi Kang

Hyperspectral Image Analysis for Precision Viticulture 730
*Marcos Ferreiro-Armán, Jean-Pierre Da Costa, Saeid Homayouni,
 Julio Martín-Herrero*

Geometric and Radiometric Improvement of an Ikonos Panchromatic
 Image Using a Digital Surface Model 742
José Gonçalves

Applications

| | |
|--|-----|
| Defect Detection in Random Colour Textures Using the MIA T ² Defect Maps | 752 |
| <i>Fernando López, José Manuel Prats, Alberto Ferrer, José Miguel Valiente</i> | |
| Joint Spatial and Tonal Mosaic Alignment for Motion Detection with PTZ Camera | 764 |
| <i>Pietro Azzari, Alessandro Bevilacqua</i> | |
| Handwriting Similarities as Features for the Characterization of Writer's Style Invariants and Image Compression | 776 |
| <i>Djamel Gaceb, Véronique Eglin, Stéphane Bres, Hubert Emptoz</i> | |
| NN Automated Defect Detection Based on Optimized Thresholding | 790 |
| <i>Hugo Peres Castilho, João Rogério Caldas Pinto, António Limas Serafim</i> | |
| Pedestrian Detection Using Stereo and Biometric Information | 802 |
| <i>Philip Kelly, Eddie Cooke, Noel O'Connor, Alan Smeaton</i> | |
| A System for Automatic Counting the Number of Collembola Individuals on Petri Disk Images | 814 |
| <i>André R.S. Marçal, Cristina M.R. Caridade</i> | |
| Combining Template Matching and Model Fitting for Human Body Segmentation and Tracking with Applications to Sports Training | 823 |
| <i>Hao-Jie Li, Shou-Xun Lin, Yong-Dong Zhang</i> | |
| Integrating Low-Level and Semantic Visual Cues for Improved Image-to-Video Experiences | 832 |
| <i>Pedro Pinho, Joel Baltazar, Fernando Pereira</i> | |
| Multi-font Script Identification Using Texture-Based Features | 844 |
| <i>Andrew Busch</i> | |
| Comparison of Region and Edge Segmentation Approaches to Recognize Fish Oocytes in Histological Images | 853 |
| <i>S. Alén, E. Cernadas, A. Formella, R. Domínguez, F. Saborido-Rey</i> | |
| Fundamental Region Based Indexing and Classification of Islamic Star Pattern Images | 865 |
| <i>Mohamed Ould Djibril, Youssef Hadi, Rachid Oulad Haj Thami</i> | |

| | |
|---|------------|
| Automating Visual Inspection of Print Quality | 877 |
| <i>J. Vartiainen, S. Lyden, A. Sadovnikov, J.-K. Kamarainen, L. Lensu, P. Paalanen, H. Kalviainen</i> | |
| BigBatch – An Environment for Processing Monochromatic Documents | 886 |
| <i>Rafael Dueire Lins, Bruno Tenório Ávila, Andrei de Araújo Formiga</i> | |
| An HMM-SNN Method for Online Handwriting Symbol Recognition.... | 897 |
| <i>Bing Quan Huang, M.-T. Kechadi</i> | |
| A Novel Shadow Detection Algorithm for Real Time Visual Surveillance Applications | 906 |
| <i>Alessandro Bevilacqua</i> | |
| Author Index | 919 |

Self-Organizing Trees and Forests: A Powerful Tool in Pattern Clustering and Recognition

Ling Guan

Multimedia Research Laboratory, Ryerson University, Toronto, ON Canada
lguan@ee.ryerson.ca

Abstract. As the fruit of the Information Age comes to bare, the question of how such information, especially visual information, might be effectively harvested, archived and analyzed, remains a monumental challenge facing today's research community. The processing of such information, however, is often fraught with the need for conceptual interpretation: a relatively simple task for humans, yet arduous for computers. In attempting to handle oppressive volumes of visual information becoming readily accessible within consumer and industrial sectors, some level of automation remains a highly desired goal. To achieve such a goal requires computational systems that exhibit some degree of intelligence in terms of being able to formulate their own models of the data in question with little or no user intervention – a process popularly referred to as *Pattern Clustering* or *Unsupervised Pattern Classification*. One powerful tool in pattern clustering is the computational technologies based on principles of Self-Organization. In this talk, we explore a new family of computing architectures that have a basis in self organization, yet are somewhat free from many of the constraints typical of other well known self-organizing architectures. The basic processing unit in the family is known as the Self-Organizing Tree Map (SOTM). We will look at how this model has evolved since its inception in 1995, how it has inspired new models, and how it is being applied to complex pattern clustering problems in image processing and retrieval, and three dimensional data analysis and visualization.

1 Unsupervised Learning and Self Organization

The goal of Unsupervised Learning is to discover significant patterns or features in a given set of data, without the guidance of a teacher. The patterns are usually stored as a set of prototypes or clusters: representations or groupings of similar data.

In describing an unknown set of data, such techniques find much application across a wide range of industries, particularly in bioinformatics (clustering of genetic data, protein structure analysis), image processing (segmentation, image retrieval), and other applications that warrant a significant need for pattern discovery.

Unsupervised Learning & Self-Organization are inherently related. In general, self-organizing systems are typified by the union of local interactions and competition over some limited resource. In his book [2], Haykin identifies four major principles of self-organization: a) Synaptic Self-Amplification; b) Synaptic Competition; c) Cooperation; and d) Knowledge through Redundancy

1.1 Synaptic Self-amplification and Competition

The first principle of self organization is expressed through Hebb’s postulate of learning [3], which is supported by neurobiological evidence. This states that (on the cellular level) when two cells are within significant proximity enabling one to excite another, and furthermore, do so persistently, some form of physiological/metabolic growth process results. This process works to enhance the firing cell’s efficiency in triggering the second cell. This action of strengthening the association between two nodes functions as a correlation between their two states.

Typically, the Hebbian learning rule that ensues from this principle would eventually drive synapses into saturation were it not for the second principle, wherein some competition occurs for limited resources. These properties combined, have led to the modified Hebbian adaptive rule, as proposed by Kohonen [4]:

$$\Delta w_{k^*} = \alpha \varphi(k^*, x_i)[x_i - w_{k^*}], \tag{1}$$

where w_{k^*} is the synaptic weight vector of the *winning* neuron k^* (see below), α is the learning rate, and some scalar response $\varphi(\cdot)$ to the firing of neuron k^* (activation).

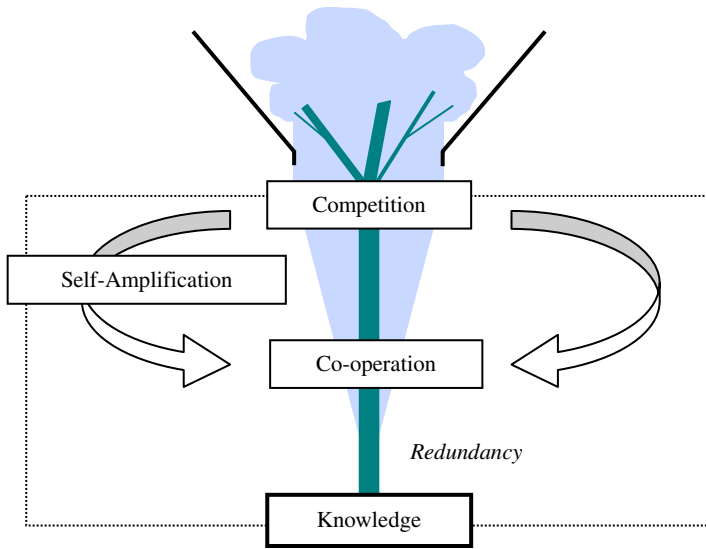


Fig. 1. The 4 Principles of Self-Organization – work to filter & encode the redundant (ordered/ structured patterns from a set of input data)

The activation function $\varphi(\cdot)$ is the result of the second principle (competition). Neurons generally compete to see which is most representative of a given input pattern presented to the network. Some form of discriminative function oversees this process (for example, choosing a neuron as a winner if its synaptic vector minimizes Euclidean distance over the set of all neurons). This process is often termed *Competitive Learning*.

1.2 Co-operation

Hebb's postulate is also suggestive of the lateral or associative aspect to the way knowledge is then captured in the network, i.e. not just through the winner, but also through nearby neurons in the output layer. This property is generally implemented in self-organizing architectures by virtue of the way in which nodes are interconnected. Often the strength of connection is not considered, but rather a simple link functions as an indicator of which other nodes in the network will more readily be associated with a winning node. Thus a local neighborhood is defined, and it is through this that knowledge may be imparted. Local adaptation usually follows the simple Kohonen update rule, whereby a portion of information is learned by the winning node, with neighboring nodes extracting lesser portions from the same input.

1.3 Knowledge Through Redundancy

Although not explicitly encoded, the final principle of self-organization implicitly results from the action of the first three. When exposed, any order or structure inherent within a series of activation patterns represents redundant information that is ultimately encoded by the network as knowledge. In other words, the network will evolve such that similar patterns will be captured and encoded by similar output nodes, while neighboring nodes organize themselves according to these dominant redundancies, each in turn focusing on and encoding lesser redundant patterns across the input space.

1.4 The Kohonen Self-Organizing Map (SOM)

Architectures such as Kohonen's *Self-Organizing Map* (SOM) [4] represent one of the most fundamental realizations of these principles, and as such have been the foundation for much research in pattern clustering and discovery.

Associative connections linking prototypes within SOM-based clustering algorithms are generally responsible for their innate ability to infer an *ordered* mapping of the underlying data space. Associations among nodes are advantageous as they help guide the evolution of such networks, and may also assist in formulating post-processing strategies or for extracting higher-level properties of any clusters discovered (e.g., inter-cluster relationships). This type of property is often used for the visualization of high-dimensional data – where multivariate data (of dimension greater than 3) is mapped onto a two-dimensional grid such as the lattice of a SOM. Since topology is preserved, neighbouring nodes in the lattice are often representative of related properties in the original data [5]. Unsupervised learning, however, is ill-posed: the nature of the underlying data is unknown, thus it is difficult to infer what an appropriate number of classes might be, or how they should be related.

Dynamically generating self-organizing networks attempt to address this issue by formulating a set of dynamic associations as they grow to represent the data space. Among the many proposed dynamic extensions to the classic SOM algorithm, there exist two principal approaches: *hierarchical* and *non-stationary*, as well as *hybrids* of these two. Many of these structures have foundations in various stationary methods including the SOM itself, *Competitive Learning* (CL) [6, 7], *Neural Gas* (NG) [8], or the *Hierarchical Feature Map* (HFM) [9].

2 The Self-Organizing Tree Map (SOTM)

The SOTM was first proposed in 1995 [10] and later developed [11, 12]. In the basic SOTM process, Competitive Learning is implemented just as in the SOM; however, the structure of the network is dynamic and is grown from a single node. In many ways, the SOTM process draws inspiration from Adaptive Resonance Theory (ART) applied to an SOM-type framework.

In ART [13-15], the network has an outer layer that is capable of storing a certain capacity of prototypes (output layer). With each input, competitive learning locates the nearest prototype in the outer layer, and a test is performed to establish how similar the input is to the existing prototype(s). This test is known as a vigilance test. If the input is within a tolerance of the winning prototype, then resonance is said to occur, resulting in the refinement of the winning prototype. Otherwise, while there is still capacity, a new prototype is formed.

Node or prototype insertion in the SOTM is driven by a top-down process that effectively explores the data space from the outside-in. An ellipsoid of significant similarity [28] forms a global vigilance threshold $H(t)$ that is used to evaluate the proximity of each new input sample against the closest existing prototype (winning node) currently in the network. A hard decision is made: if the input is distant from the winner beyond the threshold, a new prototype is spawned as a child node to the current winner, and the process continues with the next input. Otherwise, the input is deemed significantly similar and the winner is updated toward the input. This insertion process is shown in Figure 16(a) (see [28] for a more detailed treatment).

Through this dynamic vigilance function, Kohonen style adaptation and topological connections, the SOTM is distinguished from ART-based models. In fact, this vigilance is in a form of hierarchical control [22], as it evaluates proximity at decreasing levels during time and hierarchically partitions the data. The hierarchical approach naturally attempts to maximize discrimination across the entire data space in early phases of discovery before considering finer differences in the data.

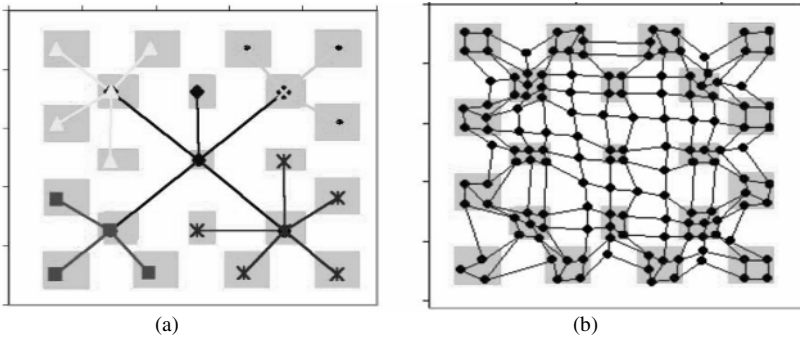


Fig. 2. Self-organizing data clustering; (a) performed by SOTM, no nodes converge to join the areas of zero data density; (b) performed by SOM, nodes converge to areas of zero data density

Error-driven mechanisms such as GNG, GCS and GG do not necessarily achieve this, as they unfold outward – expanding across the data space. As such, if the network was to exhibit a limited capacity for prototypes (i.e. be limited in the number of classes it uses), it is quite possible that nodes generated may not form a representation that adequately spans the data.

Due to the relationship among new nodes and their parents (winners at the time of generation), a tree structure naturally evolves during SOTM growth. This may also operate as an indicator of neighboring nodes, as they are generated according to proximity, as differing levels of granularity are parsed. This can be seen in Figures 3(a)-(b), which also show a comparison of the clustering efficiencies between the SOTM and the SOM on a two-dimensional feature space [11]. In these figures, the input vectors are uniformly distributed within many rectangular squares. The SOTM's clustering performance in Figure 2(a) shows that there are no neurons lying in a zero-density area. In contrast, although the SOM's topology exhibits the distribution of the structured input vectors, it also introduces several false representations outside of the distribution of the input space, as shown in Figure 2(b).

As indicated by this example, the SOTM realizes the learning such that its connections organize to form a suitable, efficient network structure. This property is necessary for a high-dimensional input space with a sparse data structure, where it is important to prioritize the formation of a model that effectively spans the data.

3 Application of SOTM to Image Retrieval

The goal of image retrieval is to find, from an image database, a set of images whose contents are similar to a given query. Each image in the database is considered as a vector point in the feature space that characterizes its color, shape, and texture information. A similarity function is applied to measure similarity among vectors.

In order to reduce the gap between low-level features used for image indexing and the high-level concepts used by users, relevance feedback (RF) techniques were introduced [17, 20]. In these techniques, a user provides initial query and trains (supervises) a search engine on what is regarded as relevant images through relevance feedback learning. The search engine is considered as a supervised learning unit that adapts itself according to user feedback.

There are two problems, however, associated with RF: a) the system requires a high degree of user workload in providing feedback samples through many cycles of relevance feedback before convergence, and b) the possible human subjective error is introduced when users provide relevance judgment to retrieved images.

In view of the above problems, SOTM was applied to minimize user workload by implementing automatic relevance feedback process [18, 19]. Repetitive user interaction steps are replaced by an SOTM module that adaptively guides RF. Experimental results have demonstrated the effectiveness of SOTM-based automatic relevance feedback [19].

The basic SOTM algorithm not only extracts global intuition from an input pattern space but also injects some degree of localization into the discriminative process such that maximal discrimination becomes a priority at any given resolution. However, it suffers from two problems: it is unable to decide on the relevant number of classes in a relevancy identification task; and often loses track of the true query position.

3.1 The Directed Self-Organizing Tree Map (DSOTM)

Losing the sense of query location within the input space can cause undesired effect on the true structure of the relevant class and can force the SOTM algorithm to spawn new clusters and form unnecessary boundaries within the query class as is illustrated in Figure 3. Therefore, retaining some degree of supervision to prevent unnecessary boundaries from forming around the query class is desirable.

A recent addition to the SOTM family, the Directed Self-Organizing Tree Map (DSOTM) has been proposed to solve the aforementioned problem. The DSOTM algorithm not only provides a partial supervision on cluster generation by forcing divisions away from the query class but also makes a gradual decision on resemblance of the input patterns by constantly modifying each sample's membership during the learning phase of the algorithm. As a result, a more robust topology with respect to the query, as well as a better sense of likeness, can be achieved [21].

On the other hand, DSOTM relies on the query position as the a-priori center of the relevant class and updates memberships according to this knowledge. Having said that, the synaptic vector adjustments in the DSOTM are not just limited to the winning node; the algorithm also constantly modifies all the centers according to the query position. Alternatively, if the winning center is not the query center, vector adjustments will affect both the winning node and the relevant center's position by moving the winning node (center of the irrelevant class) more toward the irrelevant samples and moving the relevant center more toward the query center. Thus, as more samples are exposed to the network, the DSOTM algorithm will learn less from irrelevant samples and more from the relevant ones by maintaining the integrity of a relevant center near the original query position. This action helps to foster a sense of generalization in the query class across feature subtleties that may objectively be deemed as discriminative by an independently operating SOTM. The DSOTM desensitizes partitioning in the region of a query.

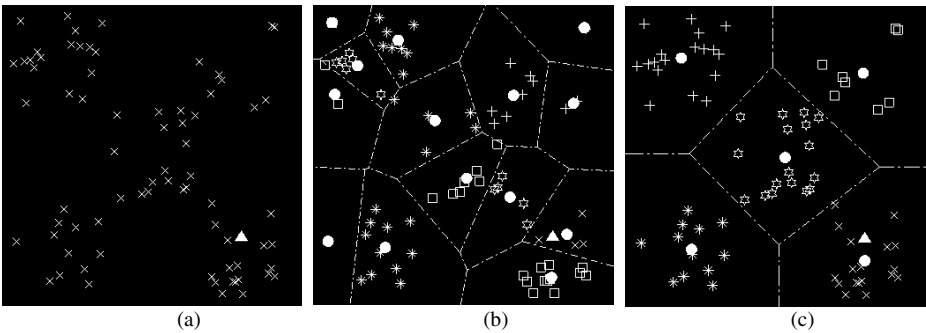


Fig. 3. 2-D mapping: (a) Input pattern with 5 distinct clusters, (b) 14 generated centers using SOTM, and (c) 5 generated centers using DSOTM. The SOTM forms a boundary near the query (triangle) contaminating relevant samples, where as some supervision is maintained in the DSOTM case, preventing unnecessary boundaries from forming.

3.2 Genetic Algorithm and Automatic CBIR System (GA-CBIR)

The block diagram of the proposed GA-CBIR is illustrated in Figure 4. Since DSOTM seeks to provide a better judgment about resemblance of input samples to one another and since GA aims to select and reproduce a better fit population of candidate solutions, this DSOTM-GA combination can effortlessly achieve a fully automated retrieval engine. The GA thus attempts to mimic human attention by working and coordinating the search subjectively, feeding and evaluating results from the more objective DSOTM. This is possible by highlighting dominant characteristics of images through feature weighting which seeks to emphasize certain characteristics in an image (i.e. its color, texture, and/or shape) that might provide significant information to the DSOTM for a more robust classification. Assigning proper weights for individual features can significantly increase performance behavior of CBIR systems, reducing the need for human supervision.

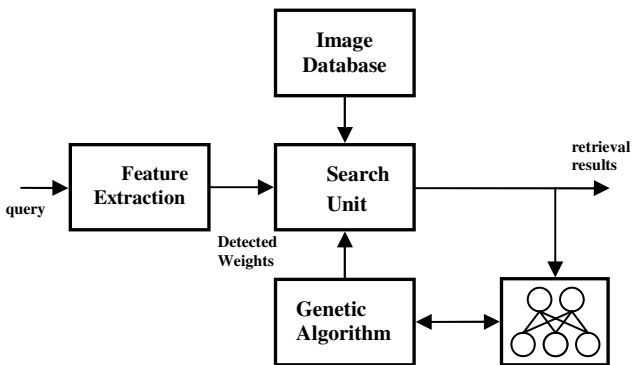


Fig. 4. Genetic Algorithm-Based CBIR

Table 1. Experimental results in terms of rr

| Query Sets | CBIR with SOTM | CBIR with DSOTM | GA-CBIR with SOTM | GA-CBIR with DSOTM |
|------------|----------------|-----------------|-------------------|--------------------|
| A | 47.5% | 58.0% | 66.8% | 78.3% |
| B | 47.4% | 59.6% | 72.1% | 76.7% |
| C | 51.0% | 56.8% | 74.4% | 80.5% |
| Ave. | 48.6% | 58.1% | 71.1% | 78.5% |

A number of experiments were conducted to compare behaviors of the automatic CBIR and GA-CBIR engines using SOTM and DSOTM clustering algorithms. Simulations were carried out using a subset of the Corel image database consisting of nearly 10,000 JPEG color images, covering a wide range of real-life photos from 100 different categories. Each category consisted of 100 visually-associated objects to simplify the measurements of the retrieval accuracy during the experiments. Three sets of 100 images were drawn from the database. In sets A and B, images were randomly selected from the entire set without regard for class; whereas in set C,

images were randomly selected such that no two images were from the same class. Retrieval results were statistically calculated from each of the three sets. In the simulations, a total of 16 most relevant images were retrieved to evaluate the performance of the retrieval. The experimental results are illustrated in Table 1. A typical retrieval result is shown in Figure 5.

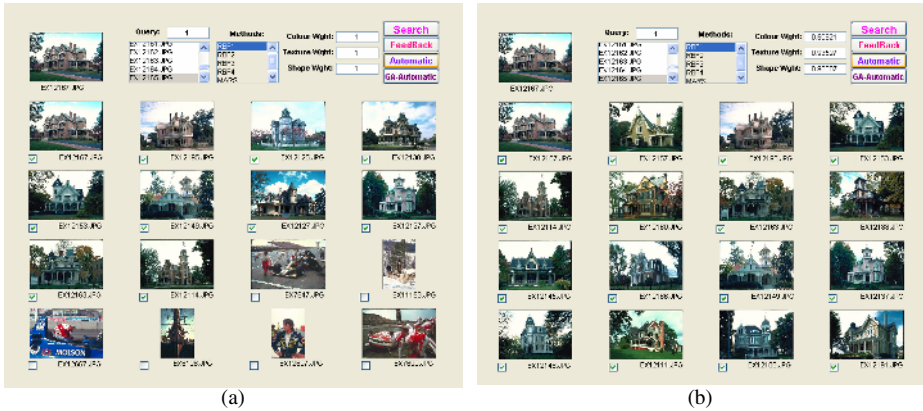


Fig. 5. Screenshot of the GA-CBIR Engine: (a) 62.5% retrieval rate for the query image (on top-left corner) using DSOTM classifier without GA feature weight detection; and (b) 100% retrieval rate for the same query using GA feature weight detection and DSOTM classifier

4 SOTM-Cluster Modeling for Microbiological Image Analysis

Microscopy has long been considered an essential investigative tool in the biological sciences. Recent advances in confocal modalities, in particular, have allowed for the non-invasive capture of spatially registered 3-D information regarding the physical and chemical structure of complex microbiological environments. With possibilities for new insights, interest is shifting from how to collect the pretty picture toward the more significant issue of how to adequately quantify this information for more systematic visualization, characterization and modeling. In such scenarios, any form of evaluation would typically be guided by an expert. Analysis often proceeds with an initial segmentation, wherein the expert attempts to identify meaningful biomass he/she wishes to analyze. Such steps become quite prohibitive if dealing with large 3-D volumes of data. In addition, they are rarely reproducible, particularly if evaluating data sets are highly heterogeneous (a common feature of biological image data). As such, an unacceptable level of subjectivity is introduced into the analysis. A systematic approach to microbiological image analysis is shown in Figure 7.

4.1 Visualization

As a visualization example, a volume rendering of a dataset could use the associations among classes to establish how to assign transparencies and color schemes to enhance or focus visualization on particular constituents of interest. Figure 8 shows such an

example of extracting orchid root tip chromosomes from a confocal stack captured using a phase-contrast modality [22]. The modality is necessary as the chromosomes are translucent in their natural state. Often, such methods yield subtleties in structural variation that cannot be detected with an epi-fluorescent modality. In the current treatment, visualization of the chromosomes is achieved through segmentation with the SOTM, an extension of [23].

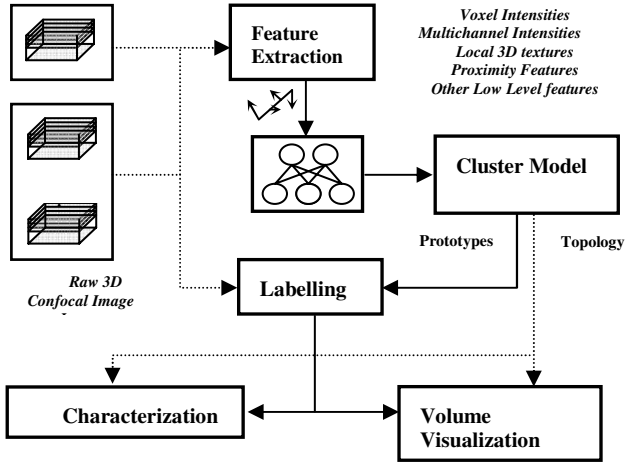


Fig. 6. Systematic Approach to Microbiological Image Analysis. SOTM forms a cluster model, which acts as a backbone for visualization and characterization tasks.

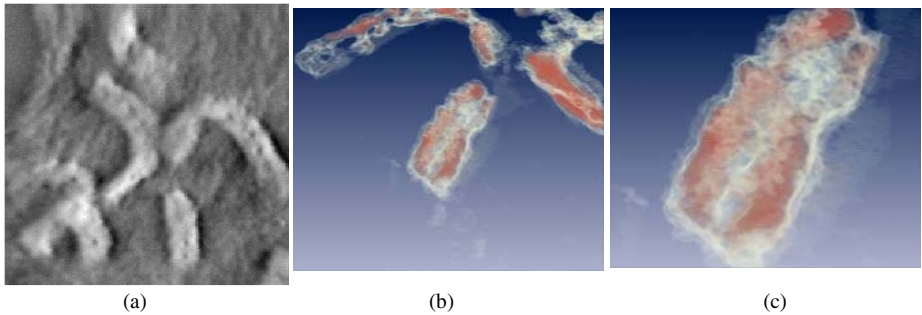


Fig. 7. SOTM Segmentation for Volume Visualisation. TOP Chromosome visualisation: (a) original slice 9/18, (b),(c) segmented and volume-rendered.

In the SOTM case, the transition into a volume visualization framework is more natural, as topological associations provide a more suitable associative backbone through which opacities of extracted microstructure types may be more readily controlled. In this instance, two dominant voxel prototypes within the chromosome are isolated and emphasized with imposed color contrast. Figure 8 shows a sample chromosome slice and the resulting volume-rendered dataset.

4.2 Characterization and Analysis

The other example highlights the problem of user-driven subjectivity for characterization. Consider two slices from different fields of view within a biofloc specimen (Figure 8 (a), (b)) which appear quite different; however, there are aspects that are similar; for instance, the localized clusters of bacteria nodules forming in surrounding fine grain material.

Typically, characterization processes would involve thresholding process [24, 25]. Such methods form a precursor to characterization software such as ISA [26] – an industry standard product for quantifying biofilm. In Biofloc, because a highly heterogeneous nature of the biomass, much potentially important information which characterizes the state of the biofilm/floc could be lost as indicated in Figure 8(d), where [24] cannot discriminate between internal constituents from Figure 8(b).

The well-known K-Means algorithm was used in Figure 8 (c), initialized with the same number of classes as discovered by the SOTM result in Figure 8(e). In this case, proximity features were incorporated in the feature extraction phase.

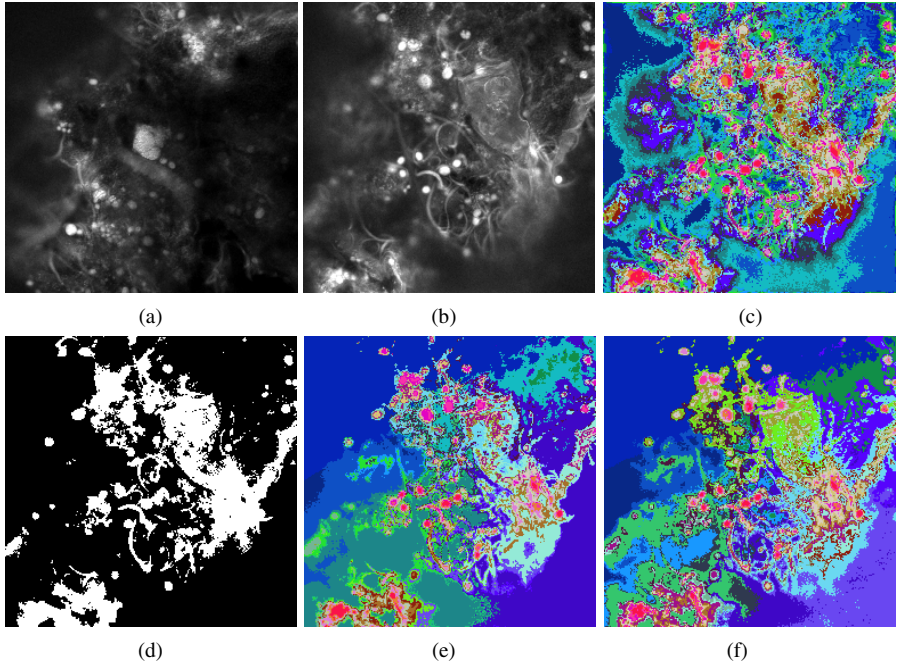


Fig. 8. SOTM Segmentations of Biofilm; from left to right, top to bottom: (a),(b) two slices of biofilm. Segmentations of (b): (c) K-means and (d) Otsu's threshold; (e) SOTM; (f) SOTM with dynamic feature weighting during learning. The final two cases show increased structural resolution and association in a manner that appears more perceptually relevant.

The SOTM shows a highly efficient allocation of classes to foreground biomass. It represents a much more perceptually relevant decomposition of the original image slice of Figure 8(b) than was achieved by K-Means. Interestingly, relationships

associating nearby particles of similar content have been discovered by the SOTM: the two bacterial nodules shown as two major spatial groupings in Figure 8(e).

Figure 8(f) shows an additional experimental result that implements a dynamic feature weighting strategy in learning. Essentially, proximity features were favored early during learning, and signal features were favored later. The flexibility of the SOTM tree structure allowed for an easy shift in allocation across these two feature subspaces. In fact, along with the hierarchical parsing, this worked to resolve even more structural subtleties as discrimination was maximized at all levels of evolution.

6 Local Variance Driven Self-Organization: Toward the Generation of Forests from Trees

We recently proposed the Self-Organizing Hierarchical Variance Map (SOHVM) which is also rooted in the principles of the SOTM [27]. This model differs from the SOTM in that it utilizes an intelligent vigilance strategy based on the underlying data distribution as it is discovered, rather than a globally decaying hierarchical function.

In addition to SOTM-based parent/children relationships, CHL is incorporated into the SOHVM's adaptive mechanism. The methodology thus shares properties common to GNG variants in that, as it partitions the input space divisively, a topological preserving set of associative connections among nodes is simultaneously constructed. In doing so, it is also possible for regions of the data space to disassociate at lower hierarchical levels. In this sense, the model begins with the growth of trees but may expand at later phases into forests of trees encoding even higher levels of association.

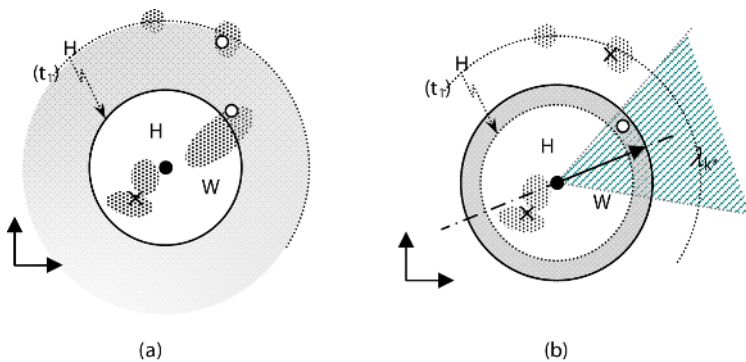


Fig. 9. A comparison of the decision regions used for node generation in an SOTM versus an SOHVM. The latter attempts to new insert nodes (open circles) at the extremities of the data described by a winning node (closed circle) – wk^* is the winning center, and its eigenvalue/vector pair (k^*, qk^*) describe local variance properties. X's are invalid sites for node insertion.

Each neuron in the network consists of a dual memory element to track information regarding a discovered prototype. In addition to center position (as in most other self-organizing structures), Hebbian-based Maximum Eigenfilters (HME) [28] simultaneously estimate the maximal variance of local data, presenting an

additional realization of the Hebbian postulate of learning to probe the nature of the covariance in the data within the vicinity of any given node. The HME embedded in each node provides for local orientation selectivity of the underlying data, by extracting the maximal eigenvector/eigenvalue pair from the locally surveyed data.

Vigilance is now assessed via interplay among local variances extracted such that more informed decisions control and naturally limit network growth. Figure 9 shows a comparison between the decision regions that result in the insertion of a new node in the SOTM versus the SOHVM. The hierarchical threshold begins as in the SOTM (very large), yet as nodes are inserted, regions of the data are isolated, thus their local variances are generally reduced. The hierarchical threshold decays in accordance with the variances discovered in the network. This continues as long as reasonable clusters are extracted. In regions where the hierarchical function falls below the local variance, the node is disabled, such that it can no longer act as a site for the spawning of new children. This limits over-fitting.

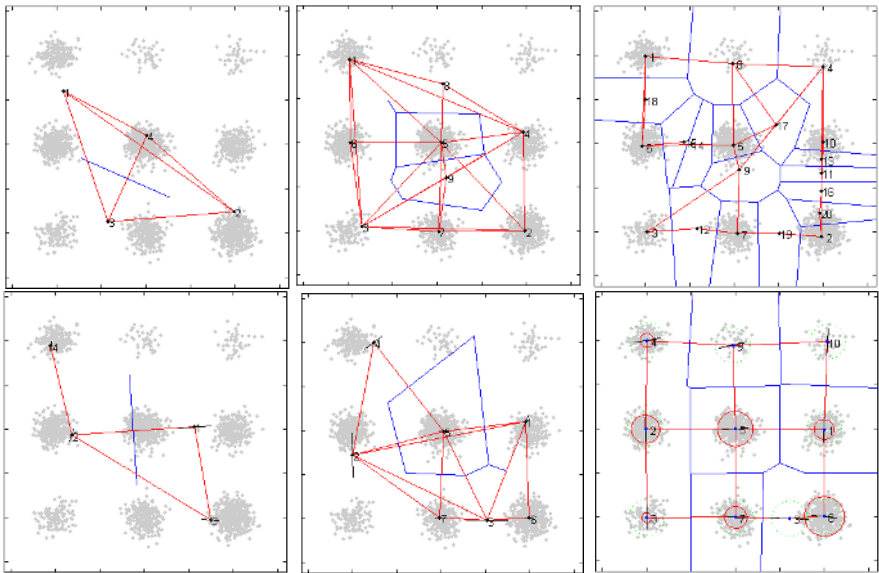


Fig. 10. GNG (top) evolves onto data space, but splits prematurely well before capturing all cluster positions. Classification thus subdivides clusters. SOHVM (bottom) maximally spans the data space through hierarchical partitioning; this property is localized through the HME units (lines show axis for orientation selectivity). Network infers more accurate number of total clusters.

Once the dense regions have been exposed, the network attempts to insert nodes into outlier or noise positions from which it is difficult to track back due to increased competition across the data space. In the SOHVM, however, the discovery of such an outlier (typified by high local variance and low probability) also drives the local variance high, disabling the node.

Ultimately, the SOHVM intelligently decides when to cease growing, thereby addressing one of the most difficult problems in unsupervised clustering – inferring

an appropriate number of clusters in which to partition the data. Figure 10 shows how the network unfolds over a sample data space in comparison to GNG.

Should complete disassociation occur between the significantly separate regions in the data, multiple trees would be produced (a forest of trees) encoding even higher levels of association. In such cases, the data may be evaluated in terms of collections of properties specific to individual tree groups, in addition to properties evaluated at the cluster level. It is anticipated that such mining approaches may thus extract inherent, but less obvious hierarchical relationships among the data.

7 Closing Remarks

Complex tasks in visual data processing that normally require extensive human interaction or guidance become prohibitive when there are overwhelming volumes of data that need to be processed. In most, a certain amount of perception is required to successfully complete them (a highly non-linear challenge). Artificial neural networks based on self-organization are particularly suited to modelling such nonlinearities, thus they have generated much research interest. Dynamically growing models of self-organization attempt to address the problem of inferring an appropriate number of classes from the underlying data distribution, and thus, exhibit greater flexibility than static approaches. In this discussion, the SOTM family of architectures is demonstrated for their propensity in formulating a topological set of classes that attempt to maximize pattern discrimination while hierarchically parsing an underlying distribution. The DSOTM and the SOHVM, newer variants of the SOTM have been proposed and applied with success to content-based image retrieval, and to analysis and visualization of 3-D microbiological data. It is demonstrated the SOTM variants are imperative components in the automation of perceptually-driven tasks. With the growing demand for such automation by many of today's visual data analysis and applications, dynamic self-organization is well-poised to leave its mark.

References

- [1] A.M. Turing, "The chemical basis of morphogenesis," *Philosophical Transactions of the Royal Society, B*, 1952 vol. 237, pp. 5-72.
- [2] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd Ed. Prentice Hall, New Jersey, 1999.
- [3] D.O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*, 1949, New York: Wiley.
- [4] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological Cybernetics*, 43:59-69.
- [5] G.G. Yen, and Z. Wu, "Ranked Centroid Projection: a Data Visualization Approach for Self-Organizing Maps," *Proc. Int. Joint Conf. on Neural Networks*, Montreal, Canada, July 31 - August 4, 2005, pp. 1587-1592.
- [6] T. Kohonen, "Self-organization and associative memory", Springer-Verlag, Berlin, 1984.
- [7] S. Grossberg, "Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural feature detectors," *Biological Cybernetics*, vol. 23, 1976, pp. 121-131.

- [8] T. Martinetz and K. Schulten, "A 'Neural-Gas' network learns topologies," *Artificial Neural Network*, vol. I, 1991, pp. 247-402.
- [9] R. Miikkulainen, "Script recognition with hierarchical feature maps," *Connection Science*, vol. 2, 1990.
- [10] H. Kong and L. Guan, "Detection and removal of impulse noise by a neural network guided adaptive median filter," *Proc. IEEE Int. Conf. on Neural Networks*, pp. 845-849, Perth, Australia, 1995.
- [11] J. Randall, L. Guan, X. Zhang, and W. Li, "Investigations of the self-organizing tree map," *Proc. Of Int. Conf. on Neural Information Processing*, vol 2, pp. 724-728, November 1999.
- [12] M. Kyan, L. Guan and S. Liss, "Dynamic Feature Fusion in the Self-Organizing Tree Map – applied to the segmentation of biofilm images," *Proc. Of Int. Joint Conf. on Neural Networks*, July 31-August 4, 2005, Montreal, Canada.
- [13] S. Grossberg, *Studies of Ming and Brain*, Boston: Reidel, 1982.
- [14] G. Carpenter, and S. Grossberg, "The ART of adaptive pattern recognition by a self-organizing neural network," *Computer*, vol. 21, no. 3, pp.77-87, 1988.
- [15] G.A. Carpenter, S. Grossberg, and J.H. Reynolds, "ARTMAP: supervised real-time learning and classification of nonstationary data by a self-organizing neural network," *Neural Networks*, vol. 4, pp. 565-588, 1991.
- [16] M. Kyan, L. Guan and S. Liss, "Refining competition in the self-organizing tree map for unsupervised biofilm segmentation," *Neural Networks*, Elsevier, July-Aug. 2005, Vol. 18(5-6), pp 850-860.
- [17] P. Muneesawang and L. Guan, "Interactive CBIR using RBF-based relevance feedback for WT/VQ coded images," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Salt Lake City, USA, pp. 1641-1644, vol. 3, May 2001.
- [18] P. Muneesawang, "Retrieval of image/video content by adaptive machine and user interaction," Ph.D. Thesis, The University of Sydney, 2002.
- [19] P. Muneesawang and L. Guan, "Automatic machine interactions for content-based image retrieval using a self-organizing tree map architecture," *IEEE Trans. on Neural Networks*, vol. 13, no. 4, pp. 821-821, July 2002.
- [20] Y. Rui and T.-S. Huang, "Optimizing learning in image retrieval," in *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, Hilton Head, SC, vol. 1, pp. 223-243, June 2000.
- [21] K. Jarrah, M. Kyan, S. Krishnan, and L. Guan, "Computational intelligence techniques and their Applications in Content-Based Image Retrieval," accepted by *IEEE Int. Conf. on Multimedia & Expo*, Toronto, July 9-12, 2006.
- [22] C.J. Cogswell and C.J.R. Sheppard, "Confocal differential interference contrast (DIC) microscopy: Including a theoretical analysis of conventional and confocal DIC imaging," *Journal of Microscopy*, vol. 165, no. 1, pp. 81–101, 1992.
- [23] M.J. Kyan, L. Guan, M.R. Arnison, and C.J. Cogswell, "Feature Extraction of Chromosomes From 3-D Confocal Microscope Images," *IEEE Trans. on Biomedical Engineering*, vol. 48, no. 11, November 2001, pp. 1306-1318.
- [24] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Trans. on Systems Man and Cybernetics*, 1979, 9 (1) 62-66.
- [25] J. Besag, "On the statistical analysis of dirty pictures (with discussion)," *J. Roy Statist Soc Ser B*, 48, 259-302, 1986.
- [26] Image Structure Analyzer ISA, Center for Biofilm Engineering, Montana State University.
- [27] M. Kyan and L. Guan, "Local Variance Driven Self-Organization for Unsupervised Clustering," accepted by *Int. Conf. on Pattern Recognition*, Hong Kong, Aug 20-24, 2006.
- [28] E. Oja, "A simplified neuron model as a principal component analyzer," *Journal of Mathematical Biology*, vol. 15, pp. 267-273.

On Optimizing Dissimilarity-Based Classification Using Prototype Reduction Schemes*

Sang-Woon Kim^{1,**} and B. John Oommen^{2,***}

¹ Dept. of Computer Science and Engineering,
Myongji University, Yongin, 449-728 Korea
kimsw@mju.ac.kr

² School of Computer Science, Carleton University,
Ottawa, Canada : K1S 5B6
oommen@scs.carleton.ca

Abstract. The aim of this paper is to present a strategy by which a new philosophy for pattern classification, namely that pertaining to Dissimilarity-Based Classifiers (DBC), can be efficiently implemented. This methodology, proposed by Duin¹ and his co-authors (see [3], [4], [5], [6], [8]), is a way of defining classifiers between the classes, and is not based on the feature measurements of the individual patterns, but rather on a suitable dissimilarity measure between them. The problem with this strategy is, however, the need to compute, store and process the inter-pattern dissimilarities for all the training samples, and thus, the accuracy of the classifier designed in the dissimilarity space is dependent on the methods used to achieve this. In this paper, we suggest a novel strategy to enhance the computation for all families of DBCs. Rather than compute, store and process the DBC based on the entire data set, we advocate that the training set be first reduced into a smaller representative subset. Also, rather than determine this subset on the basis of random selection, or clustering etc., we advocate the use of a Prototype Reduction Scheme (PRS), whose output yields the points to be utilized by the DBC. Apart from utilizing PRSs, in the paper we also propose simultaneously employing the Mahalanobis distance as the dissimilarity-measurement criterion to increase the DBC's classification accuracy. Our experimental results demonstrate that the proposed mechanism increases the classification accuracy when compared with the "conventional" approaches for samples involving real-life as well as artificial data sets.

* The work of the second author was done while visiting at Myongji University, Yongin, Korea. The second author was partially supported by NSERC, the Natural Sciences and Engineering Research Council of Canada and a grant from the Korean Research Foundation. This work was generously supported by the Korea Research Foundation Grant funded by the Korea Government (MOEHRD-KRF-2005-042-D00265).

** *Senior Member IEEE.*

*** *Fellow of the IEEE.*

¹ The authors are grateful to Bob Duin for his friendly and cooperative manner. The first author thanks him and his team for the time spent in Holland in October 2005, and the second author is grateful for the time spent together in Poland in May 2005.

1 Introduction

The field of statistical Pattern Recognition[1], [2] has matured since its infancy in the 1950's, and the aspiration to enlarge the horizons has led to numerous philosophically-new avenues of research. The fundamental questions tackled involve (among others) increasing the accuracy of the classifier system, minimizing the time required for training and testing, reducing the effects of the curse of dimensionality, and reducing the effects of peculiar data distributions.

One of the most recent novel developments in this field is the concept of Dissimilarity-Based Classifiers (DBC's) proposed by Duin and his co-authors (see [3], [4], [5], [6], [8]). Philosophically, the motivation for DBC's is the following: If we assume that "Similar" objects can be grouped together to form a class, a "class" is nothing more than a set of these "similar" objects. Based on this idea, Duin and his colleagues argue that the notion of proximity (similarity or dissimilarity) is actually more fundamental than that of a feature or a class. Indeed, it is probably more likely that the brain uses an intuitive DBC-based methodology than that of taking measurements, inverting matrices etc. Thus, DBC's are a way of defining classifiers between the classes, which are not based on the feature measurements of the individual patterns, but rather on a suitable *dissimilarity measure* between them. The advantage of this methodology is that since it does not operate on the class-conditional distributions, the accuracy can exceed the Bayes' error bound - which is, in our opinion, remarkable². Another salient advantage of such a paradigm is that it does not have to confront the problems associated with feature spaces such as the "curse of dimensionality", and the issue of estimating a large numbers of parameters. The problem with this strategy is, however, the need to compute, store and process the inter-pattern dissimilarities for (in the worst case) all the training samples, and thus, the accuracy of the classifier designed in the dissimilarity space is dependent on the methods used to achieve this.

A dissimilarity representation of a set of samples, $T = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$, is based on pairwise comparisons and is expressed, for example, as an $n \times n$ dissimilarity matrix³ $D_{T,T}[\cdot, \cdot]$, where the subscripts of D represent the set of elements on which the dissimilarities are evaluated. Thus each entry $D_{T,T}[i, j]$ corresponds to the dissimilarity between the pairs of objects $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$, $\mathbf{x}_i, \mathbf{x}_j \in T$. When it concerns testing any object in the sample space, \mathbf{x} , the latter is represented by a vector of proximities $\delta(\mathbf{x}, Z)$ to the objects in a specific set Z , which is used for the testing purposes. Thus, if $\mathbf{x} = \mathbf{x}_i$, and $Z = T$, $\delta(\mathbf{x}_i, T)$ is the i^{th} row of $D_{T,T}[\cdot, \cdot]$. The principle behind DBC's is that a new (testing) sample \mathbf{z} , if

² In our opinion, the theory of DBC's is one of the major contributions to the field of statistical PR in the last decade. Duin and his colleagues [3] have ventured to call this paradigm a *featureless* approach to PR by insisting that there is a clear distinction between feature-based and non-feature based approaches. We anticipate that a lot of new research energy will be expended in this direction in the future.

³ If the dissimilarity is not stored, but rather computed when needed, it would be more appropriate to regard it as a *function* $D_{T,T}(\cdot, \cdot)$.

represented by $\delta(\mathbf{z}, T)$, is classified to a specific class if it is sufficiently similar to one or more objects within that class.

The problem we study in this paper deals with how DBCs between classes represented in this manner can be effectively computed. The *families* of strategies investigated in this endeavour are many. First of all, by selecting a set of prototypes or support vectors, the problem of dimension reduction can be drastically simplified. In order to select such a representative set from the training set, the authors of [4] discuss a number of methods such as random selections, the k -centers method, and others which will be catalogued presently. Alternatively, some work has also gone into the area of determining appropriate measures of dissimilarity using measures such as various L_p Norms (including the Euclidean and $L_{0.8}$), the Hausdorff and Modified Hausdorff norm, and some traditional PR-based measures such as those used in Template matching, and Correlation-based analysis. These too which will be listed presently⁴.

1.1 Contributions of the Paper

We claim two modest contributions in this paper based on rigorous tests done on both established benchmark artificial and real-life data sets:

1. First of all, we show that a PRS⁵ can also be used as a *tool* to achieve an intermediate goal, namely, to minimize the number of samples that are *subsequently* used in any DBC system. This subset, will, *in turn* be utilized to design the classifier - which, as we shall argue, must be done in conjunction with an appropriate dissimilarity measure. This, in itself, is novel to the field when it concerns the *applications of PRSs*.
2. The second contribution of this paper is the fact that we have shown that using second-order distance measures, such the Mahalanobis distance, together with appropriate PRS, has a distinct advantage when they are used to implement the DBC.

2 Dissimilarity-Based Classification and Prototype Reduction Schemes

2.1 Foundations of DBCs

Let $T = \{\mathbf{x}_1, \dots, \mathbf{x}_n\} \in R^p$ be a set of n feature vectors in a p dimensional space. We assume that T is a labeled data set, so that T can be decomposed into, say, c subsets $\{T_1, \dots, T_c\}$ such that $\forall i \neq j$:

- (1) $T_i = \{\mathbf{x}_1, \dots, \mathbf{x}_{n_i}\}$, (2) $n = \sum_{i=1}^c n_i$, (3) $T = \bigcup_{k=1}^c T_k$, and (4) $T_i \cap T_j = \phi$.

Our goal is to design a DBC in an appropriate dissimilarity space constructed with this *training data* set, and to classify an input sample \mathbf{z} appropriately.

⁴ For want of a better term, DBCs enhanced with these prototype selection methods and the latter distance measures, will be referred to as “conventional” schemes.

⁵ Bezdek *et al* [9], who have composed an excellent survey of the field, report that there are “zillions!” of methods for finding prototypes (see page 1459 of [9]).

To achieve this, we assume that from T_i , the training data of class ω_i , we extract a prototype set⁶, Y_i , where,

$$(1) Y_i = \{\mathbf{y}_1, \dots, \mathbf{y}_{m_i}\}, \text{ and } (2) \quad m = \sum_{i=1}^c m_i.$$

Every DBC assumes the use of a dissimilarity measure, d , computed from the samples, where, $d(\mathbf{x}_i, \mathbf{y}_j)$ represents the dissimilarity between two samples \mathbf{x}_i and \mathbf{y}_j . Since, d , is required to be nonnegative, reflexive and symmetric⁷:

$$d(\mathbf{x}_i, \mathbf{y}_j) \geq 0 \text{ with } d(\mathbf{x}_i, \mathbf{y}_j) = 0 \text{ if } \mathbf{x}_i = \mathbf{y}_j, \text{ and } d(\mathbf{x}_i, \mathbf{y}_j) = d(\mathbf{y}_j, \mathbf{x}_i).$$

The dissimilarity computed between T and Y leads to a $n \times m$ matrix, $D_{T,Y}[i, j]$, where $\mathbf{x}_i \in T$ and $\mathbf{y}_j \in Y$. Consequently, an object \mathbf{x}_i is represented as a column vector as following :

$$[d(\mathbf{x}_i, \mathbf{y}_1), d(\mathbf{x}_i, \mathbf{y}_2), \dots, d(\mathbf{x}_i, \mathbf{y}_m)]^T, \quad 1 \leq i \leq n. \quad (1)$$

Here, we define the dissimilarity matrix $D_{T,Y}[\cdot, \cdot]$ to represent a *dissimilarity space* on which the p -dimensional object, \mathbf{x} , given in the feature space, is represented as an m -dimensional vector $\delta(\mathbf{x}, Y)$, where if $\mathbf{x} = \mathbf{x}_i$, $\delta(\mathbf{x}_i, Y)$ is the i^{th} row of $D_{T,Y}[\cdot, \cdot]$. In this paper, the column vector $\delta(\mathbf{x}, Y)$ is simply denoted by $\delta_Y(\mathbf{x})$, where the latter is an m -dimensional vector, while \mathbf{x} is p -dimensional.

For a training set $\{\mathbf{x}_i\}_{i=1}^n$, and an evaluation sample \mathbf{z} , the modified training set and sample now become $\{\delta_Y(\mathbf{x}_i)\}_{i=1}^n$ and $\delta_Y(\mathbf{z})$, respectively. From this perspective, we can see that the dissimilarity representation can be considered as a *mapping* by which any arbitrary \mathbf{x} is translated into $\delta_Y(\mathbf{x})$, and thus, if m is selected sufficiently small (i.e., $m \ll p$), we are essentially working in a space with much smaller dimensions. The literature reports the use of many traditional decision classifiers including k -NN rule and the linear/quadratic normal-density-based classifiers to the task of classifying \mathbf{z} using $\delta_Y(\mathbf{z})$ in the dissimilarity space.

2.2 Prototype Selection Methods for DBCs

We first consider the reported methods [6], [7], [8] by which each T_i is pruned to yield a set of representative *prototypes*, Y_i , where, without loss of generality $|Y_i| < |T_i|$. The intention is to guarantee a good tradeoff between the recognition accuracy and the computational complexity when the DBC is built on $D_{T,Y}(\cdot, \cdot)$ rather than $D_{T,T}(\cdot, \cdot)$. The reported comparison of [8] has been performed from the perspective of the resultant error rates and the the number of prototypes obtained. The experiments were conducted with seven artificial and real-life data sets. Eight selection methods employed for the experiments were *Random*, *Random-C*, *KCentres*, *ModeSeek*, *LinProg*, *PeatSeal*, *KCentres-LP*, and *EdiCon*. In the interest of completeness, we briefly explain below (using the notation introduced above) the methods that are pertinent to our present study.

1. *Random* : This method involves a *random* selection of m samples from the training data set T .

⁶ Since we are invoking a PRS to obtain Y_i from T_i , we do not require that $Y_i \subseteq T_i$. Rather Y_i may be created or selected from T_i , and its computation may also involve the other sets, $T_j, j \neq i$.

⁷ Note that $d(\cdot, \cdot)$ need not be a *metric* [8].

2. *RandomC*: This method involves a random selection of m_i samples per class, ω_i , from T_i .
3. *KCentres*: This method⁸ consists of a procedure that is applied to each class separately. For each class ω_i , the algorithm is invoked so as to choose m_i samples which are “evenly” distributed with respect to the dissimilarity matrix $D_{T_i, T_i}[\cdot, \cdot]$. The algorithm can be summarized as follows:
 - (a) Select an initial set $Y_i = \{\mathbf{y}_1, \dots, \mathbf{y}_{m_i}\}$ consisting of m_i objects, e.g. randomly chosen from T_i .
 - (b) For each $\mathbf{x} \in T_i$, find its nearest neighbor in Y_i . Let $N_j, j = 1, \dots, m_i$, be a subset of T_i consisting of objects that yield the same nearest neighbor \mathbf{y}_j in Y_i . This means that $T_i = \cup_{j=1}^{m_i} N_j$.
 - (c) For each N_j , find its center \mathbf{c}_j , which is the object for which the maximum distance to all other objects in N_j is minimum (this value is called the radius of N_j).
 - (d) For each center \mathbf{c}_j , if $\mathbf{c}_j \neq \mathbf{y}_j$, then replace \mathbf{y}_j by \mathbf{c}_j in Y_i . If any replacement is done, then return to Step (b). Otherwise exit.
 - (e) Return the final representation set Y consisting of all the final sets Y_i .

From the experimental results of [8], the authors seem to have deliberated that systematic approaches lead to better results than those which rely on random selection, especially when the number of prototypes is small. Furthermore, although there is no single winner (inasmuch as the results depend on the characteristics of the data), they indicate that, in general, the *KCentres* works well.

The details of the other methods (see [8]) such as *ModeSeek*, *FeatSel*, *LinProg*, *KCentres-LP*, and *EdiCon* are omitted here as they are not directly related to the premise of our work. Whereas our present work and the above three methods pursue a pruning in the *original feature space*, the methods omitted here attempt the same in the *dissimilarity space*.

2.3 Dissimilarity Measures Used in DBCs

Fundamental to DBCs is the measure used to quantify the dissimilarity between two vectors⁹. The work in [8] reports extensive experiments conducted using various dissimilarity measures (see Table 2 of [8]). A list of these measures where we quantify the dissimilarity between \mathbf{v} and $\mathbf{w} \in R^q$, is given below:

1. *City Block Norm*: $D_1 = \sum_{i=1}^q |v_i - w_i|$.
2. *Euclidean Norm*: D_E (or D_2) = $\sqrt{(\mathbf{v} - \mathbf{w})^T (\mathbf{v} - \mathbf{w})}$.
3. *Max Norm*: $D_{max} = \text{Max}_i |v_i - w_i|$.
4. *L_p or Minkowski Norm*: $D_p = (\sum_{i=1}^q |v_i - w_i|^p)^{1/p}, p \geq 1, p \neq 2$.
5. *Hausdorff Norm*:

$$h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\| \quad (2)$$

⁸ This procedure is essentially identical to the *k*-means clustering algorithm performed in a vector space, and is thus heavily dependent on the initialization.

⁹ The details of the binary, categorical, ordinal, symbolic and quantitative features are omitted here, but can be found in [8].

The measures which were tested in [8] essentially fall into three categories : (a) The City Block, $L_{0.8}$, Euclidean, and Max Norm, which are special cases of the L_p metric for $p = 1, 0.8, 2$ and ∞ respectively, (b) The Hausdorff Norm, and its variants, which involve Max-Min computations, and (c) Traditional pattern recognition norms such as the Template matching and Correlation Norms. The details of the other measures such as the *Median* and *Cosine*, are omitted here in the interest of compactness, but can be found in [6].

2.4 State-of-the-Art DBC Optimization

Based on the above, in all brevity, we state that the state-of-the-art strategy applicable for optimizing DBCs involves the following steps:

1. Select the representative set, Y , from the training set T by resorting to one of the methods given in Section 2.2
2. Compute the dissimilarity matrix $D_{T,Y}[\cdot, \cdot]$, using Eq. (1), in which each individual dissimilarity is computed using one of the measures described in Section 2.
3. For a testing sample \mathbf{z} , compute a dissimilarity column vector, $\delta_Y(\mathbf{z})$, by using the same measure used in Step 2 above.
4. Achieve the classification based on invoking a classifier built in the dissimilarity space and operating on the dissimilarity vector $\delta_Y(\mathbf{z})$.

2.5 State-of-the-Art Prototype Reduction Schemes

In non-parametric pattern classification which use the Nearest Neighbour (NN) or the k -NN rule, each class is described using a set of sample prototypes, and the class of an unknown vector is decided based on the identity of the closest neighbour(s) which are found among all the prototypes. To reduce the number of training vectors, various PRSs have been reported in the literature - two excellent surveys are found in [9], [10]. Rather than embark on yet another survey of the field, we mention here a *few* representative methods of the “zillions” that have been reported. One of the first of its kind is the Condensed Nearest Neighbour (CNN) rule [11]. The reduced set produced by the CNN, however, customarily includes “interior” samples, which can be completely eliminated, without altering the performance of the resultant classifier. Accordingly, other methods have been proposed successively, such as the Reduced Nearest Neighbour (RNN) rule, the Prototypes for Nearest Neighbour (PNN) classifiers [13], the Selective Nearest Neighbour (SNN) rule, two modifications of the CNN [17] Neighbour (ENN) rule and the non-parametric data reduction method [12]. Besides these, the Vector Quantization (VQ) and the Bootstrap [17] techniques and Support Vector Machines (SVM) [15] have also been reported as being extremely effective approaches to data reduction.

In selecting prototypes, vectors near the boundaries between the classes have to be considered to be more significant, and the created prototypes need to be adjusted towards the classification boundaries so as to yield a higher performance. Based on this philosophy, we recently proposed a new hybrid approach

that involved two distinct phases, namely, selecting and adjusting [18], [19]. To overcome the computational burden for “large” datasets, we also proposed a recursive PRS mechanism in [20]. In [20], the data set is sub-divided recursively into smaller subsets to filter out the “useless” internal points. Subsequently, a conventional PRS processes the smaller subsets of data points that effectively sample the entire space to yield *subsets* of prototypes – one set of prototypes for each subset. The prototypes, which result from each subset, are then coalesced, and processed again by the PRS to yield more refined prototypes. In this manner, prototypes which are in the interior of the Voronoi boundaries, and are thus ineffective in the classification, are eliminated at the subsequent invocations of the PRS. As a result, the processing time of the PRS is *significantly* reduced.

Changing now the emphasis, we observe that with regard to designing classifiers, PRS can be employed as a pre-processing module to reduce the data set into a smaller representative subset, and have thus been reported to optimize the design of KNS classifiers in [21], [22]. The details of these are omitted here as they are irrelevant.

3 Proposed Optimization of DBC’s

We have already seen (in Section 2) that the two fundamental avenues by which DBCs can be optimized involve those of reducing the size of T^{10} , and determining a suitable dissimilarity measure. The drawbacks which the reported methods have, are the following:

1. The reported methods reduce the set T (to design Y) by merely *selecting* elements from the former.
2. When the reported methods compute the dissimilarity measure between two vectors, they ignore the second-order properties of the data.

With regard to reducing the size of the representative points, rather than deciding to discard or retain the training points, we permit the user the choice of either *selecting* some of the training samples using methods such as the CNN, or *creating* a smaller set of samples using the methods such as those advocated in the PNN, VQ, and HYB. This reduced set effectively serves as a new “representative” set for the dissimilarity representation. Additionally, we also permit the user to migrate the resultant set by an LVQ3-type method to further enhance the quality of the reduced samples. To investigate the computational advantage gained by resorting to such a PRS preprocessing phase, we observe, first of all, that the number of the reduced prototypes is *fractional* compared to that of the conventional ones, namely those obtained by random selection or clustering-based operations. Once the reduced prototypes are obtained, the dissimilarity matrix computation is significantly smaller since the computation is now done for a much smaller set, i.e., for an $n \times m$ matrix, *versus* an $n \times n$ one.

¹⁰ In general, increasing the cardinality of the representative subset, drastically improves the average classification accuracy of the resultant DBC.

We propose to enhance DBC's by modifying each of the above as follows:

1. Reduce the set T (to design Y) by invoking a PRS on the former. The advantages of doing this (over the methods given in Section 2.2) are:
 - (a) A PRS permits us to obtain Y by either *selecting* the representative set or *creating* it.
 - (b) By choosing an appropriate PRS, we are able to obtain the representative samples of each class Y_i by also including the information in the other Y_j 's ($j \neq i$). This is especially true of the classes of PRSs which also invoke *LVQ3*-type perturbations on the representative points.
 - (c) Most PRSs are designed with the specific task of determining the representative subset of points so as to maximize the class-discriminating properties. But incorporating such information in the subset used for DBCs, we believe that the resultant optimized DBC will be superior to the corresponding DBC which excludes this information. This is, indeed, our experience.
2. Compute the dissimilarity measure between two vectors using the Mahalanobis distance, where the estimated covariance matrix of each class is obtained by using the training samples of that class in T . The advantages of doing this (over the methods given in Section 2.3) are:
 - (a) Defining a well-discriminating dissimilarity measure for a non-trivial learning problem has always been known to be difficult. Indeed, designing such a measure in a DBC is equivalent to defining good *features* in a traditional feature-based classification problem. If a good measure is found and the training set T is representative, the authors of [8] report that the performance of the DBC can be enhanced.
 - (b) The dissimilarity measures used in [8] (referred to in Section 2.3) do not utilize the actual spread of the data in the feature space. It is well known in the field that computing distances and achieving classification using the available covariance information can lead to results superior to those obtained by ignoring this information. We intend to take advantage of this.
 - (c) Although the reduced set of points Y is used in the DBC, the information concerning the spread of the original data points is contained in the sets $\{T_i\}$. Thus, we believe that we can take advantage of this information by using the DBC obtained by the subset of points in $\{Y_i\}$, but by simultaneously incorporating the variance information contained in the original sets, the T_i 's.
 - (d) The basic premise for the DBC methodology is that since it does not operate on the class-conditional distributions, the accuracy can exceed the Bayes' errors bound. However, in attempting to achieve this, the state-of-the-art DBC methods ignore the information contained in the class-conditional distributions. Since this information can be summarized in the moments, we intend to use the second-order moments to optimize the design of DBCs. This is done, in our present scheme, by incorporating it in the computation of the Mahalanobis distances.

- (e) Recently, Horikawa [7] experimented on the properties of NN classifiers for high-dimensional patterns in DBCs. From the experimental results reported, the author demonstrated that the performance of the NN classifiers constructed with DBCs increases with the dimensionality of the pattern when the categorical pattern distributions are different from each other. This is, indeed, what we want to take advantage of incorporating *this* distinct information *via* the Mahalanobis distance computations.

Based on the above, our proposed strategy applicable for optimizing DBCs involves the following steps:

1. From each T_i compute the estimate of Σ_i , the covariance matrix of the class conditional density.
2. Select the representative set Y from the training set T by resorting to one of the PRS methods given in Section 2.5.
3. Compute the dissimilarity matrix $D_{T,Y}[\cdot, \cdot]$, using Eq. (1), in which each individual dissimilarity is computed as the Mahalanobis distance evaluated using the value for Σ_i estimated in Step 2 above.
4. For a testing sample \mathbf{z} , compute a dissimilarity column vector, $\delta_Y(\mathbf{z})$, by using the Mahalanobis distance used in Step 3 above.
5. Achieve the classification based on invoking a classifier built in the dissimilarity space and operating on *this* dissimilarity vector, $\delta_Y(\mathbf{z})$.

4 Experimental Results : Artificial/Real-Life Data Sets

Experimental Data: The proposed method has been tested and compared with the conventional ones. This was done by performing experiments on a number of data sets. The sample vectors of each data set are divided into two subsets of equal size, and used for training and validation, alternately. The training set was used for computing the prototypes and the respective covariance matrices, and the test set was used for evaluating the quality of the corresponding classifier.

In our experiments, the three artificial data sets “Random”, “Non_normal 2”, and “Non_linear 2” were generated with different sizes of testing and training sets of cardinality 400, 1,000, and 1,000 respectively. The data set described as “Random” is generated randomly with a uniform distribution, but with irregular decision boundaries. In this case, the points are generated uniformly, and the assignment of the points to the respective classes is achieved by *artificially* assigning them to the region they fall into, as per the manually created “irregular decision boundary”.

The data set named “Non_normal2”, which has also been employed as a benchmark experimental data set [1] for numerous experimental set-ups was generated from a mixture of four 8-dimensional Gaussian distributions.

The data set named “Non_linear2”, which has a strong non-linearity at its boundary, was generated artificially from a mixture of four variables as follows: $p_1(x) = \{x_1, \frac{1}{2}x_1^2 + y_1\}$, $p_2(x) = \{x_2, -\frac{1}{2}x_2^2 + y_2\}$, where x_1, x_2, y_1, y_2 are normal

random variables whose means and variances are (0, 10), (10, 5), (3, 10) and (20, 5), respectively. The total number of vectors per class is 500.

On the other hand, the data sets “Iris2”, “Tonosphere” (in short, “Tono”), “Sonar”, “Arrhythmia” (in short, “Arrhy”) and “Adult4”, which are real benchmark data sets, are cited from the UCI Machine Learning Repository¹¹. Their details can be found in the latter site, and also in [17].

In the above, all of the vectors were normalized to be within the range $[-1, 1]$ using their standard deviations, and the data set for class j was randomly split into two subsets, $T_{j,t}$ and $T_{j,v}$, of equal size. One of them was used for choosing the initial prototypes and training the classifiers, and the other one was used in their validation (or testing). Later, the role of these sets were interchanged.

Experimental Parameters: As in all algorithms, choosing the parameters¹² of the PRS and the conventional prototype selection schemes play an important role in determining the quality of the solution. The parameters for the reported conventional schemes such as the RAND, RAND_C, and KCentres (the methods referred to as *Random*, *RandomC* and *KCentres* in Section 2.2 respectively) and the PRS-based schemes such as the CNN, PNN, and HYB, are summarized as:

1. Parameters for the RAND, RAND_C, and KCentres : In RAND, a total of 10 % of the samples were randomly selected from the original training data set. In RAND_C, for each class, 10 % of the samples were randomly selected as prototypes. In KCentres, initially, 10 % of the samples (for each class) were arbitrarily chosen as the initial cluster centers, after which a k -means clustering algorithm was invoked, as explained earlier.

2. Parameters for the CNN and the PNN : None.

3. Parameters for the HYB : The initial code book size was determined by the SVM. In this experiment, we hybridized the SVM and an LVQ3-type algorithm. The parameters for the LVQ3 learning, such as the α , the ϵ , the window length, w , and the iteration length, η , were specified as described in [18].

Selecting Prototype Vectors: In order to evaluate the proposed classification mechanisms, we first selected the prototype vectors from the experimental data sets using the CNN, the PNN and the HYB algorithms. In the HYB, we selected initial prototypes using a SVM algorithm. After this selection, we invoked a phase in which the optimal positions (i.e., with regard to classification) were learned with an LVQ3-type scheme. For the SVM and LVQ3 programs, we utilized publicly-available software packages¹³. Table 1 shows a comparison of the number of prototype vectors extracted from the artificial and real-life data sets using the CNN, PNN, and HYB methods.

¹¹ <http://www.ics.uci.edu/mllearn/MLRepository.html>

¹² The same parameters were used for both the artificial and real-life data sets.

¹³ These packages can be available from: http://www-ai.cs.uni-dortmund.de/SOFTWARE/SVM_LIGHT/svm_light.eng.html and http://cochlea.hut.fi/research/som_lvq-pak.shtml, respectively.

Table 1. The number of prototype vectors extracted from experimental data sets using the CNN, PNN, and HYB methods. The two values for each data set are the numbers of prototype vectors obtained from the training and test subsets, respectively.

| Dataset Types | Dataset Names | Whole Dataset ($n1, n2$) | Selected Prototypes ($m1, m2$) | | |
|-----------------|---------------|----------------------------|----------------------------------|----------|----------|
| | | | CNN | PNN | HYB |
| Artificial Data | Random | 200, 200 | 36, 30 | 30, 25 | 18, 15 |
| | Non_normal2 | 500, 500 | 64, 66 | 56, 380 | 63, 57 |
| | Non_linear2 | 500, 500 | 96, 109 | 87, 90 | 82, 58 |
| Real-life Data | Iris2 | 50, 50 | 15, 12 | 10, 7 | 6, 8 |
| | Ionosphere | 176, 176 | 51, 42 | 37, 33 | 44, 46 |
| | Sonar | 104, 104 | 52, 53 | 34, 33 | 53, 59 |
| | Arrhythmia | 226, 226 | 32, 28 | 8, 7 | 65, 69 |
| | Adult4 | 4168, 4168 | 755, 752 | 659, 658 | 430, 448 |

From Table 1, for example, we see that the numbers of selected prototype vectors of the ‘‘Random’’ dataset, $(m1, m2)$, are $(36, 30)$, $(30, 25)$ and $(18, 15)$, respectively. Each of them is considerably smaller than the size of the original data set. Using the selected vectors as a representative of the training data set, we can significantly reduce the dimensionality of the dataset (and the consequential computations) without degrading the performance. Once the reduced prototype set $Y = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_m\}$, is obtained, the dimensionality (and the classification processing time) of the matrix $D_{T,Y}[\cdot, \cdot]$ can be reduced into $n \times m$, where the dimensionality of the column vector is m , not n (e.g., 15 not 200).

Experimental Results: We report below the run-time characteristics of the proposed algorithm for the artificial and real-life data sets as shown in Table 2, where the ‘‘Wholeset’’ approach represents the experimental results for the entire original data sets, that is, $D_{T,T}[\cdot, \cdot]$, without employing any selection method. On the other hand, the results of the RAND, RAND_C, and KCentres, and the CNN, PNN, and HYB are obtained by calculating the dissimilarity matrix, $D_{T,Y}[\cdot, \cdot]$, using the representatives obtained with each respective method. Also, each result is the averaged one for the training and the test sets, respectively.

Table 2 shows the DBC accuracy rates (%) of the classifiers designed with the conventional prototype selection schemes, such as the RAND, RAND_C, and KCentres methods, on the artificial and real-life data sets, in which the dissimilarity measure used is the Euclidean distance. It also presents the optimized DBC accuracy rates (%) on the same data sets, in which the prototype selection schemes are the PRS-based ones such as the CNN, PNN, and HYB methods, which, as explained earlier, use the Mahalanobis distance.

A Comparison of Conventional DBCs and PRS-based Schemes: First of all, it is worth mentioning that the data set ‘‘Adult4’’ possesses a noticeable class imbalance¹⁴. Thus, although the number of prototypes obtained using the HYB scheme are 430 and 448 for the training and test sets respectively, the number

¹⁴ For example, the sample points of two classes ω_1 and ω_2 of its training set are 3961 and 206, respectively.

Table 2. The accuracy rates (%) of the DBCs for the artificial and real-life data sets. The results reported concern the schemes designed with the conventional prototype selection methods such as the RAND, RAND_C, and KCentres methods which use the Euclidean distance, and the optimized ones which use PRS-based schemes such as the CNN, PNN, and HYB methods and the Mahalanobis distance. The other notation is discussed in the text.

| Dataset Types | Dataset Names | Conventional Schemes | | | | Proposed Schemes | | |
|---------------|---------------|----------------------|-------|--------|----------|------------------|-------|-------|
| | | Wholeset | RAND | RAND_C | KCentres | CNN | PNN | HYB |
| Artificial | Random | 83.75 | 83.98 | 82.25 | 83.50 | 84.00 | 84.50 | 84.25 |
| | Non_normal2 | 95.10 | 95.09 | 95.05 | 95.40 | 89.50 | 91.00 | 90.10 |
| | Non_linear2 | 59.50 | 59.80 | 60.23 | 60.00 | 74.90 | 76.40 | 76.30 |
| Real-life | Iris2 | 77.00 | 76.90 | 71.00 | 83.00 | 90.00 | 88.00 | 88.00 |
| | Ionosphere | 71.31 | 71.70 | 69.89 | 69.32 | 86.08 | 86.08 | 86.08 |
| | Sonar | 60.10 | 58.61 | 56.11 | 55.29 | 70.19 | 69.23 | 69.71 |
| | Arrhythmia | 92.92 | 87.23 | 85.40 | 82.31 | 92.92 | 92.92 | 95.13 |
| | Adult4 | 72.08 | - | 71.98 | 71.57 | - | - | - |

of prototypes in the second class is only about 10% of the number of prototypes in the first. Since this precludes a meaningful comparison, we shall omit it in further discussions. Suffice it to mention that the accuracy of the conventional DBC (71.98%) is quite comparable to the accuracy for the ‘Wholeset’ (72.08%).

The overall remark that we can make from Table 2 is that, for every data set, the accuracy of the conventionally optimized DBC is quite comparable to (and sometimes even more accurate than) the accuracy when the ‘Wholeset’ is utilized. This is true for both the artificial and real-life data sets. The reason for this increased accuracy is that when a k -NN scheme is used in the dissimilarity space, the effects of the outliers become much more prominent when the ‘Wholeset’ is utilized. Thus, for the set “Iris2”, the accuracy for the DBC using the ‘Wholeset’ is 77%, and this increases to 83% if the KCentres method is used.

With regard to comparing the conventional and the new schemes, we again refer the reader to Table 2. Generally speaking, we note that if we consider the *average* accuracy obtained by using any of the conventional prototypes selection schemes (namely, RAND, RAND_C, KCentres), and compare them with the *average* accuracy obtained by using any of the PRSs (namely, CNN, PNN, HYB), the latter is almost always superior. Thus, to render the comparison more fair, in what follows, we shall consider the *best* accuracy that a conventional scheme yields, and compare it with the *best* accuracy that a PRS would yield.

Consider the artificial data set, “Non_linear2”. In this case, the RAND_C method yields the best accuracy (60.23%) of the three conventional selection methods when the Euclidean distance is used. The corresponding accuracy for the optimized methods is obtained when the PNN is the PRS used, and it yields an accuracy of 76.40%. Similarly, consider the real-life “Arrhythmia” data set. In this case, the RAND method yields the best accuracy (87.23%) of the three conventional selection methods when the Euclidean distance is used. The corresponding accuracy for the optimized methods is obtained when the HYB is the PRS used, leading to an accuracy of 95.13%. The conclusion that we can

make is that the optimized methods (the PRSs used in conjunction with the Mahalanobis distance) are almost always superior (and sometimes, much more superior) to the conventional schemes.

It is also interesting to note how a conventional selection scheme (such as the RAND, RAND_C, KCentres) would perform if the Mahalanobis distance is used. The details of the results are omitted here in the interest of compactness, but can be found in [17]. Here too, if the average accuracy of the schemes (RAND, RAND_C, KCentres) is compared to the average accuracy of the PRSs (CNN, PNN and HYB), the latter is almost always the better option. The optimized methods continue to be the superior ones if the best of the method is chosen in each case, although the advantage is not so marked.

The general conclusion that we can make from the results is the following: It is always advantageous to use a PRS to select the subset of representative points, but when a PRS is used, *one must not resort to using the Euclidean distance to compute the dissimilarities*. Rather, since the PRS implicitly takes the data distribution into consideration, it must be used in conjunction with a distribution-based dissimilarity measure, like the Mahalanobis distance.

From the above considerations, it is also worth mentioning that it is not so easy to crown any one scheme to be superior to the others in the context of the PRS method used. But a general observation seems to be that for artificial data the PNN is the most advantageous, and the HYB seems to yield the best results for the real-life data sets. From the other results given in [17], we also see that the processing CPU-times can also be reduced significantly by employing a PRS such as the CNN, PNN, and HYB without sacrificing the accuracy so much.

5 Conclusions

In this paper, we have suggested a novel strategy to enhance the computation for all families of Dissimilarity-Based Classifiers (DBC). Rather than compute, store and process the DBC based on the entire data set, we advocate that the training set be first reduced into a smaller representative subset obtained by invoking a Prototype Reduction Scheme (PRS), whose output yields the points to be utilized by the DBC. Apart from utilizing PRSs, in the paper we have also proposed simultaneously employing the Mahalanobis distance as the dissimilarity-measurement criterion to increase the DBC's classification accuracy. Our experimental results demonstrate that the proposed mechanism increases the classification accuracy when compared with the "conventional" approaches for samples involving real-life as well as artificial data sets.

References

1. K. Fukunaga, *Introduction to Statistical Pattern Recognition, Second Edition*, Academic Press, San Diego, 1990.
2. A. K. Jain, R. P. W. Duin and J. Mao, "Statistical pattern recognition: A review", *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. PAMI-22, no. 1, pp. 4 - 37, Jan. 2000.

3. R. P. W. Duin, D. Ridder and D. M. J. Tax, "Experiments with a featureless approach to pattern recognition", *Pattern Recognition Letters*, vol. 18, pp. 1159 - 1166, 1997.
4. R. P. W. Duin, E. Pekalska and D. de Ridder, "Relational discriminant analysis", *Pattern Recognition Letters*, vol. 20, pp. 1175 - 1181, 1999.
5. E. Pekalska and R. P. W. Duin, "Dissimilarity representations allow for building good classifiers", *Pattern Recognition Letters*, vol. 23, pp. 943 - 956, 2002.
6. E. Pekalska, *Dissimilarity representations in pattern recognition. Concepts, theory and applications*, Ph.D. thesis, Delft University of Technology, Delft, The Netherlands, 2005.
7. Y. Horikawa, "On properties of nearest neighbor classifiers for high-dimensional patterns in dissimilarity-based classification", *IEICE Trans. Information & Systems*, vol. J88-D-II, no. 4, pp. 813 - 817, Apr. 2005, in Japanese.
8. E. Pekalska, R. P. W. Duin, and P. Paclik, "Prototype selection for dissimilarity-based classifiers", *Pattern Recognition*, vol. 39, pp. 189 - 208, 2006.
9. J. C. Bezdek and L. I. Kuncheva, "Nearest prototype classifier designs: An experimental study", *International Journal of Intelligent Systems*, vol. 16, no. 12, pp. 1445 - 1473, 2001.
10. B. V. Dasarathy, *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*, IEEE Computer Society Press, Los Alamitos, 1991.
11. P. E. Hart, "The condensed nearest neighbor rule", *IEEE Trans. Inform. Theory*, vol. IT-14, pp. 515 - 516, May 1968.
12. K. Fukunaga and J. M. Mantock, "Nonparametric data reduction", *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. PAMI-6, no. 1, pp. 115 - 118, Jan. 1984.
13. C. L. Chang, "Finding prototypes for nearest neighbor classifiers", *IEEE Trans. Computers*, vol. C-23, no. 11, pp. 1179 - 1184, Nov. 1974.
14. J. C. Bezdek, T. R. Reichherzer, G. S. Lim, and Y. Attikiouzel, "Multiple-prototype classifier design", *IEEE Trans. Systems, Man, and Cybernetics - Part C*, vol. SMC-28, no. 1, pp. 67 - 79, Feb. 1998.
15. C. J. C. Burges, "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121 - 167, 1998.
16. T. Kohonen, *Self-Organizing Maps*, Berlin, Springer - Verlag, 1995.
17. S.-W. Kim and B. J. Oommen, "On Optimizing Dissimilarity-based Classification Using Prototype Reduction Schemes". *Unabridged version of this paper*.
18. S. -W. Kim and B. J. Oommen, "Enhancing prototype reduction schemes with LVQ3-type algorithms", *Pattern Recognition*, vol. 36, no. 5, pp. 1083 - 1093, 2003.
19. S. -W. Kim and B. J. Oommen, "A Brief Taxonomy and Ranking of Creative Prototype Reduction Schemes", *Pattern Analysis and Applications Journal*, vol. 6, no. 3, pp. 232 - 244, December 2003.
20. S. -W. Kim and B. J. Oommen, "Enhancing Prototype Reduction Schemes with Recursion : A Method Applicable for "Large" Data Sets", *IEEE Trans. Systems, Man, and Cybernetics - Part B*, vol. SMC-34, no. 3, pp. 1384 - 1397, June 2004.
21. S. -W. Kim and B. J. Oommen, "On using prototype reduction schemes to optimize kernel-based nonlinear subspace methods", *Pattern Recognition*, vol. 37, no. 2, pp. 227 - 239, 2004.
22. S. -W. Kim and B. J. Oommen, "On using prototype reduction schemes and classifier fusion strategies to optimize kernel-based nonlinear subspace methods", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 455 - 460, March 2005.

General Adaptive Neighborhood Image Restoration, Enhancement and Segmentation

Johan Debayle, Yann Gavet, and Jean-Charles Pinoli

Ecole Nationale Supérieure des Mines de Saint-Etienne
Centre Ingénierie et Santé (CIS), Laboratoire LPMG, UMR CNRS 5148
42023 Saint-Etienne Cedex 2, France

Abstract. This paper aims to outline the General Adaptive Neighborhood Image Processing (GANIP) approach [1–3], which has been recently introduced. An intensity image is represented with a set of local neighborhoods defined for each point of the image to be studied. These so-called General Adaptive Neighborhoods (GANs) are simultaneously adaptive with the spatial structures, the analyzing scales and the physical settings of the image to be addressed and/or the human visual system. After a brief theoretical introductory survey, the GANIP approach will be successfully applied on real application examples in image restoration, enhancement and segmentation.

1 The General Adaptive Neighborhood (GAN) Paradigm

This paper deals with 2D intensity images, that is to say image mappings defined on a spatial support D in the Euclidean space \mathbb{R}^2 and valued into a gray tone range, which is a real numbers interval. The General Adaptive Neighborhood paradigm has been introduced in order to propose an original image representation for adaptive processing and analysis. The central idea is the notion of adaptivity which is simultaneously associated to the analyzing scales, the spatial structures and the intensity values of the image to be addressed.

1.1 Adaptivity with Analyzing Scales

A multiscale image representation such as wavelet decomposition [4] or isotropic scale-space [5], generally takes into account analyzing scales which are global and a priori defined, that is to say *extrinsic* scales. This kind of multiscale analysis presents a main drawback since a priori knowledge, relating to the features of the studied image, is consequently required. On the contrary, an *intrinsic* multiscale representation such as anisotropic scale-space [6], takes advantage of scales which are self-determined by the local image structures. Such a decomposition does not need any a priori information.

1.2 Adaptivity with Spatial Structures

The image processing techniques using *spatially invariant* transformations, with fixed operational windows, give efficient and compact computing structures, in

the sense where data and operators are independent. Nevertheless, they consequently have several drawbacks such as creating artificial patterns, changing the detailed parts of large objects, damaging transitions or removing significant details. Alternative approaches towards context depend processing have been proposed [7]. A *spatially adaptive* image processing implies that operators are no longer spatially invariant, but must vary over the whole image with adaptive windows, taking locally into account the image context.

1.3 Adaptivity with Intensity Values

In order to develop powerful image processing operators, it is necessary to represent intensity images within mathematical frameworks (most of the time of a vectorial nature) based on a *physically and/or psychophysically relevant image formation process*. In addition, their mathematical structures and operations (the vector addition and then the scalar multiplication) have to be consistent with the physical nature of the images and/or the human visual system, and computationally effective. Thus, although the *Classical Linear Image Processing Framework* (CLIP), i.e. with the usual vectorial operations, has played a central role in image processing, it is not necessarily the best choice. Indeed, it was shown [8] that the usual addition is not a satisfactory solution in some non-linear physical settings, such as that based on multiplicative or convolutive image formation model. The reasons are that the classical addition operation and consequently the usual scalar multiplication are not consistent with the combination and amplification laws to which such physical settings obey. However, using the power of abstract linear algebra, it is possible to go up to the abstract level and to explore *General Linear Image Processing* (GLIP) frameworks [9, 10], in order to include situations in which signals or images are combined by processes other than the usual vector addition. Consequently, operators based on such *intensity-based image processing frameworks* should be consistent with the physical and/or physiological settings of the images to be processed. For instance, the *Logarithmic Image Processing* (LIP) framework of intensity images (f, g, \dots) has been introduced [11, 12] with its vector addition \triangle , its vector subtraction \triangleleft and its scalar multiplication \triangleleft defined respectively as following:

$$f \triangle g = f + g - \frac{fg}{M} \quad (1)$$

$$f \triangleleft g = M \left(\frac{f - g}{M - g} \right) \quad (2)$$

$$\alpha \triangleleft f = M - M \left(1 - \frac{f}{M} \right)^\alpha, \quad \alpha \in \mathbb{R} \quad (3)$$

where $M \in \mathbb{R}$ denotes the upper bound of the range where intensity images are digitized and stored.

The LIP framework has been proved to be consistent with the transmittance image formation model, the multiplicative reflectance image formation model, the multiplicative transmittance image formation model, and with several laws and characteristics of human brightness perception [10, 13].

2 GANs Sets

In the so-called General Adaptive Neighborhood Image Processing (GANIP) approach [1–3], a set of General Adaptive Neighborhoods (GANs set) is identified about each point in the image to be analyzed. A GAN is a subset of the spatial support constituted by connected points whose measurement values, in relation to a selected criterion (such as luminance, contrast, thickness, . . .), fit within a specified homogeneity tolerance. These GANs are used as adaptive windows for image transformations or quantitative image analysis.

The space of *image* (resp. *criterion*) *mappings*, defined on the spatial support D and valued in a real numbers interval \bar{E} (resp. E), is represented in a GLIP framework, denoted \mathcal{I} (resp. \mathcal{C}). The GLIP framework \mathcal{I} (resp. \mathcal{C}) is then supplied with an ordered vectorial structure, using the formal vector addition \oplus (resp. \oplus), the formal scalar multiplication \otimes (resp. \otimes) and the classical total order relation \geq defined directly from those of real numbers:

$$\forall (f, g) \in \mathcal{I}^2 \text{ or } \mathcal{C}^2 \quad f \geq g \Leftrightarrow (\forall x \in D \quad f(x) \geq g(x)) \quad (4)$$

There exists several GANs sets, whose each collection satisfies specific properties [1]. This paper presents the most elementary kind of these ones, denoted $V_{m_{\square}}^h(x)$. For each point $x \in D$ and for an image $f \in \mathcal{I}$, the GANs $V_{m_{\square}}^h(x)$ are included as subsets in D . They are built upon a *criterion mapping* $h \in \mathcal{C}$ (based on a local measurement such as luminance, contrast, thickness, . . . related to f), in relation with an *homogeneity tolerance* m_{\square} belonging to the positive intensity value range E^{\oplus} . More precisely, $V_{m_{\square}}^h(x)$ is a subset of D which fulfills two conditions :

1. its points have a measurement value close to that of the point $x : \forall y \in V_{m_{\square}}^h(x) \quad |h(y) \ominus h(x)|_{\square} \leq m_{\square}$, where \ominus and $|\cdot|_{\square}$ denote the considered GLIP subtraction and GLIP modulus, respectively,
2. the set is path-connected (with the usual Euclidean topology on $D \subseteq \mathbb{R}^2$).

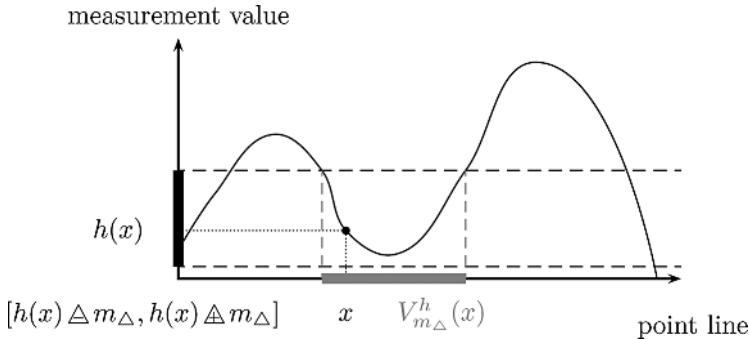


Fig. 1. One-dimensional computation of an adaptive neighborhood set $V_{m_{\Delta}}^h(x)$ in the LIP framework. For a point x , a tube of tolerance m_{\square} is first computed around $h(x)$. Secondly, the inverse map of this interval gives a subset of the 1-D spatial support. Finally, the path-connected component holding x provides the GAN $V_{m_{\Delta}}^h(x)$.

The GANs are thus defined as following:

$$\forall (m_{\square}, h, x) \in E^+ \times \mathcal{C} \times D \quad V_{m_{\square}}^h(x) = C_{h^{-1}([h(x) \ominus m_{\square}, h(x) \oplus m_{\square}])}(x) \quad (5)$$

where $C_X(x)$ denotes the path-connected component (with the usual Euclidean topology on $D \subseteq \mathbb{R}^2$) of $X \subseteq D$ containing $x \in D$.

Figure 1 gives a visual impression, on a 1-D example, of the computation of a GAN in the LIP framework (i.e. with the \triangle vector addition (1) and the \triangle vector subtraction (2)).

3 GAN Mean and Rank Filtering

Usual image to image transformations generally work on fixed-size and fixed-shape operational windows, either they are convolution filters (mean, ...) or rank operators (min, max, median, ...). This kind of operators removes thin details and displaces contours. In the GANIP approach, adaptive filters are introduced in substituting the usual disks $B_r(\cdot)$ of radius r as isotropic operational windows by the anisotropic GANs $V_{m_{\square}}^h(\cdot)$ (Fig. 2).

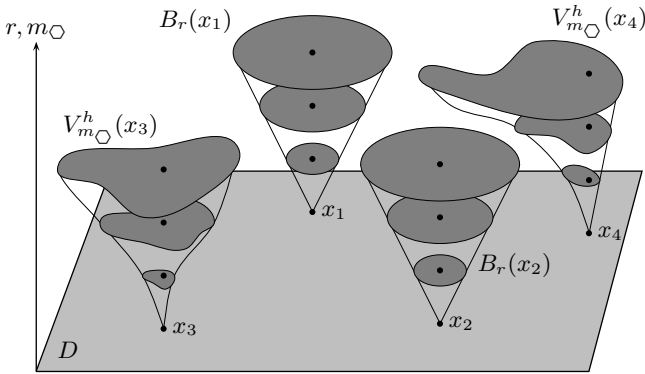


Fig. 2. Example of adaptive $V_{m_{\square}}^h(\cdot)$ and non-adaptive $B_r(\cdot)$ operational windows with three values both for the homogeneity tolerance parameter m_{\square} , and for the disks radius r . The shape of $B_r(x_1)$ and $B_r(x_2)$ are identical and $\{B_r(x)\}_r$ is a family of homothetic sets for each point $x \in D$. On the contrary, the shape of $V_{m_{\square}}^h(x_3)$ and $V_{m_{\square}}^h(x_4)$ are dissimilar and $\{V_{m_{\square}}^h(x)\}_m$ is not a family of homothetic sets.

The resulting GANIP-based operators perform consistent image processing, such as adaptive mean, median or morphological filters which are introduced in the following. Mean and rank filtering are simple, intuitive and easy to implement methods for smoothing images, i.e. reducing the amount of intensity variation between one pixel and the next. They are often used to reduce noise effects in images [14].

The idea of mean filtering consists in replacing the gray tone of every point in an image with the mean ('average') gray tone of its neighbors, including itself. This has the effect of eliminating point values which are unrepresentative of their

surroundings. Mean filtering is usually thought of as a convolution filter. Like The idea of mean filtering consists in replacing the gray tone of every point in an image with the mean ('average') gray tone of its neighbors, including itself. This has the effect of eliminating point values which are unrepresentative of their surroundings. Mean filtering is usually thought of as a convolution filter. Like other convolutions it is based around a kernel, which represents the shape and size of the neighborhood to be sampled when calculating the mean. Often an isotropic kernel is used, as a disk of radius 1, although larger kernels (e.g. disk of radius 2) can be used for more severe smoothing. (Note that a small kernel can be applied more than once in order to produce a similar - but not identical - effect as a single pass with a large kernel).

Rank filters in image processing sort (rank) the gray tones in some neighborhood of every point in ascending order, and replace the seed point by some value k in the sorted list of gray tones. When performing the well-known median filtering [14], each point to be processed is determined by the median value of all points in the selected neighborhood. The median value k of a population (set of points in a neighborhood) is that value for which half of the population has smaller values than k , and the other half has larger values than k .

So, the GANs mean and rank filters are introduced by substituting the isotropic neighborhoods, generally used for this kind of filtering, with the (anisotropic) general adaptive neighborhoods (GANs).

4 GAN Mathematical Morphology

Mathematical Morphology (MM) [15] is an important and nowadays a traditional theory in image processing. Its development leads to several image processing tools that are extremely useful in image enhancement, image segmentation and classification, pattern recognition, texture analysis and synthesis. The elementary morphological operators of dilation and erosion use an operational window named *Structuring Element* (SE). Generally, the SEs are extrinsically defined and have consequently fixed shape and size.

4.1 Adaptive Structuring Elements

The basic idea in the General Adaptive Neighborhood Mathematical Morphology (GANMM) is to replace the usual SEs by GANs, providing adaptive operators and filters. More precisely the Adaptive Structuring Elements (ASEs), denoted $R_{m_{\square}}^h(x)$, are defined as following:

$$\forall(m_{\square}, h, x) \in E^{\oplus} \times \mathcal{C} \times D \quad R_{m_{\square}}^h(x) = \bigcup_{z \in D} \{V_{m_{\square}}^h(z) | x \in V_{m_{\square}}^h(z)\} \quad (6)$$

The GANs $V_{m_{\square}}^h(x)$ are not directly used as ASEs, because they do not satisfy the symmetry property contrary to the $R_{m_{\square}}^h(x)$: $x \in R_{m_{\square}}^h(y) \Leftrightarrow y \in R_{m_{\square}}^h(x)$. This symmetry condition is relevant for visual, topological, morphological and practical reasons as explained in [3].

4.2 Adaptive Morphological Operators

The elementary dual operators of adaptive dilation and adaptive erosion are defined accordingly to the ASEs:

$$\forall(m_{\square}, h, f) \in E^{\oplus} \times \mathcal{C} \times \mathcal{I}$$

$$D_{m_{\square}}^h(f) : \begin{cases} D \rightarrow \tilde{E} \\ x \mapsto \sup_{w \in R_{m_{\square}}^h(x)} f(w) \end{cases} \quad (7)$$

$$E_{m_{\square}}^h(f) : \begin{cases} D \rightarrow \tilde{E} \\ x \mapsto \inf_{w \in R_{m_{\square}}^h(x)} f(w) \end{cases} \quad (8)$$

Then, several adaptive morphological filters can be defined by combination of these two elementary adaptive morphological operators, in particular:

- adaptive closing: $C_{m_{\square}}^h(f) = E_{m_{\square}}^h \circ D_{m_{\square}}^h(f)$
- adaptive opening: $O_{m_{\square}}^h(f) = D_{m_{\square}}^h \circ E_{m_{\square}}^h(f)$
- adaptive closing-opening: $CO_{m_{\square}}^h(f) = C_{m_{\square}}^h \circ O_{m_{\square}}^h(f)$
- adaptive opening-closing: $CO_{m_{\square}}^h(f) = C_{m_{\square}}^h \circ O_{m_{\square}}^h(f)$

Those resulting GAN morphological operators perform a really spatially-adaptive image processing and notably, in several and important practical cases, are connected [3, 1], which is a great advantage compared to the usual ones that fail to this property.

5 Practical Application Examples

Most of the time, image filtering is a necessary step in image pre-processing, such as restoration, pre-segmentation, enhancement, sharpening, brightness correction, . . . The GAN-based filtering allows such transformations to be defined [16, 2]. Three results are here exposed in image restoration, image enhancement and image segmentation.

5.1 Image Restoration

This section addresses the image restoration area with a concrete application example in visual image denoising. The aim is to suppress noise as much as possible while preserving image features. Figure 3 exposes results of a denoising process applied on the Edouard Manet's painting 'Le Fife'. It is realized both with classical mean filters, denoted Mean_r , using disks of radius r as classical operational windows, and with GAN mean filters, denoted $\text{Mean}_{m_{\Delta}}^f$, using GANs computed with the luminance criterion in the LIP framework.

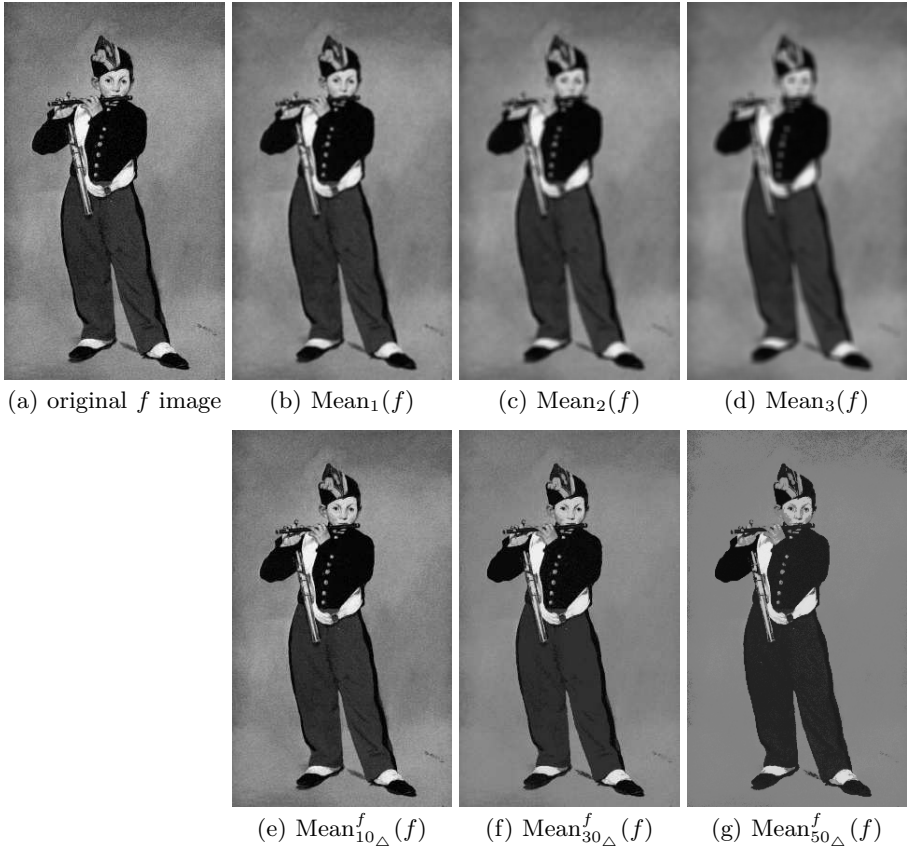


Fig. 3. Image restoration through usual (b-d) and adaptive (b-d) mean filtering applied on the original (a) image. The adaptive filters (connected) do not damage edges contrary to the classical filters which blur the image.

Those adaptive filters using the elementary GANs work well if the processed images are noise free or a bit corrupted .

In the presence of impulse noise, such as salt and pepper noise, the GANs need to be combined so as to provide efficient filtering operators [2]. Indeed, the elementary GAN of a corrupted point by such a noise is generally not representative of the region of which it belongs, for any homogeneity tolerance value m_{\square} .

5.2 Image Enhancement

Image enhancement is the improvement of image quality [14], wanted e.g. for visual inspection or for machine analysis. Physiological experiments have shown that very small changes in luminance are recognized by the human visual system in regions of continuous gray tones, and not at all seen in regions of some discontinuities [17]. Therefore, a design goal for image enhancement is often to smooth images into more uniform regions, while preserving edges. On the other

hand, it has also been shown that somehow degraded images with enhancement of certain features, e.g. edges, can simplify image interpretation both for a human observer and for machine recognition [17]. A second design goal, therefore, is image sharpening [14].

In this paper, the considered image enhancement technique is an edge sharpening process: the approach is similar with unsharp masking [18] type enhancement where a high pass portion is added to the original image. The contrast enhancement process is realized through the *toggle contrast* [19], whose operator κ_r is defined in the following:

$$\forall(f, x, r) \in \mathcal{I} \times D \times \mathbb{R}^+$$

$$\kappa_r(f)(x) = \begin{cases} D_r(f)(x) & \text{if } D_r(f)(x) - f(x) < f(x) - E_r(f)(x) \\ E_r(f)(x) & \text{otherwise} \end{cases} \quad (9)$$

where D_r and E_r denote the classical dilation and erosion, respectively, using a disk of radius r as structuring element.

This (non-adaptive) toggle contrast will be compared with the *adaptive LIP toggle contrast*, using a 'contrast' criterion. This transformation requires a 'contrast' definition which is introduced in the digital setting of the LIP framework [20]: The LIP contrast at a point $x \in D$ of an image $f \in \mathcal{I}$, denoted $C(f)(x)$, is defined with the help of the gray values of its neighbors included in a disk $V(x)$ of radius 1, centered in x :

$$C(f)(x) = \frac{1}{\#V(x)} \triangle \sum_{y \in V(x)} (\max(f(x), f(y)) \triangle \min(f(x), f(y))) \quad (10)$$

where \sum^{\triangle} and $\#$ denote the sum in the LIP sense [20], and the cardinal symbol, respectively.

Consequently, the so-called adaptive toggle LIP contrast is the transformation $\kappa_{m_\Delta}^{C(f)}$, where $C(f)$ and m_Δ represent the criterion mapping and the homogeneity tolerance within the LIP framework (required for the GANs definition), respectively. It is defined as following:

$$\forall(f, x, m_\Delta) \in \mathcal{I} \times D \times E^{\triangle}$$

$$\kappa_{m_\Delta}^{C(f)}(f)(x) = \begin{cases} D_{m_\Delta}^{C(f)}(f)(x) & \text{if } D_{m_\Delta}^{C(f)}(f)(x) - f(x) < f(x) - E_{m_\Delta}^{C(f)}(f)(x) \\ E_{m_\Delta}^{C(f)}(f)(x) & \text{otherwise} \end{cases} \quad (11)$$

where $D_{m_\Delta}^{C(f)}$ and $E_{m_\Delta}^{C(f)}$ denote the adaptive dilation and adaptive erosion, respectively, using ASEs computed on the criterion mapping $C(f)$ with the homogeneity tolerance m_Δ .

Figure 4 illustrates an application example of image enhancement through usual and adaptive toggle contrast, respectively. The process is applied on a real image acquired on the retina of a human eye.

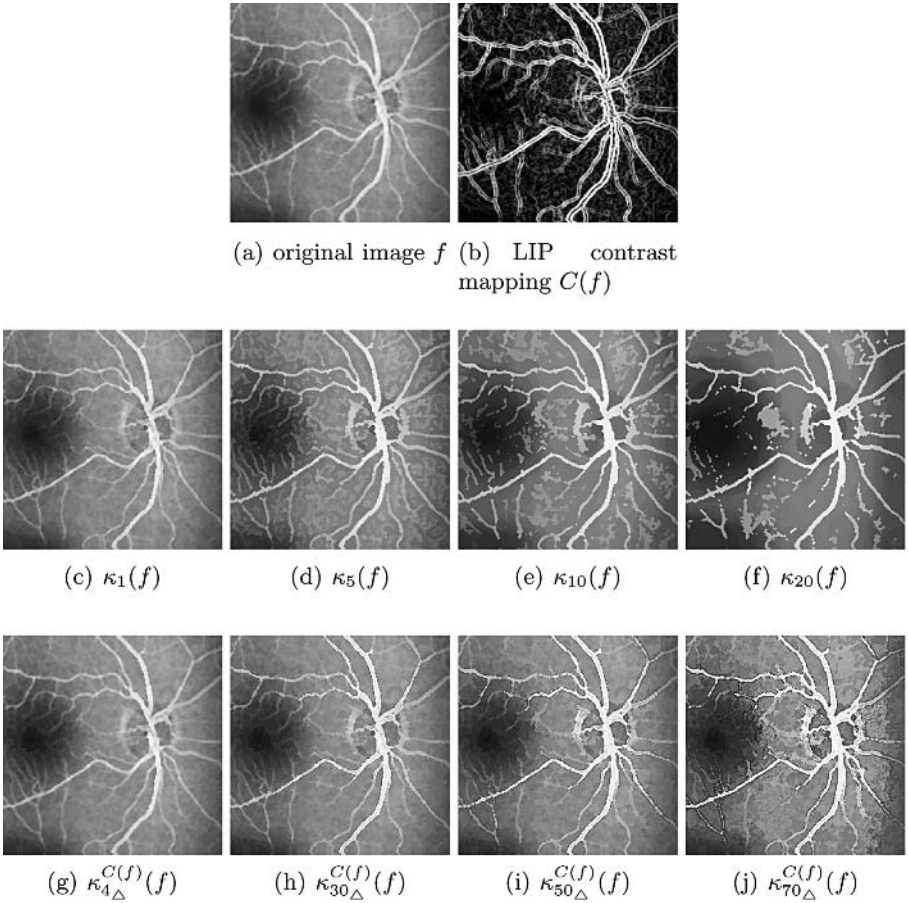


Fig. 4. Image enhancement through the toggle contrast process. The operator is applied on a real (a) image acquired on the retina of a human eye. The enhancement is achieved with the usual toggle contrast (c-f) and the GANIP-based toggle LIP contrast (g-j). Using the usual toggle contrast, the edges are disconnected as soon as the filtering becomes too strong. On the contrary, such structures are preserved and sharpened with the GAN filters.

This image enhancement application example confirms that the GANIP operators are more effective than the corresponding classical ones. Indeed, the adaptive toggle LIP contrast performs a locally accurate image enhancement, taking into account the notion of contrast within spatial structures of the image. Consequently, only the transitions are sharpened while preserving the homogeneous regions. On the contrary, the usual toggle contrast enhances the image in a uniform way. Thus, the spatial zones around transitions are rapidly damaged as soon as the filtering becomes too strong.

5.3 Image Segmentation

The segmentation of an intensity image can be defined as its *partition* (in fact the partition of the spatial support D) into different connected regions, relating to an homogeneity condition [14]. In this paper, the segmentation process is based on a morphological transformation called *watershed* [21] and a GANIP-based decomposition process. It will be illustrated on a human corneal endothelial image provided by the University Hospital Center of Saint-Etienne in France.

The cornea is the transparent surface in the front of the eye. It has a role of protection of the eye, and with the lens, of focusing light into the retina. It is constituted of several layers, such as the epithelium (at the front of the cornea), the stroma and the endothelium (at the back of the cornea). The endothelium contains non-regenerative cells tiled in a monolayer and hexagonal mosaic. This layer pumps water from the cornea, keeping it clear. A high cell density and a regular morphometry of this layer characterize the good quality of a cornea before transplantation, the most common transplantation in the world. Herein lays the importance of the endothelial control. Ex vivo controls are done by optical microscopy on corneal button before grafting. That image acquisition equipment give gray tones images which are segmented, for example by the SAMBATM software [22], into regions representing cells. These ones are used to compute statistics in order to quantify the corneal quality before transplantation.

The authors proposed a GANIP-based approach to segment the cornea cells. The process is achieved by a closing-opening morphological filtering using the GAN sets with the luminance criterion in the CLIP framework, followed by a watershed transformation, denoted W . A comparison with the results provided by the SAMBATM software, whose process is achieved by thresholding, filtering and skeletonization [22], is proposed (Fig. 5). The parameter m_{\square} of the adaptive morphological filter has been tuned to visually provide the best possible segmentation.

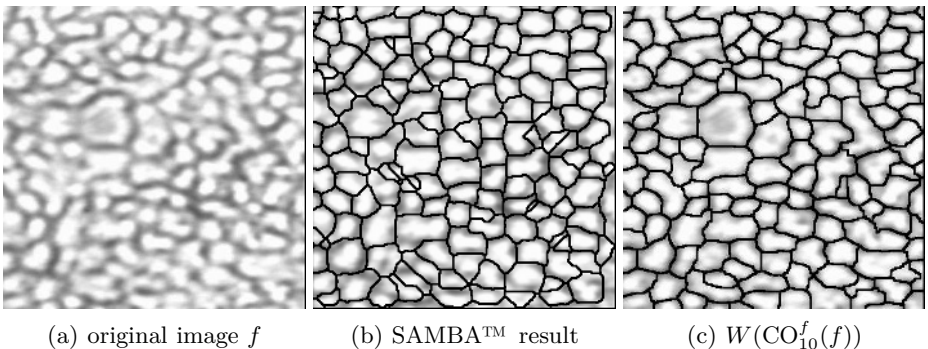


Fig. 5. Segmentation of human endothelial cornea cells (a). The process achieved by the GANIP-based morphological approach (c) provides better results (from the point of view of ophthalmologists) than the SAMBATM software [22] (b).

The detection process achieved by the GANIP-based morphological approach provides better results (from the point of view of ophthalmologists) than the SAMBA™ software. Those results highlight the spatially-variant adaptivity of the GANIP-based operators. A more specific study should be investigated for this promising cells detection process.

6 Conclusion and Prospects

The General Adaptive Neighborhood Image Processing (GANIP) approach allows efficient image processing operators to be built. The GAN-based representation of an image is simultaneously adaptive with its analyzing scales, its spatial structures and its intensity values. In this way, the resulting adaptive operators are spatially variant, relevant from a physical and/or psychophysical point of view and are intrinsically multiscale in the sense where processing scales are locally determined by the image context. These theoretical aspects have been practically confirmed on real application examples in image restoration, enhancement and segmentation. From a practical point of view, the computation of the GANs sets increases the running time of the adaptive operators [3]. Currently, the authors work on GANIP-based topological approaches.

Acknowledgments

The authors wish to thank their colleague P. Gain from the University Hospital Center of Saint-Etienne in France, who has kindly provided the human corneal endothelial images (Fig. 5-a,b).

References

1. Debayle, J., Pinoli, J.C.: General Adaptive Neighborhood Image Processing - Part I: Introduction and Theoretical Aspects. (Journal of Mathematical Imaging and Vision) paper accepted on January 26, 2006.
2. Debayle, J., Pinoli, J.C.: General Adaptive Neighborhood Image Processing - Part II: Practical Application Examples. (Journal of Mathematical Imaging and Vision) paper accepted on January 26, 2006.
3. Debayle, J., Pinoli, J.C.: Spatially Adaptive Morphological Image Filtering using Intrinsic Structuring Elements. *Image Analysis and Stereology* **24**(3) (2005) 145–158
4. Mallat, S.G.: A Theory for Multiresolution Decomposition : The Wavelet Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11** (1989) 674–693
5. Lindeberg, T.: Scale-Space Theory: a Basic Tool for Analysing Structures at Different Scales. *Journal of Applied Statistics* **21**(2) (1994) 225–270
6. Perona, P., Malik, J.: Scale-Space and Edge Detection using Anisotropic Diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **12**(7) (1990) 629–639
7. Gordon, R., Rangayyan, R.M.: Feature Enhancement of Mammograms using Fixed and Adaptive Neighborhoods. *Applied Optics* **23**(4) (1984) 560–564

8. Rosenfeld, A.: *Picture Processing by Computers*. Academic Press, New-York, U.S.A. (1969)
9. Oppenheim, A.V.: Generalized Superposition. *Information and Control* **11** (1967) 528–536
10. Pinoli, J.C.: A General Comparative Study of the Multiplicative Homomorphic, Log-Ratio and Logarithmic Image Processing Approaches. *Signal Processing* **58** (1997) 11–45
11. Jourlin, M., Pinoli, J.C.: A model for logarithmic image processing. *Journal of Microscopy* **149** (1988) 21–35
12. Jourlin, M., Pinoli, J.C.: Logarithmic Image Processing : The Mathematical and Physical Framework for the Representation and Processing of Transmitted Images. *Advances in Imaging and Electron Physics* **115** (2001) 129–196
13. Pinoli, J.C.: The Logarithmic Image Processing Model : Connections with Human Brightness Perception and Contrast Estimators. *Journal of Mathematical Imaging and Vision* **7**(4) (1997) 341–358
14. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*. Addison-Wesley (1992)
15. Serra, J.: *Image Analysis and Mathematical Morphology*. Academic Press, London, U.K. (1982)
16. Debayle, J., Pinoli, J.C.: Multiscale Image Filtering and Segmentation by means of Adaptive Neighborhood Mathematical Morphology. In: *Proc. of the IEEE ICIP*. Volume III., Genova, Italy (2005) 537–540
17. Stockham, T.G.: Image Processing in the Context of a Visual Model. In: *Proceedings of the IEEE*. Volume 60. (1972) 825–842
18. Ramponi, G., Strobel, N., Mitra, S.K., Yu, T.H.: Nonlinear Unsharp Masking Methods for Image-Contrast Enhancement. *Journal of Electronic Imaging* **5**(3) (1996) 353–366
19. Soille, P.: *Filtering*. In: *Morphological Image Analysis. Principles and Applications*. Springer Verlag, New York (2003) 241–266
20. Jourlin, M., Pinoli, J.C., Zeboudj, R.: Contrast definition and contour detection for logarithmic images. *Journal of Microscopy* **156** (1988) 33–40
21. Beucher, S., Lantuejoul, C.: Use of watersheds in contour detection. In: *International Workshop on image processing, real-time edge and motion detection/estimation*, Rennes, France (1979)
22. Gain, P., Thuret, G., Kodjikian, L., Gavet, Y., Turc, P.H., Theillere, C., Acquart, S., LePetit, J.C., Maugery, J., Campos, L.: Automated tri-image analysis of stored corneal endothelium. *British Journal of Ophthalmology* **86** (2002) 801–808

Frozen-State Hierarchical Annealing

Wesley R. Campaigne¹, Paul Fieguth¹, and Simon K. Alexander²

¹ Department of Systems Design Engineering, University of Waterloo
Waterloo, Ontario, Canada, N2L 3G1
{wrcampai, pfieguth}@uwaterloo.ca

² Department of Mathematics, University of Houston
Houston, Texas, U.S.A., 77204
simon@math.uh.edu

Abstract. There is growing demand for methods to synthesize large images of porous media. Binary porous media generally contain structures with a wide range of scales. This poses difficulties for generating accurate samples using statistical techniques such as simulated annealing. Hierarchical methods have previously been found quite effective for such problems. In this paper, a frozen-state approach to hierarchical annealing is presented that offers over an order of magnitude reduction in computational complexity versus existing hierarchical techniques. Current limitations to this approach and areas of further research are discussed.

1 Introduction

Scientific imaging plays a significant role in research advancement, especially with the increasing availability of sophisticated imaging tools, including magnetic resonance imaging, scanning electron microscopy, confocal microscopy, computer aided X-ray tomography, and ultrasound, to name only a few. Because of the significant research funding and public interest in medical imaging and remote sensing, these aspects of scientific imaging have seen considerable attention and success.

However there is an enormous variety of imaging problems outside of medicine and remote sensing, where we would argue the current image processing practice to be relatively rudimentary, and where substantial contributions remain to be made. One such area is that of porous media — the science of water-porous materials such as cement, concrete, cartilage, bone, wood, and soil, with corresponding significance in the construction, medical, and environmental industries.

Because samples of porous media may, in some cases, be expensive to produce, handle, and measure, there is a growing literature [10,11,13] on the simulation and synthesis of large-scale 2D and 3D binary random fields. Some past approaches used a Markov-FFT approach, however the current state-of-the-art focuses on methods of statistical sampling, especially simulated annealing. Although annealing possesses attractive convergence properties *in principle*, in practice the approach is computationally very slow, particularly for images containing a mix of large-scale and small-scale structures — very common in porous media.

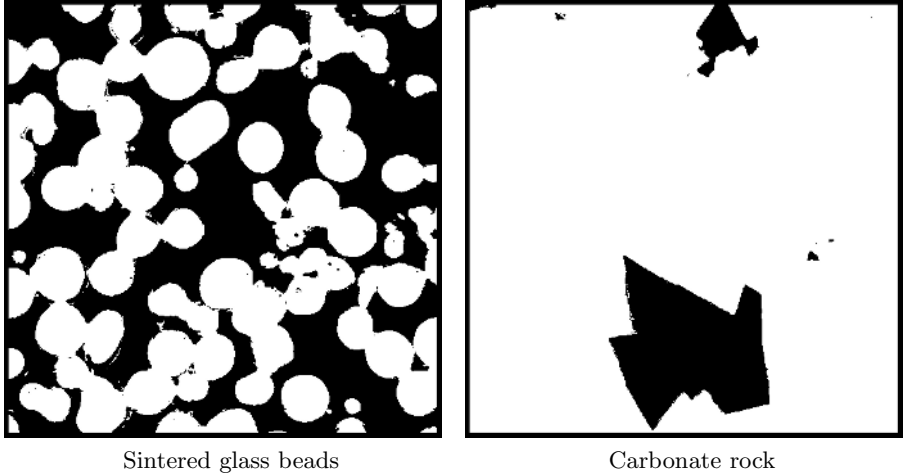


Fig. 1. Excerpts of two large images of physical porous media. The range of scales of pore structure (black) is clearly evident and poses challenges for pixel-based synthesis.

Our long-term research objective is the development of accelerated, hierarchical approaches to annealing-based image synthesis. We have had past success in such hierarchical synthesis, however even these approaches were hampered by having to, eventually, perform some number of annealing passes on the very finest scale. In this paper we propose a truly hierarchical method, in which large-scale structures are synthesized *and fixed in place* at coarse scales, meaning that only local, fine-resolution details remain to be refined at finer scales.

2 Motivation

Suppose we consider a two-dimensional domain of $n = m \times m$ pixels. Further suppose that the domain contains regions (solids or pore voids) having a fractional size of γ relative to the entire domain; that is, the regions have an approximate diameter $d = \gamma m$, such that γ is a fixed fraction, but that d increases as we seek to synthesize finer and finer images (larger and larger m).

Very approximately, at least $\mathcal{O}(d^2)$ passes of simulated annealing are required for convergence, thus the total computational complexity is $\mathcal{O}(\gamma^2 m^4)$. Therefore to achieve one finer scale of resolution increases the computational time by a factor of sixteen, very quickly limiting the size of image which can be considered, and therefore also limiting the range of scales which can be produced — a key consideration in porous media, many of which exhibit structure on scales many orders of magnitude apart!

Existing approaches to hierarchical simulated annealing [1,2,3] seek to ameliorate the above condition by synthesizing large-scale structure at relatively coarse scales, when the structures measure only a few pixels, allowing for much more

rapid convergence at finer scales where only small-scale details remain to be determined. Such an approach has led to two orders of magnitude improvement in computational complexity [2], however two key problems remain:

1. Because each scale is treated as a separate annealing problem, great care must be taken in selecting an annealing schedule to ensure that the structures synthesized at coarser scales are not randomly eroded.
2. Because the method continues to visit every pixel at every scale, including the finest scale, there remains a substantial computational burden of visiting very large numbers of pixels at the finest scale.

Consider the porous media such as those in Fig. 1, and suppose we wish to produce a very detailed synthesis, such as $n = 8000 \times 8000$ pixels. Clearly the overwhelming fraction of these 64-million pixels lie far within large regions, pixels which are extremely unlikely to be changed at the finest scale. That is, the overwhelming fraction of site visits at the finest scale are unnecessary. The crux of our approach is to reduce computational complexity by allowing only certain pixels to be sampled and changed at a given scale, giving rise to two benefits paralleling the above problems:

1. Because the larger-scale structures are *fixed* at coarser scales, finer scales cannot destroy or erode these structures, so our proposed method is not sensitive to a choice of annealing schedule.
2. Because our proposed method visits only a small subset of pixels at a given scale, much larger domains become computationally tractable.

3 Method

Let $x^{(s)}$ represent an image at scale s where increasing s denotes progressively finer scales, and let $x_i^{(s)}$ denote the value of $x^{(s)}$ at pixel i ; $x_i^{(s)} \in \{0, \frac{1}{2}, 1\}$ (i.e., a ternary state: black, gray, or white). Let $\{i_1, i_2, i_3, i_4\}$ be the indices of the children of $x_i^{(s-1)}$ at scale s . Then given training image $x^{(s)}$, the coarser-scale representation $x^{(s-1)}$ can be derived for modelling using

$$x_i^{(s-1)} = \begin{cases} 0 & \text{if } x_{i_j}^{(s)} = 0 \quad \forall j \\ 1 & \text{if } x_{i_j}^{(s)} = 1 \quad \forall j \\ \frac{1}{2} & \text{otherwise} \end{cases} \quad (1)$$

The repeated use of (1) allows a single sufficiently large binary image to provide data for modelling at arbitrarily coarse scales, as demonstrated in Fig. 2.

In synthesis, a corresponding set of rules is asserted as constraints on the annealing:

$$\text{If } x_i^{(s-1)} \in \{0, 1\}, \quad \text{then } x_{i_j}^{(s)} = x_i^{(s-1)} \quad (2a)$$

$$\begin{aligned} \text{If } x_i^{(s-1)} = \frac{1}{2}, \quad \text{then neither } & x_{i_j}^{(s)} = 0, \quad \forall j \\ & \text{nor } x_{i_j}^{(s)} = 1, \quad \forall j \end{aligned} \quad (2b)$$

Modelling

Synthesis

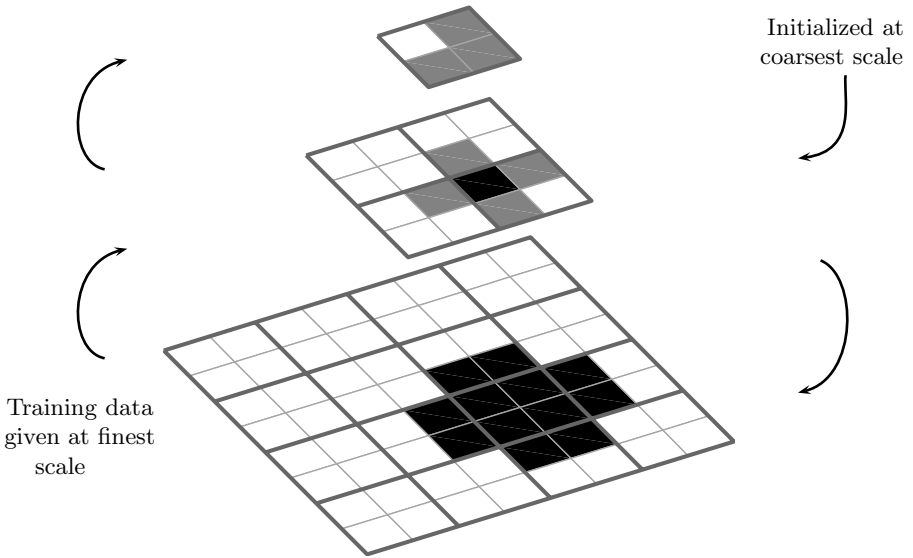


Fig. 2. Example up-scaling and down-scaling behaviour. The heavier dividing lines indicate groups of pixels that are expressed as a single pixel at the next coarser scale, using the rule expressed in (1).

The consequence of (2) is that each state in the sample space visited by the annealer must be consistent under (1) with the result at all coarser scales. Black and white pixels represent frozen states in the image domain that are preserved at all finer scales. If a coarse-scale pixel is gray, then the corresponding set of pixels at finer scales cannot be either entirely black or entirely white. All pixels subject to (2a) can be ignored in the annealing sampler, as their values have been previously frozen.

Figure 2 demonstrates this coarsening method in practice. Progressively coarser model information is extracted via (1) based on a large binary training image. Synthesis starts from some coarse initialization, and annealing is performed at progressively finer scales (each until convergence), constrained by the previous scale’s result according to (2). When annealing the final, finest scale result, it is additionally asserted that all values must be 0 or 1.

Further constraints on x are made using some energy function $E(x)$. Choices of x leading to low energies are believed to be plausible rest states of the process. Many such energy models have been proposed, however adapting existing energy models used with binary images for use with the ternary frozen-state model is a non-trivial task. The research presented here uses an extension of the local neighbourhood model described in [1], where it was shown to be sufficient for implicitly matching a number of properties from training images.

Consider a local neighbourhood around some pixel at position i . For convenience, let us assume it is a 3×3 neighbourhood structure as in Fig. 3, with

| | | |
|-------|-------|-------|
| P_1 | P_2 | P_3 |
| P_4 | P_0 | P_5 |
| P_6 | P_7 | P_8 |

Fig. 3. Indices for a second-order neighbourhood

position i in the central P_0 location. (Other neighbourhood structures can be used instead; this assumption is merely to simplify the explanation.)

Let c denote one of the 3^9 possible unique configurations of the neighbourhood given ternary values. The value for c , $0 \leq c < 3^9$ corresponding to the neighbourhood about pixel $x_i^{(s)}$ can be evaluated using

$$c_i^{(s)} = \sum_k (2 \cdot x_{P_k}^{(s)}) \cdot 3^k \quad (3)$$

Let $H_c^{(s)}$ then express the probability of configuration c at scale s , calculated by observation of its frequency in the training data. Figure 4 offers an example of this distribution at one scale. Let $\mathcal{H}_c(x^{(s)})$ produce histogram entry c given image $x^{(s)}$. Supposing the training data at scale s was expressed as image $y^{(s)}$, then $H_c^{(s)} = \mathcal{H}_c(y^{(s)})$. Given these terms, the energy function is then

$$E(x^{(s)}) = \sum_c (\mathcal{H}_c(x^{(s)}) - H_c^{(s)})^2 \quad (4)$$

This is a non-parametric, highly local model. To effectively capture the nature of pore structures of the sort seen in Fig. 1, this model must be paired with a hierarchical approach to image synthesis.

It was observed in early testing that at finer scales the energy function would attempt to contradict the coarse-scale results, finding it more advantageous to overfit some aspects of the training data (for instance, matching the precise probability of homogenous black or white neighbourhoods) at the expense of accurate pore shapes. For this reason, two modifications were made to the energy function.

The first modification is to weight configuration c in the energy function by its standard deviation. Let $\sigma_c^{(s)}$ express the standard deviation of the value of $H_c^{(s)}$. This is inferred by determining $H_c^{(s)}$ for many subsections of the training data at scale s with the same size as $x^{(s)}$ and evaluating the standard deviation across this set of $H_c^{(s)}$ values. Using this weighting, with some small value ϵ , a new energy function was proposed:

$$E(x^{(s)}) = \sum_c \frac{(\mathcal{H}_c(x^{(s)}) - H_c^{(s)})^2}{(\sigma_c^{(s)})^2 + \epsilon} \quad (5)$$

This modification also expresses more specifically the purpose of what the annealing is trying to accomplish: we are not trying to create an image simply

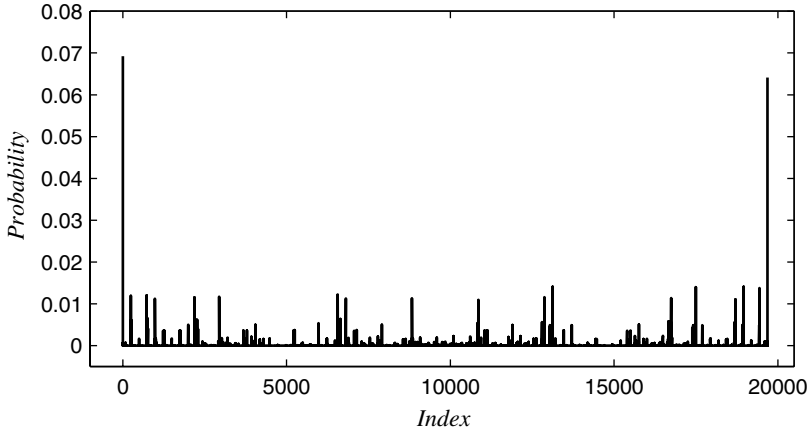


Fig. 4. A target histogram model at 64×64 . The horizontal axis indexes the 3^9 state configurations possible using the neighbourhood structure of fig. 3. The peaks at the first and last indices correspond respectively to the strong probability of homogeneously black and homogeneously white neighbourhoods.

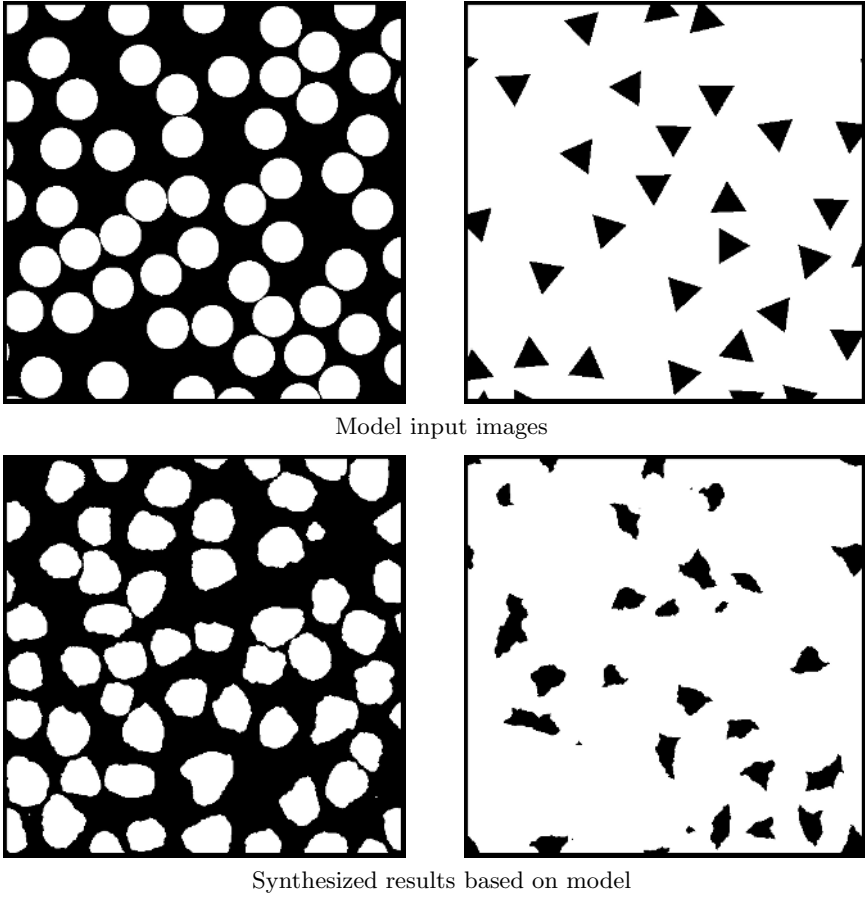
to match one specific distribution neighbourhood *exactly*; we are trying to sample images from a distribution of plausible images. The nature of this distribution is inferred from the training data. Weighting by $\sigma_c^{(s)}$ is a means of using the variations within the training data to infer how important the precision of each value $H_c^{(s)}$ is in that greater distribution.

A second modification was to restrict the scope of the histogram operator to only those pixels whose neighbourhood configurations can be affected by the annealing process. The annealer can only affect the image as much as the constraints on the current scale permit, so the energy function should be evaluating the image *given those constraints*. Effectively, this means that $\mathcal{H}_c(x^{(s)})$ operates only on those pixels in $x^{(s)}$ that lie within the radius of the neighbourhood structure of some i_j for which $x_i^{(s-1)} = \frac{1}{2}$. This allows the energy function $E(x^{(s)})$ to evaluate the fitness of $x^{(s)}$ independently from $E(x^{(s-1)})$.

For this paper, all results came from the use of a second-order 3×3 neighbourhood, although the use of a third-order 13-pixel neighbourhood with the ternary model is also feasible. Future research may focus on adaptive methods for expanding the neighbourhood structures in a manner that actively balances the issues of memory requirements and data sparseness with increased descriptive power.

4 Results

As an initial test for the proposed algorithm, a set of artificial test images were created (Fig. 5). Each image in this set is entirely composed of a single geometric shape repeated across the image at random, non-overlapping locations with



Model input images

Synthesized results based on model

Fig. 5. Multiple images composed of equally sized shapes, with random rotations and random non-overlapping locations, were used to test the proposed algorithm’s performance in shape discrimination

random orientation. Figure 5 contains excerpts and results for the ‘circles’ and ‘triangles’ members of this set. In both cases, the proposed method, despite being non-parametric, is able to recreate the pore size and density of the respective training data. The disjointness of the pores is also preserved. The structures in the ‘circles’ image clearly resemble the circular phenomena of the training data, although the small 3×3 local neighbourhood model was unable to recreate the smooth contours at the finest scale. Similarly, for the ‘triangles’ image, the pore structure exhibits peaked points, often three per pore, but the highly local nature of the energy measurement was unable to discriminate the long, straight edges of the original image.

In both results, there are also some very small structures present — so much smaller than anything in the input images that their presence suggests an error.

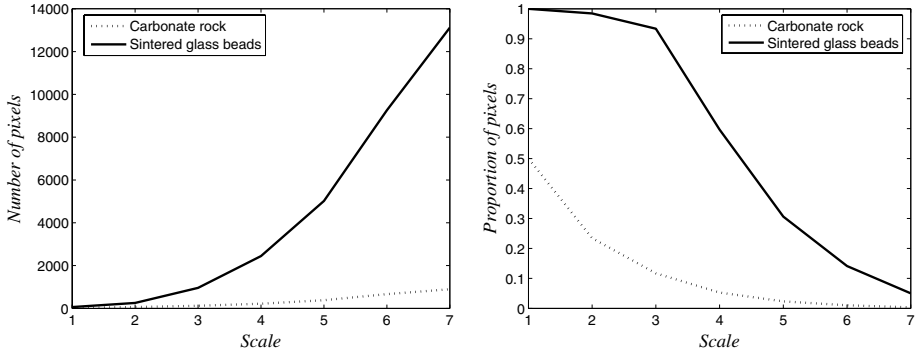


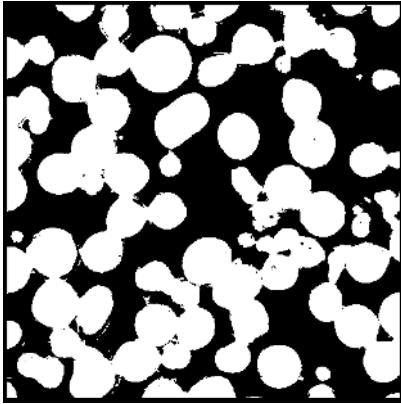
Fig. 6. The number and proportion of pixels to be visited in a single pass of the annealer, as the scale increases from 8×8 to 512×512 . The number of pixels to visit at a scale equals the number of pixels which are not fixed; that is, those having a grey-valued pixel as a parent. As seen in the right plot, the pixels to visit are normally only a small fraction of the total, especially at fine scales (where the bulk of the computational effort appears) and particularly for sparsely detailed images.

They exist as a result of (2b): *some* mixture of black and white pixels must be present there to satisfy the constraint posed by a gray element at a coarser scale.

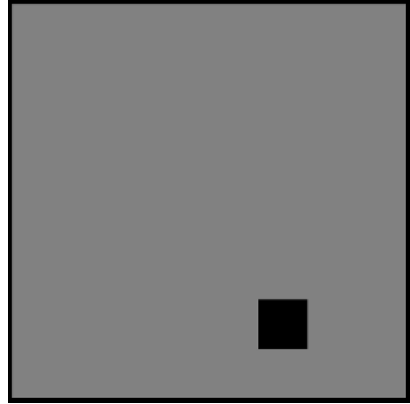
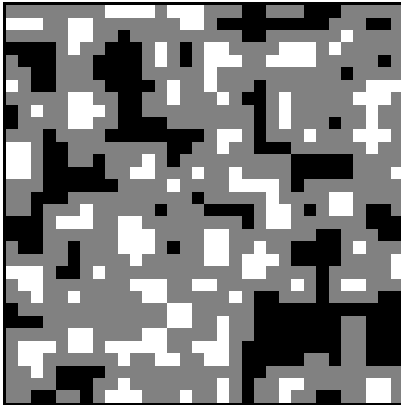
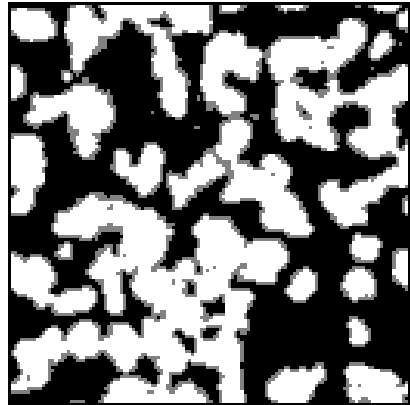
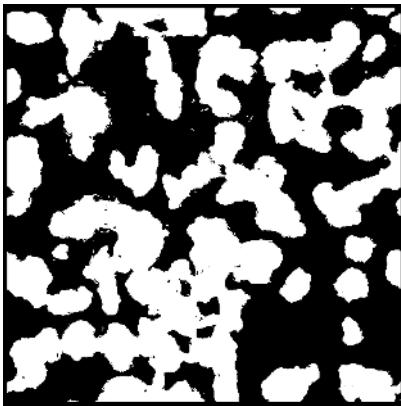
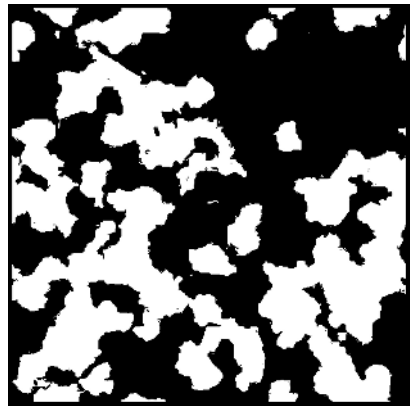
These two initial test images were designed to emulate the morphology of two physical porous media. Figure 7 is an image of a surface of a medium created using densely packed spherical glass beads. The appearance of these beads, when imaged in cross-section, is approximated in the ‘circles’ image of Fig. 5. In doing so, the ‘circles’ image tested the proposed method’s performance on this morphological aspect independently from other features present in the physical image. Similarly, the ‘triangles’ image resembles the angular, crystalline pore structures of carbonate rock in Fig. 8.

The synthesis results of Figs. 7 and 8 exhibit the same properties observed previously: the size, density, and general morphology of pore structure is maintained although the pore edges do not match the precise nature seen in the respective training images. The smaller, orphaned structures previously attributed to ternary constraints are also present, however in this instance it is not necessarily a fault: both sets of training data contain similar tiny structures. The non-ternary results, seen in Figs. 7(f) and 8(f), also exhibit such structures.

In the intermediate scales in Figs. 7 and 8, gray pixels are present on the interface between black and white regions. Gray pixels identify where structures *will* exist at finer scales; they are those pixels whose values have not yet been frozen. The progressive thinning of the gray mediation demonstrates the significance in the reduction in computational complexity, since the number of sites sampled at scale s is directly limited by the number of gray pixels present in the converged result at scale $s - 1$, plotted in Figure 6.



(a) Excerpt of training data

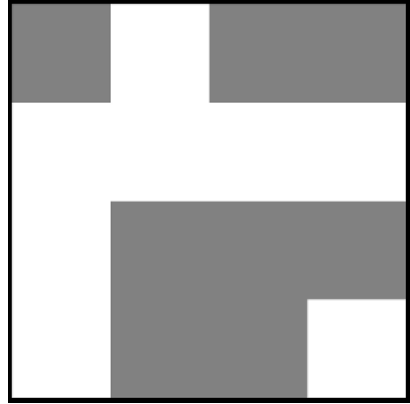
(b) Initial ternary constraint, 8×8 (c) Ternary result at 32×32 (d) Ternary result at 128×128 (e) Ternary result at 512×512
(full resolution)

(f) Non-ternary result

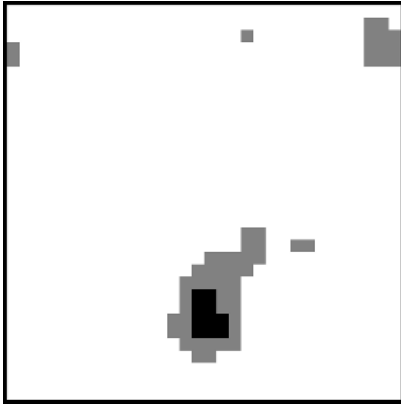
Fig. 7. Training data and synthesis results for an image of sintered glass beads surface



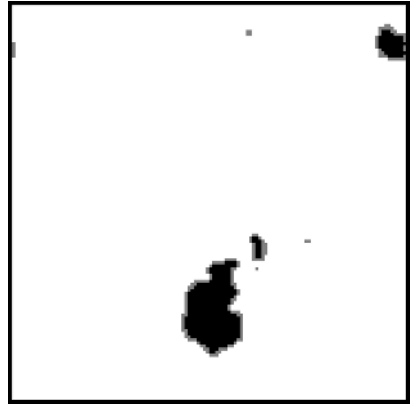
(a) Excerpt of training data



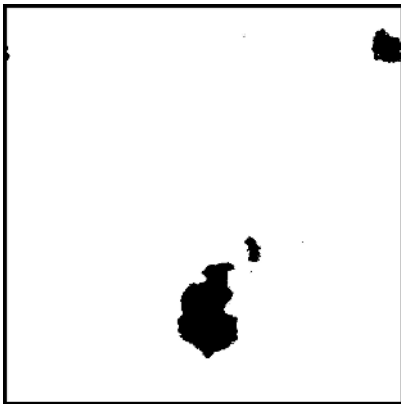
(b) Initial ternary constraint, 4×4



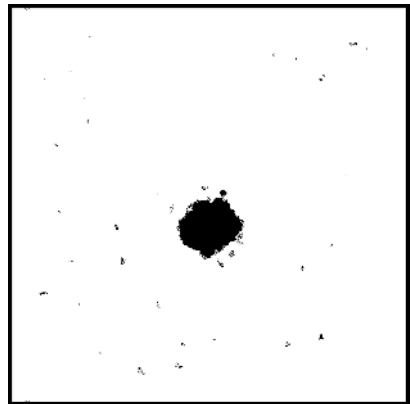
(c) Ternary result at 32×32



(d) Ternary result at 128×128



(e) Ternary result at 512×512
(full resolution)



(f) Non-ternary result

Fig. 8. Training data and synthesis results for an image of carbonate rock surface

4.1 Advantages

The most significant advantage to the proposed model is the reduction in the number of proposed changes faced by the annealer in a single pass across the image at all but the coarsest scales. The reduction experienced for the results presented is shown in Fig. 6. For the sintered glass beads results of Fig. 7, only 8.9% of the pixels in the overall hierarchy needed to be considered for changing by the annealer — an order of magnitude reduction in the computational requirements. Using the sparser carbonate rock data of Fig. 8, only a meagre 0.68% of the total pixels were examined.

The hard assertion that black and white never change at finer scales prevents large structures created at coarse scales from eroding when annealed at finer scales at high temperatures. By enabling the possibility of annealing with arbitrarily high initial temperatures at *each* scale, the annealing process is able to explore the sample space more thoroughly.

4.2 Limitations

The most immediate limitation of the frozen-state method is that other energy functions which assume a binary field, such as correlation and chordlength distributions, cannot be directly used and it is not immediately obvious how to extend such functions to work in a ternary domain.

Another consequence of using a frozen-state model is an increase in the memory needed to store the energy model. With the local neighbourhood model used here, the memory requirement for using a neighbourhood of n pixels increases as $\mathcal{O}(3^n)$, whereas it was only $\mathcal{O}(2^n)$ in the binary case. Increases to the neighbourhood size also pose the problem of increased sparseness: given limited training data, particularly at coarse scales, what is the significance of a neighbourhood configuration that shows up only once in the training set? Expanding the neighbourhood size magnifies this issue.

The ternary method is also subject to unreliable behaviour at very coarse scales. The amount of training data present at each scale is reduced by a factor of four with each level of coarseness; concerns that the data are not able to express the spatially stationary distribution of pore structures at that scale progressively increase. Consider the example in Fig. 2: Would it be accurate to assert that *each* sample from its distribution must have a single white pixel in the upper-left corner at the coarsest scale, or is that scenario merely the result of the positioning of the pore structure at the finest scale? When dealing with very limited coarse-scale data, the latter case would be a sensible assumption. Since the energy model complexity remains the same while the amount of training data decreases exponentially, the annealer is increasingly likely to overfit the energy model. This issue is amplified by the frozen-state nature of the coarse results: the hierarchical constraints asserted by the ternary model force an initial coarse overfitting to be maintained at all finer scales, reducing the diversity of synthesis results.

Although there are some limitations to the frozen-state approach, they are minor. The resistance to erosion of large structures at fine scales and, more importantly, the order-of-magnitude or greater reduction in computational complexity are of great practical benefit.

References

1. S.K. Alexander, P. Fieguth, and E.R. Vrscay, *Hierarchical annealing for random image synthesis*, EMMCVPR 2003, LNCS, no. 2683, Springer, 2003.
2. S.K. Alexander, P. Fieguth, and E.R. Vrscay, *Parameterized hierarchical annealing for scientific models*, ICIAR 2004, LNCS, no. 3211, Springer, 2004.
3. S.K. Alexander, P. Fieguth, and E.R. Vrscay, *Hierarchical Annealing for Scientific Models*, IEEE ICASSP **3** (2004), 33–36.
4. C. Bouman and M. Shapiro, *A multiscale random field model for Bayesian image segmentation*, IEEE Image Processing **3** (1994), no. 2, 162-177.
5. S. Geman and D. Geman, *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*, IEEE PAMI **6** (1984), 721-741.
6. B. Gidas, *A renormalization group approach to image processing problems*, IEEE PAMI **11** (1989), no. 2, 164-180.
7. T. Hofmann, J. Puzicha, and J.M. Buhmann, *Unsupervised texture segmentation in a deterministic annealing framework*, IEEE PAMI **20** (1998), no. 8, 803-818.
8. M.E. Kainourgiakis, E.S. Kikkinides, G.Ch. Charalambopoulou, and A.K. Stubos, *Simulated annealing as a method for the determination of the spatial distribution of a condensable adsorbate in mesoporous materials*, Langmuir **19** (2003), no. 8, 3333-3337.
9. Z. Kato, M. Berthod, and J. Zerubia, *A hierarchical Markov random field model and multitemperature annealing for parallel image classification*, Graphical Models and Image Processing **58** (1996), no. 1, 18-37.
10. Z. Liang, M.A. Ioannidis, and I. Chatzis, *Reconstruction of 3d porous media using simulated annealing*, Computational Methods in Water Resources XIII (Balkema, Rotterdam) (Bentley et al., ed.), 2000.
11. M.G. Rozman and M. Utz, *Efficient reconstruction of multiphase morphologies from correlation functions*, Physical Review E **63** (2001), no. 066701.
12. H. Szu and R. Hartley, *Fast simulated annealing*, Physics Letters A **122** (1987), 157-162.
13. M.S. Talukdar, O. Torsaeter, and M.A. Ioannidis, *Stochastic reconstruction of particulate media from two-dimensional images*, Journal of Colloid and Interface Science **248** (2002), no. 2, 419-428.

Fast and Robust Filtering-Based Image Magnification

Wenze Shao¹ and Zihui Wei²

¹ Computer Science and Engineering, Nanjing University of Science and Technology,
210094 Nanjing, China
shaowenze@qianlong.com

² Graduate School, Nanjing University of Science and Technology,
210094 Nanjing, China
gswei@mail.njust.edu.cn

Abstract. Image magnification, or interpolation, produces a high resolution image from a low resolution, and perhaps noisy image. There have been proposed a variety of magnification algorithms. However, they are either sensitive to the noise, or non-robust to the blocking artifacts, or of high computational complexity, which hence limits their utility. In this paper, we propose an alternative magnification approach utilizing a filtering-based implementation scheme and novel regularization through coupling bilateral filtering with the digital total variation model. The approach is simple, fast, and robust to both the noise and blocking artifacts. Experiment results demonstrate the effectiveness of our approach.

Keywords: Magnification, bilateral filtering, total variation, regularization.

1 Introduction

Image magnification, or interpolation, is desired and often required in many actual applications, such as medical imaging, astronomical imaging, law enforcement, and so on [1, 2]. Briefly speaking, image magnification is to produce a high-resolution (HR) image from a low-resolution (LR), and perhaps noisy image. One goal in this paper is to estimate a HR image of high visual quality, while with much low computational complexity.

Image magnification can be formulated as the following degradation model [3]

$$\mathbf{g} = \mathbf{D}\mathbf{u} + \boldsymbol{\eta}. \quad (1)$$

where \mathbf{g} , \mathbf{u} , and $\boldsymbol{\eta}$ are respectively the lexicographically ordered vectors of the observed $N \times N$ LR image \mathbf{g} , the unknown $qN \times qN$ HR image \mathbf{u} , and the additive sensor noise $\boldsymbol{\eta}$. The constant q denotes the down-sampling factor. The $N^2 \times q^2 N^2$ matrix \mathbf{D} represents the low-pass filtering and down-sampling process. The matrix can be decomposed as $\mathbf{D} = \mathbf{D}_1 \otimes \mathbf{D}_1$, where \otimes denotes the Kronecker product, and \mathbf{D}_1 represents 1-D filtering and down-sampling effect. Hence, (1) can be explicitly written as

$$g_{m,n} = \frac{1}{q^2} \left(\sum_{k=q(m-1)+1}^{qm} \sum_{l=q(n-1)+1}^{qn} u_{k,l} \right) + \eta_{m,n}. \quad (2)$$

There have been many solution methods for image magnification in literatures [1~11]. Standard approaches rely on direct interpolation, linear [4, 10, 11] or nonlinear [6, 7]. Linear methods, such as the nearest-neighbor, bilinear and bicubic interpolation, correspond to fitting the image in spline kernel spaces with low computational complexity, while apt to generate blocking artifacts and blurring. Nonlinear ones adapt the interpolation process according to the edges in the LR image, which are hence called edge-directed interpolation [7], while with high computational complexity. Besides, both linear and nonlinear direct approaches completely fail when the LR image is degraded by noise. On the other hand, it is obvious that the direct inversion of the matrix \mathbf{D} in (1) is not feasible, because \mathbf{D} is not square and the noise η is unknown either. Hence, image magnification is mathematically an ill-posed inverse problem in the sense of Hardmard [12]. The regularization theory is often exploited to solve such a problem, whose principle is to approximate the solution of an ill-posed problem using a well-posed problem obtained by imposing some kind of *a-priori* constraints [13]. In image processing, the *a-priori* constraints are essentially the image models, i.e., the locations of sharp edges, fine textures, etc.

Assume the noise η is independently and identically Gaussian distributed, image magnification can be then formulated as the following minimization functional

$$\mathbf{u} = \arg \min_{\mathbf{u}} E[\mathbf{u}, \lambda] = \arg \min_{\mathbf{u}} \left\{ \frac{\lambda}{2} \cdot E_1[\mathbf{g}, \mathbf{u}] + E_2[\mathbf{u}] \right\}. \quad (3)$$

where λ is the regularization parameter controlling the tradeoff between $E_1[\mathbf{g}, \mathbf{u}]$ and $E_2[\mathbf{u}]$, and $E_1[\mathbf{g}, \mathbf{u}]$ is the data fidelity term penalizing the inconsistency between the estimated HR image and the observed LR image, represented as

$$E_1[\mathbf{g}, \mathbf{u}] = \|\mathbf{g} - \mathbf{D}\mathbf{u}\|^2 = \sum_{m=1}^N \sum_{n=1}^N \left\{ g_{m,n} - \frac{1}{q^2} \left(\sum_{k=q(m-1)+1}^{qm} \sum_{l=q(n-1)+1}^{qn} u_{k,l} \right) \right\}^2. \quad (4)$$

As for $E_2[\mathbf{u}]$, it is the regularization term imposing *a-priori* constraints on the HR image \mathbf{u} , which determines the visual quality of the magnified HR image to a great degree. Many regularization terms have been proposed for magnification, such as the Tikhonov regularization [1, 7], the Huber-MRF regularization [3, 9], the digital total variation (TV) regularization [2, 14, 15], whose performance will be analyzed a little later in the paper. Except for the visual quality of the magnified HR image, the computational complexity of minimizing $E[\mathbf{u}, \lambda]$ is another critical issue in image magnification. Currently, gradient descent [3, 9, 15] is one of the most popular methods to solve the minimization functional (3), while involving a large memory of the matrix $\mathbf{D}^T \mathbf{D}$, the high computational complexity of the adaptive time step, the issue of computational stability, and so on.

Although there have been proposed many magnification algorithms, they are either sensitive to the noise, or non-robust to the blocking artifacts, or of high computational complexity, which hence limits their utility. In this paper, we propose an alternative approach utilizing a filtering-based implementation scheme and novel regularization through coupling bilateral filtering with the digital TV model. The method is simple, fast, and robust to both the noise and blocking artifacts. Experiment results demonstrate the effectiveness of our approach.

The paper is organized as follows. Section 2 induces the origin of bilateral filtering. Section 3 proposes a digital bilateral TV model and a filtering-based magnification scheme. Experiments are shown in section 4, and concluding remarks are given in section 5.

2 Origin of Bilateral Filtering

Definition 2.1 (Neighborhood system)

A neighborhood system on 2-D discrete grid Z is a family $N = \{N_\alpha\}_{\alpha \in Z}$ of subsets of Z such that for all $\alpha, \beta \in Z$,

- i) $\alpha \in N_\alpha$;
- ii) $\beta \in N_\alpha \Leftrightarrow \alpha \in N_\beta$;

The subset N_α is called the neighborhood of α . And if $\beta \in N_\alpha$, we write $\beta \sim \alpha$ just for simplification.

Definition 2.2 (Neighborhood order)

The order of a neighborhood N_α is s (>1) if and only if the size of N_α is $(1 + 2(s-1)) \times (1 + 2(s-1))$, and $s = 1$ corresponds to a 4-neighborhood.

2.1 Bilateral Filtering and Its Origin

Bilateral filtering, proposed by Tomasi and Manduchi [16], smoothes images while preserving the edges by means of a nonlinear combination of nearby image values in a large neighborhood, preferring near values to distant values in both domain and range. In fact, the same idea also can be found in the previous works [20]. A single iteration of bilateral filtering yields:

$$u_x^{t+1} = \frac{\sum_{y \sim x} g(u_x^t - u_y^t) \cdot w(x-y) \cdot u_y^t}{\sum_{y \sim x} g(u_x^t - u_y^t) \cdot w(x-y)}. \quad (5)$$

where $x = (x_1, x_2)$, $y = (y_1, y_2)$ are the coordinates in the image domain Ω , the order of $y \sim x$ is $s \geq 2$. The function g measures the photometric similarity of gray levels, called range weight; and the function w measures the geometric closeness of space coordinates, called domain weight. Both functions are defined as (6) in [16]

$$f(r) = \exp\{-|r|^2 / 2\sigma^2\}. \quad (6)$$

Several developments have been made on bilateral filtering in both theory and actual applications [17, 18, 15]. Based on the principle of bilateral filtering, Barash extended anisotropic diffusion and adaptive smoothing to an arbitrary neighborhood [17], showing their equivalence and stressing the importance of using large neighborhoods for edge-preserving smoothing. Another interesting conclusion given by Elad [18] is that, bilateral filtering essentially emerges from the weighted least squares minimization functional, as a single iteration of the Jacobi algorithm. Quite recently, a new regularization term was proposed for super-resolution [15], through coupling

bilateral filtering with the l^1 norm. Here, we prove that bilateral filtering emerges from the Bayesian framework, but with a more concise penalty functional.

The kernel of the domain weight is its measurement of the geometric closeness of space coordinates. Hence, the choice of the domain weight is much intuitionistic, such as (6) and other nonincreasing measurements on the half positive axis. In the paper, we choose (6) for the experiments. For gray images, the kernel of the range weight is its measurement of the photometric similarity of gray levels. Hence, the choice of the range weight is a critical issue for edge-preserving smoothing, as we will see, which is equivalent to the designing of potential functions in the Markov Random Field (MRF). Assume the steady state of (5) is u , i.e., $\lim_{l \rightarrow \infty} u^l(x) = u(x)$ for any $x \in \Omega$, then (5) is reduced to

$$u_x = \frac{\sum_{y \sim x} g(u_x - u_y) \cdot w(x - y) \cdot u_y}{\sum_{y \sim x} g(u_x - u_y) \cdot w(x - y)}. \quad (7)$$

where g needs not to be (6). Obviously, (7) can be rewritten as

$$\sum_{y \sim x} w(x - y) \cdot g(u_x - u_y) \cdot (u_x - u_y) = 0. \quad (8)$$

If there exists some differentiable function ρ such that $g(r) = \rho'(r)/r$, then

$$u = \arg \min_u \left\{ \sum_{x \in \Omega} \sum_{y \sim x} w(x - y) \cdot \rho(u_x - u_y) \right\}. \quad (9)$$

As a matter of fact, the function ρ can be chosen as the potential function in MRF. And several efforts have been made on the choice of edge-preserving potential functions [12, 19]. Therefore, the steady state u of bilateral filtering is essentially a local or global minimum of (9) depending on the function ρ . On the other hand, bilateral filtering is reconfirmed to emerge from the Bayesian framework, but with a more concise penalty functional than that of Elad [18].

3 Nonlinear Filtering-Based Image Magnification

Bilateral filtering is capable of edge-preserving in image denoising, mainly consisting in two important factors, i.e., firstly its origin in the Bayesian framework with edge-preserving potential functions, and secondly the large neighborhood exploited in the filtering process. In fact, the large neighborhood-based filtering has its close relationship with the human visual system and the human perception law, such as the Helmholtz hypothesis, the similarity law, the closeness law and the continuity law. Hence, the idea of bilateral filtering can be viewed as a basic principle in various image processing tasks, such as filtering, restoration, inpainting, magnification, super-resolution, and so on.

3.1 Digital Bilateral Total Variation

Recently, Chan *et al.* [14] proposed the so-called digital total-variation (TV) filter for noise removal, and applied it to image inpainting and magnification (The up-sampling factor is 2). However, unlike bilateral filtering, they restricted the filter to a small support ($s = 1$ or 2), thus losing its prime origin of strength. In this section, we

propose a novel regularization operator through coupling bilateral filtering with the digital TV model, denoted as

$$E_2(u) = \sum_{x \in \Omega} \sqrt{\sum_{y \sim x} w(x-y) \cdot (u_x - u_y)^2}. \quad (10)$$

Obviously, (10) reduces to the digital TV model when the domain weight w equals to 1 and the order of $y \sim x$ is equal to 1 or 2. Given the degradation model $g = u + \eta$, consider the following minimization functional

$$u = \arg \min_u \left\{ \frac{\lambda}{2} \sum_{x \in \Omega} (g_x - u_x)^2 + E_2(u) \right\}. \quad (11)$$

For any $x \in \Omega$, the steady state of (11) yields

$$\lambda \cdot (u_x - g_x) + \sum_{y \sim x} \left(1/|\nabla_x u|_e + 1/|\nabla_y u|_e \right) \cdot w(x-y) \cdot (u_x - u_y) = 0. \quad (12)$$

where

$$|\nabla_x u|_e = \sqrt{\sum_{y \sim x} w(x-y) \cdot (u_x - u_y)^2 + e}. \quad (13)$$

and e is any small positive constant. (12) can be further rewritten as

$$u_x = \sum_{y \sim x} w_{xy} \cdot u_y + w_{xx} \cdot g_x. \quad (14)$$

where

$$w_{xy} = h_{xy} / (\lambda + \sum_{y \sim x} h_{xy}). \quad (15)$$

$$w_{xx} = \lambda / (\lambda + \sum_{y \sim x} h_{xy}). \quad (16)$$

$$h_{xy} = \left(1/|\nabla_x u|_e + 1/|\nabla_y u|_e \right) \cdot w(x-y). \quad (17)$$

We name (14) the digital bilateral TV filtering, which is obviously the generalization of digital TV filtering [14] and much more powerful because of the large neighbourhood exploited in the filtering process.

3.2 Nonlinear Filtering-Based Image Magnification

Based on the degradation model (2), as stated in section 1, image magnification can be formulated as the minimization functional (3). Particularly, when the digital bilateral TV model is chosen as the regularization term in (3), we obtain

$$u = \arg \min_u \left\{ \frac{\lambda}{2} \cdot \sum_{m=1}^N \sum_{n=1}^N \left\{ g_{m,n} - \frac{1}{Q^2} \left(\sum_{k=q(m-1)+1}^{qm} \sum_{l=q(n-1)+1}^{qn} u_{k,l} \right) \right\}^2 + \sum_{x \in \Omega} |\nabla_x u|_e \right\}. \quad (18)$$

The estimated HR image is just the steady state of (18). That is, u satisfies

$$\nabla E[u, \lambda] \Big|_x = \frac{\lambda}{2} \cdot \nabla E_1[g, u] \Big|_x + \nabla E_2[u] \Big|_x = 0. \quad (19)$$

for any $x = (x_1, x_2) \in \Omega$. A simple calculation yields

$$\nabla E_2[\mathbf{u}]|_x = \sum_{y=x} (1/|\nabla_x u|_e + 1/|\nabla_y u|_e) \cdot w(x-y) \cdot (u_x - u_y). \quad (20)$$

$$\nabla E_1[\mathbf{g}, \mathbf{u}]|_x = 2 \cdot \left(g_{m,n} - \frac{1}{q^2} \left(\sum_{k=q(m-1)+1}^{qm} \sum_{l=q(n-1)+1}^{qn} u_{k,l} \right) \right) \cdot \left(-\frac{1}{q^2} \right). \quad (21)$$

where $m = \lceil x_1 / q \rceil$, and $n = \lceil x_2 / q \rceil$. Therefore, (19) can be rewritten as

$$u_x = \sum_{y=x} \varpi_{xy} \cdot u_y + \varpi_{xx} \cdot v. \quad (22)$$

where ϖ_{xy} , ϖ_{xx} , and h_{xy} is the same as (15) ~ (17) respectively when $2\lambda/q^2$ is denoted as λ , and

$$v = \sum_{k=q(m-1)+1}^{qm} \sum_{l=q(n-1)+1}^{qn} ((u_x/q^2 - u_{k,l}) + g_{m,n}). \quad (23)$$

Image magnification can be finally implemented by the filtering process (22), with an arbitrary initial guess u^0 . Since the implementation scheme (22) is filtering-based, our magnification approach is of much low computational complexity.

4 Experiment Results and Performance Analysis

To achieve the ‘‘best’’ visual quality of the estimated HR image, image sharpening is added to the end of filtering-based magnification scheme. In the experiment, an original HR image is locally averaged and under-sampled by the factor of 4 in each direction (see Fig. 1). Magnification results are shown in Fig. 2, utilizing bilinear interpolation, bicubic interpolation, Tikhonov regularization, Huber-MRF regularization, digital TV regularization, and digital bilateral TV regularization. They are all implemented by the filtering-based scheme, just similar to the process (22). The initial guess is chosen as the pixel replication of the LR image.

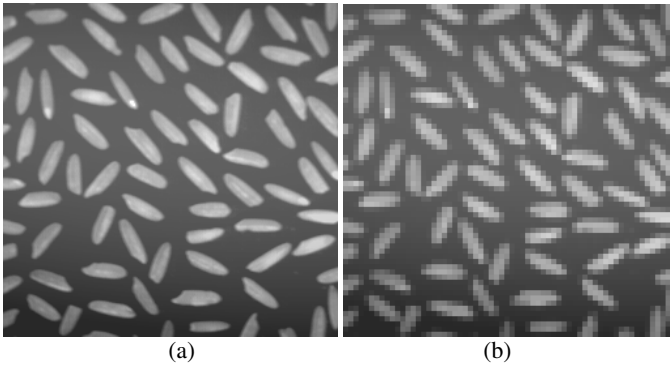


Fig. 1. Original image (256×256) and LR image (64×64)

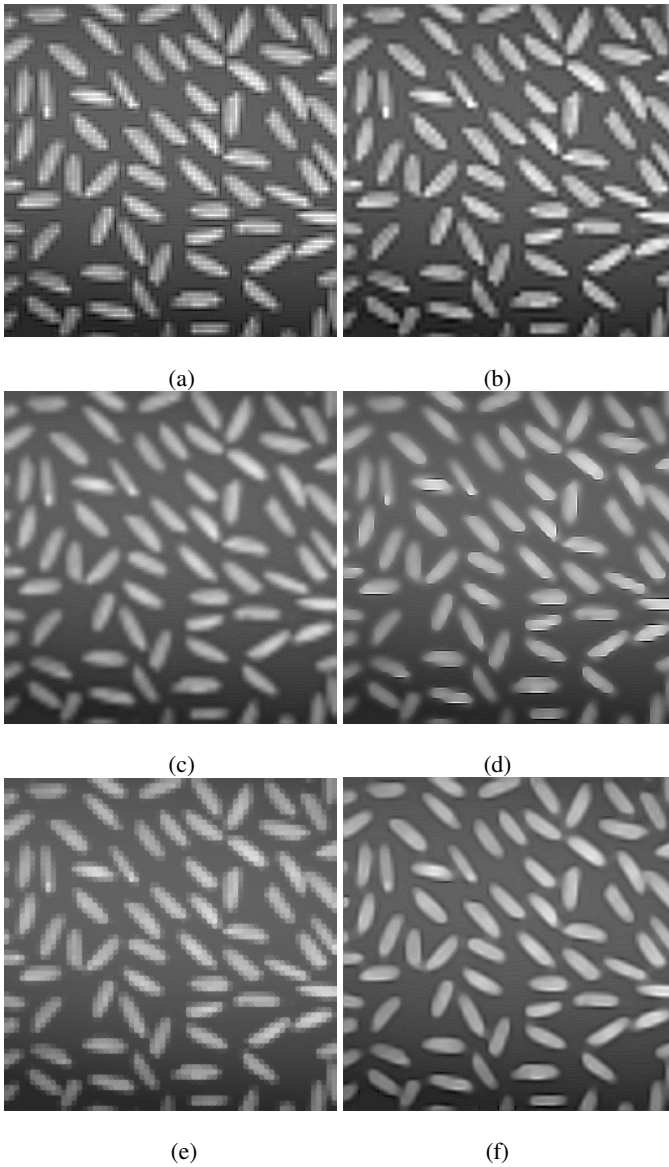


Fig. 2. (a) Bilinear interpolation, (b) Bicubic interpolation, (c) Tikhonov regularization (the 5th iteration, the neighborhood order s is 2, $\lambda = 0.0005$), (d) Huber-MRF regularization (the 10th iteration, the neighborhood order s is 2, the breakdown point is 10, $\lambda = 0.0005$), (e) Digital TV regularization (the 10th iteration, the neighborhood order s is 2, $\lambda = 0.0005$), (f) Digital bilateral TV regularization (the 3rd iteration, the neighborhood order s is 3, the variance σ of the domain weight is 1.5, $\lambda = 0.0005$). Each iteration costs about 2~3 seconds.

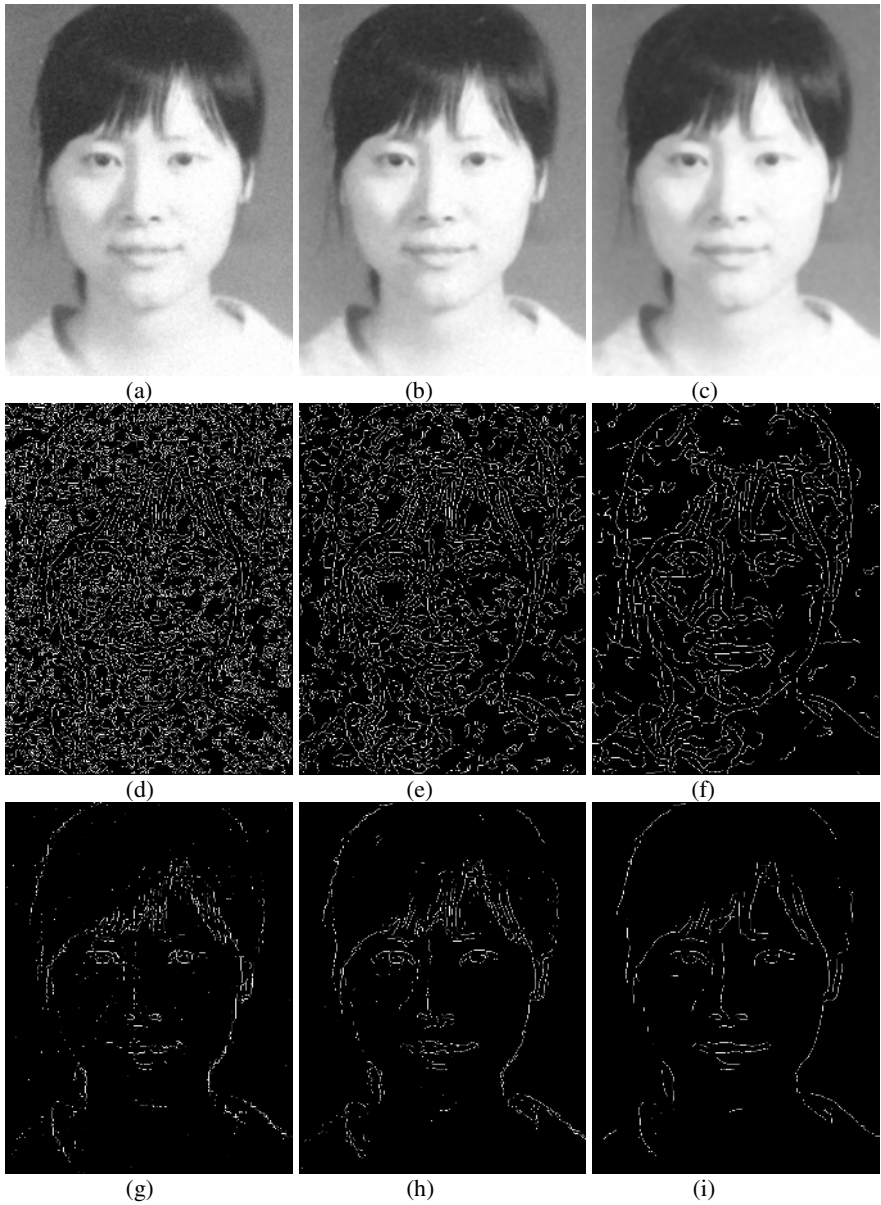


Fig. 3. Magnification results ($q=3$) (a) Bicubic interpolation, (b) Digital TV regularization (the 10th iteration, the neighborhood order s is 2, $\lambda=0.0005$), (c) Digital bilateral TV regularization (the 6th iteration, the neighborhood order s is 3, the variance σ of the domain weight is 1.5, $\lambda=0.0005$), (d) ~ (f) Edge detection of (a)~(c) using Canny operator, (g) ~ (i) Edge detection of (a)~(c) using Prewitt operator

Obviously, our magnification approach achieves much better visual quality than other methods. Specifically, bilinear and bicubic interpolation produces severe blocking artifacts. The Tikhonov regularization makes the estimated HR image much blurring. There exists the problem of tradeoff between blurring and edge-preservation as for the Huber-MRF regularization. The digital TV regularization completely fails when the down-sampling factor is 4, although it behaves well in the case of 2. The success of the digital bilateral TV regularization in image magnification demonstrates again the strength of large neighborhoods exploited in image processing. Besides, the proposed filtering-based scheme (22) is fast, in which each iteration costs about 2~3 seconds. Fig. 3 shows the magnification results of a real scanned photo as $q=3$, utilizing bicubic interpolation, digital TV regularization and our proposed approach. Obviously, our proposed approach behaves more robust in enhancing the edges and suppressing the random noises.

5 Conclusions

In this paper, we propose an alternative magnification approach utilizing a filtering-based implementation scheme and novel regularization through coupling bilateral filtering with the digital TV model. The method is simple, fast, and robust to both noise and blocking artifacts. Experiment results demonstrate the effectiveness of our method.

References

1. El-Khamy, S.E., Hadhoud, M.M., Dessouky, M.I., Salam, B.M., El-Samie F.E.: Efficient Implementation of Image Interpolation as An Inverse Problem. *Digital Signal Processing*, Vol. 15 (2005) 137–152
2. Aly, H. A., Dubois, E.: Specification of the Observation Model for Regularized Image Up Sampling. *IEEE Transactions on Image Processing*, Vol. 14, No. 5 (2005) 567–576
3. Schultz, R.R., Stevenson, R.L.: A Bayesian Approach to Image Expansion for Improved Definition. *IEEE Transactions on Image Processing*, Vol. 3, No. 3 (1994) 233–242
4. Blu, T., Th´evenaz, P., Unser, M.: Linear Interpolation Revisited. *IEEE Transactions on Image Processing*, Vol. 13 (2004) 710–719
5. Li, X., Orchard, T.: New Edge-Directed Interpolation. *IEEE Transactions on Image Processing*, Vol. 10 (2001) 1521–1527
6. Carrey, W.K., Chuang, D.B., Hemami, S.S.: Regularity-Preserving Image Interpolation. *IEEE Transactions on Image Processing*, Vol. 8 (1999) 1293–1297
7. Mu˜noz, A., Blu, T., Unser, M.: Least-Squares Image Resizing Using Finite Differences. *IEEE Transactions on Image Processing*, Vol.10, No. 9 (2001) 1365–1378
8. Chan, T.F., Shen, J.H.: Mathematical Models for Local Nontexture Inpaintings. *SIAM J. Appl. Math.*, Vol. 62, No. 3 (2002) 1019–1043
9. Borman, S., Stevenson, R. L.: Super-Resolution for Image Sequences—A Review. In *Proc. IEEE Int. Symp. Circuits and Systems (1998)* 374–378
10. Unser, M., Aldroubi, A., Eden, M.: B-Spline Signal Processing: Part I—Theory. *IEEE Trans. Signal Processing*, Vol. 41, No. 2 (1993) 821–833

11. Unser, M., Aldroubi, A., Eden, M.: B-Spline Signal Processing: Part II—Efficient Design Applications. *IEEE Trans. Signal Processing*, Vol. 41, No. 2 (1993) 834–848
12. Charbonnier, P., Fe`raud, L.B., Aubert, G., Barlaud, M.: Deterministic Edge-preserving Regularization in Computed Imaging. *IEEE Transactions on Image Processing*, Vol. 6, No. 2 (1997) 298–311
13. Tikhonov, A.N., Arsenin V.Y.: *Solutions of Ill-Posed Problems*. New York: Wiley (1977)
14. Chan, T. F.: The Digital TV Filter and Nonlinear Denoising. *IEEE Transactions on Image Processing*, Vol.10, No. 2 (2001) 231–241
15. Farsiu, S., Robinson, M.D., Elad, M., Milanfar, P.: Fast and Robust Multiframe Super Resolution. *IEEE Transactions on Image Processing*, Vol. 13, No.10 (2004) 1327–1344
16. Tomasi, C., Manduchi, R.: Bilateral Filtering for Gray and Color Images. In *Proc. 6th Int. Conf. Computer Vision*, New Delhi, India (1998) 839–846
17. Barash, D.: A Fundamental Relationship between Bilateral Filtering, Adaptive Smoothing, and the Nonlinear Diffusion Equation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 6 (2002) p.844
18. Elad, M.: On the Bilateral Filter and Ways to Improve It. *IEEE Transactions on Image Processing*, Vol. 11, No. 10 (2002) 1141–1151
19. Li, S.Z.: Discontinuous MRF Prior and Robust Statistics: A Comparative Study. *Image and Vision Computing*, Vol.13, No. 3 (1995) 227–233
20. Overton, K.J., Weymouth, T.E.: A Noise Reducing Preprocessing Algorithm. In *Proc. IEEE Computer Science Conf. on Pattern Recognition and Image Processing* (1979) 498–507

An Efficient Post-processing Using DCT Domain Projections Onto Convex Sets

Changhoon Yim*

Department of Internet and Multimedia Engineering,
Konkuk University, Seoul 143-701, Korea
cyim@konkuk.ac.kr

Abstract. Post-processing methods using projections onto convex sets (POCS) have shown good performance for blocking artifact reduction. The iterative operations in POCS require infeasible amount of computations for practical real-time applications. In this paper, we propose an efficient non-iterative post-processing method using DCT domain POCS, namely DPOCS. In DPOCS, the inverse DCT and the forward DCT need not be performed by performing the low-pass filtering (LPF) in the DCT domain. Through the investigation of LPF at each iteration in the conventional POCS, the k -th order LPF is defined that is equivalent to the first order LPF with k iterations. By combining DCT domain filtering and the k -th order LPF, we define k -th order DCT domain LPF. The k -th order DCT domain LPF is performed only once to have the equivalent performance to the conventional POCS method with k iterations. Simulation results show that the proposed DPOCS without iteration gives very close PSNR and subjective quality performance compared to the conventional POCS with iterations, while it requires much less computational complexity. If we take into account typical sparseness in DCT coefficients, the DPOCS method gives tremendous complexity reduction compared to the conventional POCS method. Hence the proposed DPOCS is an attractive method for practical real-time post-processing applications.

1 Introduction

There have been considerable researches to reduce blocking artifacts in block DCT coding methods, which can be classified into two categories [1], [2]. One is variation of transform structure such as interleaved block transform [3] and lapped transform [4]. The other one is postprocessing techniques such as filtering approaches [5] and iterative approaches based on the theory of projections onto convex sets (POCS) [1], [2], [6], [7], [8], [9].

The post-processing techniques are attractive because these would keep the encoder and decoder unchanged [1], [2]. The simplest approach among post-processing techniques would be just low-pass filtering. The simple low-pass filtering might result in over-loss of high frequency components, which would result in over-blurring. On the other hand, POCS approach adopts quantization

* This paper was supported by Konkuk University in 2006.

constraint set (QCS) to ensure the post-processed image located in the same quantization set as the quantized image. The QCS can be simply obtained from the quantized image and the quantization table. The projection onto QCS (simply projection) is performed as clipping operation. The projection operation is effective to prevent post-processed images from diverging. In POCS, low-pass filtering (LPF) is performed for smoothing. The projection and LPF is performed iteratively in POCS. Note that the projection operation is performed in the DCT domain and the LPF is performed in the spatial domain. The IDCT and DCT would be required to go back and forth between the DCT domain and the spatial domain. Hence the following operations would be required at each iteration in POCS: IDCT, LPF, DCT, projection.

Recently, blocking artifact reduction techniques are demanded in practical applications at low bit rates such as video over internet and wireless networks. Even though POCS-based approaches have demonstrated good performance to reduce blocking artifacts, it is infeasible to employ conventional POCS in practical applications, especially in real-time applications. It is essential for blocking artifact reduction method to be simple or reasonably complex for practical applications.

In this paper, we propose an efficient post-processing using DCT domain POCS, namely DPOCS. In conventional POCS, the LPF is performed in the spatial domain. Hence IDCT and DCT would be required in this approach. In DPOCS, the LPF is performed in the DCT domain. Hence IDCT and DCT would not be required. We also investigate LPF at each iteration. For the equivalent LPF without iteration, we define k -th order LPF. By combining DCT domain filtering and k -th order LPF concept, we define k -th order DCT domain LPF (DLPF). In the proposed DPOCS, the projection operation is performed *only once* after the k -th order DLPF is performed *only once*.

The proposed DPOCS method gives very close PSNR and subjective quality performance compared to the conventional spatial domain POCS method, while it requires much less computational complexity. The proposed DPOCS method would be an attractive scheme for blocking artifact reduction in practical real-time post-processing applications.

The organization of this paper is as follows. Section 2 introduces conventional spatial domain POCS as background. In Section 3, we investigate low-pass filtering (LPF) in POCS and define k -th order LPF. Section 4 presents DCT domain separable symmetric linear filtering. In Section 5, we propose the DCT domain POCS. Section 6 presents simulation results. In Section 7, we discuss on complexity comparison between the conventional POCS method and the proposed DPOCS method. Finally, Section 8 provides brief conclusion.

2 Conventional Spatial Domain POCS

Fig. 1 represents the block diagram of conventional POCS [6], [7], [8] *for one iteration*. First, IDCT is performed to obtain spatial domain images from DCT coefficients. Second, spatial domain low-pass filtering (SLPF) is performed for

smoothing images. Third, forward DCT operation need to be performed to obtain DCT coefficients. Fourth, projection operation is performed to satisfy the quantization constraint as clipping operation.

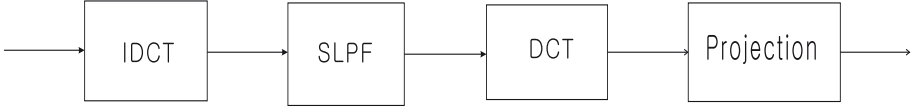


Fig. 1. Block diagram of conventional POCS for one iteration

In conventional spatial domain POCS algorithm, these four operations need to be performed iteratively. In other words, these four operations in Fig. 1 need to be performed k times for k iterations in conventional spatial domain POCS.

3 LPF in POCS

In POCS, low-pass filtering (LPF) is performed in the spatial domain for smoothing. Let $\{x(m, n)\}$ and $\{y(m, n)\}$ be the input image and filtered image, respectively. Let $\{f(m, n)\}$ be the filter coefficients for 2-D LPF. The LPF by 2-D filter coefficients $\{f(m, n)\}$ can be represented as

$$y(m, n) = f(m, n) * x(m, n) = \sum_k \sum_l f(k, l)x(m - k, n - l) \quad (1)$$

where $*$ represents the convolution operation and the range of summation over k and l is given by the support of the filter coefficients $\{f(m, n)\}$. We assume that the 2-D filter is *separable*, that is, $\{f(m, n)\}$ can be factorized as $\{f(m, n)\} = v_m h_n$ for some 1-D filters $\{v_m\}$ and $\{h_n\}$. Let the supports of $\{v_m\}$ and $\{h_n\}$ be $-M \leq m \leq M$ and $-N \leq n \leq N$, respectively.

In POCS, the LPF is performed in each iteration. In k -th iteration, the filtering operation is performed over the filtered output after the $(k - 1)$ -th iteration. Let $y^{(k)}(m, n)$ be the filtered output after the k -th iteration. Then

$$y^{(k)}(m, n) = f(m, n) * y^{(k-1)}(m, n). \quad (2)$$

Let $f^{(1)}(m, n) \equiv f(m, n)$ and $f^{(k)}(m, n) \equiv f^{(1)}(m, n) * f^{(k-1)}(m, n)$. Then it can be shown that

$$y^{(k)}(m, n) = f^{(k)}(m, n) * x(m, n). \quad (3)$$

We call $\{f^{(k)}(m, n)\}$ the k -th order low-pass filter for smoothing. Note that the k -th order low-pass filter $\{f^{(k)}(m, n)\}$ is the equivalent filter with the low-pass filter at the k -th iteration in the conventional POCS method.

The k -th order low-pass filter kernel $\{f^{(k)}(m, n)\}$ is also separable and can be obtained from the 1-D filter coefficients $\{v_m\}$ and $\{h_n\}$. Let $v_m^{(1)} \equiv v_m$,

$v_m^{(k)} \equiv v_m^{(1)} * v_m^{(k-1)}$, $h_n^{(1)} \equiv h_n$, and $h_n^{(k)} \equiv h_n^{(1)} * h_n^{(k-1)}$. Then it can be shown that

$$f^{(k)}(m, n) = v_m^{(k)} h_n^{(k)}. \quad (4)$$

The 2-D filter kernel in [6], which is used in many POCS works [7], [9], is 3×3 with $f(0, 0) = 0.2042$, $f(0, 1) = f(0, -1) = f(1, 0) = f(-1, 0) = 0.1259$, $f(0, 2) = f(0, -2) = f(2, 0) = f(-2, 0) = 0.0751$. This 2-D filter is separable and symmetric, and the corresponding 1-D filter kernel is $v_0 = h_0 = 0.4518$, $v_1 = v_{-1} = h_1 = h_{-1} = 0.2741$.

We can calculate the k -th order 1-D filter coefficients from the given first order 1-D filter coefficients $\{v_m^{(1)}\}$. Table 1 shows the filter coefficients of $\{v_m^{(2)}\}$, $\{v_m^{(5)}\}$, and $\{v_m^{(8)}\}$ from the given $\{v_m^{(1)}\}$. Note that $h_m^{(k)} = v_m^{(k)}$ and $v_{-m}^{(k)} = v_m^{(k)}$ for all k and m .

Table 1. 1-D filter kernels of k -th order filters $\{v_m^{(k)}\}$ for some k

| | $v_m^{(1)}$ | $v_m^{(2)}$ | $v_m^{(5)}$ | $v_m^{(8)}$ |
|-------|-------------|-------------|-------------|-------------|
| v_0 | 0.4518 | 0.3514 | 0.2339 | 0.1870 |
| v_1 | 0.2741 | 0.2477 | 0.1987 | 0.1682 |
| v_2 | 0 | 0.0751 | 0.1203 | 0.1219 |
| v_3 | 0 | 0 | 0.0498 | 0.0705 |
| v_4 | 0 | 0 | 0.0128 | 0.0319 |
| v_5 | 0 | 0 | 0.0015 | 0.0109 |
| v_6 | 0 | 0 | 0 | 0.0027 |
| v_7 | 0 | 0 | 0 | 0.0004 |
| v_8 | 0 | 0 | 0 | 0.0000 |

Fig. 2 shows the frequency response of $\{v_m^{(1)}\}$, $\{v_m^{(2)}\}$, $\{v_m^{(5)}\}$, and $\{v_m^{(8)}\}$. In Fig. 2, we can see that low-pass bandwidth becomes smaller as the order becomes larger. This explains that the filtered image becomes smoother as the iteration number increases in POCS.

4 DCT Domain Separable Symmetric Linear Filtering

We assume that the DCT block size is 8×8 , which is common in DCT-based compression standards. Let $\{x(m, n)\}$ and $\{y(m, n)\}$ be composed of non-overlapping 8×8 matrices $X_{i,j}$ and $Y_{i,j}$, respectively, for block-based matrix form. Let $V \equiv [V_{-1} \ V_0 \ V_1]$ and $H \equiv [H_{-1} \ H_0 \ H_1]$, where

$$V_{-1} = \begin{bmatrix} v_8 & v_7 & \cdots & v_2 & v_1 \\ 0 & v_8 & \ddots & \ddots & v_2 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & v_8 & v_7 \\ 0 & 0 & \cdots & 0 & v_8 \end{bmatrix}, \quad (5)$$

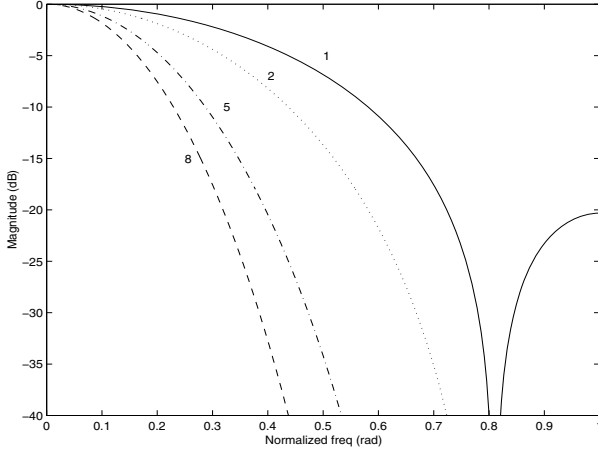


Fig. 2. Frequency response of $\{v_m^{(k)}\}$ for some k

$$V_0 = \begin{bmatrix} v_0 & v_{-1} & \cdots & v_{-6} & v_{-7} \\ v_1 & v_0 & \ddots & \ddots & v_{-6} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & v_0 & v_{-1} \\ v_7 & v_6 & \cdots & v_1 & v_0 \end{bmatrix}, \quad (6)$$

$$V_1 = \begin{bmatrix} v_{-8} & 0 & \cdots & 0 & 0 \\ v_{-7} & v_{-8} & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ v_{-2} & \ddots & \ddots & v_{-8} & 0 \\ v_{-1} & v_{-2} & \cdots & v_{-7} & v_{-8} \end{bmatrix}. \quad (7)$$

The H_{-1} , H_0 , and H_1 matrices are similarly defined as V_{-1} , V_0 , and V_1 , respectively, by replacing v_k to h_k for all k . In this paper, we assume that M and N do not exceed 8 so that the filtering operation for $X_{i,j}$ is only related to the neighboring blocks. The 2-D separable linear filtering can be represented as

$$Y_{i,j} = \sum_{m=-1}^1 \sum_{n=-1}^1 V_m X_{i+m,j+n} H_n^t. \quad (8)$$

Note that $Y_{i,j}$, $X_{i+m,j+n}$, V_m , and H_n are all matrices of size 8×8 .

Let C be the DCT transform matrix. Since the DCT is unitary, $C^{-1} = C^t$. Let $\mathbf{Y}_{i,j}$, $\mathbf{X}_{i+m,j+n}$, \mathbf{V}_m , and \mathbf{H}_n be the DCT of $Y_{i,j}$, $X_{i+m,j+n}$, V_m , and H_n , respectively. Then $\mathbf{Y}_{i,j} = C Y_{i,j} C^t$, $\mathbf{X}_{i+m,j+n} = C X_{i+m,j+n} C^t$, $\mathbf{V}_m = C V_m C^t$,

and $\mathbf{H}_n = CH_nC^t$. It can be shown that $\mathbf{Y}_{i,j}$ can be represented in terms of $\mathbf{X}_{i+m,j+n}$'s as [10], [14]

$$\mathbf{Y}_{i,j} = \sum_{m=-1}^1 \sum_{n=-1}^1 \mathbf{V}_m \mathbf{X}_{i+m,j+n} \mathbf{H}_n^t. \quad (9)$$

We call the matrices \mathbf{V}_m 's and \mathbf{H}_n 's, the DCT domain filtering matrices. Note that the DCT domain filtering matrices \mathbf{V}_m 's and \mathbf{H}_n 's can be pre-calculated given the filter coefficients $\{v_k\}$ and $\{h_l\}$. Equation (9) represents the 2-D linear filtering in the DCT domain.

The separable 2-D linear filtering in the DCT domain can be performed separately in the vertical and in the horizontal directions. The separable 2-D linear filtering in the DCT domain in (9) can be decomposed as [14]

$$\mathbf{Z}_{i,j} = \mathbf{V}_{-1} \mathbf{X}_{i-1,j} + \mathbf{V}_0 \mathbf{X}_{i,j} + \mathbf{V}_1 \mathbf{X}_{i+1,j} \quad (10)$$

$$\mathbf{Y}_{i,j} = \mathbf{Z}_{i,j-1} \mathbf{H}_{-1}^t + \mathbf{Z}_{i,j} \mathbf{H}_0^t + \mathbf{Z}_{i,j+1} \mathbf{H}_1^t \quad (11)$$

where $\mathbf{Z}_{i,j}$ represents the 1-D linear filtered matrix in the vertical direction for the DCT domain input matrix. The separable 2-D linear filtering in the DCT domain as in (10) and (11) would require 3072 multiplications for one 8×8 DCT block.

The 2-D filter kernel in many POCS works [6], [7], [8], [9] is separable and symmetric. For separable symmetric 2-D filter kernel, we can perform the DCT domain 2-D linear filtering with 1536 multiplications for one DCT block [14].

5 Proposed DCT Domain POCS

In this section, we propose the DCT domain POCS (DPOCS). Fig. 3 shows the block diagram of the proposed DPOCS. In Fig. 3, DLPF is the DCT domain low-pass filtering. In Section 3, we define the k -th order LPF for POCS. The k -th order low-pass filtering corresponds to the low-pass filtering at the k -th iteration in POCS. If the k -th order low-pass filter kernel is selected for DLPF, we call it the k -th order DLPF. Note that DCT domain filtering matrices can be pre-calculated for any order. We define the k -th order DPOCS when DLPF is k -th order. The projection operation in Fig. 3 is identical to the projection operation in the conventional POCS. In DPOCS, the projection operation is performed *only once* after the k -th order DLPF operation is performed *only once*. The block diagram in Fig. 3 shows the whole operations for DPOCS, not just for one iteration as in Fig. 1.

The proposed DPOCS method would require much less computations compared to the conventional POCS due to the following reasons. First, the proposed DPOCS method does not require any iteration. Second, the proposed DPOCS method does not require IDCT and DCT operations. Third, DCT coefficients are generally very sparse after the quantization, especially at low bit rates. More detailed discussion on complexity comparison would be presented in Section 7.

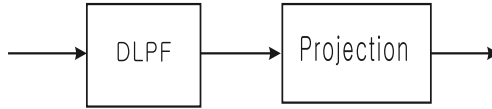


Fig. 3. Block diagram of the proposed DCT domain POCS

6 Simulation Results

In this section, simulation results are presented to evaluate the proposed DPOCS method compared with the conventional spatial domain POCS method. Table 2 shows the quantization table that was used for simulations.

Table 2. Quantization table [6], [7]

| | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 50 | 60 | 70 | 70 | 90 | 120 | 255 | 255 |
| 60 | 60 | 70 | 96 | 130 | 255 | 255 | 255 |
| 70 | 70 | 80 | 120 | 200 | 255 | 255 | 255 |
| 70 | 96 | 120 | 145 | 255 | 255 | 255 | 255 |
| 90 | 130 | 200 | 255 | 255 | 255 | 255 | 255 |
| 120 | 255 | 255 | 255 | 255 | 255 | 255 | 255 |
| 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 |
| 255 | 255 | 255 | 255 | 255 | 255 | 255 | 255 |

Fig. 4 shows the peak signal-to-noise (PSNR) ratio of the post-processed images for ‘Lena’, ‘Peppers’, ‘Barbara’, and ‘Baboon’ images. The horizontal axis represents the number of iterations in the conventional spatial domain POCS and the order DLPF in the proposed DPOCS. The dotted line with ‘+’ points represents the PSNR of the conventional spatial domain POCS, and the solid line with ‘o’ points represents the the PSNR of DPOCS. We compare the conventional POCS and the proposed DPOCS up-to 8 iterations and the corresponding 8-th order DLPF. Simulation results on conventional POCS show that the PSNR values are mostly converged with 8 iterations in previous works [6], [7], [8]. In these results, we can see that the proposed DPOCS method yields very close PSNR performance at each corresponding DLPF order compared to the conventional POCS method at each iteration.

Fig. 5 represents the simulated images for ‘Barbara’ image. Fig. 5(a) represents the original image and Fig. 5(b) represents the quantized image with blocking artifacts. Fig. 5(c) represents the post-processed image using the conventional spatial domain POCS method with 8 iterations. Fig. 5(d) represents the post-processed image by the proposed DPOCS method with the corresponding 8-th order DLPF. These results show that the DPOCS as well as the conventional POCS remove most blocking artifacts successfully while preserving the edge components. From these simulation results, we can see that the proposed DPOCS method gives very close performance compared to the conventional spatial domain POCS method.

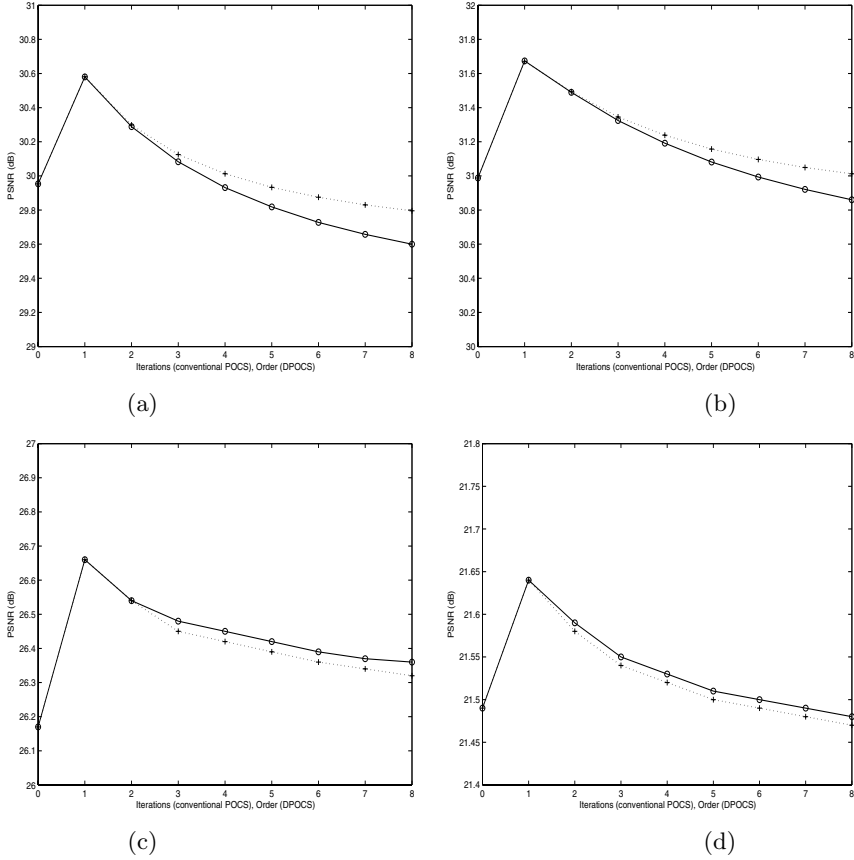


Fig. 4. PSNR of post-processed images versus the number of iterations in spatial domain POCS case and the order in DPOCS case (spatial domain POCS: ‘+’ dotted, proposed DPOCS: ‘o’ solid). (a) Lena (b) Pepper (c) Barbara (d) Baboon.

7 Complexity Comparison

In this section, we discuss on complexity comparison between the conventional POCS method and the proposed DPOCS method. As represented in Fig. 1, the conventional POCS method need to go through 4 steps: IDCT, SLPF, DCT, and projection. These 4 steps need to be performed in each iteration. Hence the complexity of conventional POCS method depends on the number of iterations. First, we consider about SLPF. We assume that 3×3 filter kernel is used as in [6], [7], [9]. If we calculate SLPF as separable 1-D filtering, then it would require 384 multiplications for one 8×8 block. Second, we consider about DCT/IDCT computations. There are many fast algorithms and VLSI architectures for 2-D DCT implementations. The most popular approach for 2-D DCT/IDCT VLSI implementation is row-column method [11]. Typical 2-D DCT as row-column method would require 192 multiplications for one 8×8 block [12]. The same number of multiplications

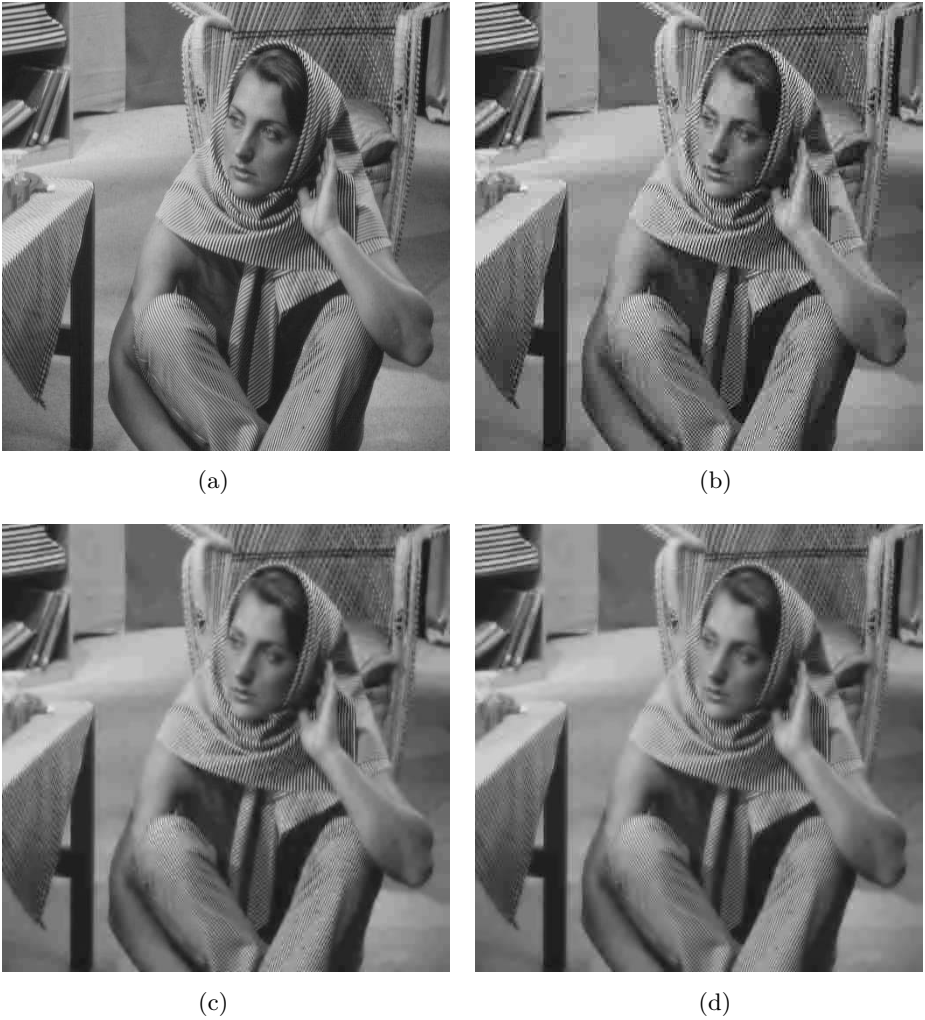


Fig. 5. Simulated images for ‘Barbara’ image. (a) Original image (b) Quantized image (c) Post-processed image using the conventional spatial domain POCS with 8 iterations (d) Post-processed image using the proposed DPOCS with the corresponding 8-th order DLPF.

would be required for IDCT. Overall it would require 768 multiplications for one iteration in conventional POCS method. Let m represent the number of multiplications for one 8×8 block. If we perform k iterations in conventional POCS method, then $m = 768k$. Table 3 shows the number of operations in conventional POCS method for 2, 4, 6, and 8 iterations. In conventional POCS, the number of operations increases linearly as the number of iterations increases.

Now we consider about the complexity of the proposed DPOCS method. The number of multiplications for DLPF depends on the sparseness of DCT coefficients. Actually high percentage of coefficients in a typical DCT block have zero

Table 3. The number of multiplications in conventional POCS method for 2, 4, 6, and 8 iterations

| Iterations | Number of multiplications |
|------------|---------------------------|
| 2 | 1536 |
| 4 | 3072 |
| 6 | 4608 |
| 8 | 6144 |

Table 4. The number of multiplications in the proposed DPOCS method for 2, 4, 6, and 8 orders

| Orders | Number of multiplications without sparseness | Number of multiplications with sparseness |
|--------|--|---|
| 2 | 1536 | 384 |
| 4 | 1536 | 384 |
| 6 | 1536 | 384 |
| 8 | 1536 | 384 |

values after the quantization, especially at low bit rate compression. Let α and β denote the ratio of non-zero coefficients in input DCT matrices $\mathbf{X}_{i,j}$ and 1-D filtered DCT matrices $\mathbf{Z}_{i,j}$, respectively. The number of multiplications m for DCT domain linear filtering as in (10) and (11) would be

$$m = 768\alpha + 768\beta. \quad (12)$$

The sparseness of DCT blocks is obtained after the quantization. The amount of sparseness is dependent on many factors such as image characteristics and quantization operation. In [13], they define a DCT block as sparse if only its 4×4 upper left quadrant coefficients are nonzero. In this case, $\alpha = \beta = 0.25$, and the number of multiplications in (12) is 384. We call this case as typical sparse case. Actually α and β values would be smaller than 0.25 in low bit rate compression. Table 4 shows the number of multiplications in the proposed DPOCS method for orders 2, 4, 6, and 8. The number of multiplications without sparseness is 1536 for any order, which is the same as the 2 iteration case in the conventional POCS. If we compare the number of multiplications between the DPOCS without sparseness in order 8 and the conventional POCS in iteration 8, it is reduced from 6144 to 1536, which is 25% of multiplications. If we take into account the sparseness, the number of multiplications would be further reduced. The number of multiplications for DPOCS with typical sparseness is 384 in any order. If we compare the number of multiplications between the D-POCS with typical sparseness in order 8 and the conventional POCS in iteration 8, it is reduced from 6144 to 384, which is 6.3% of multiplications. Note that both PSNR and subjective quality performance is almost the same at each iteration in conventional POCS and the corresponding order in DPOCS.

8 Conclusion

We proposed the DCT domain POCS method (DPOCS) for post-processing images with blocking artifacts. In DPOCS, the low-pass filtering (LPF) is per-

formed in the DCT domain. In this way, we removed the IDCT and DCT from POCS. We also investigated LPF at each iteration, and defined k -th order LPF for the LPF to be equivalent without iteration. By combining DCT domain filtering and k -th order LPF concept, we defined k -th order DCT domain LPF (DLPF). In the proposed DPOCS, the projection operation is performed only once after the k -th order DLPF.

Simulation results show that the proposed DPOCS method without iteration give almost the same result as the conventional POCS method with iterations. The proposed DPOCS method requires much less computational complexity compared to the conventional POCS method. The proposed DPOCS method would be an attractive scheme for blocking artifact reduction in practical post-processing applications.

References

1. Paek, H., Kim, R.-C., Lee, S. U.: On the POCS-based postprocessing technique to reduce the blocking artifacts in transform coded images. *IEEE Trans. Circuits Syst. Video Technol.* **8(3)** (1998) 358–367
2. Park, S. H., Kim, D. S.: Theory of projection onto narrow quantization constraint set and its applications. *IEEE Trans. Image Processing* **8(10)** (1999) 1361–1373
3. Pearson, D., Whybray, M.: Transform coding of images using interleaved blocks. *Proc. Inst. Elect. Eng.* **131** (1984) 466–472
4. Malvar, H. S., Staelin, D. H.: The LOT: Transform coding without blocking artifacts. *IEEE Trans. Acoust., Speech, Signal Processing* **37** (1989) 553–559
5. Ramamurthi, B., Gersho, A.: Nonlinear space-variant postprocessing of block coded images. *IEEE Trans. Acoust., Speech, Signal Processing.* **34** (1986) 1258–1267
6. Zakhor, A.: Iterative procedure for reduction of blocking effects in transform image coding. *IEEE Trans. Circuits Syst. Video Technol.* **2(1)** (1992) 91–95
7. Yang, Y., Galatanos, N. P., Katsaggelos, A. K.: Regularized reconstruction to reduce blocking artifacts of block discrete cosine transform compressed image. *IEEE Trans. Circuits Syst. Video Technol.* **3(6)** (1993) 421–432
8. Yang, Y., Galatanos, N. P., Katsaggelos, A. K.: Projection-based spatially adaptive reconstruction of block-transform compressed images. *IEEE Trans. Image Processing* **4(7)** (1995) 896–908
9. Jeong, Y., Kim, I., Kang, H.: A practical projection-based postprocessing of block-coded images with fast convergence rate. *IEEE Trans. Circuits and Syst. Video Technol.* **10(4)** (2000) 617–623
10. Merhav, N., Kresch, R.: Approximate convolution using DCT coefficient multipliers. *IEEE Trans. Circuits Syst. Video Technol.* **8** (1998) 468–476
11. Madisetti, A., Wilson, JR., A. N.: A 100 MHz 2-D DCT/IDCT processor for HDTV applications. *IEEE Trans. Circuits Syst. Video Technol.* **5** (1995) 158–165
12. Cho, N. I., Lee, S. U.: Fast algorithm and implementation of 2-D discrete cosine transform. *IEEE Trans. Circuits and Syst.* **38(3)** (1991) 297–305
13. Kresch, R., Merhav, N.: Fast DCT domain filtering using the DCT and the DST. *IEEE Trans. Image Processing* **8(6)** (1999) 378–384
14. Yim, C.: An efficient method for DCT-domain separable symmetric 2-D linear filtering. *IEEE Trans. Circuits Syst. Video Technol.* **14(4)** (2004) 517–521

Rank-Ordered Differences Statistic Based Switching Vector Filter

Guillermo Peris-Fajarnés, Bernardino Roig*, and Anna Vidal

Universidad Politécnica de Valencia, E.P.S. de Gandia, Carretera Nazaret-Oliva s/n, 46730 Grau de Gandia (València), Spain
broig@mat.upv.es

Abstract. In this paper a new switching vector filter for impulsive noise removal in color images is proposed. The new method is based on a recently introduced impulse detector named *Rank-Ordered Differences Statistic* (ROD) which is adapted to be used in color images. The switching scheme between the identity operation and the Arithmetic Mean Filter is defined using the mentioned impulse detector. Extensive experiments show that the proposed technique is simple, easy to implement and significantly outperforms the classical vector filters presenting a competitive performance respect to recent impulsive noise vector filters.

1 Introduction

During image formation, acquisition, storage and transmission many types of distortions limit the quality of digital images. Transmission errors, periodic or random motion of the camera system during exposure, electronic instability of the image signal, electromagnetic interferences, sensor malfunctions, optic imperfections or aging of the storage material, all disturb the image quality by the introduction of noise and other undesired effects. In particular, a kind of noise which is mainly introduced during the transmission process is the so-called impulsive noise [9,13]. In this context, the filtering process becomes an essential task to avoid possible drawbacks in the subsequent image processing steps.

When the images are contaminated with impulsive noise the switching approaches are widely used due to their sufficient performance and proven computational simplicity. On the basis of the classical vector filters as the *Arithmetic Mean Filter* (AMF) [13], the *Vector Median Filter* (VMF) [2], the *Basic Vector Directional Filter* (BVDF) [17,18], or the *Distance Directional Filter* (DDF) [5], the switching approaches aim at selecting a set of pixels of the image to be filtered leaving the rest of the pixels unchanged. A series of methods for selecting the noise-likely pixels have been proposed to date [1,6,7,8,11,12,14,15,16]. For example, the *Adaptative Vector Median Filter* (AVMF) is proposed in [8], the *Fast Impulsive Vector Filter* (FIVF) in [12], and the *Peer Group Switching Arithmetic Mean Filter* (PGSAMF) in [15]. In [1] the authors propose to determine if the vector in consideration is likely to be noisy using cluster

* Corresponding author.

analysis. The standard deviation, the sample mean and various distance measures are used in [6,10] to form the adaptive noise detection rule. [8] proposes to use an statistical test and [3] uses statistical confidence limits. In [11,14,15] a neighborhood test is applied. Then, the filtering operation is performed only when it is necessary. In a similar way, in [19] a genetic algorithm is used to decide in each image position to perform the *Vector Median Filter* operation, the *Basic Vector Directional Filter* operation or the identity operation. In [7,12,16], it is proposed to privilege the central pixel in each filtering window to reduce the number of unnecessary substitutions.

In this paper, a novel approach to the detection of impulsive noise in color images is presented. The detection of corrupted pixels is performed by calculating a generalization to color images of the statistic developed in [4] which will be called *Rank-Ordered Differences Statistic (ROD)*. The function of this impulse detection mechanism is to check each pixel in order to find out whether it is distorted or not. It will be seen that the magnitude of the *Rank-Ordered Differences Statistic* serves as an indicator whether the pixel is corrupted by impulsive noise or is undisturbed by the noise process. When the ROD statistic indicates corruption, the switching filter performs the *Arithmetic Mean Filter* on the corrupted pixels, while the noise-free pixels are left unaltered in order to avoid excessive distortion.

In the following, Section 2 explains the *Rank-Ordered Differences Statistic*. The proposed filtering scheme is described in Section 3. Experimental results, performance comparison and discussion are included in Section 4. Finally, Section 5 presents the conclusions.

2 Rank-Ordered Differences Statistic for Color Images

Attending to its location in the image, we can consider for each pixel \mathbf{x} a $n \times n$ window $\Omega_{\mathbf{x}}$ centered at \mathbf{x} , and we identify each pixel $\mathbf{y} \in \Omega_{\mathbf{x}}$ with its RGB color vector $(y(1), y(2), y(3)) \in X^3$ where $X = \{0, 1, \dots, 255\}$. Denote by $d_{\mathbf{x},\mathbf{y}}^1$ and $d_{\mathbf{x},\mathbf{y}}^2$ the city-block distance and the euclidean distance between the color vectors \mathbf{x} and \mathbf{y} in $\Omega_{\mathbf{x}}$, respectively.

Denote by $\Omega_{\mathbf{x}}^0$ the set of neighbors of \mathbf{x} in the window $\Omega_{\mathbf{x}}$, excepting \mathbf{x} , i.e. $\Omega_{\mathbf{x}}^0 = \Omega_{\mathbf{x}} - \{\mathbf{x}\}$. Fixed $k \in \{1, 2\}$, we compute the distances $d_{\mathbf{x},\mathbf{y}}^k$ for all $\mathbf{y} \in \Omega_{\mathbf{x}}^0$, so we obtain a collection of non-negative real values $r_j(\mathbf{x})$, not necessarily distinct, which we order in an ascending sequence

$$r_1(\mathbf{x}) \leq r_2(\mathbf{x}) \leq \dots \leq r_{n^2-1}(\mathbf{x}) \quad (1)$$

(Roughly speaking, $r_i(\mathbf{x})$ is the i th smallest $d_{\mathbf{x},\mathbf{y}}^k$ for $\mathbf{y} \in \Omega_{\mathbf{x}}^0$).

Now, fixed a positive integer $m \leq n^2 - 1$, the m rank order difference statistic (ROD $_m$) is defined as follows.

$$\text{ROD}_m(\mathbf{x}) = \sum_{i=1}^m r_i(\mathbf{x}). \quad (2)$$

The ROD statistic here defined is a generalization to RGB color images of the ROAD presented in [4] for gray scale images. In this paper, we only consider $m = 4$, and for simplicity we set $\text{ROD}(\mathbf{x}) = \text{ROD}_4(\mathbf{x})$.

The ROD statistic provides a measure of how close a pixel is to its four most similar neighbors in $\Omega_{\mathbf{x}}^0$ attending to their RGB color vectors. The logic underlying of the statistic is that unwanted impulses will vary greatly in intensity in one or more colors from most of their neighboring pixels, whereas most pixels composing the actual image should have at least some of their neighboring pixels of similar intensity in each color, even pixels on an edge in the image. Thus, noise-free pixels has a significantly lower ROD value than corrupted pixels.

3 Proposed Filtering Method

The proposed filtering scheme, which will be called from now on *Rank-Ordered Differences Switching Arithmetic Mean Filter* (RODSAMF), performs in two steps. In the first step, the detection process of *corrupted* pixels is performed and, in the second step, each pixel detected as *corrupted* is replaced by the output of the *Arithmetic Mean Filter* performing over its *uncorrupted* pixels neighbors.

The noise detection process is done as follows.

1. For each pixel \mathbf{x} of the color image consider a $\Omega_{\mathbf{x}}^0$ window and compute the $\text{ROD}(\mathbf{x})$ statistic for $\Omega_{\mathbf{x}}^0$ with the chosen distance d^k ($k \in \{1, 2\}$).
2. If $\text{ROD}(\mathbf{x})$ is less than a fixed quantity d_R , then mark the pixel \mathbf{x} as *uncorrupted*.
3. Elsewhere, mark \mathbf{x} as *corrupted*.

The filtering operation is done on the basis of the result of the detection process. If y_{RODSAMF} denotes the output of the filtering operation, \mathbf{x} is the central pixel in $\Omega_{\mathbf{x}}$ and AMF_{out} is the output of the AMF performing (imitating Smolka [14]) over the *non-corrupted* pixels in $\Omega_{\mathbf{x}}^0$, then, we can design the following switching algorithm:

$$y_{\text{RODSAMF}} = \begin{cases} \mathbf{x} & \text{if } \mathbf{x} \text{ is } \textit{uncorrupted} \\ \text{AMF}_{\text{out}} & \text{if } \mathbf{x} \text{ is } \textit{corrupted} \end{cases} \quad (3)$$

The *Arithmetic Mean Filter* is applied because of its computational simplicity, however other filters as the *Vector Median Filter* could be applied in the above conditions, as well.

4 Experimental Results

In this section, several images have been considered to evaluate the performance of the proposed filter explained in the previous Section. The impulsive noise model for the transmission noise, as defined in [13], has been used to add noise to the images of Babbon, Lenna and Peppers. In the following we denote by p the noise intensity of the image and it will be given as a probability of noise appearance in the image. We have chosen in our experiences values $p \in \{0.05, 0.1, 0.2, 0.3\}$. In view of a better visualization, a detail of the previous images is considered. As in [4], here we have select $n = 3$ because the corresponding $\Omega_{\mathbf{x}}^0$ is the least window that contains the four nearest neighbors of \mathbf{x} .

Table 1. Performance comparison in terms of NCD (10^{-2}) using the Baboon detail image

| Filter | $p = 0.05$ | $p = 0.1$ | $p = 0.2$ | $p = 0.3$ |
|----------------|------------|-----------|-----------|-----------|
| AMF [13] | 8.5056 | 9.8406 | 12.7770 | 15.9292 |
| VMF [2] | 6.0155 | 6.1204 | 6.3798 | 6.7354 |
| BVDF [18] | 6.1232 | 6.5440 | 7.0415 | 7.4589 |
| DDF [5] | 5.3678 | 5.6541 | 6.0996 | 6.6314 |
| FIVF [12] | 1.6295 | 2.6733 | 4.0031 | 5.1747 |
| AVMF [8] | 1.9899 | 2.4758 | 4.0463 | 6.9469 |
| PGSAMF [15] | 1.2163 | 2.0720 | 3.6963 | 5.5190 |
| RODSAMF- d^1 | 1.3750 | 2.5933 | 4.1569 | 5.9039 |
| RODSAMF- d^2 | 1.0668 | 2.0082 | 3.4923 | 5.2020 |

Table 2. Performance comparison in terms of NCD (10^{-2}) using the Lenna detail image

| Filter | $p = 0.05$ | $p = 0.1$ | $p = 0.2$ | $p = 0.3$ |
|----------------|------------|-----------|-----------|-----------|
| AMF [13] | 5.7908 | 7.7173 | 10.9955 | 13.4624 |
| VMF [2] | 2.7293 | 2.8567 | 3.0753 | 3.3286 |
| BVDF [18] | 2.6464 | 2.7837 | 3.2616 | 3.7658 |
| DDF [5] | 2.5758 | 2.7340 | 3.0946 | 3.5190 |
| FIVF [12] | 0.5260 | 0.8971 | 1.7809 | 2.4939 |
| AVMF [8] | 0.7056 | 0.9622 | 2.1853 | 3.7892 |
| PGSAMF [15] | 0.4278 | 0.8296 | 1.9300 | 3.0338 |
| RODSAMF- d^1 | 0.6771 | 1.1971 | 2.2614 | 3.3583 |
| RODSAMF- d^2 | 0.4444 | 0.8719 | 1.7566 | 2.7564 |

Table 3. Performance comparison in terms of NCD (10^{-2}) using the Peppers detail image

| Filter | $p = 0.05$ | $p = 0.1$ | $p = 0.2$ | $p = 0.3$ |
|----------------|------------|-----------|-----------|-----------|
| AMF [13] | 5.5300 | 7.2128 | 9.2101 | 11.6675 |
| VMF [2] | 2.8054 | 2.9228 | 3.0679 | 3.3538 |
| BVDF [18] | 2.6198 | 2.7605 | 3.0748 | 3.9547 |
| DDF [5] | 2.5129 | 2.6486 | 2.8638 | 3.5977 |
| FIVF [12] | 0.6858 | 0.8401 | 1.2901 | 2.3454 |
| AVMF [8] | 0.6893 | 0.7670 | 1.9624 | 4.0513 |
| PGSAMF [15] | 0.5911 | 1.0097 | 1.5203 | 3.3459 |
| RODSAMF- d^1 | 0.8829 | 1.2568 | 2.0836 | 3.5640 |
| RODSAMF- d^2 | 0.6216 | 0.9844 | 1.5112 | 2.8268 |

In order to assess the performance of the proposed filters, the *Normalized Color Difference* (NCD) [13] have been used due to it properly approaches human perception. This objective quality measure is defined as follows

$$NCD_{Lab} = \frac{\sum_{i=1}^N \sum_{j=1}^M \Delta E_{Lab}}{\sum_{i=1}^N \sum_{j=1}^M E_{Lab}^*} \quad (4)$$

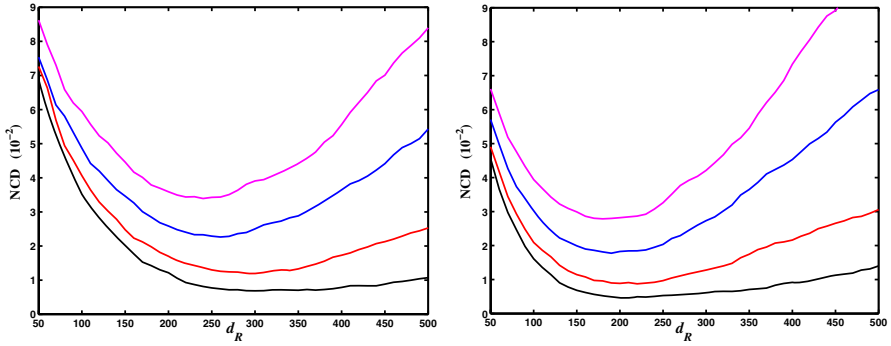


Fig. 1. Dependence of the NCD on the parameter d_R for the Lenna detail image with d^1 (left) and d^2 (right) distances. The different graphs correspond to $p = 0.05, 0.1, 0.2$ and 0.3 in ascending order.



Fig. 2. Results for the Baboon detail image. From left to right and from top to bottom: original image, image with impulsive noise with $p = 0.2$, images filtered with: RODSAMF- d^1 , RODSAMF- d^2 , AMF, VMF, BVDF, DDF, FIVF, AVMF, and PGSAMF.

where M, N are the image dimensions and $\Delta E_{Lab} = [(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2]^{\frac{1}{2}}$ denotes the perceptual color error and $E_{Lab}^* = [(L^*)^2 + (a^*)^2 + (b^*)^2]^{\frac{1}{2}}$ is the *norm* or *magnitude* of the original image color vector in the $L^*a^*b^*$ color space.

Experiments varying the values of d_R have been carried out to determine the best filter performance. We have seen that the value of d_R that minimizes the value of NCD

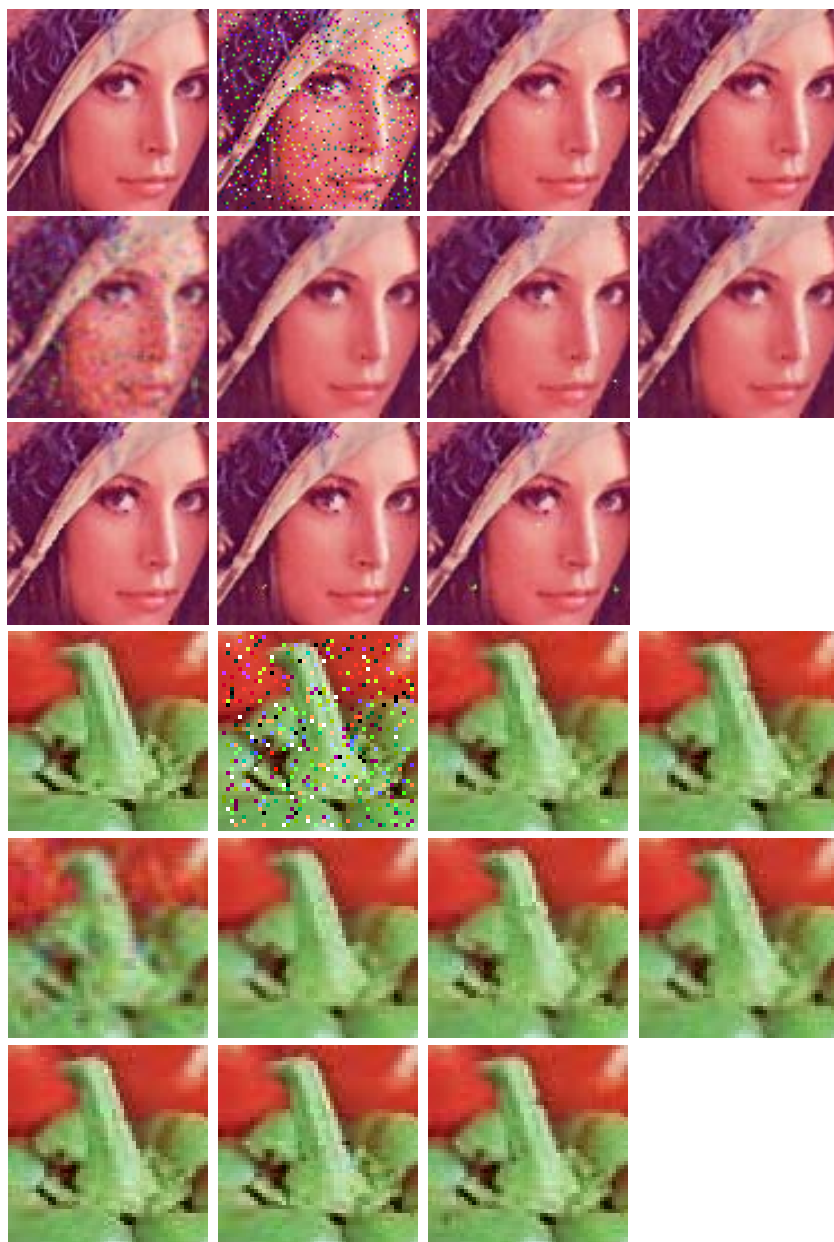


Fig. 3. Results for the Lenna and Peppers detail images. From left to right and from top to bottom: original image, image with impulsive noise with $p = 0.2$, images filtered with: RODSAMF- d^1 , RODSAMF- d^2 , AMF, VMF, BVDF, DDF, FIVF, AVMF, and PGSAMF.

lies in $[220, 300]$ for the city-block distance, and a good selection of d_R for different values of p is $d_R = 260$. For the euclidean distance the value of d_R that minimizes the value of NCD lies in the interval $[160, 240]$, and a good selection of d_R for different values of p is $d_R = 200$. We also have seen for this filter that the euclidean distance performs better than the city-block. Figure 1 shows those results for the Lenna detail image.

Experimental results are shown in Tables 1-3 for the different values of p and for the filters considered in the comparisons. We also show for comparison reasons the results with the well-known filters AMF, VMF, BVDF and DDF, and with the more recently developed filters FIVF, AVMF, PGSAMF. Figures 2 and 3 show the filtering efficiency for the images considered when $p = 0.2$.

It can be seen that the RODSAMF- d^2 performs much better than AMF, VMF, BVDF and DDF, and performs better than AVMF and PGSAMF when the noise increase. It also performs better than FIVF for images presenting texture areas and fine details and it is competitive in the rest of the cases.

5 Conclusions

In order to detect impulsive noise pixels in RGB color images, we introduce the ROD statistic based in some neighborhood of a pixel. This statistic has a significantly lower ROD value for noise-free pixels than for noise impulsive pixels.

The proposed filtering scheme of this paper, RODSAMF, performs in two steps. First, the detection of *corrupted* pixels is performed in function of the ROD value, and second, each pixel detected as *corrupted* is replaced by the output of the *Arithmetic Mean Filter* performing over its *uncorrupted* pixels neighbors.

The RODSAMF with the euclidean distance in the RGB space performs better than with the city-block distance, and much better than the well-known classic filters. It also has a competitive performance in front of more recently developed filters.

The proposed technique is easy to implement and it preserves fine details.

References

1. H. Allende, J. Galbiati, "A non-parametric filter for image restoration using cluster analysis", *Pattern Recognition Letters*, vol. 25 num. 8, pp. 841–847, 2004.
2. J. Astola, P. Haavisto, Y. Neuvo, "Vector Median Filters", *Proc. IEEE*, vol. 78 num. 4, pp. 678–689, 1990.
3. J. Camacho, S. Morillas, P. Latorre, "Efficient impulsive noise suppression based on statistical confidence limits", *Journal of Imaging Science and Technology*, to appear.
4. R. Garnett, T. Huegerich, C. Chui, W. He, "A universal noise removal algorithm with an impulse detector", *IEEE Transactions on Image Processing*, vol. 14 num. 11, 1747-1754, 2005.
5. D.G. Karakos, P.E. Trahanias, "Generalized multichannel image-filtering structure", *IEEE Transactions on Image Processing*, vol. 6 num. 7, pp. 1038–1045, 1997.
6. R. Lukac, "Adaptive vector median filtering", *Pattern Recognition Letters*, vol. 24 num. 12, pp. 1889–1899, 2003.

7. R. Lukac, "Adaptive Color Image Filtering Based on Center Weighted Vector Directional Filters", *Multidimensional Systems and Signal Processing*, vol. 15, pp. 169–196, 2004.
8. R. Lukac, K.N. Plataniotis, A.N. Venetsanopoulos, B. Smolka, "A Statistically-Switched Adaptive Vector Median Filter", *Journal of Intelligent and Robotic Systems*, vol. 42, pp. 361–391, 2005.
9. R. Lukac, B. Smolka, K. Martin, K.N. Plataniotis, A.N. Venetsanopoulos, "Vector Filtering for Color Imaging", *IEEE Signal Processing Magazine, Special Issue on Color Image Processing*, vol. 22 num. 1, pp. 74–86, 2005.
10. R. Lukac, B. Smolka, K.N. Plataniotis, A.N. Venetsanopoulos, "Vector sigma filters for noise detection and removal in color images", *Journal of Visual Communication and Image Representation*, vol. 17 num. 1, pp. 1–26, 2006.
11. Z. Ma, D. Feng, H.R. Wu, "A neighborhood evaluated adaptive vector filter for suppression of impulsive noise in color images", *Real-Time Imaging*, vol. 11 num. 5-6, pp. 403–416, 2005.
12. S. Morillas, V. Gregori, G. Peris-Fajarnés, P. Latorre, "A fast impulsive noise color image filter using fuzzy metrics", *Real-Time Imaging*, vol. 11 num. 5-6, pp. 417–428, 2005.
13. K.N. Plataniotis, A.N. Venetsanopoulos, *Color Image processing and applications*, Springer-Verlag, Berlin, 2000.
14. B. Smolka, A. Chydzinski, "Fast detection and impulsive noise removal in color images", *Real-Time Imaging*, vol. 11 num. 5-6, pp. 389–402, 2005.
15. B. Smolka, K.N. Plataniotis, "Ultrafast technique of impulsive noise removal with application to microarray image denoising", *Lecture Notes in Computer Science*, vol. 3656, pp. 990–997, 2005.
16. B. Smolka, K.N. Plataniotis, A. Chydzinski, M. Szczepanski, A.N. Venetsanopoulos, K. Wojciechowski, "Self-adaptive algorithm of impulsive noise reduction in color images", *Pattern Recognition*, vol. 35 num. 8, pp. 1771–1784, 2002.
17. P.E. Trahanias, D. Karakos, A.N. Venetsanopoulos, "Directional processing of color images: theory and experimental results", *IEEE Transactions on Image Processing*, vol. 5 num. 6, pp. 868–880, 1996.
18. P.E. Trahanias, A.N. Venetsanopoulos, "Vector directional filters-a new class of multichannel image processing filters", *IEEE Transactions on Image Processing*, vol. 2 num. 4, pp. 528–534, 1993.
19. H.H. Tsai, P.T. Yu, "Genetic-based fuzzy hybrid multichannel filters for color image restoration", *Fuzzy Sets and Systems*, vol. 114 num. 2, pp. 203–224, 2000.

Mathematical Analysis of “Phase Ramping” for Super-Resolution Magnetic Resonance Imaging

Gregory S. Mayer and Edward R. Vrscay

Department of Applied Mathematics
Faculty of Mathematics
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1
gsmayer@uwaterloo.ca, ervrscay@uwaterloo.ca

Abstract. Super-resolution image processing algorithms are based on the principle that repeated imaging together with information about the acquisition process may be used to enhance spatial resolution. In the usual implementation, a series of low-resolution images shifted by typically subpixel distances are acquired. The pixels of these low-resolution images are then interleaved and modeled as a blurred image of higher resolution and the same field-of-view. A high-resolution image is then obtained using a standard deconvolution algorithm. Although this approach has been applied in magnetic resonance imaging (MRI), some controversy has surfaced regarding the validity and circumstances under which super-resolution may be applicable. We investigate the factors that limit the applicability of super-resolution MRI.

1 Introduction

Increasing the spatial resolution in magnetic resonance imaging (MRI) has most frequently been performed by optimizing the capabilities of the imaging hardware. However, any improvement in the resolving power of MRI would further increase the efficacy of this versatile modality. Recently, new approaches to resolution enhancement, based on the methodologies of super-resolution (SR) imaging, have been proposed and show promise in increasing the spatial resolution of MRI [5,6,13,19,21].

Super-resolution (SR) image processing is concerned with the enhancement of spatial resolution of images. For example, one may wish to use a number of low resolution images (of various positions and possibly differing resolutions) to produce a higher resolution image of an object. In the classic SR procedure, multiple realizations are collected with a moving field of view (FOV). These multiple images are then used as input to a post-processing reconstruction algorithm that enhances the resolution of the image, overcoming limitations in the system hardware. Readers are referred to the following standard references on the subject [1,3,11,28]. This paper deals with the super-resolution problem in MRI.

One of the first SRMRI papers [22] that was published was received with controversy. On a qualitative level, results in this article, by Yeshurun and Peled, showed that spatial resolution could be enhanced by acquiring multiple MR images and using a SR technique. However, in [27] it was stated that these results were not reliable. For frequency and phase encoded data, which Yeshurun and Peled implemented (we will describe these encoding methods briefly in the next section), it was noted in [27] that there could be no new information present in each additional image acquired beyond the first image. Therefore, there was no justification for acquiring multiple data sets, and the results obtained in [22] could have been reproduced with a standard interpolation scheme. In response, Yeshurun and Peled agreed with this argument [23]. As a result all subsequent SRMRI papers have avoided using frequency and phase encoded data [5,6,13,19,21]. The purpose of this paper is to revisit the arguments made in [27] and [23] in order to provide a more detailed and mathematical analysis of the basis for super-resolution algorithms when applied to these two encoding schemes, often used in MRI.

There exists an enormous amount of research on resolution enhancement methods in MRI that employ information from only a single acquisition (reviewed extensively in [14,15]). However, we would expect (or hope) that by using more data in an SR approach, more information could be incorporated to increase the image resolution. If this were always true, then SRMRI could have a significant edge over single acquisition techniques. Otherwise, as pointed out in [27], there would be an insufficient justification for the increased scan time needed to acquire more data. It is therefore of great importance to determine any conditions under which additional data sets can yield more information. To the best of our knowledge, this paper represents a first step in this direction. We attempt to analyze mathematically the validity of using multiple acquisitions to enhance spatial resolution in magnetic resonance imaging. In particular, we investigate the possible role of frequency shifting, or “phase ramping,” in performing super-resolution MRI. We shall, in fact, show that the amount of independent information that each data set holds is related to the spatial shift applied to the original data.

2 Basic Procedures and Equations in MRI

Let us first briefly outline some typical procedures of acquiring a signal in magnetic resonance imaging. Firstly, MRI data is produced in the form of a continuous analog signal in frequency space. The region of frequency space often takes the form of a rectangular domain, typically centered at the origin and bounded by lines parallel to the coordinate axes.

One of the most common acquisition strategies in MRI is the so-called *2D spin-echo sequence*. In this method, the two image directions in k -space are referred to as the *readout* and *phase encode* directions. The MRI data is usually acquired along a set of trajectories in the readout direction, while assuming a set of discrete values in the phase encode direction. In the readout direction, the signal briefly exists as a continuous entity before it is discretized and used to produce the final image.

The *slice-encoding scheme* is another commonly used technique to spatially encode the measured raw data. SRMRI research has been applied almost exclusively in the slice encoding direction [5,6,13,19,21]. However, in this paper, we shall discuss super-resolution as applied to the method of *frequency encoding*.

For simplicity of notation and presentation, we consider only one-dimensional MRI procedures in this paper. We shall assume that the object of concern is located within the interval $[-R, R]$ on the x -axis. It is the *proton density* of the object, to be denoted as $\rho(x)$ for $x \in [-R, R]$, to which the magnetic resonance spectrometer responds. The fundamental basis of MRI is that different portions of the object, for example, different organs or tissues, possess different proton densities. Visual representations of these differing densities then account for the magnetic resonance “images” of an object.

The (spatially) linearly varying magnetic field in the magnetic resonance spectrometer produces a complex-valued signal $s(k)$ of the real-valued frequency parameter k . The relation between $s(k)$ and the proton density function $\rho(x)$ may be expressed as follows [10,15,9]:

$$s(k) = \int_{-\infty}^{+\infty} \rho(x) \exp(-2\pi i k x) dx. \quad (1)$$

In other words, $s(k)$ is the Fourier transform of $\rho(x)$. If life were ideal, then $\rho(x)$ could be reconstructed from $s(k)$ by simple inverse Fourier transformation. However, there are a number of complications. For example, since the object being imaged has a finite size, its Fourier transform would have to have an infinite support. In practice, however, the frequency response $s(k)$ is obtained for only a finite range of k values. In what follows, we shall assume that the domain over which the signal is measured is a symmetric interval, i.e., $k \in [-k_{max}, k_{max}]$. Following a series of standard post-processing steps to be discussed below, a variety of reconstruction algorithms allow us to estimate $\rho(x)$.

There is also the problem that any practical algorithm must work with discrete data. As such, the analog signal $s(k)$ must be converted into a discrete series. This discretization is accomplished in several steps. Firstly, the signal $s(k)$ must be convolved with an anti-aliasing filter $\psi(k)$ to reduce any aliasing that may occur from sampling. An analog-to-digital converter extracts values of the (frequency) signal at integer multiples of the sampling period, to be denoted as Δk . The entire process may be modelled as follows:

$$L_1(k) = \text{III}\left(\frac{k}{\Delta k}\right) \int_{-k_{max}}^{k_{max}} s(\kappa) \psi\left(\frac{1}{K}(k - \kappa)\right) d\kappa, \quad k \in [-k_{max}, k_{max}], \quad (2)$$

where K represents the (frequency) width of the low-pass filter ψ and $\text{III}(x)$ denotes the so-called *Shah function*, defined as [2]

$$\text{III}(x) = \sum_{n=-\infty}^{\infty} \delta(x - n), \quad (3)$$

where $\delta(x)$ denotes the standard Dirac delta function.

The above integration is performed electronically in a continuous manner on the analog signal. The discrete series $L_1(k)$ represents the filtered and sampled signal that is used in the subsequent digital imaging process. A standard inverse discrete Fourier transform is then applied to $L_1(k)$ to produce a discrete data set $l_1(n\Delta x)$, which is the digital MRI “image” that provides a visual representation of the the proton density function $\rho(x)$ throughout the object.

Finally, we mention that in normal medical imaging applications, e.g., commercial MRI machines, one does not have the luxury of being able to alter the sampling size Δk and anti-aliasing filter $\psi(k)$. This is the assumption that we make in the discussion below.

3 Super-Resolution MRI in the Spatial Domain

Currently, the standard approach to performing super-resolution in MRI is to “fuse” two or more sets of lower resolution spatial data, i.e. magnetic resonance “images” of the proton density function $\rho(x)$, to produce a higher resolution image. In this section, we describe a simple and idealized method of spatial MRI super-resolution.

We assume that two low-resolution, one-dimensional discrete data sets have been acquired from the MRI. For simplicity, we assume that each of these two signals, or “channels”, has a uniform sample spacing of Δx and that the sampling of one channel is performed at shifts of Δx from the other. We shall denote these samples as

$$l_1(n\Delta x) \quad \text{and} \quad l_2(n\Delta x + \Delta x/2), \quad n_l = 0, 1, 2, \dots, N_l - 1. \quad (4)$$

These two channels, which are vectors in \mathbf{R}^{N_l-1} , represent *spatially* sampled data. For simplicity, we may consider the discrete l_1 series as a “reference” data set that represents the object being imaged. The second set, l_2 , is then obtained by sampling after the object has been shifted by $\Delta x/2$, or one-half the pixel size.

These low-resolution samples are then simply interleaved to produced a merged data set of higher resolution. The discrete merged signal, $m(n\Delta x/2) \in \mathbf{R}^{N_h}$, $n = 0, 1, 2, \dots, N_h - 1$, where $N_h = 2N_l$, is defined as follows:

$$m(n\Delta x/2) = \begin{cases} l_1(n\Delta x/2) & n \text{ even} \\ l_2(n\Delta x/2) & n \text{ odd.} \end{cases} \quad (5)$$

A convolution is then used to model the relationship between $m(n\Delta x)$ and the desired high resolution data set, $h(n\Delta x/2) \in \mathbf{R}^{N_h}$:

$$m(n\Delta x/2) = \sum_{n'=0}^{N_h-1} h(n'\Delta x/2)\phi((n-n')\Delta x/2), \quad n = 0, 1, 2, \dots, N_h - 1. \quad (6)$$

The vector $\phi(n\Delta x/2)$ is a point spread function that is estimated using information about the acquisition sequence and/or the object being imaged. After

$\phi(n\Delta x/2)$ has been estimated and $m(n\Delta x)$ has been formed, $h(n\Delta x)$ is found using a chosen deconvolution algorithm.

This is the standard approach adopted for SRMRI [5,6,13,19,21]. However, the “shifted” series l_2 is rarely, if ever, obtained by physically shifting the object being imaged. Normally, the shift is performed “electronically” using the Fourier shift theorem, which will be discussed below.

Finally, the requirement that the two data sets l_1 and l_2 represent one-half pixel shifts can be relaxed to fractional pixel shifts.

4 Super-Resolution MRI in the Frequency Domain

As stated earlier, in order to perform super-resolution, we need at least two data sets that are shifted by fractional pixel lengths. In practice, however, it is difficult to guarantee the accuracy of a spatial shift of the object being imaged (e.g., a patient), especially at subpixel levels. Therefore most, if not all, SRMRI methods simulate the spatial shift by means of the following standard Fourier shift theorem [2]: If $F(k)$ denotes the Fourier transform of a signal $f(x)$, then

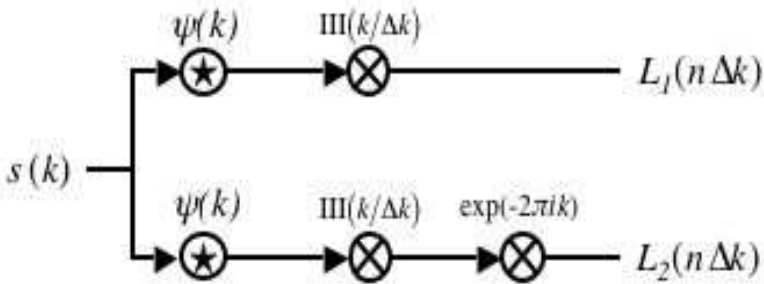
$$\mathcal{F}[f(x - \Delta x)] = e^{-i2\pi k\Delta x} F(k). \tag{7}$$

The term $e^{-i2\pi k\Delta x}$ is known as a *linear phase ramp* – the linearity refers to the k variable.

The main problem in magnetic resonance imaging is that one does not work with the raw frequency signal $s(k)$ but rather with a filtered and sampled version of it, namely, the discrete series $L_1(k)$. It is necessary to compare the effects of (i) phase ramping the filtered/sampled $L_1(k)$ series and (ii) phase ramping the raw signal $s(k)$ before filtering/sampling.

- **Case I: Phase ramp applied after filtering/sampling of raw signal $s(k)$**

The process is illustrated schematically below.



Using Eq. (2), we have the following two series, defined for $k \in [-k_{max}, k_{max}]$,

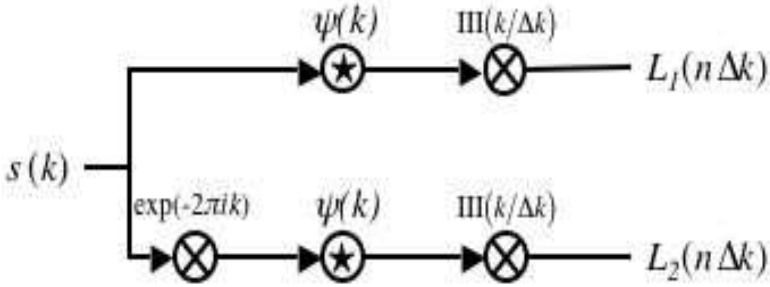
$$L_1(k) = \text{III}\left(\frac{k}{\Delta k}\right) \int_{-k_{max}}^{k_{max}} s(\kappa)\psi(k - \kappa)d\kappa, \tag{8}$$

$$L_2(k) = e^{-i2\pi k \Delta x} \text{III}\left(\frac{k}{\Delta k}\right) \int_{-k_{max}}^{k_{max}} s(\kappa) \psi(k - \kappa) d\kappa. \quad (9)$$

Without loss of generality, we have used $K = 1$. Clearly, one can obtain $L_2(k)$ from $L_1(k)$ by multiplying $L_1(k)$ with another phase ramp *after* $\text{III}\left(\frac{k}{\Delta k}\right)$ is applied. Therefore, no new information is obtained by constructing $L_2(k)$. This point has been recognized in the literature [23,27].

– **Case II: Phase ramp applied before filtering/sampling of raw signal $s(k)$**

The process is illustrated schematically below.



In this case, we have the following two series, defined for $k \in [-k_{max}, k_{max}]$,

$$L_1(k) = \text{III}\left(\frac{k}{\Delta k}\right) \int_{-k_{max}}^{k_{max}} s(\kappa) \psi(k - \kappa) d\kappa, \quad (10)$$

$$L_2(k) = \text{III}\left(\frac{k}{\Delta k}\right) \int_{-k_{max}}^{k_{max}} s(\kappa) e^{-i2\pi \kappa \Delta x} \psi(k - \kappa) d\kappa \quad (11)$$

Note that the $L_2(k)$ series corresponds to the sampled and filtered data that would be obtained if the object being imaged were physically shifted by $\Delta x/2$ [17].

Unlike Case I, we cannot, in general, obtain $L_2(k)$ from $L_1(k)$ because of the noninvertibility of the operations behind the processing of the signal $s(k)$, namely, convolution and sampling, as well as the “degradation” that is produced by the limited range of integration. These operations are unavoidable in the acquisition process. It would then seem that at least *some* new information is present in $L_2(k)$, which is contrary to what has been claimed in the literature [27]. Unfortunately, it is not clear *how much* more information is added by the second acquisition. This is an open question that we explore in the next section.

Before ending this section, however, we briefly describe how super-resolution MRI can be performed using the L_1 and L_2 series of Case II above. There are at least two avenues:

1. **Performing super-resolution in the spatial domain:** Using the discrete inverse Fourier transform, one can first convert each of the L_1 and L_2 series to the spatial series l_1 and l_2 introduced in Section 2. One then performs super-resolution using the interleaving approach described in that section. In other words, we are performing a standard SR technique to enhance the image quality.
2. **Performing super-resolution in the frequency domain:** In the frequency domain, super-resolution may be accomplished by *frequency extrapolation*, i.e., using low frequency information to estimate high frequency behaviour. There are some standard methods to accomplish such extrapolation, e.g., the Papoulis-Gerchberg algorithm [4,12,20,24,25,26] and projection methods [7,8,14,15,18,29]. We are currently investigating the application of these methods to multiple data sets in the frequency domain, a subject that is beyond the scope of this paper.

4.1 Further Analysis of Case II: Phase Ramping Before Filtering/Sampling

To pursue the question of how much *new* information about $\rho(x)$ is present in the second acquisition, $L_2(k)$ of Case II above, let us generalize our model of the raw MRI data (Eqs. (10) and (11)). We shall consider the spatial shift Δx as a variable parameter which will be denoted as β . The shifted low-resolution signal L_2 is now a function of two variables, i.e.,

$$L_2(k, \beta) = \text{III}\left(\frac{k}{\Delta k}\right) I_2(k, \beta), \quad (12)$$

$$I_2(k, \beta) = \int_{-k_{max}}^{k_{max}} s(\kappa) e^{-i2\pi\kappa\beta} \psi(k - \kappa) d\kappa. \quad (13)$$

Clearly, $L_2(k, 0) = L_1(k)$. The β -dependence of L_2 lies in the integral I_2 .

Assuming that the filter $\psi(k)$ is continuous in k , the integral $I_2(k, \beta)$ is a continuous function of both k and β . However, due to the Shah sampling function, $L_2(k, \beta)$ is nonzero for discrete values of $k \in [-k_{max}, k_{min}]$, which we shall denote as $k_n = n\Delta k$ for appropriate integer values of n , say $-N \leq n \leq N$, where

$$N = \text{int} \left[\frac{k_{max}}{\Delta k} \right]. \quad (14)$$

Note that the associated sample spacing in the (spatial) image domain is given by

$$\Delta x = \frac{1}{N\Delta k}. \quad (15)$$

In what follows, we shall focus on the amplitudes $I_2(k_n, \beta)$ of the delta function spikes produced by the Shah-function sampling. These amplitudes comprise a $2N + 1$ -dimensional vector that defines the filtered and sampled signal. For simplicity of notation, we shall denote this vector as $\mathbf{v}(\beta)$, with components

$$v_n(\beta) = I_2(k_n, \beta), \quad n = -N, -N + 1, \dots, N - 1, N. \quad (16)$$

The fact that the sampled signal values $v_n(\beta)$ are continuous functions of the spatial shift parameter β does not seem to have been mentioned in the literature. In particular,

$$v_n(\beta) \rightarrow v_n(0) = I_1(k_n) \quad \text{as } \beta \rightarrow 0, \quad (17)$$

where

$$I_1(k) = \int_{-k_{max}}^{k_{max}} s(\kappa)\psi(k - \kappa)d\kappa. \quad (18)$$

In fact, it is rather straightforward to show that the function $I_2(k, \beta)$, hence each component of the sampled signal, admits a Taylor series expansion in β which can be generated by formal expansion of the exponential in Eq. (12) followed by termwise integration. (The boundedness of the integration interval $k \in [-k_{max}, k_{max}]$ and the uniform convergence of the Taylor expansion of the exponential over this interval guarantees the convergence of the Taylor expansion for $I_2(k, \beta)$.) The resulting complex-valued series will be written as

$$I_2(k, \beta) = \sum_{m=0}^{\infty} c_m \beta^m, \quad (19)$$

where

$$c_m = \frac{(-2i\pi)^m}{m!} \int_{-k_{max}}^{k_{max}} s(\kappa)(\kappa)^m \psi(k - \kappa) d\kappa, \quad m = 0, 1, 2, \dots \quad (20)$$

This result would then imply that we could construct the phase ramped signal $v_p(\beta)$ for nonzero values of β from the Taylor series to arbitrary accuracy. But – and this is the important point – we would have to know the coefficients c_m , which means having access to the raw frequency signal $s(k)$, which is not generally possible. In fact, if we had access to $s(k)$, we could compute $L_2(k, \beta)$ directly using Eq.(12)!

We now return to the question of whether additional “information” is being provided by phase ramping. The answer is in the affirmative: Some kind of additional “information” is provided – at least for sufficiently small β values – because the signals $\mathbf{v}(\beta)$ and $\mathbf{v}(0)$ are generally linearly independent for $\beta \neq 0$. (We use the word “generally” because there are exceptional cases, e.g., $s(k) = 0$.) This follows from the definition of the integral $I_2(k, \beta)$ in Eq. (13). It is not generally possible that for a $\beta > 0$, $I_2(k, \beta) = CI_2(k, 0)$ for all values of k (or even k_n).

Unfortunately, at this time we have no method of characterizing this additional “information” in terms of standard information theoretic concepts, e.g., entropy. This implies, of course, that we have no way of quantifying it at present.

As β increases from zero to $\beta = \Delta x/2$, the spatial counterpart of the sampled signal $L_2(k, \beta)$ is shifted from zero pixels to half a pixel. From a spatial resolution viewpoint, with recourse to the equation for L_2 in Case II, we expect that information is added as β increases from zero. Indeed, we are led to conjecture, but cannot prove at present, the following:

Given the two discrete data sets $\mathbf{v}(0)$ and $\mathbf{v}(\beta)$ associated with, respectively, signals $L_1(k)$ and $L_2(k, \beta)$ defined in Eqs. (10), the amount of information added by phase ramping is an increasing function of the shift β over the interval $\beta \in [0, \Delta x/2]$.

In an attempt to study this conjecture, we present a typical result from a series of numerical experiments over a range of β values. In particular, we consider $N = 50$ and $k_{max} = 1$ so that the pixel width is given by $\Delta x = 1/2$. We have chosen

$$s(k) = \text{sinc}^2(10\Delta x k) = \text{sinc}^2(5k), \quad (21)$$

which is the Fourier transform of the triangular proton density function

$$\rho(x) = \begin{cases} \frac{x}{5}, & -5 \leq x \leq 0, \\ 1 - \frac{x}{5}, & 0 \leq x \leq 5. \end{cases} \quad (22)$$

The anti-aliasing filter $\psi(k/K)$ is simply a Gaussian function with width $K = \Delta k = 2k_{max}/(2N + 1)$. In order to compare $\mathbf{v}(\beta)$ and $\mathbf{v}(0)$, we simply compare the ‘‘angle’’ between the two complex-valued vectors using their complex-valued scalar product,

$$C(\beta) = \cos \theta(\beta) = \frac{\overline{\mathbf{v}(\beta)} \cdot \mathbf{v}(0)}{|\mathbf{v}(\beta)||\mathbf{v}(0)|}, \quad (23)$$

which can be viewed as a kind of ‘‘cross correlation’’ function between the two vectors. In the figure below, we plot numerical values of $C(\beta)$ for $\beta \in [0, 2]$. In all cases, the imaginary parts of $C(\beta)$ are negligible (i.e., zero to at least eight decimal places). At $\beta = 0$, $C(0) = 1$, as expected. As β increases, $C(\beta)$ decreases toward zero, showing that the ‘‘angle’’ between them is increasing toward $\pi/2$.

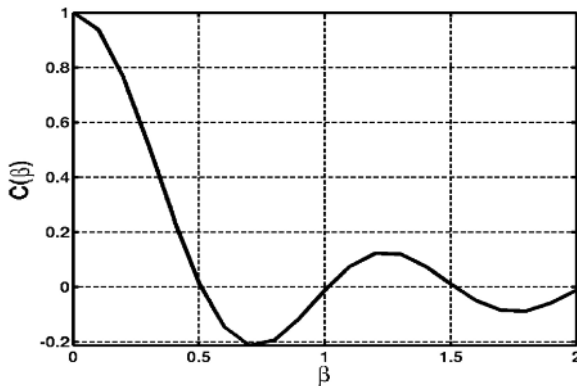


Fig. 1. Plot of angle $\cos \theta(\beta)$, defined in Eq. (23), between vectors $\mathbf{v}(\beta)$ and $\mathbf{v}(0)$ for phase ramping values $0 \leq \beta \leq 2.0$

5 Discussion and Future Work

In this paper, we have investigated the role of multiple data sets in enhancing the spatial resolution of magnetic resonance imaging, i.e. super-resolution MRI. We agree with the conclusions of [27] and [23] that frequency shifting/phase ramping of postprocessed data does not provide any new information. On the other hand, we have shown that phase ramping of the raw or unprocessed frequency signal $s(k)$ can yield new information, thereby making it possible to perform SRMRI. Of course, it remains to see how much super-resolution can actually be accomplished in the presence of machine noise and other degradation factors. This is the subject of our current work. As well, we are interested in quantifying the additional information provided by phase shifting of the raw frequency signal, so that our conjecture could be stated more rigorously.

Another question concerns the role of the anti-aliasing filter ψ . In the numerical experiments performed to date, a sample of which was presented above, we have used a simple Gaussian filter. It remains to investigate the effects of using other filters as well as their sampling widths. For example, consider the limit in which ψ is a Dirac delta function. Then from Eq. (13), ramping of the frequency signal $v_n = s(k_n)$ produces the signal $v_n = s(k_n)e^{-2\pi i\beta k_n}$ and the cross correlation between signals is given by

$$C(\beta) = \frac{\sum_{n=-N}^N s(k_n)^2 e^{-2\pi i\beta k_n}}{\sum_{n=-N}^N s(k_n)^2}. \quad (24)$$

We have observed that the values of $C(\beta)$ for this case differ from the experimental results presented in the previous section. Clearly, further analysis is required if we wish to understand and/or quantify the difference.

Acknowledgements

This research was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) in the form of a Discovery Grant (ERV) and a Postgraduate Scholarship (GSM), which are hereby gratefully acknowledged. GSM thanks Prof. J. Ross Mitchell and Dr. Louis Lauzon of the Seaman Family MR Research Center, Foothills Hospital, University of Calgary, Alberta, Canada, for many helpful conversations during his Master’s programme. GSM also thanks Dr. Hongmei Zhu, formerly at the Seaman Center and now at York University, Toronto, Ontario, Canada.

References

1. Borman S, Stevenson R, Spatial resolution enhancement of low-resolution image sequences - a review. Proceedings of the 1998 Midwest Symposium on Circuits and Systems, Notre Dame IN, 1998

2. Bracewell R, The Fourier Transform and its Applications. McGraw-Hill, 2nd Edition, 1978
3. Chaudhuri S (Editor), Super-Resolution Imaging. Kluwer Academic Publishers, 2001
4. Gerchberg R W, Super-resolution through Error Energy Reduction. *Optica Acta*, Vol. 21, No. 9, 709-720, 1974
5. Greenspan H, Peled S, Oz G, Kiryati N, MRI inter-slice reconstruction using super-resolution. Proceedings of MICCAI 2001, Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention (Lecture Notes in Computer Science, Springer, October), 2001
6. Greenspan H, Oz G, Kiryati N, Peled S, MRI inter-slice reconstruction using super-resolution Magnetic Resonance Imaging, 20, 437-446, 2002
7. Haacke M E, Mitchell J, Doohi L, Improved Contrast at 1.5 Tesla Using Half-Fourier Imaging: Application to Spin-Echo and Angiographic Imaging. *Magnetic Resonance Imaging*, 8, 79-90, 1990
8. Haacke M E, Lindskog E D, Lin W, A Fast, Iterative, Partial-Fourier Technique Capable of Local Phase Recovery. *Journal of Magnetic Resonance*, 92, 126-145, 1991
9. Haacke M E, Brown R W, Thompson M R, Venkatesan R, *Magnetic Resonance Imaging: Physical Principles and Sequence Design*. John Wiley & Sons, Inc., USA, 1999
10. Hinshaw W, Lent A, An Introduction to NMR Imaging: From the Bloch Equation to the Imaging Equation. Proceedings of the IEEE, Vol. 71, No. 3, 338-350, 1983
11. Irani M, Peleg S, Motion analysis for image enhancement: resolution, occlusion, and transparency. *Journal of Visual Communication and Image Representation*, December, 4(4), 324-335, 1993
12. Jain A, Ranganath S, Extrapolation Algorithms for Discrete Signals with Application in Spectral Estimation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-29, 4, 830-845, 1981
13. Kornprobst P, Peeters R, Nikolova M, Deriche R, Ng M, Hecke P V, A superresolution framework for fMRI sequences and its impact on resulting activation maps. In : *Medical Image Computing and Computer-Assisted Intervention-MICCAI 2003*, Lecture Notes in Computer Science, 2, Springer-Verlag, 117-125, 2003
14. Liang Z, Boada F E, Constable R T, Haacke M E, Lauterbur P C, Smith M R, Constrained Reconstruction Methods in MR Imaging. *Reviews of Magnetic Resonance in Medicine*, 4, 67-185, 1992
15. Liang Z, Lauterbur P C, *Principles of Magnetic Resonance Imaging, A Signal Processing Perspective*. IEEE Press, New York, 2000
16. Margosian P, Schmitt F, Faster MR Imaging: Imaging with Half the Data. *Health Care Instrumentation*, 1, 195-197, 1986
17. Mayer G S, *Synthetic Aperture MRI*. M.Sc. Thesis, The University of Calgary, 2003
18. McGibney G, Smith M R, Nichols S T, Crawley A, Quantitative Evaluation of Several Partial Fourier Reconstruction Algorithms Used in MRI. *Magnetic Resonance in Medicine*, 30, 51-59, 1993
19. Ng K P, Deriche R, Kornprobst P, Nikolova M, Half-Quadratic Regularization for MRI Image Restoration. *IEEE Signal Processing Conference*, 585-588 (Publication No. : 76681), 2003
20. Papoulis A, A New Algorithm in Spectral Analysis and Band-Limited Extrapolation. *IEEE Transactions on Circuits and Systems*, Vol. CAS-22, 9, 735-742, 1975
21. Peeters R, et al, The Use of Super-Resolution Techniques to Reduce Slice Thickness in Functional MRI. *International Journal of Imaging Systems and Technology*, Vol. 14, 131-138, 2004

22. Peled S, Yeshurun Y, Superresolution in MRI: Application to Human White Matter Fiber Tract Visualization by Diffusion Tensor Imaging. *Magnetic Resonance in Medicine*, 45, 29-35, 2001
23. Peled S, Yeshurun Y, Superresolution in MRI - Perhaps Sometimes. *Magnetic Resonance in Medicine*, 48, 409, 2002
24. Sanz J, Huang T, Discrete and Continuous Band-Limited Signal Extrapolation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-31, No. 5, 1276-1285, 1983
25. Sanz J, Huang T, Some Aspects of Band-Limited Signal Extrapolation: Models, Discrete Approximations, and Noise. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-31, No. 6, 1492-1501, 1983
26. Sabri M S, Steenaart W, An Approach to Band-Limited Signal Extrapolation: The Extrapolation Matrix. *IEEE Transactions on Circuits and Systems*, Vol. CAS-25, No. 2, 1978
27. Scheffler K, Superresolution in MRI? *Magnetic Resonance in Medicine*, Vol. 48, 408, 2002
28. Tsai R, Huang T, Multiframe image restoration and registration. In: *Advances in Computer Vision and Image Processing*, JAI Press Inc., 1, 317-339, 1984
29. Youla D, Generalized Image Restoration by the Method of Alternating Orthogonal Projections. *IEEE Transactions on Circuits and Systems*, Vol. CAS-25, No. 9, 1978

Blind Blur Estimation Using Low Rank Approximation of Cepstrum

Adeel A. Bhutta and Hassan Foroosh

School of Electrical Engineering and Computer Science,
University of Central Florida,
4000 Central Florida Boulevard,
Orlando, FL 32816-2666, USA
{abhutta, foroosh}@eecs.ucf.edu
<http://cil.eecs.ucf.edu/>

Abstract. The quality of image restoration from degraded images is highly dependent upon a reliable estimate of blur. This paper proposes a blind blur estimation technique based on the low rank approximation of cepstrum. The key idea that this paper presents is that the blur functions usually have low ranks when compared with ranks of real images and can be estimated from cepstrum of degraded images. We extend this idea and propose a general framework for estimation of any type of blur. We show that the proposed technique can correctly estimate commonly used blur types both in noiseless and noisy cases. Experimental results for a wide variety of conditions i.e., when images have low resolution, large blur support, and low signal-to-noise ratio, have been presented to validate our proposed method.

1 Introduction

The first and foremost step in any image restoration technique is blur estimation which is referred to as blind blur estimation when partial or no information about imaging system is known. Numerous techniques [1] have been proposed over the years which try to estimate Point Spread Function (PSF) of blur either separately from the image [2, 3, 4, 5], or simultaneously with the image [6, 7]. The first set of these techniques, also known as Fourier-based techniques, normally use the idea of finding zeros in the frequency domain. These techniques are widely popular because these tend to be computationally simple and require minimal assumptions on images. But these techniques require small blur support or relatively high signal-to-noise ratio. A lot of past research has focused on these requirements and how to relax them. One such technique [2] proposed reducing the effect of noise by using an averaging scheme over smaller portions of the image, thus requiring the original image to be of high resolution. Another technique [8] suggested that the bispectrum of the ‘central slice’ should be used in order to estimate the blur. In that work, only motion blur with small support was estimated at low signal-to-noise ratios. Another related approach was proposed in [9], where image restoration through operations on singular vectors and singular

values of degraded image was proposed. In their work several images were used to remove the effect of noise. Thus far, no single Fourier-based technique has been able to estimate common blur functions correctly from *single* degraded image while ensuring that the technique works not only for various amounts of blur support in low resolution images but also at low signal-to-noise ratios.

In this work, we address the blind blur estimation problem for images that may not be of high resolution and may have noise present during the blurring process. We estimate the blur parameters using low rank constraints on blur functions without performing any operations on singular vectors directly. Our technique assumes that the type of blur added is known but no other assumptions on any prior knowledge about the original image is made. We observe that the blurs are usually low rank functions and thus can be separated from images of high ranks. Based on this observation, we quantify the exact rank approximation for different blur types. These low rank constraints, when applied to log-power spectrum (the power cepstrum) of degraded image, allow us to estimate the blur functions. We also demonstrate that this technique works for a wide variety of blur types (e.g., Gaussian, Motion and Out of focus blurs), when blur support is not small and also when additive noise is present. Experimental results for real images (both noiseless and noisy cases) are presented for various signal-to-noise ratios.

The rest of the paper is organized as follows: Section 2 reviews general concepts of low rank approximation and image degradation and also introduces different blur models. Section 3 presents means to estimate blur parameters using low rank approximation of cepstrum whereas results for different blur types in several scenarios are presented in Section 4. Section 5 presents conclusion of this work.

2 Theory

2.1 Low Rank Approximation

Low rank approximation is a well known tool for dimension reduction and has been widely used in many areas of research. For a given data A , its Singular Value Decomposition (SVD) can be written as,

$$A = U \Sigma V^T$$

where U and V are orthogonal and Σ is diagonal. A can be approximated as,

$$A \approx U_k \Sigma_k V_k^T \tag{1}$$

where k is the rank that best approximates A ; U_k and V_k are the matrices formed by the first k columns of the matrices U and V respectively; and Σ_k is the k -th principal submatrix of Σ . We use this formulation along with an observation, that blurs are usually low rank functions, to quantify their exact rank approximation in Section 3.

2.2 Image Degradation and Blur Models

The goal in a general blind blur estimation problem is, to separate two convolved signals, \mathbf{i} and \mathbf{b} , when both are either unknown or partially known. The image degradation process is mathematically represented by convolution of image \mathbf{i} and blur \mathbf{b} in time domain i.e., $h(x, y) = i(x, y) * b(x, y)$. The frequency domain representation of this system is given by point-by-point multiplication of image spectrum with blur spectrum.

$$H(m, n) = I(m, n)B(m, n) \quad (2)$$

The system in (2) depicts a noiseless case when image degradation occurred only due to the blur function. The noisy counterpart of such degradation can be modelled as,

$$H(m, n) = I(m, n)B(m, n) + N(m, n) \quad (3)$$

where $N(m, n)$ denotes the spectrum of an additive noise.

A list of optical transfer functions (OTF) of frequently used blurs has been given in Table 1. It should be noted that OTF is the Fourier transform version of PSF. In the experiments presented in this paper, we have estimated OTFs of blur functions.

Table 1. Frequently used blur models

| Blur Model | Rank of Blur | Optical Transfer Function (OTF) |
|--------------------|--------------|--|
| Linear Motion Blur | 1 | $\frac{\sin(\pi a f_x)}{\pi a f_x}$ |
| Gaussian Blur | 1 | $\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(f-\mu)^2}{2\sigma^2}}$ |
| Out of focus Blur | 10 | $\frac{J_1(2\pi r f)}{\pi r f}$ |

3 Blur Estimation Using Low Rank Approximation of Cepstrum

The logarithm of the absolute spectra of degraded image (also known as power cepstrum) can be written as the summation of logarithms of absolute spectra of image and blur as in (4). We propose that these can be separated when low rank constraints are applied. Table 1 summarizes ranks for commonly used blur functions¹.

Power-Cepstrum of (2), can be written as,

$$H_p = I_p + B_p \quad (4)$$

¹ Exact value of rank for out of focus blur depends on the value of blur parameter.

where H_p is the log-power spectrum or power-cepstrum of $h(x, y)$. It should be noted that H_p is the summation of two matrices I_p and B_p where B_p has low rank and I_p has high rank. Therefore, by performing low rank approximation using SVD of (4), we can get a close approximation for the blur. This can be written as,

$$LR[H_p] = LR[I_p + B_p] \simeq B_p \quad (5)$$

where LR represents low rank approximation using SVD. It should be emphasized that only the first few singular vectors will characterize blur functions whereas remaining will represent image. For estimation of any blur, only a few singular vectors are sufficient. Blur parameters can then be estimated from exponent of (5) by finding location of zeros (for uniform motion or out of focus blur) or by fitting appropriate Gaussian (for Gaussian blur). In order to have a more reliable estimate of the blur parameters, we use several (candidates of) blur estimates obtained from single image. It should be noted that when the degraded image has additive noise, the low rank approximation will characterize blur as well as added noise.

3.1 Gaussian Blur

When an image is blurred using Gaussian blur, the parameters of Gaussian can be estimated from low rank approximation as described above. These parameters can be derived from a scaled 2D version of Gaussian blur ($B_{m,n}$) in Table 1 as below:

$$B_{m,n} = \frac{\alpha}{2\pi\sigma^2} e^{-\frac{(m^2+n^2)}{2\sigma^2}} \quad (6)$$

Where m and n are Fourier counter-parts of x and y .

If σ is defined as blur parameter and α as the scale factor, Blur parameters (σ) can be calculated by

$$\frac{\frac{m^2+n^2}{2}}{-1 \pm \sqrt{1 - (\frac{m^2+n^2}{2})(-\log(B_{m,n}) - \log(\alpha) - \log(2\pi))}} \quad (7)$$

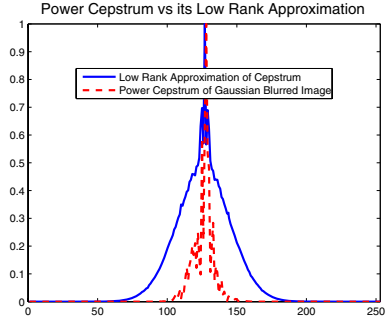
A complete proof of (7) is provided in Appendix. Once blur estimates are obtained from (5), we can directly find the parameters using (7) for noiseless image degradation. The mode of the distribution of candidate parameters is used to calculate the final blur parameters.

When the blurred image has additive noise, it can be suppressed by integrating the noisy blurred image in one direction. The parameters for blur (σ) in this case can be estimated using (8). A complete proof of (8) is also provided in Appendix. The parameters for blur in noisy case, are given by,

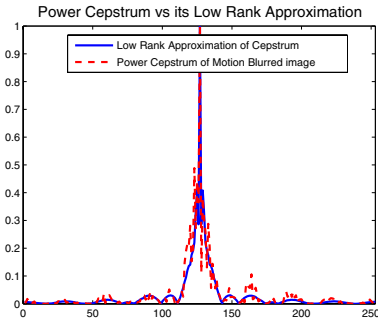
$$\frac{n^2}{1 \pm \sqrt{1 + (2n^2)(-\log(B_m) - \log(\alpha) - 1 - \frac{\log(2\pi)}{2})}} \quad (8)$$



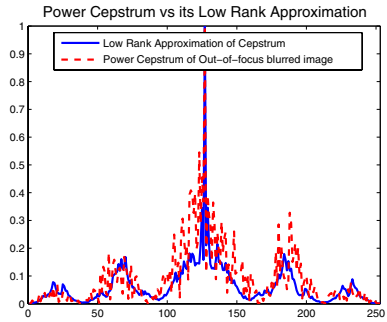
(a)



(b)



(c)



(d)

Fig. 1. Comparison of Blur Estimation using Cepstrum vs its Low Rank approximation when input image (a) is degraded by blur type: (b) Gaussian, (c) Linear Motion, and (d) Out of focus

For noisy Gaussian blur, low rank approximation characterizes both blur as well as noise. Since the blurred image had additive noise, the parameters estimated will be noisy. Therefore, it is reasonable to assume that the correct blur parameters can be estimated by fitting Gaussian mixture model [10].

3.2 Motion Blur

When an image is blurred with motion blur, blur parameters for motion blur are calculated using (9). These parameters are derived by finding the location of zeros of the blur estimates as below:

$$\frac{\sin(\pi a f_x)}{\pi a f_x} = \sin(\pi k) = 0 = \pi a f_x = \pi k$$

$$\implies f_x a = k \tag{9}$$

where \mathbf{a} is the parameter to be estimated, \mathbf{k} is the number of zero crossings whereas \mathbf{f}_x relates to the location of zero crossing. A reliable estimate of blur parameter can be found using a system as given below:

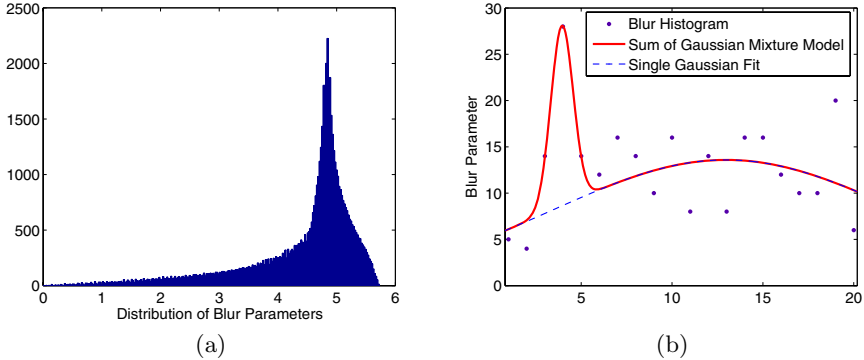


Fig. 2. Example of Gaussian Blur Estimation: (a) Distribution of all blur parameters, and (b) Maximum-likelihood estimate of parameters by fitting Gaussian mixture model

$$\mathbf{F}_x \cdot \mathbf{a} = \mathbf{K} \quad (10)$$

where \mathbf{F}_x and \mathbf{K} are matrices with several values of \mathbf{f}_x and \mathbf{k} from (9) stacked respectively together.

3.3 Out of Focus Blur

When an image is blurred with out of focus blur, blur parameters are calculated in a manner similar to (9) where a least square system for out of focus blur can be created as in (10). These parameters are derived by finding the location of zeros of the blur estimates as below:

$$\frac{J_1(2\pi a f)}{\pi a f} = 0 \implies r f = k$$

where \mathbf{r} is the parameter to be estimated, \mathbf{k} is the number of zero crossings, and \mathbf{f} is related to the location of zero crossings. It should be mentioned that in our results, we have used the value of \mathbf{f} as provided in [3].

4 Results

The proposed method can be used to estimate any type of blur function but results for Gaussian, uniform motion, and out of focus blurs have been presented here. Once low rank approximation of cepstrum is obtained, initial blur parameters are estimated followed by their robust estimates (from several candidates). We have presented the results both in the noiseless and noisy cases (with different signal-to-noise ratios).

Figure 1(a) shows the original image before any degradation. Figure 1(b) shows the comparison of Gaussian blur parameter estimation for cepstrum verses the low rank approximation of cepstrum. It is clearly obvious that the low rank

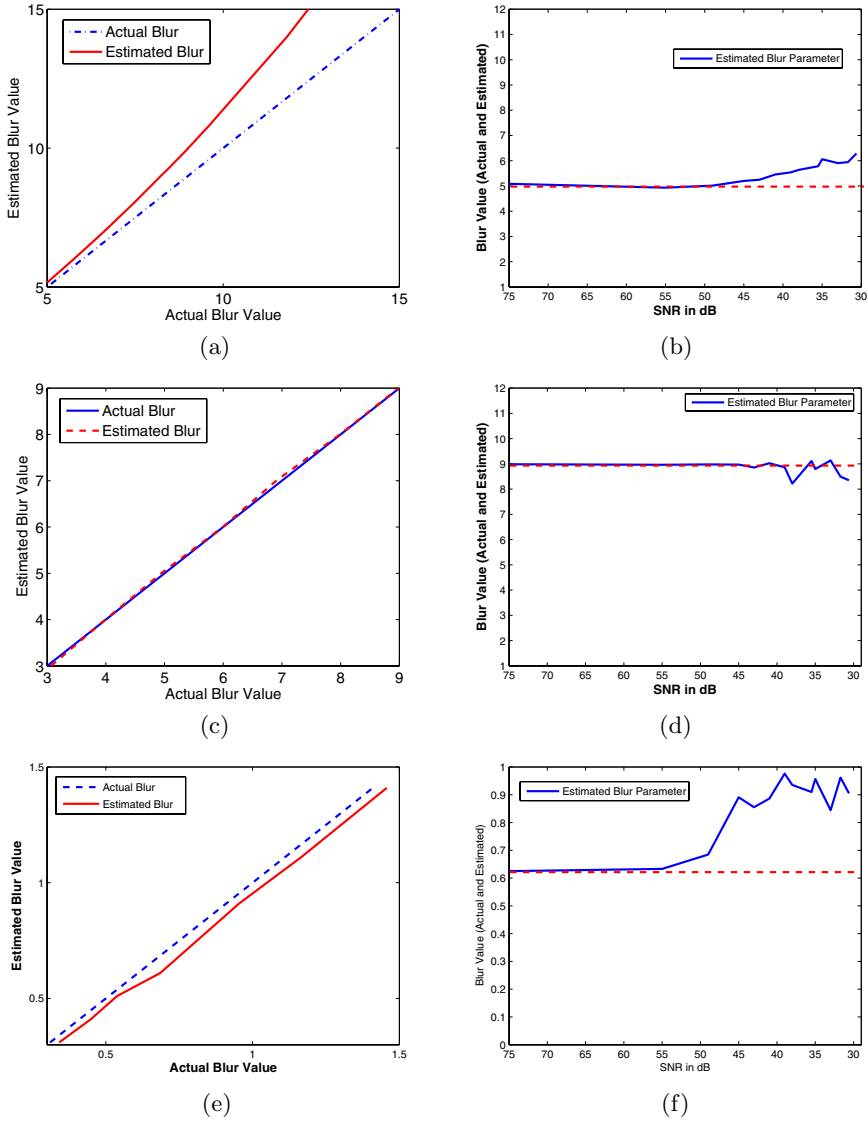


Fig. 3. Comparison of actual vs estimated blur parameters when blur type is: (a) Noiseless Gaussian Blur, (b) Noisy Gaussian Blur, (c) Noiseless Motion Blur, (d) Noisy Motion Blur, (e) Noiseless Out of Focus Blur, (f) Noisy Out of Focus Blur

(approximation) version is less erratic and easy to use. The observation holds true for Linear Motion Figure 1(c) and Out of focus Figure 1(d) blurs. Using several slices of our blur estimates, we can find the parameters using (7). The mode of the distribution of these candidate parameters, shown in Figure 2(a), is used to calculate the final blur parameter. The comparison of actual and estimated

blur values has been shown in Figure 3(a). For noisy Gaussian blur, the low rank approximations have both noise as well as blur parameters. In order to estimate and subsequently separate both distributions, Gaussian curve fitting is used. Figure 2(b) shows the fitting of Gaussian mixture model on the distribution of noisy parameters. Since the noise usually has small mean value, the mode of larger distribution gives us the correct estimate of blur parameters. The comparison of actual and estimated blur values has been shown in Figure 3(b). It should be noted that, for the noisy case, results for average values of Gaussian blur estimates over 100 independent trials have been plotted against the actual values under different noise levels.

Next, an image was blurred using uniform motion blur in one direction, as proposed in [3], and blur parameters were estimated using low rank approximation of cepstrum. These motion blur parameters are calculated using (9) after ‘properly’ thresholding the low rank approximations. It should be emphasized that motion blur is only in one direction therefore, several estimates of blur parameters are available along the direction orthogonal to the motion. To estimate robust parameters having consensus over all available estimates, we build a over-determined system of equations and find the least square solution as in (10). The comparison of actual and estimated values of motion blur using our method is shown in Figure 3(c) and (d) for noiseless and noisy cases respectively. It should again be noted that in the noisy case, results for average values of motion blur estimates over 100 independent trials have been plotted against actual values under different noise levels. Similarly, blur parameters for images blurred by out of focus blur are estimated and are shown in Figure 3(e) and (f) both for noiseless and noisy cases respectively.

4.1 Discussion

Most Fourier transform based techniques require high resolution of input images [8]. Their performance also reduces when the blur amount is too high or signal-to-noise ratio is too low [1]. We have presented a technique which does not require high resolution images. The images used in this paper were of smaller size 253x253 as compared to the resolution required by [8] i.e., 512x512. Moreover, we are able to estimate the blur when the blur value is high and signal-to-noise ratio is low. Figure 3 shows the results for a wide range of SNR ratios. It should also be emphasized that our technique uses low rank approximation of cepstrum and therefore requires only fewer singular vectors for blur estimation as compared with any other known technique.

5 Conclusion

We have presented a novel technique for blind blur estimation using low rank constraints on blur functions. The main idea this paper presents is that the blur functions usually have low ranks when compared with images, allowing us to estimate them robustly from a single image. We have also quantified the exact rank that should be used for different blur types. One major strength of our

approach is that it presents a general framework for low rank approximation since rank constraints apply to all blur functions in practice. Results for Gaussian, motion, and out of focus blurs are presented to show that the technique works for most common blur types and at various noise levels. It has also been shown that the technique does not require any assumptions on imaging system and works well for low resolution images.

References

1. D. Kundur and D. Hatzinakos: Blind image deconvolution. *IEEE Signal Processing Magazine*. **13(3)** (1996) 43–64
2. M. Cannon: Blind deconvolution of spatially invariant image blurs with phase. *IEEE Trans. Acoust., Speech, Signal Processing*. **24** (1976) 58–63
3. D. Gennery: Determination of optical transfer function by inspection of frequency-domain plot. *Journal of SA*. **63** (1973) 1571–1577
4. R. Hummel, K. Zucker, and S. Zucker: Deblurring gaussian blur. *CVGIP* **38** (1987) 66–80
5. R. Lane and R. Bates: Automatic multidimensional deconvolution. *JOSA* **4(1)** (1987) 180–188
6. A. Tekalp, H. Kaufman, and J. Wood: Identification of image and blur parameters for the restoration of noncausal blurs. *IEEE Trans. Acoust., Speech, Signal Processing* **34(4)** (1986) 963–972
7. S. Reeves and R. Mersereau: Blur identification by the method of generalized cross-validation. *IEEE Trans. Image Processing*, **1(3)** (1992) 301–311
8. M. Chang, A. Tekalp, and A. Erdem: Blur identification using the bispectrum. *IEEE Trans. Signal Processing*. **39**, (1991) 2323–2325
9. D. Z. and L. S.: Blind Restoration of Space-Invariant Image Degradations in the SVD domain. *Proc. IEEE ICIP*, **2** (2001) 49–52
10. J. Bilmes: A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hmms. U.C.Berkely TR-97-021 (1998)

Appendix

1: Proof of Equation (7)

Taking logarithm of (6) we get,

$$\log(B_{m,n}) = \log(\alpha) - \log(2\pi\sigma^2) + \frac{-m^2}{2\sigma^2} + \frac{-n^2}{2\sigma^2} \quad (11)$$

which can be written as,

$$\log(B_{m,n}) = \frac{1}{\sigma^2} \left(\frac{-m^2 - n^2}{2} \right) + \log(2) + 2\log\left(\frac{1}{\sigma}\right) - \log(\alpha) - \log(2\pi)$$

or

$$\log(B_{m,n}) = \frac{1}{\sigma^2} \left(\frac{-m^2 - n^2}{2} \right) + 2\log\left(1 + \frac{1}{\sigma}\right) - \log(\alpha) - \log(2\pi)$$

If σ and α are constants, the 1st order Taylor series approximation of above equation gives,

$$\frac{1}{\sigma^2} \left(\frac{-m^2 - n^2}{2} \right) + 2 \left(\frac{1}{\sigma} \right) - \log(B_{m,n}) - \log(\alpha) - \log(2\pi) = 0$$

hence,

$$\frac{1}{\sigma} = \frac{-1 \pm \sqrt{1 - \left(\frac{m^2+n^2}{2} \right) (-\log(B) - \log(\alpha) - \log(2\pi))}}{\frac{m^2+n^2}{2}} \quad (12)$$

Therefore, (σ) for noiseless case is given by,

$$\frac{\frac{m^2+n^2}{2}}{-1 \pm \sqrt{1 - \left(\frac{m^2+n^2}{2} \right) (-\log(B_{m,n}) - \log(\alpha) - \log(2\pi))}} \quad (13)$$

2: Proof of Equation (8)

Integrating (6) along the direction orthogonal to motion and taking logarithm gives,

$$\log(B_x) = \log(\alpha) - \frac{1}{2} \log(2\pi) + \log\left(\frac{1}{\sigma}\right) - \frac{y^2}{2\sigma^2} \quad (14)$$

If σ and α are constants, the 1st order Taylor series approximation of above equation gives,

$$\frac{1}{\sigma^2} \left(\frac{-y^2}{2} \right) - 1 + \frac{1}{\sigma} - \log(B_{m,n}) + \log(\alpha) - \frac{1}{2} \log(2\pi) = 0$$

hence,

$$\frac{1}{\sigma} = \frac{-1 \pm \sqrt{1 - 2y^2(-\log(B_x) + \log(\alpha) + (\frac{1}{2})\log(2\pi) - 1)}}{-y^2} \quad (15)$$

Therefore, (σ) for noisy case is given by,

$$\frac{-y^2}{-1 \pm \sqrt{1 - 2y^2(-\log(B_x) + \log(\alpha) + (\frac{1}{2})\log(2\pi) - 1)}} \quad (16)$$

An Image Interpolation Scheme for Repetitive Structures

Hiệp Luong, Alessandro Ledda, and Wilfried Philips

Ghent University, Department of Telecommunications and Information Processing,
B-9000 Ghent, Belgium

Abstract. In this paper we present a novel method for interpolating images with repetitive structures. Unlike other conventional interpolation methods, the unknown pixel value is not estimated based on its local surrounding neighbourhood, but on the whole image. In particular, we exploit the repetitive character of the image. A great advantage of our proposed approach is that we have more information at our disposal, which leads to a better reconstruction of the interpolated image. Results show the effectiveness of our proposed method and its superiority at very large magnifications to other traditional interpolation methods.

1 Introduction

Many applications nowadays rely on digital image interpolation. Some examples are simple spatial magnification of images or video sequences (e.g. printing low resolution documents on high resolution (HR) printer devices, digital zoom in digital cameras or displaying Standard Definition video material on High Definition television (HDTV)), geometric transformation and registration (e.g. affine transformations or computer-assisted alignment in modern X-ray imaging systems), demosaicing (reconstruction of colour images from CCD samples), etc.

Many interpolation methods already have been proposed in the literature, but all suffer from one or more artefacts. Linear or non-adaptive interpolation methods deal with aliasing (e.g. jagged edges in the up scaling process), blurring and/or ringing effects. Well-known and popular linear interpolation methods are nearest neighbour, bilinear, bicubic and interpolation with higher order (piece-wise) polynomials, B-splines, truncated or windowed sinc functions, etc. [7,10].

Non-linear or adaptive interpolation methods incorporate a priori knowledge about images. Dependent on this knowledge, the interpolation methods could be classified in different categories. The edge-directed based techniques follow a philosophy that no interpolation across the edges in the image is allowed or that interpolation has to be performed along the edges. This rule is employed for instance in Allebach's EDI method, Li's NEDI method and Muresan's AQUA method [1,8,12]. The restoration-based techniques tackle unwanted interpolation artefacts like aliasing, blurring and ringing. Examples are methods based on isophote smoothing, level curve mapping and mathematical morphology [11,9,6]. Some other adaptive techniques exploit the self-similarity property of an image, e.g. iterated function systems [5,15]. Another class of adaptive interpolation

methods is the training-based approach, which maps blocks of the low resolution image into predefined high resolution blocks [4].

Adaptive methods still suffer from artefacts: their results often look segmented, yield important visual degradation in fine textured areas or random pixels are created in smooth areas [9]. When we use very big enlargements (i.e. linear magnification factors of 8 and more), then all these artefacts become more visible and hence more annoying.

In §2 we will motivate the exploitation of repetitive structures in image interpolation. In §3 we give an image interpolation scheme for repetitive structures and in §4 we show some experiments and results of our proposed method compared to other interpolation techniques.

2 Repetitive Structures

Fractal-based interpolation methods suppose that many things in nature possess fractalness, i.e. scale invariance [5]. This means that parts of the image repeat themselves on an ever-diminishing scale, hence the term self-similarity. This self-similarity property is exploited for image compression and interpolation by mapping the similar parts at different scales. Due to the recursive application of these mappings at the decoder stage, the notion of *iterated function systems* (IFS) is introduced.

Unlike IFS, we exploit the similarity of small patches in the same scale, i.e. spatially. In order to avoid confusion, we will use the term repetitively. Another class of upscaling methods which also takes advantage of repetitively, is called *super resolution* (SR) reconstruction. SR is a signal processing technique that obtains a HR image from multiple noisy and blurred low resolution images (e.g. from a video sequence). Contrary to image interpolation, SR uses multiple source images instead of a single source image. It is well known that SR produces superior results to conventional interpolation methods [3].

It is often assumed that true motion is needed for SR, however many registration methods do not yield true motion: their results are optimal to some proposed cost criterion, which are not necessarily equal to true motion. With this in mind, we can hypothetically assume that repetitive structures could serve as multiple noisy observations of the same structure (after proper registration). Results of our experiments in §4 will confirm that this hypothesis holds for real situations. The concept of repetitive structures has already successfully been used for image denoising [2]. Besides repetitively in texture, we can also find this recurrent property in other parts of the image, some examples are illustrated in figure 1.

Our method is found perfectly suitable to some applications: some examples are interpolation of scanned text images (availability of multiple repeated characters regardless of their font and the scan orientation) and gigantic satellite images (long roads and a lot of texture provide a huge amount of training data). A special application is SR in surveillance: when very few low resolution images are available, missing data could be filled up with the use of repetitive

structures. Because of the close relationship with SR, we can also denote our method as an intra-frame SR technique. In that way, surveillance applications could use a combination of inter- and intra-frame SR.



(a) Repetition in different objects.



(b) Repetition along edges.



(c) Repetition in uniform areas.

Fig. 1. Examples of repetitive structures in images

3 Proposed Reconstruction Scheme

We propose a simple interpolation method which exploits this repetitive behaviour. Our interpolation method is based on our camera model as shown in figure 2. Our scheme is quite straightforward and consists of three consecutive steps:

1. Matching repetitive structures and subpixel registration of these structures on the HR grid.
2. Robust data fusion.
3. Restoration (i.e. denoising and deblurring).

In the rest of this paper we will simply treat each R,G,B-channel of colour images separately. For an enhanced colour treatment we refer to [14]. The following sections will discuss each component carefully.

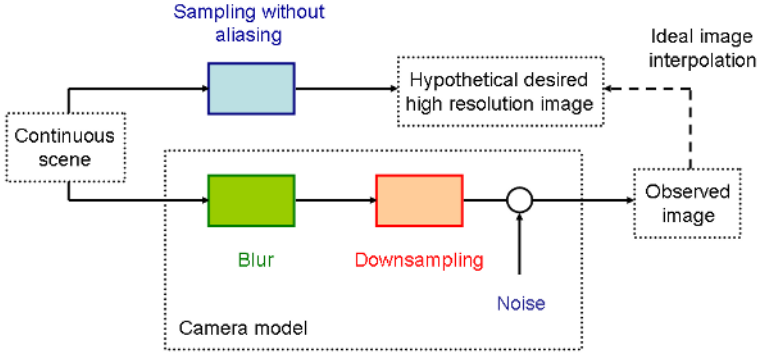


Fig. 2. Observation model of the image acquisition

3.1 Matching and Registration of Repetitive Structures

For the sake of simplicity, we define small rectangular windows B as basic structure elements. Two criteria are used in our algorithm to find matching windows or blocks across the whole image, namely the *zero-mean normalized cross correlation* (CC) and the *mean absolute differences* (MAD):

$$E_{CC} = \frac{\sum_{\mathbf{x} \in \Omega} (B(m(\mathbf{x})) - \overline{B})(B_{\text{ref}}(\mathbf{x}) - \overline{B_{\text{ref}}})}{\sqrt{\sum_{\mathbf{x} \in \Omega} (B(m(\mathbf{x})) - \overline{B})^2 \sum_{\mathbf{x} \in \Omega} (B_{\text{ref}}(\mathbf{x}) - \overline{B_{\text{ref}}})^2}} \quad (1)$$

$$E_{MAD} = \frac{1}{\kappa(\Omega)} \sum_{\mathbf{x} \in \Omega} |B(m(\mathbf{x})) - B_{\text{ref}}(\mathbf{x})| \quad (2)$$

where Ω contains all the pixels of the reference window B_{ref} and $\kappa(\Omega)$ is the cardinality (i.e. the number of pixels) of Ω . \overline{B} and $\overline{B_{\text{ref}}}$ are denoted as the mean values of respectively B and B_{ref} . The transformation of the coordinates is characterized by the mapping function m . To simplify the registration problem and in particular to save computation time, we assume that we are only dealing with pure translational motions of B . The main motive to use the CC and MAD criteria is because they are somewhat complementary: CC emphasizes the similarity of the structural or geometrical content of the windows, while MAD underlines the similarity of the luminance and colour information. A matched window is accepted if the two measures E_{CC} and E_{MAD} satisfy to the respective thresholds τ_{CC} and τ_{MAD} , more specifically: $E_{CC} > \tau_{CC}$ and $E_{MAD} < \tau_{MAD}$. Since we only want to have positive correlation, E_{CC} must lay between 0.0 and 1.0 and τ_{MAD} denotes the maximum mean absolute pixel difference between two windows. The choice of E_{CC} and τ_{MAD} depends heavily on the noise content of the image (e.g. due to additive noise or due to quantization noise in DCT-based compressed images such as JPEG): the higher the noise variance, the lower E_{CC}

and the higher τ_{MAD} must be chosen in order to match the same repetitive structures. Our implementation uses a simple exhaustive search in order to find the matching windows, but more intelligent (pattern-based) search algorithms could reduce the computation time enormously.

Common ways to achieve subpixel registration in the spatial domain, is to interpolate either the image data or the correlation data. In order to save computation time we only resample the reference window B_{ref} on a higher resolution. In this way we represent the downsampling operator in the camera model in figure 2 as a simple decimation operator as illustrated in figure 3. We estimate the subpixel shifts by minimizing the MAD criterion in equation 2. As a simplification of the optimization problem, we use the HR grid as the discrete search space for the subpixel shifts. After the subpixel registration, the pixel values of B are mapped onto the HR grid.

Most existing techniques use linear methods to resample B_{ref} . However, these interpolation methods typically suffer from blurring, staircasing and/or ringing. These artefacts not only degrade the visual quality but also affect the registration accuracy. Therefore we adopt a fast non-linear restoration-based interpolation technique based on level curve mapping, which suffers less from these artefacts [9].

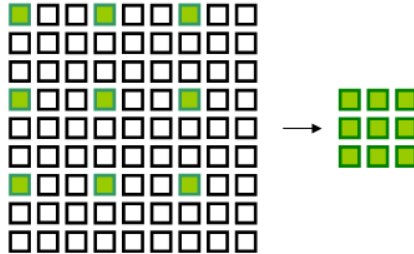


Fig. 3. The 3 : 1 decimation operator maps a $3M \times 3N$ image to an $M \times N$ image

3.2 Data Fusion

In this step we determine an initial pixel value for every pixel of the HR grid. In the previous registration step we already obtained zero or more observations for these pixels.

Several observations are available. Starting from the maximum likelihood principle, it can be shown that minimizing the norm of the residuals is equivalent to median estimation [3]. A residual is the difference between an observed pixel value and the predicted pixel value. The median is very robust to outliers, such as noise and errors due to misregistration. For this reason we adopt the median estimate of all observations for every pixel in the HR grid for which we have at least one observation.

No observation is available. These unknown pixel values are initialised with the values of the interpolated reference windows B_{ref} . We do not need additional computations since this image is already constructed for the registration step (see §3.1).

Restoring the original pixels. These original pixel values are simply mapped back on the HR grid, because we know that these pixel values contain the least noise.

In a nutshell, the HR grid consists of three classes: the original pixels (OR), the unknown pixels (UN) and the fused pixels (FU). The last mentioned class will provide the extra information which gives us better interpolation results compared to conventional upscaling techniques.

3.3 Denoising and Deblurring

We assume that the blur in the camera model in figure 2 is characterized by a shift-invariant *point spread function* (PSF). The inverse problem becomes highly unstable in the presence of noise. This can be solved by imposing some prior knowledge about the image. Typically we will try to force spatial smoothness in the desired HR solution. This is usually implemented as a penalty factor in the generalized minimization cost function:

$$\hat{I}(\mathbf{x}) = \arg \min_{I(\mathbf{x})} [\rho_{\text{R}}(I(\mathbf{x})) + \lambda \rho_{\text{D}}(H * I(\mathbf{x}) - I(\mathbf{x}, 0))] \quad (3)$$

where H denotes the PSF-kernel (typically Gaussian blur, which is characterized by its standard deviation σ_{blur}) and λ is the regularization parameter between the two terms, respectively called the regularization term ρ_{R} and the data fidelity term ρ_{D} . Image $I(\mathbf{x}, 0)$ is the HR image obtained in §3.2.

The minimization problem of equation 3 could be transformed to the following partial differential equation (PDE) which produces iteratively diffused images $I(\mathbf{x}, t)$ starting from the inialisation image $I(\mathbf{x}, 0)$:

$$\frac{\partial I(\mathbf{x}, t)}{\partial t} = \rho'_{\text{R}}(I(\mathbf{x}, t)) + \lambda \rho'_{\text{D}}(H * I(\mathbf{x}, t) - I(\mathbf{x}, 0)) \quad (4)$$

where the chain rule is applied on both ρ -terms: $\rho'(I(t)) = \frac{\partial \rho(I(t))}{\partial I(t)} \cdot \frac{\partial I(t)}{\partial t}$.

The use of the so-called edge-stopping functions in the regularization term is very popular, because it suppresses the noise better while retaining important edge information [13]. Therefore we apply one of the most successful edge-preserving regularization terms proposed for image denoising, namely the total variation (TV):

$$\rho_{\text{R}}(I(\mathbf{x}, t)) = |\nabla I(\mathbf{x}, t)| \quad (5)$$

Since the interpolation of repetitive structures is closely related to super resolution problems, we could assume that the outliers (due to the inaccuracy of the image registration, blur, additive noise and other kinds of error which are

not explicitly modeled in the fused images) are better modeled by a Laplacian probability density function (PDF) rather than a Gaussian PDF according to [3]. The maximum likelihood estimate of data in the presence of Laplacian noise is obtained through the L_1 -norm minimization. That is why we will use the L_1 -norm function for the data fidelity term:

$$\rho_D(H * I(\mathbf{x}, t) - I(\mathbf{x}, 0)) = |H * I(\mathbf{x}, t) - I(\mathbf{x}, 0)| \quad (6)$$

These ρ -functions are very easy to implement and are very computationally efficient. Other robust ρ -functions could also be employed [13]. We now adapt the PDE in equation 4 locally to the several classes of pixels on the HR grid:

- Class OR: since these pixels contain the least noise as discussed in §3.2, very little regularization has to be applied. This means that these pixels depend mainly on the data fidelity term and thus λ is set to λ_{MAX} .
- Class UN: these pixels are most likely noise and depend only on the regularization term ($\lambda = 0$).
- Class FU: these pixels contain noise and relevant information. The more observations we have, the more robust the initial estimation will be and hence the lesser noise the pixel will contain. That is why we apply equation 4 with λ proportional to the number of available observations α : in our implementation we use $\lambda = \min(\lambda_{\text{MAX}}, \frac{\alpha}{10}\lambda_{\text{MAX}})$. Where 10 is the buffer length of the observations.

Finally the PDE of equation 4 is iteratively applied to update the blurred and noisy image in the restoration process.

4 Experiments and Results

As a first realistic experiment we have both printed and scanned one A4 paper containing the *Lorem ipsum* text with the HP PSC 2175 machine at 75 dpi as shown in figure 4. The Lorem ipsum text is very popular as default model text, but additionally it has a more-or-less normal distribution of letters [16].

The basic structure element was chosen to be a 18×12 rectangular window. We have enlarged the *region of interest* (ROI) with a linear magnification factor of 8 and compared our result with the popular cubic B-spline interpolation technique in figure 4. The parameters for our method are $\sigma_{\text{blur}} = 4.0$, $\tau_{\text{CC}} = 0.6$, $\tau_{\text{MAD}} = 40.0$, $\lambda_{\text{MAX}} = 100$ and 100 iterations for the restoration process. The parameter selection was based on trial and error, i.e. to produce the visually most appealing results. Also the detection rate of the word *leo* was perfectly, this means a recall and a precision of both 100% (on 13 samples). We can clearly see in figure 4 that our method outperforms traditional interpolation techniques: the letters are much better readable and reconstructed, noise and JPEG-artefacts are heavily reduced and much less blurring, staircasing and ringing artefacts are created.

nec turpis nec risus tristique porttitor. In dapibus nisl vel fermentum gravida, ullamcorper at, tortor. Fusce id fel placerat justo porttitor sem. Phasellus tincidunt, eros non vehicula sem mi ut leo. Morbi in ligula at justo rhon adipiscing ultrices leo. In quis sem et nisl aliquam ullam eget ultricies sodales, nunc magna convallis nibh, vel he mi. Integer at tortor in magna dapibus vulputate. Donec e at leo. Nam auctor feugiat justo. Maecenas lacinia con auctor libero vitae enim. Maecenas luctus, ante quis viv mauris sed est. Fusce nulla. In orci eros, elementum et, sc vel mauris. Pellentesque eu purus. Aliquam erat volutpat.

(a)



(b)



(c)



(d)

Fig. 4. Interpolation of the Lorem ipsum text ($8\times$ enlargement): (a) a part of the original scanned text image (at 75 dpi), (b) nearest neighbour interpolation of the ROI of (a), (c) cubic B-spline interpolation, (d) our proposed method (partition of the HR grid: 1.5625% (OR), 14.0625% (FU) and 84.375% (UN))



(a)



(b)



(c)

Fig. 5. Experiment with the scanned Lorem ipsum text ($2\times$ enlargement): (a) original scanned text at 150 dpi, (b) cubic B-spline interpolation of the ROI of figure 4a at 75 dpi, (c) our proposed method applied on the ROI of figure 4a at 75 dpi (partition of the HR grid: 25% (OR), 50% (FU) and 25% (UN))

We have scanned the same text at 150 dpi as shown in figure 5. If we visually inspect the $2\times$ linear enlargement of our method and the cubic B-spline interpolation to this ground truth data, we can conclude that our method manage to reconstruct the characters much better. The images in figure 5 are $4\times$ enlarged with the nearest neighbour interpolation in order to achieve a better visibility.

As a second experiment we have interpolated a real image. In figure 6 we show a part of the original image with a $8\times$ nearest neighbour interpolation of the



(a) A part of the original image.



(b) Nearest neighbour of the region of interest of (a).

Fig. 6. An example with a real image

region of interest. As basic structure elements we use 5×5 windows and we have enlarged the image with a linear magnification factor of 8. For the matching step we have used the following threshold parameters: $\tau_{CC} = 0.9$ and $\tau_{MAD} = 9.0$. For the denoising and deblurring we have applied the PDE within 100 iterations and with $\sigma_{\text{blur}} = 4.0$ and $\lambda_{\text{MAX}} = 10$. With these parameters we obtained the following partition of the different classes: 1.5625% (OR), 30.9091% (FU) and 67.5284% (UN).



(a) Cubic B-spline.



(b) IFS [15].



(c) Our proposed method.

Fig. 7. Results of several interpolation methods for the ROI of figure 6a

Figure 7 shows our result compared to a linear interpolation (cubic B-spline) and an IFS method (obtained from commercial software [15]). Significant improvements in visual quality can be noticed in our method: there is a very good reconstruction of the edges and our result contains much less annoying artefacts. The result produced with our method is also better denoised while important edges are preserved.

5 Conclusion

In this paper we have presented a novel interpolation scheme based on the repetitive character of the image. Exploiting repetitivity brings more information at our disposal, which leads to much better estimates of the unknown pixel values. Results show the effectiveness of our proposed interpolation technique and its superiority at very large magnifications to other interpolation methods: edges are reconstructed well and artefacts are heavily reduced. In special applications with text images, we can achieve excellent results: characters could be made much better readable again. This could be very advantageous for optical character recognition (OCR) applications or when the image resolution can not be improved at the sensor because of technological limitations or because of high costs.

References

1. Allebach, J., Wong, P.W.: Edge-Directed Interpolation. Proc. of IEEE International Conference on Image Processing (1996) 707–710
2. Buades, A., Coll, B., Morel, J.: Image Denoising By Non-Local Averaging. Proc. of IEEE International Conference of Acoustics, Speech and Signal Processing **2** (2005) 25–28
3. Farsiu, S., Robinson, M.D., Elad, M., Milanfar, P.: Fast and Robust Multiframe Super Resolution. IEEE Trans. on Image Processing **13** (2004) 1327–1344
4. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-Based Super-Resolution. IEEE Computer Graphics and Applications **22** (2002) 56–65
5. Honda, H., Haseyama, M., Kitajima, H.: Fractal Interpolation For Natural Images. Proc. of IEEE International Conference of Image Processing **3** (1999) 657–661
6. Ledda, A., Luong, H.Q., Philips, W., De Witte, V., Kerre, E.E.: Image Interpolation Using Mathematical Morphology. Proc. of 2nd IEEE International Conference On Document Image Analysis For Libraries (2006) (to appear)
7. Lehmann, T., Gönner, C., Spitzer, K.: Survey: Interpolations Methods In Medical Image Processing. IEEE Trans. on Medical Imaging **18** (1999) 1049–1075
8. Li, X., Orchard, M.T.: New Edge-Directed Interpolation. IEEE Trans. on Image Processing **10** (2001) 1521–1527
9. Luong, H.Q., De Smet, P., Philips, W.: Image Interpolation Using Constrained Adaptive Contrast Enhancement Techniques. Proc. of IEEE International Conference of Image Processing **2** (2005) 998–1001
10. Meijering, E.H.W., Niessen, W.J., Viergever, M.A.: Quantitative Evaluation Of Convolution-Based Methods For Medical Image Interpolation. Medical Image Analysis **5** (2001) 111–126

11. Morse, B.S., Schwartzwald, D.: Isophote-Based Interpolation. Proc. of IEEE International Conference on Image Processing (1998) 227–231
12. Muresan, D.: Fast Edge Directed Polynomial Interpolation. Proc. of IEEE International Conference of Image Processing **2** (2005) 990–993
13. Pižurica, A., Vanhamel, I., Sahli, H., Philips, W., Katartzis, A.: A Bayesian Approach To Nonlinear Diffusion Based On A Laplacian Prior For Ideal Image Gradient. Proc. of IEEE Workshop On Statistical Signal Processing (2005)
14. Vrhel, M.: Color Image Resolution Conversion IEEE Trans. on Image Processing **14** (2005) 328–333
15. Genuine Fractals 4: <http://www.ononesoftware.com>
16. Lorem Ipsum generator: <http://www.lipsum.com>

A Discontinuous Finite Element Method for Image Denoising

Zhaozhong Wang¹, Feihu Qi², and Fugen Zhou¹

¹ Image Processing Center, Beihang University, Beijing 100083, China

² Dept. Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200030, China

Abstract. Discontinuous finite element methods are powerful mathematical tools for solving partial differential equations with discontinuous solutions. In this paper such a method is presented to denoise digital images, while preserving discontinuous image patterns like edges and corners. This method is theoretically superior to the commonly used anisotropic diffusion approach in many aspects such as convergence and robustness. Denoising experiments are provided to demonstrate the effectiveness of this method.

1 Introduction

Preserving sharp image patterns such as contours and corners is a major challenge in image denoising and other related processing issues. Partial differential equations (PDEs) have raised a strong interest in the past decades because of their ability to smooth data while preserving discontinuities of images. Many PDE schemes have been presented so far in the literature [1,2,3,4]. It can be found that almost all of them belong to the class of (isotropic and anisotropic) diffusion differential equations.

Let $u : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ be a 2D scalar image defined on the image domain Ω . The image denoising problem via a diffusion equation can be formulated as:

$$\frac{\partial u}{\partial t} = \nabla \cdot (c \nabla u), \quad u_0 = f, \quad (1)$$

where u_0 is the initial value of u at time $t = 0$, f the initial noisy image, and c denotes the diffusion coefficient or diffusion tensor. If c is constant, the equation is isotropic. Equations like (1) have been successfully used in many problems including image denoising [3,4].

However, some inherent drawbacks of the anisotropic diffusion equation (1) was discovered in the past. Mathematically speaking, Eq. (1) is a nonlinear parabolic equation. To achieve the convergence of such an equation (i.e., to achieve that $\frac{\partial u}{\partial t} = 0$), it may need infinite steps of iteration in theory. But the large number of iterations can blur image edges heavily, losing the property of preserving sharp structures. So Eq. (1) generally does not converge towards an interesting solution. On the other hand, less number of iterations may leave noise

unremoved. Thus the performance of the anisotropic diffusion is very sensitive to algorithm parameters like the number of iterations.

With the aim of avoiding the above drawbacks of the anisotropic diffusion approaches, we solve the denoising problem by developing a completely new algorithm based on discontinuous finite element methods (DFEMs). Discontinuous finite element methods belong to a subclass of finite element methods (FEMs), which are frequently used to solve PDEs. Solutions of common FEMs are continuous, and cannot preserve discontinuities like image edges. But DFEMs are preferred over more standard continuous finite element methods because of their capability in preserving discontinuous patterns in solutions.

DFEMs have become very widely used for solving a large range of computational fluid problems [5]. Here we shall use them in the filed of image processing. Aided by this kind of methods, we can model the image denoising problem using a linear elliptic equation, which is simpler and more effective than the anisotropic diffusion (nonlinear parabolic) equations. In the following parts, we first give this model in Section 2, then introduce the DFEM method in Section 3. Section 4 provides experiments and discussions, and the last section draws the conclusion.

2 Variational and PDE Formulations of Denoising

We still denote u as the solution image and f the input noisy image. Image denoising can be formulated as the following variational problem:

$$\min_{u: \Omega \rightarrow \mathbb{R}} \int_{\Omega} [(u - f)^2 + \lambda \|\nabla u\|^2] dx, \quad (2)$$

where λ is a penalty parameter, which controls the strength of the smoothness constraint term. A larger λ imposes stronger smoothness on the solution. Both the two terms in Eq. (2) use the L^2 -like norms, which will result in a linear PDE. Using other norms may reduce to a nonlinear anisotropic equation, e.g. using the L^1 norm generates a total variation formula [6].

Now we derive the PDE model based on Eq. (2). It is known that the Euler-Lagrange equation

$$\frac{\partial F}{\partial u} - \nabla \cdot \left[\frac{\partial F}{\partial u_x} \quad \frac{\partial F}{\partial u_y} \right]^T = 0, \quad (3)$$

where $F = (u - f)^2 + \lambda \|\nabla u\|^2$, gives a necessary condition that must be verified by u to reach a minimum of the above functional. Equation (3) reduces to the following PDE

$$-\lambda \Delta u + u = f, \quad (4)$$

where $\Delta := \nabla^2$ is the Laplace operator. Note that this equation does not declare boundary conditions. We shall impose the Dirichlet boundary condition $u = g$ on $\partial\Omega$, where g may take as the 1D limit of the original image on the boundary, i.e. $g = f|_{\partial\Omega}$, or take as a 1D smoothed version of $f|_{\partial\Omega}$ to reduce noise on boundary. Thus we have the complete problem:

$$-\lambda \Delta u + u = f \quad \text{in } \Omega, \quad u = g \quad \text{on } \partial\Omega. \quad (5)$$

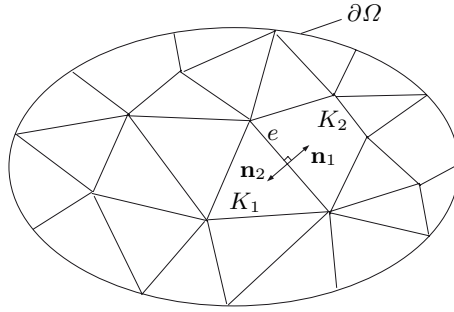


Fig. 1. Illustration for the geometry of (discontinuous) finite element methods

This is the linear elliptic equation we need.

Most diffusion equations in the literature were derived from a similar way as the above [3]. But after an elliptic equation like (4) hading been obtained, an auxiliary time variable t was introduced and the *gradient descent* method was used to achieve a parabolic equation like (1). In this paper, we do not need to introduce the auxiliary time variable, and shall solve directly the elliptic equation (5) using algorithms based on finite element.

3 Finite Element and Discontinuous Finite Element Methods

In this section, we first review basic concepts of typical finite element methods, then introduce the discontinuous finite element methods, which has a major advantage of preserving discontinuities in solutions.

3.1 Review of Finite Element Methods

Finite element methods [7] are viewed as a procedure for obtaining numerical approximations to the solution of continuous problems posed over a domain Ω . This domain is replaced by the union of disjoint sub-domains called *finite elements*. Such a subdivision operation of the domain is known as a *triangulation*. The triangulation of the space results in a *mesh* and the vertices of the element are *nodes*. See Fig. 1 for illustration.

The unknown function (or functions) is locally approximated over each element by an interpolation formula expressed in terms of values taken by the function(s), and possibly their derivatives, at a set of nodes. The states of the assumed unknown function(s) determined by unit nodal values are called *shape functions*. A continuous physical problem is transformed into a discrete finite element problem with unknown nodal values. For a linear problem a system of linear algebraic equations should be solved. Then values inside finite elements can be recovered using nodal values.

A discrete finite element model may be generated from a variational or a *weak form* of the continuous mathematical model. For example, the weak form for the FEM of Eq. (4) can be obtained in the following way: We multiply Eq. (4) by a *test function* v which vanishes on $\partial\Omega$, and integrate over Ω , then apply the Green's formula on the first term of the left-hand side, to obtain

$$\int_{\Omega} \lambda \nabla u \cdot \nabla v \, d\mathbf{x} + \int_{\Omega} u v \, d\mathbf{x} = \int_{\Omega} f v \, d\mathbf{x}. \tag{6}$$

Then in the FEM it needs to find an approximate solution u_h on the mesh which satisfy the above weak form for any test function v . The weak form can in turn be reduced to a linear system of equations for solving. This way of defining an approximate solution is referred to as the *Galerkin finite element method*.

3.2 Discontinuous Finite Element Methods

In 1973, Reed and Hill [8] introduced the first *discontinuous Galerkin method* (DGM) for hyperbolic equations of neutron transport problems, and since that time there has been an active development of DGMs for hyperbolic and nearly hyperbolic problems. Recently, these methods also have been applied to purely elliptic problems. Also in the 1970s, Galerkin methods for elliptic and parabolic equations using discontinuous finite elements were independently proposed and a number of variants were introduced and studied [9,10]. These discontinuous methods were then usually called *interior penalty* (IP) methods, and their development remained independent of the development of the DGMs for hyperbolic equations.

For elliptic problems, it was then found that the DGMs and the IP methods can be unified in the same framework [11]. They all belong to discontinuous finite element methods (DFEMs). The difference between DFEMs and FEMs mainly lies in that DFEMs allow function values to be discontinuous across adjoining elements, but this is not allowed in FEMs. To describe this property of DFEMs, some notations need to be introduced and described in more detail.

Denote by \mathcal{T}_h a triangulation of Ω in polygons K , and by $P(K)$ a finite dimensional space of smooth functions, typically polynomials, defined on the polygon K . This space will be used to approximate the variable u . Specifically, the space is given by

$$V_h := \{v \in L^2(\Omega) : v|_K \in P(K), \forall K \in \mathcal{T}_h\},$$

where $L^2(\Omega)$ denotes the space of squared integrable functions on Ω . Note that $L^2(\Omega)$ allows its functions to be discontinuous.

Referring to Fig. 1, let e be an interior edge shared by elements K_1 and K_2 . Define the unit normal vectors \mathbf{n}_1 and \mathbf{n}_2 on e pointing exterior to K_1 and K_2 , respectively. It is easy to know that $\mathbf{n}_1 = -\mathbf{n}_2$. If v is a scalar function on $K_1 \cup K_2$, but possibly discontinuous across e , denoting $v_i := (v|_{K_i})|_e, i = 1, 2$, then we can define

$$\{v\} := \frac{1}{2}(v_1 + v_2), \quad \llbracket v \rrbracket := v_1 \mathbf{n}_1 + v_2 \mathbf{n}_2,$$

where $\{\!\{v\}\!\}$ and $\llbracket v \rrbracket$ are called the *average* and *jump* of the function v [11], respectively. If \mathbf{q} is a vector-valued function on $K_1 \cup K_2$, we then have

$$\{\!\{\mathbf{q}\}\!\} := \frac{1}{2}(\mathbf{q}_1 + \mathbf{q}_2), \quad \llbracket \mathbf{q} \rrbracket := \mathbf{q}_1 \cdot \mathbf{n}_1 + \mathbf{q}_2 \cdot \mathbf{n}_2.$$

Note that the jump $\llbracket v \rrbracket$ of the scalar function v is a vector, and the jump $\llbracket \mathbf{q} \rrbracket$ of the vector function \mathbf{q} is a scalar. The advantage of these definitions is that they do not depend on assigning an ordering to the element K_i [11]. For an edge e on the boundary, i.e., $e \subset \partial\Omega$, we define

$$\llbracket v \rrbracket := v\mathbf{n}, \quad \{\!\{\mathbf{q}\}\!\} := \mathbf{q},$$

where \mathbf{n} is the outward unit normal to the boundary edge. The quantities $\{\!\{v\}\!\}$ and $\llbracket \mathbf{q} \rrbracket$ on boundary edges are undefined, since they are unnecessary in this paper.

There are many formulations for the DFEMs. Here we shall use the nonsymmetric interior penalty formulation [12]. The weak form of this DFEM for Eq. (5) can be stated as: Find $u_h \in V_h$ such that

$$\begin{aligned} & \sum_{K \in \mathcal{T}_h} \int_K (\lambda \nabla u_h \cdot \nabla v_h + u_h v_h) \, d\mathbf{x} \\ & + \lambda \sum_{e \in \mathcal{E}_h} \int_e (\mu_e \llbracket u_h \rrbracket \cdot \llbracket v_h \rrbracket - \{\!\{\nabla u_h\}\!\} \cdot \llbracket v_h \rrbracket + \llbracket u_h \rrbracket \cdot \{\!\{\nabla v_h\}\!\}) \, ds \\ & = \int_{\Omega} f v_h \, d\mathbf{x} + \lambda \int_{\partial\Omega} (\mu_e g v_h + g \nabla v_h \cdot \mathbf{n}) \, ds \end{aligned} \quad (7)$$

for all $v_h \in V_h$, where \mathcal{E}_h denotes the set of all edges. In this equation, the penalty weighting parameter μ_e is given by η/h_e on each edge $e \in \mathcal{E}_h$ with η a positive constant and h_e the edge size. The derivation of the above equation is omitted here, see [12,11] for detail. It is proven that this formulation achieves optimal rates of convergence by using large penalty terms [13].

Comparing Eq. (7) with (6), it can be found that the major change from FEMs to DFEMs is the appearance of the integral terms on all edges. So the discontinuities across edges are taken into consideration, and will be preserved in the solving process. The last integral on the right-hand side of (7) imposes the Dirichlet boundary condition.

For the solution is meaningful to the image denoising problem, we should preserve image discontinuous features. Thus the finite element edges should coincide with image object edges. This can be realized by a mesh refinement procedure, which splits initial coarse meshes containing image edges to finer ones. The Hessian of the solution [14] can be used as a criterion to split meshes.

3.3 Solution from Linear System of Equations

Equation (7) can be reduced to a linear system of algebraic equations. First, we have the finite element expression:

$$u_h(\mathbf{x}) = \sum_{i=1}^{N_h} \alpha_i \phi_i(\mathbf{x}), \quad v_h(\mathbf{x}) = \sum_{j=1}^{N_h} \beta_j \phi_j(\mathbf{x}), \quad (8)$$

where α_i and β_j are unknown coefficients, N_h is equal to the *degrees of freedom* of the problem, and $\phi_i(\mathbf{x})$ are basis functions (i.e. shape functions), which are defined in terms of polynomials on \mathbb{R}^2 , such as the P_1 polynomials (with degrees ≤ 1). For each element K_k , the P_1 basis functions $\phi_i(\mathbf{x})$ are given by

$$\begin{aligned} \phi_i(\mathbf{x}) &= \phi_i(x, y) = a_i^k + b_i^k x + c_i^k y, \quad \mathbf{x} \in K_k \\ \phi_i(\mathbf{p}^i) &= 1, \quad \phi_i(\mathbf{p}^j) = 0 \text{ if } i \neq j, \end{aligned}$$

where a_i^k , b_i^k and c_i^k are known coefficients, and \mathbf{p}^i the vertices of K_k .

Aided by (8), Eq. (7) can be written as the following matrix-vector notation:

$$\mathbf{v}^T B \mathbf{u} = \mathbf{v}^T \mathbf{f}, \quad (9)$$

where the *stiffness matrix* B and the vector \mathbf{f} are both derived from the integrals with respect to the basis functions $\phi_i(\mathbf{x})$, and the vectors \mathbf{u} and \mathbf{v} are defined as

$$\mathbf{u} := [\alpha_1 \ \alpha_2 \ \cdots \ \alpha_{N_h}]^T, \quad \mathbf{v} := [\beta_1 \ \beta_2 \ \cdots \ \beta_{N_h}]^T.$$

Since Eq. (9) holds for any \mathbf{v} , it can be reduced to the linear system of equations

$$B \mathbf{u} = \mathbf{f}. \quad (10)$$

Detailed discussions for this equation and the corresponding (7) on the existence and uniqueness of solutions can be found in [12,11]. Since the matrix B in Eq. (10) is sparse and nonsymmetric, we use a UMFPAK algorithm [15] to solve the equations. Thus we obtain the coefficients α_i , then using Eq. (8) we finally get the solution $u_h(\mathbf{x})$.

4 Experiments and Discussions

We now test our algorithm using real image data. We set the parameters in Eq. (7) as follows: $\lambda = 0.1$, and $\mu_e = 1.0 \times 10^5 / h_e$ with h_e the element edge size. Fig. 2 is an example for Lena image, where we compare our algorithm with the state of the art, e.g. the method of Tschumperle and Deriche [4]. Note that their method is primarily designed for handling color images, but can also be used for scalar images, and reports almost the best results at the present.

It can be seen that our algorithm obtains a denoising result similar to that of [4], both of the two preserve the edge information of Lena. Fig. 2(d) shows that our algorithm performs even better than [4] in the balance of smoothing

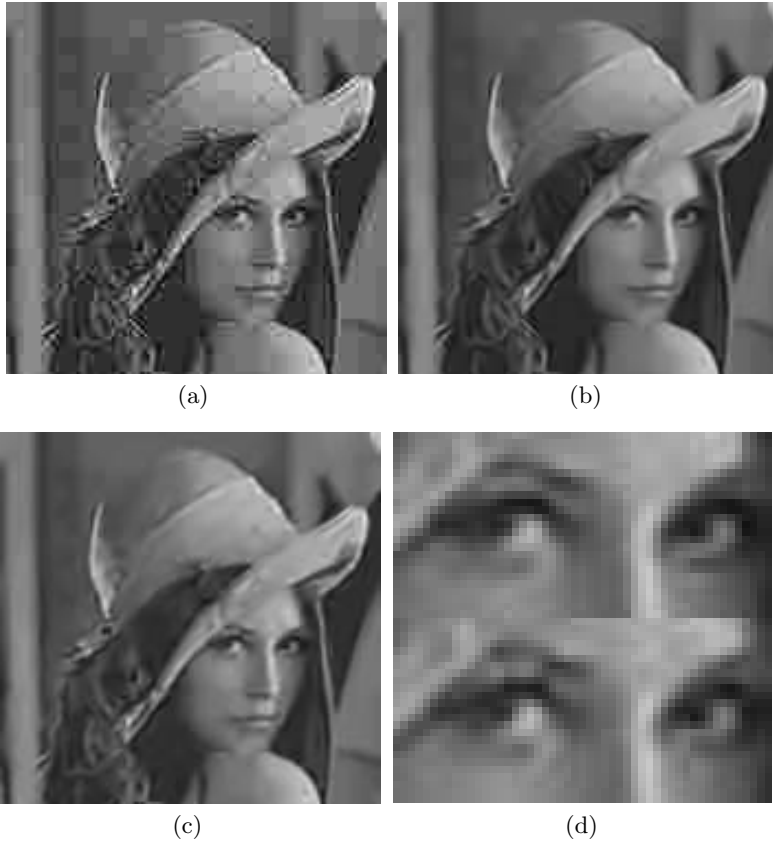


Fig. 2. Denoising Lena image. (a) The original image disturbed by block effects in decompressing. (b) The result of [4]. (c) The result of our algorithm. (d) The top is the local enlargement of (b), and the bottom is that of (c).

noise and preserving discontinuities. The result of [4] still has some block effects; smoothing them may cause the blur of edges. In Fig. 3, we depict the adaptively refined mesh by our DFEM for denoising the Lena image. Near object edges, the mesh elements are of small size, and the discontinuities of the solution are maintained. Fig. 4 provides another example for denoising the car image. The quality of our result is similar to the state of the art.

Based on the experiments, we now discuss and compare the DFEM with the diffusion-based methods, such as those in [3,4].

First, the DFEM has the inherent feature of preserving solutions' discontinuities. However, anisotropic diffusion actually smooths edges, though in a small quantity. After a large number of iterations, the edges are blurred. An inverse diffusion process may deblur and enhance edges, but it is unstable in theory. Second, the DFEM has a good property of convergence, which is theoretically guaranteed. But it is difficult to determine the convergence criteria for diffusion

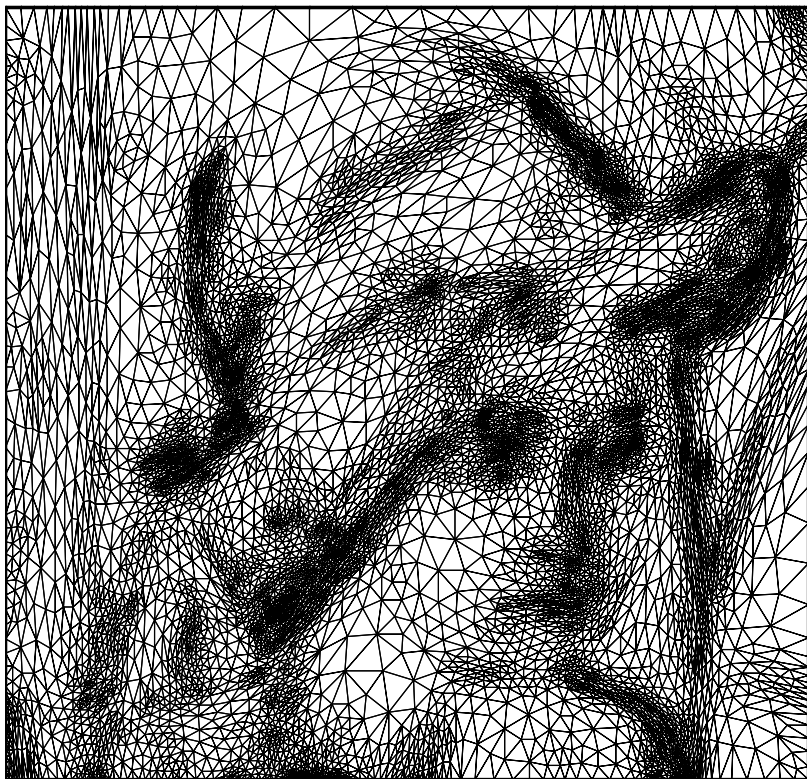


Fig. 3. The adaptively generated finite element mesh for denoising the Lena image

equations, and their solutions are sensitive to iteration steps. Third, the DFEM (including FEM) is theoretically faster than the finite difference scheme used in anisotropic diffusions, since it needs not to perform computations on every pixels. Only values at element vertices are set as unknowns and to be solved, the function values within elements can then be interpolated. In a flat image region, only a few elements are sufficient to achieve an accurate solution. And last, the DFEM uses integral formulations, rather than image derivative information, so it is more robust to noise than diffusion equations based on image gradient. In addition, there is no obvious boundary effects in the DFEM, but diffusion-based methods suffer from gradient errors near image boundary.

5 Conclusion and Future Work

In this paper we propose a new denoising algorithm based on the discontinuous finite element method. The denoising quality of our algorithm is similar to the state of the art using anisotropic diffusions. But the convergence performance of our algorithm is better, since it is not sensitive to algorithm parameters such

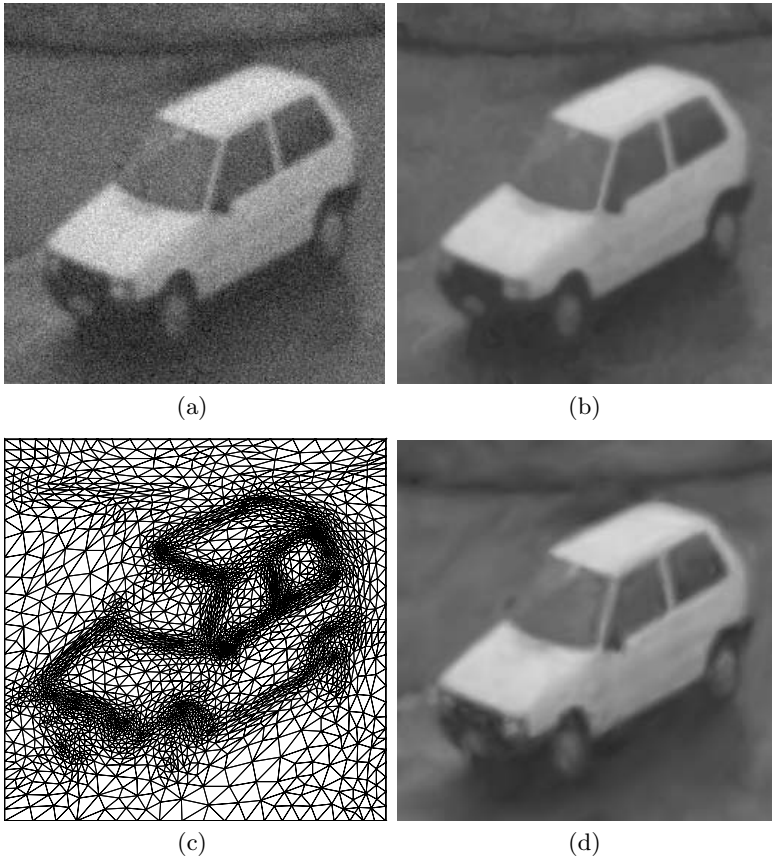


Fig. 4. Denoising Car image. (a) The original noisy image. (b) The result of [4]. (c) The mesh generated by the DFEM. (d) The result of the DFEM.

as the number of iterations. In addition, the algorithm is potentially faster and more robust than the diffusion-based approaches.

Benefit from the DFEM, we use a linear elliptic PDE for the edge-preserved denoising. But it should be emphasize that DFEMs are also applicable to non-linear equations [12]. We shall study this situation, and generalize the algorithm to color image denoising problems.

References

1. Aubert, G., Kornprobst, P.: *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*. Volume 147 of Applied Mathematical Sciences. Springer-Verlag (2002)
2. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Analysis and Machine Intelligence* **12** (1990) 629–639

3. Sapiro, G.: Geometric Partial Differential Equations and Image Analysis. Cambridge University Press, Cambridge, UK (2001)
4. Tschumperle, D., Deriche, R.: Vector-valued image regularization with PDE's: A common framework for different applications. *IEEE Trans. Pattern Analysis and Machine Intelligence* **27** (2005) 506–517
5. Cockburn, B., Karniadakis, G., Shu, C.W., eds.: Discontinuous Galerkin Methods: Theory, Computation and Applications. Springer-Verlag, New York (2000)
6. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D* **60** (1992) 259–268
7. Bathe, K.J.: Finite Element Procedures. Prentice-Hall, Englewood Cliffs (1996)
8. Reed, W.H., Hill, T.R.: Triangular mesh methods for the neutron transport equation. Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, Los Alamos, NM (1973)
9. Douglas, J., Dupont, T.: Interior penalty procedures for elliptic and parabolic Galerkin methods. In: *Lecture Notes in Phys.* 58, Berlin, Springer-Verlag (1976)
10. Arnold, D.N.: An Interior Penalty Finite Element Method with Discontinuous Elements. PhD thesis, The University of Chicago, Chicago, IL (1979)
11. Arnold, D.N., Brezzi, F., Cockburn, B., Martin, L.D.: Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numerical Analysis* **39** (2002) 1749–1779
12. Rivière, B., Wheeler, M.F., Girault, V.: A priori error estimates for finite element methods based on discontinuous approximation spaces for elliptic problems. *SIAM J. Numerical Analysis* **39** (2001) 902–931
13. Castillo, P.: Performance of discontinuous Galerkin methods for elliptic PDE's. *SIAM J. Scientific Computing* **24** (2002) 524–547
14. George, P.L.: Automatic triangulation. Wiley & Sons, New York (1996)
15. Davis, T.A.: Algorithm 8xx: UMFPACK v4.1, an unsymmetric-pattern multi-frontal method TOMS. <http://www.cise.ufl.edu/research/sparse/umfpack> (2003)

An Edge-Preserving Multigrid-Like Technique for Image Denoising

Carolina Toledo Ferraz, Luis Gustavo Nonato, and José Alberto Cuminato

São Paulo University - ICMC, PO Box 668, 13560-970, São Carlos - SP, Brazil

Abstract. Techniques based on the Well-Balanced Flow Equation have been employed as efficient tools for edge preserving noise removal. Although effective, this technique demands high computational effort, rendering it not practical in several applications. This work aims at proposing a multigrid like technique for speeding up the solution of the Well-Balanced Flow equation. In fact, the diffusion equation is solved in a coarse grid and a coarse-to-fine error correction is applied in order to generate the desired solution. The transfer between coarser and finer grids is made by the Mitchell-Filter, a well known interpolation scheme that is designed for preserving edges. Numerical results are compared quantitative and qualitatively with other approaches, showing that our method produces similar image quality with much smaller computational time.

1 Introduction

Over the years the use of partial differential equations (PDE) in image denoising has grown significantly. This research area started with the formulation proposed by Marr and Hildreth [10] that introduced the concept of optimal filter smoothing through Gaussian kernels and the heat equation. From this seminal work, more sophisticated models have been derived, as the Perona-Malik model [13], selective smooth by nonlinear diffusion model [2] and the Well-Balanced Flow Equation model (WBFE) [4]. These models are able to process an image while keeping edges and small features, a desirable property to many applications. An important feature of all these models is that they consider image evolution as a time parameter, representing the image at multiple scales [9].

Depending on the equation to be solved and the set of associated parameters, the solution of the PDE involved in such modeling can demand high computational effort, impairing the application of such methods in problems that require fast processing.

This work is concerned with the problem of reduction of the computational time involved in solving PDE's in image processing. In fact, we present a new scheme that speeds up the solution of evolution equations in the image processing context. Based on Multigrid techniques [5], our approach employ a hierarchical structure so as to solve the equations in different image resolutions, using the results in a resolution to solve the discrete equation in the next one. Besides

improving the computational time, the proposed method reduces artifacts associated with the iterations [1], eliminating high frequency errors (noise) and low frequency errors such as blotches and false edges [11].

In order to illustrate the effectiveness of the proposed framework in practical applications, we have employed it in an edge preserving noise elimination problem. The Well-Balanced Flow Equation, by Barcelos et al. [4], has been chosen as the PDE model due to the good results obtained with this equation in preserving edges and the high computational cost involved in its solutions.

Comparisons in terms of computational time as well as image quality between the conventional implementation and the proposed method for WBFE is presented in section 4. Before presenting the results we discuss, in section 2, the properties of WBFE. Section 3 is devoted to describing our approach. Conclusion and future work are in section 5.

2 Well-Balanced Flow Equation (WBFE)

As mentioned before, we shall adopt the Well-Balanced Flow Equation as the test model for validating our approach. The WBFE is based on the Perona-Malik model [13], and considers an image as a function $I(\mathbf{x}) : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$, that can be obtained from the solution of the following equation:

$$u_t = \text{div}(C(|\nabla u|)\nabla u), \quad \text{in } \Omega \times R_+ \quad u(\mathbf{x}, 0) = \tilde{I}(\mathbf{x}) \tag{1}$$

where C is a nonincreasing smooth function such that $C(0) = 1$, $C(s) \geq 0$, $C(s) \rightarrow 0$ when $s \rightarrow \infty$, and $\tilde{I}(x)$ is the original degraded image.

The function C aims at reducing the diffusion where $|\nabla u|$ is large, thus preserving edges. An example for the function C is

$$C(|\nabla u|) = \frac{1}{1 + \frac{|\nabla u|^2}{k^2}}$$

where k is a constant intrinsically related to the amount of detail we want to preserve.

The Perona-Malik model produces bad results when the image is very noisy, as the function C will be close to zero in almost every point, keeping the noise during the process.

In order to overcome this drawback, Barcelos et al. [4] proposed the Well-Balanced Flow equation for image recovery, which can be written as:

$$\begin{aligned} u_t &= g|\nabla u|\text{div}\left(\frac{\nabla u}{|\nabla u|}\right) - (1 - g)(u - \tilde{I}), \quad \mathbf{x} \in \Omega, \quad t > 0, \\ u(\mathbf{x}, 0) &= \tilde{I}(\mathbf{x}), \quad \mathbf{x} \in \Omega \\ \frac{\partial u}{\partial n} &|_{\partial\Omega \times R_+} = 0, \quad \mathbf{x} \in \partial\Omega, \quad t > 0 \end{aligned}$$

where $g = g(|G_\sigma * \nabla u|)$, G_σ is a Gaussian function and $G_\sigma * \nabla u$ is the local estimate of ∇u used for noise elimination.

The term $|\nabla u|\text{div}(\nabla u/|\nabla u|) = \Delta u - \nabla^2 u(\nabla u, \nabla u)/|\nabla u|^2$ diffuses u in the orthogonal direction to its gradient ∇u , preserving thus the edges. $g(|G_\sigma * \nabla u|)$

is used for edge detection in order to control the diffusion behavior. In fact, a small value of $|G_\sigma * \nabla u|$ means that ∇u is also small in a neighborhood of a point \mathbf{x} , so \mathbf{x} is not close to an edge. On the other hand, when $|G_\sigma * \nabla u|$ is large, i.e., the average of ∇u is large, the point \mathbf{x} is close to an edge. As $g(s) \rightarrow 0$, $s \rightarrow \infty$, a “weak” diffusion is applied in regions close to edges.

The term $(1-g)(u-\tilde{I})$ intends to keep the smoothed image close to the initial image \tilde{I} in the areas where $g \sim 0$. On the other hand, in homogeneous areas, where $g \sim 1$, the forcing term will have an inexpressive effect, which allows for a better suavization of the image.

3 Multigrid Based Approach

Multigrid methods refer to a family of iterative techniques for solving linear systems that are, in general, obtained from the discretization of partial differential equations [5]. Multigrid methods are based on two main ideas: nested iteration and coarse grid correction. Nested iteration means that an approximation to the solution is computed in different grid resolution and coarse grid correction means that the error in coarser grids is used to improve the solution in finer grids.

An important property of multigrid methods is that the exchanges between fine-to-coarse and coarse-to-fine grids are made by smooth interpolation functions, eliminating high-frequency components. Furthermore, coarse grid correction is expected to reduce the low-frequency components without introducing new high-frequency ones.

The above characteristics provides the motivation for adapting it to the image processing context. How such adaption has been conducted is described below.

3.1 Overview of the Proposed Method

Before presenting the complete description of the method, we introduce some notation that will be useful in the remaining text.

Let $(\cdot)_\uparrow$ denote the interpolation operator from the coarse to the fine grid and $(\cdot)_\downarrow$ denote the restriction operator (fine-to-coarse). Let G^h , $h = 0, \dots, m$ be a set of grids where G^0 is the original grid (finest grid corresponding to the original image) and G^m the coarsest grid employed. We denote by $E^h = u^h - I^h$ the error estimate in G^h , where u^h and I^h are the approximated and the exact solutions in G^h .

Given u^h in G_h , let $u^{h+1} \leftarrow (u^h)_\downarrow$ be the restriction of u^h in G^{h+1} and u_b^{h+1} and u_a^{h+1} be approximations of I before and after applying the diffusion process in level $h + 1$.

From the above definitions we can state our multigrid based method as follows:

1. $u_b^1 \leftarrow (\tilde{I}^0)_\downarrow$
2. For $h = 1$ to $m - 1$
 - a) Compute u_a^h (from eq. 2)
 - b) $u_b^{h+1} \leftarrow (u_a^h)_\downarrow$

3. $E^m = u_b^m - u_a^m$
4. For $h = m$ to 2
 - a) $E^{h-1} \leftarrow (E^h)_\uparrow$
 - b) $u_b^{h-1} \leftarrow u_a^{h-1} + E^{h-1}$
 - c) Compute u_a^{h-1} (from eq. 2)
 - d) $E^{h-1} = u_b^{h-1} - u_a^{h-1}$
5. $u^0 \leftarrow (u_a^1)_\uparrow$

The above algorithm deserves some comments. In steps 2a and 4c the diffusion process governed by equation (2) is applied in a discretized form to produce the approximation u^h . The time derivative in equation 2 is discretized by the explicit Euler's method, and the second order spatial derivatives are discretized by central differences. Other critical points of the algorithm are the interpolation and restriction operators (steps 1, 2b, 4a, and 5), which will be discussed in the next sub-section.

It is important to note that the computational cost of the diffusion process is reduced because equation (2) is solved only in the coarse grid. Furthermore, the number of iterations employed to compute the approximation in each level is usually much smaller than the number of iterations needed to solve equation (2) directly on the original grid. For example, the number of iterations employed by Barcelos [4] is about 200 (depending on the image). In our implementation we have applied about 20 iteration per level, drastically reducing the computational time. The time complexity for our algorithm is $\left((K + n_{it}) \left(\frac{2^{m-1}-1}{2^{m-1}} \right) + \frac{n_{it}}{2^m} \right) O(N^2)$, where N is the image size, K is the mask size, n_{it} is the number of iterations and m is the number of levels. We point out that the complexity constant for our method is much smaller than that in the conventional method ($n_{it}O(N^2)$).

3.2 Interpolation and Restriction Filters

The interpolation and restriction operations employed in steps 1, 2a, 4a and 5 of the above algorithm are of primordial importance in our context, as we intend to preserve as much as possible, the original edges in the processed image. Therefore, an appropriate interpolation function must be chosen so as to properly perform the interpolation and restriction operations.

It is not difficult to see that interpolation and restriction operations are closely related to image resampling. Therefore, the functions usually employed in resampling are good candidates for interpolation, as for example cubic splines. Such functions in the image resampling context [7,8], are defined as a weighted sum of pixel values where the weights are derived from kernel functions, as for example:

$$k(t) = \frac{1}{6} \begin{cases} (12-9B-6C)|t^3| + (-18+12B+6C)|t^2| + (6-2B) & \text{if } |t| < 1 \\ (-B-6C)|t^3| + (6B+30C)|t^2| + (-12B-48C)|t| + (8B+24C) & \text{if } 1 \leq |t| < 2 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Notice that by changing the values of B and C in (2) we obtain different cubic filters. For example, $B = 1$ and $C = 0$ give rise to cubic B-splines [7], $B = 0$ generates an one-parameter family of cardinal cubics [8], and $C = 0$ generates Duff's tensioned B-Splines [6]. Michell and Netravali [12] investigated the design of cubic filters in image synthesis, concluding that $B = C = 1/3$ yields the best quality images.

In fact, the Michell filter turned out to be very robust in the tests we carried out, showing its effectiveness in keeping edges.

4 Results

This section is devoted to presenting the results obtained with the proposed multigrid based method, comparing it with the conventional implementation (see [4]) of equation (2). A comparison between the Michell filter [12] and other interpolation kernels is also presented.

Our experiments make use of 256×256 synthetic and MRI images. We limit the number of grids in $m = 5$, thus the coarsest grid contains 16×16 pixels. This limit was chosen in such way that the image do not lose resolution completely.

In equation 2 we take $g(|G_\sigma * \nabla u|) = (1/(1 + k|G_{\sigma_r} * \nabla u|^2))$, where σ_r is the standart deviation of \tilde{I} . The parameter k has been chosen as in [3], that is:

$$k(\sigma) = \begin{cases} 1277.175e^{0.040104385\sigma} & \text{if } 0 \leq \sigma \leq 201 \\ 2429901.476e^{0.004721247\sigma} & \text{if } 0 < \sigma \leq 305 \end{cases}$$

The degraded images \tilde{I} has been obtained by adding noise to each pixel of the original image I . The noise is computed so as to produce a desired SNR (Signal to Noise Ratio), given by:

$$SNR = 10 \log_{10} \left(\frac{\text{Variance of the original image}}{\text{Variance of the noisy image}} \right) dB$$

Our first experiment purportes showing the effectiveness of our approach in keeping edges. Images of geometric objects (figure 1a) and text characters (figure 2a) have been chosen because edges can easily be recognized from these pictures. Figures 1b) and 2b) are the degraded version of 1a) and 2a), where a gaussian noise level of SNR=5.3727 db and SNR=2.2822 db have been applied, respectively. Figures 1c) and 2c) show reconstructed images by the conventional implementation of equation (2) and figures 1e) and 2e) present the result of applying our multigrid based approach with the Mitchell-Filter to figures 1b) and 2b). Comparing figures 1c) and 2c) with figures 1e) and 2e) we can see that the quality of the images produced by our method is similar to the conventional approach. Furthermore, the edges are preserved very satisfactorily, as can be observed from figures 1f) and 2f) (compare with figures 1d) and 2d). Computational times involved in both approaches can be seen in table 1. The third and fifth columns in this table show the number of iterations taken by the two methods. Iterations per level mean that in each level

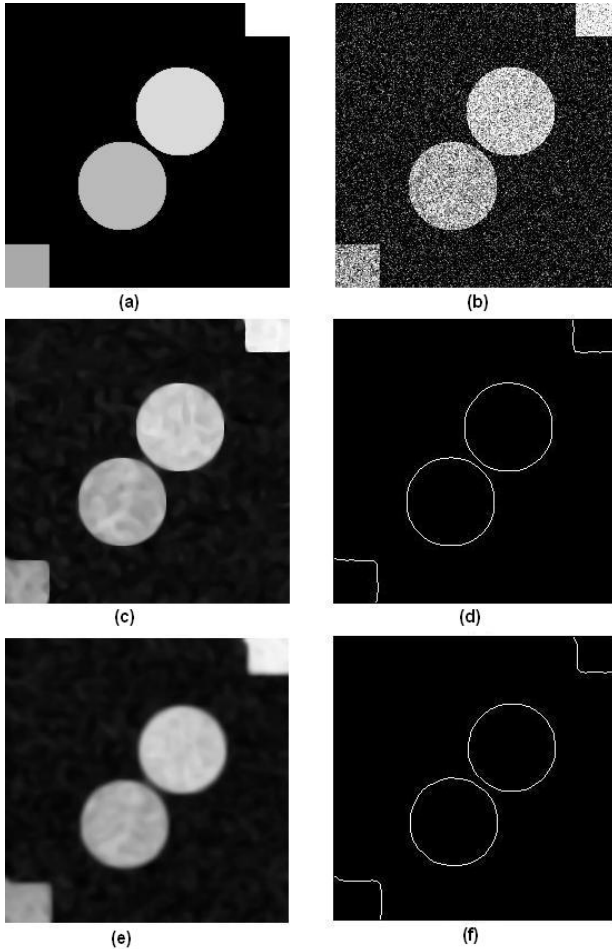


Fig. 1. (a) Original image; (b) Noisy image (SNR=5.3727 db); (c) Reconstructed image using the Well-Balanced Flow Equation (traditional approach); (d) Its segmentation; (e) Reconstructed image by the proposed method; (f) Its segmentation

of the multigrid method the indicated number of iteration is performed. Notice that the multigrid based method is about 7 times faster than the conventional implementation.

Figures 3 and 4 show the behavior of our method in MRI images. Figures 3a) and 4a) present MRI images of a papaia fruit and a knee, respectively. The corresponding degraded images by added Gaussian noise are shown in figures 3b) (SNR=3.3236 db) and 4b) (SNR=4.0495 db). As in the previous experiment, the images obtained by our method (figures 3e) and 4e) and by the conventional implementation (figures 3c) and 4c)) are quite similar. The effectiveness of multigrid in preserving edges can be seen in figures 3f) and 4f).

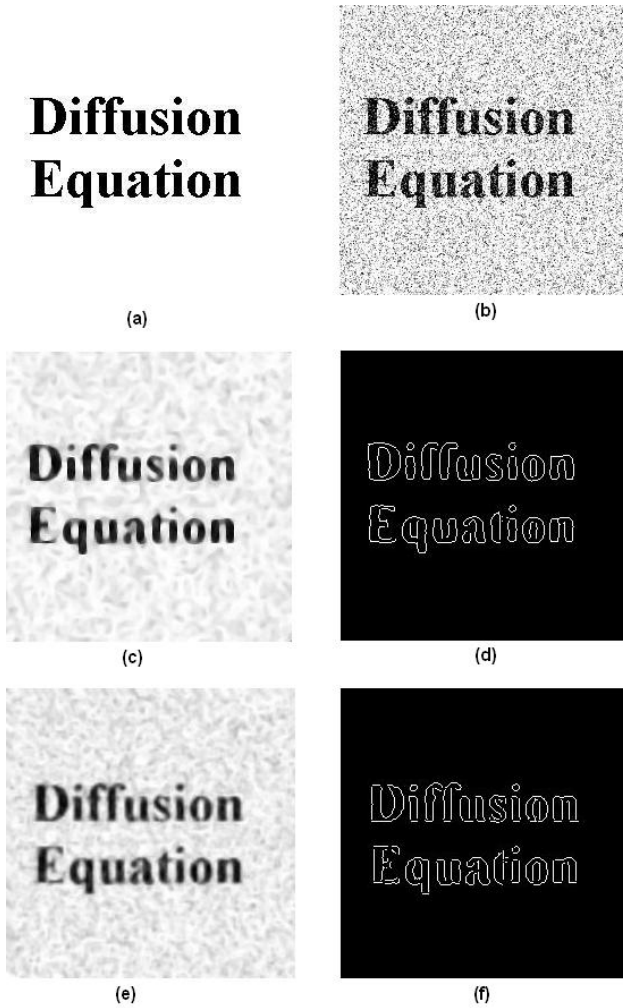


Fig. 2. (a) Original image; (b) Noisy image (SNR=2.2822 db); (c) Reconstructed image using the Well-Balanced Flow Equation (traditional approach); (d) Its segmentation; (e) Reconstructed image by the proposed method; (f) Its segmentation

Computational times are presented in the third and fourth rows of table 1. Notice that a six fold speed up has been obtained by the multigrid based method.

We also tested our multigrid based method adding other types of noise to the image. In figures 5a) and 5d) multiplicative and salt & pepper noise were added respectively. Figures 5b) and 5e) present the results of applying our approach with the Mitchell-Filter. The preservation of edges are shown in figures 5c) and 5f). As can be noticed, with only 15 iterations starting from figure 5a) and 5d) our method eliminated these types of noise from the image.

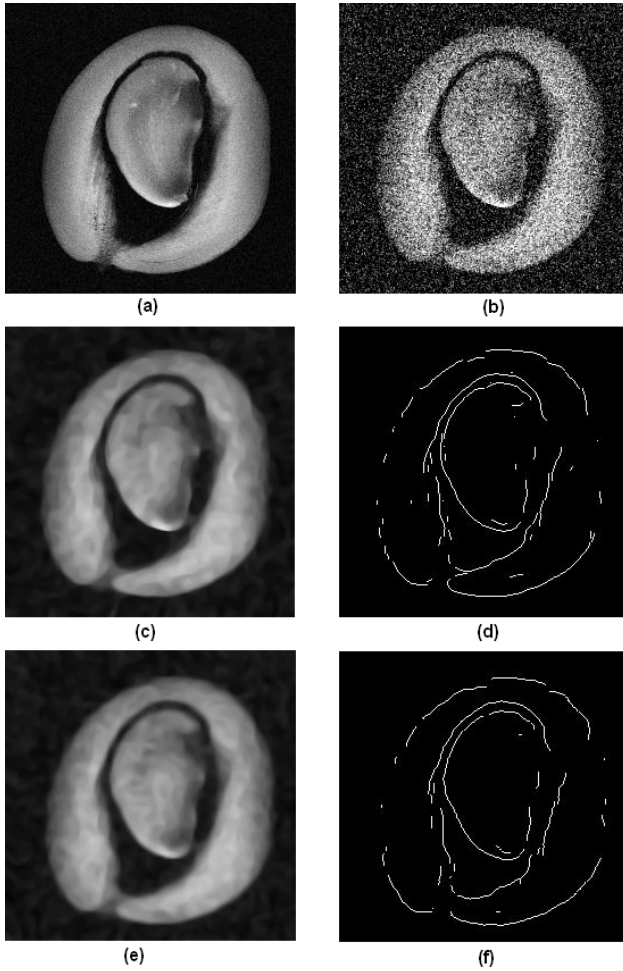


Fig. 3. (a) Ultrasonography of a papaia fruit original; (b) Noisy Image (SNR=3.3236 db); (c) Image reconstructed using the Well-Balanced Flow Equation (traditional approach); (d)Its segmentation; (e) Image reconstructed by the proposed method; (f)Its segmentation

Table 1. Computation Times (in seconds) for the traditional and multigrid approaches

| Figure | Traditional Approach | Iterations | Multigrid | Iterations per Level |
|--------|----------------------|------------|-----------|----------------------|
| 1 | 173 | 200 | 22 | 20 |
| 2 | 60 | 50 | 10 | 5 |
| 3 | 169 | 100 | 22 | 20 |
| 4 | 110 | 100 | 15 | 10 |

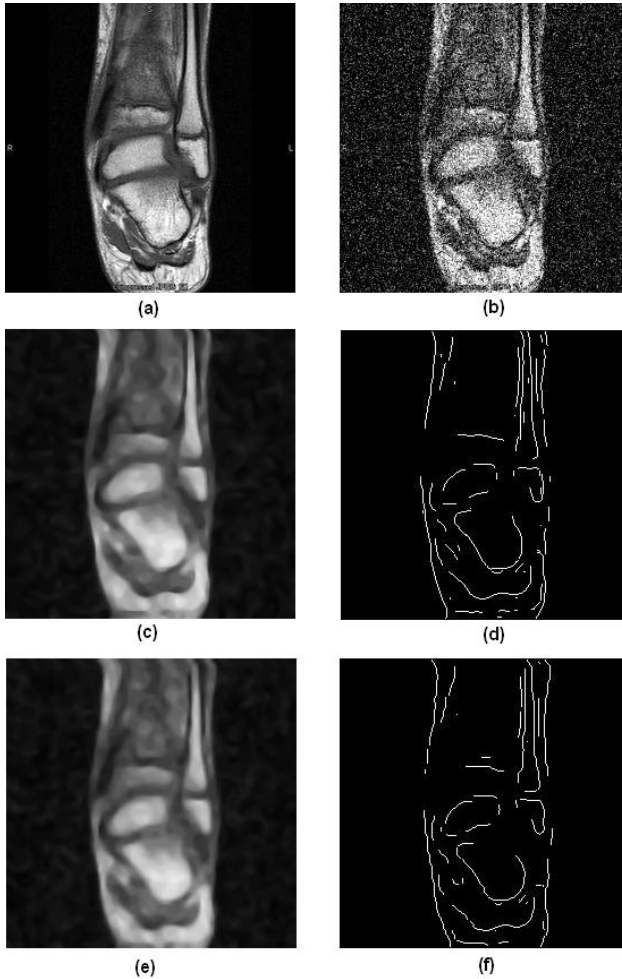


Fig. 4. (a) Ultrasonography of knee original; (b) Noisy image; (c) Image reconstructed using the Well-Balanced Flow Equation (traditional approach); (d) Its segmentation; (e) Image reconstructed by the proposed method; (f) Its segmentation

Our last example shows the results of applying our multigrid based approach with other kernels as interpolation and restriction operators. The SNR of the noisy image is the same of image 4b). Figures 6a) and 6b) present the reconstructed image and edges obtained by a Box-Filter kernel, where 10 iterations per level have been performed. Notice from figure 6a) that the Box-filter shifts some edges (see figure 4d) and 4f)) while introducing non smooth curves. The results obtained with a Lanczos-Filter are shown in figure 6c) and 6d). Notice the improvement in the preservation of the edges (figure 6d)) compared to

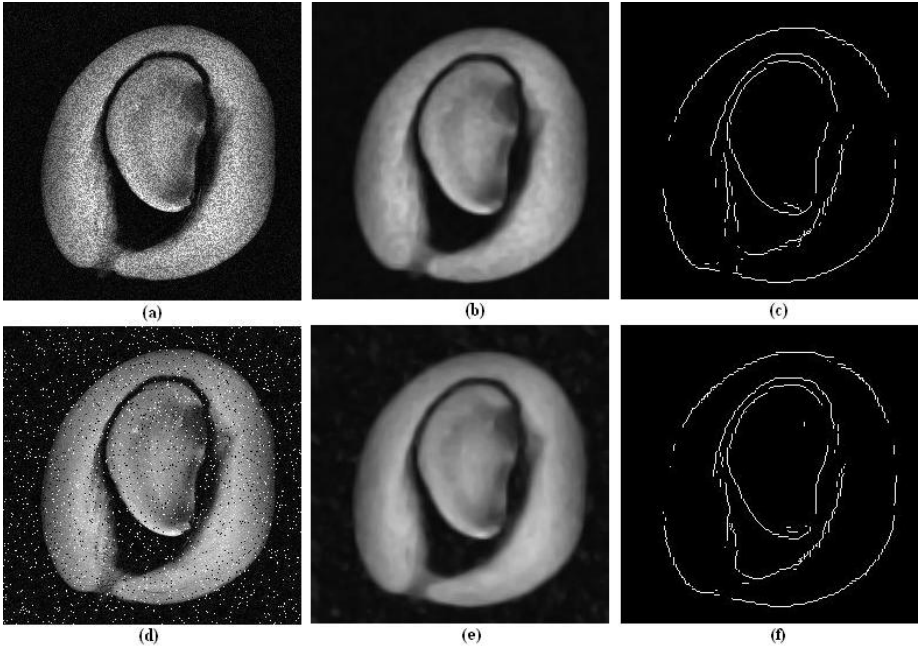


Fig. 5. (a) Multiplicative noisy image (SNR=9.3261); (b) Reconstructed image by the proposed method; (c) Its segmentation; (d) Salt & pepper noisy image (SNR=5.2731); (e) Reconstructed image by the proposed method; (f) Its segmentation.

figure 6b), however the Mitchell-Filter continues giving the best results (figures 4d) and 4f)).

A objective measure to compare quality of filtering is the mean error:

$$e_m = \frac{\sum_{i=0}^m \sum_{j=0}^n |u_{i,j}^n - I_{i,j}|}{(m+1) \times (n+1)}, \quad (3)$$

where m and n are the dimensions of the image, $u_{i,j}^n$ is the reconstructed image and $I_{i,j}$ is the original image. Table 2 shows this error for the two approaches tested. In the majority of the images, the error is smaller for our method. Only for figure 2 the error is larger for our method, but the processing time for this image is 6 times faster than for the conventional method.

Table 2. Mean Error (in percentage) of the traditional and multigrid approach

| Figure | Traditional Approach | Multigrid |
|--------|----------------------|-----------|
| 1 | 12.0484% | 8.1089% |
| 2 | 7.9528% | 13.8109% |
| 3 | 6.8719% | 6.4972% |
| 4 | 9.4049% | 8.1772% |

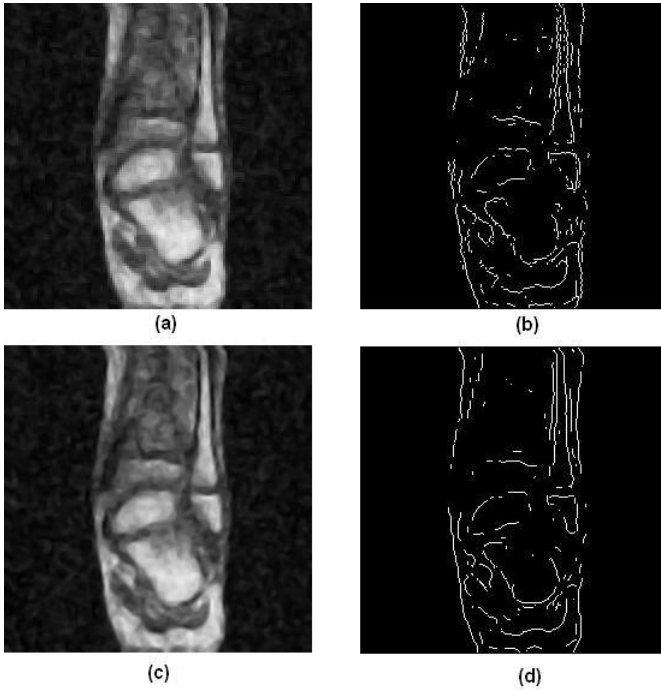


Fig. 6. (a)Image reconstructed using the Well-Balanced Flow Equation (multigrid based approach using the Box-Filter); (b) Its segmentation (c)Image reconstructed using the Well-Balanced Flow Equation (multigrid based approach using the Lanczos-Filter); (d)Its segmentation

The results were obtained on a Pentium III processor with 1 GB Ram.

5 Conclusions and Future Work

Based on the results of section 4 we are fairly confident that the multigrid-like approach is a competitive technique for image denoising while preserving edges.

Using the appropriated interpolation method we have obtained similar image quality compared to the traditional approach.

The use of a group of images including synthetic and medical images all with different level of noise, brings out the robustness of the PDE approach for image denoising and edge preservation. The multigrid approach inherits this robustness adding speed to its computation.

As a future work, we intend to implement the transport equation for image inpainting and denoising under the multigrid based approach.

References

1. S. T. Acton. Multigrid anisotropic diffusion. *IEEE Transaction on Image Processing*, Vol.7, NO.3, 7(3):280–291, 1998.
2. L. Alvarez, P. Lions, and J. Morel. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, 29(3):182–193, 1992.
3. C. A. Z. Barcelos, M. Boaventura, and E. C. S. Jr. Edge detection and noise removal by use of a partial differential equation with automatic selection of parameters. *Computational and Applied Mathematics*, 24(1):131–150, 2005.
4. C. A. Z. Barcelos, M. Boaventura, and E. C. S. Júnior. A well-balanced flow equation for noise removal and edge detection. *IEEE Transaction on Image Processing*, 12(7):751–763, 2003.
5. W. L. Briggs. *A Multigrid Tutorial*. Society for Industrial and Applied Mathematics, 1987.
6. T. Duff. *Splines in Animation and Modeling*. State of the Art in Image Synthesis, SIGGRAPH 86 Course Notes, 1986.
7. H. S. Hou and H. C. Andrews. Cubic splines for image interpolation and digital filtering. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-26:508–517, 1978.
8. R. Keys. Cubic convolution interpolation for digital image processing. *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-29:1153–1160, 1981.
9. J. Koenderink. The structure of images. *Biol. Cybernet*, 50:363–370, 1984.
10. D. Marr and E. Hildreth. Theory of edge detection. *Proc. Roy. Soc.*, 207:187–217, 1980.
11. S. F. McCormick, editor. *Multigrid methods*, volume 3 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1987.
12. D. Mitchell and A. Netravali. Reconstruction filters in computer graphics. *Computer Graphics*, 22(4):221–228,, 1988.
13. P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

Fuzzy Bilateral Filtering for Color Images

Samuel Morillas*, Valentín Gregori**, and Almanzor Sapena

Universidad Politécnica de Valencia, E.P.S. de Gandia, Carretera Nazaret-Oliva s/n, 46730 Grao de Gandia (Valencia), Spain

Abstract. Bilateral filtering is a well-known technique for smoothing gray-scale and color images while preserving edges and image details by means of an appropriate nonlinear combination of the color vectors in a neighborhood. The pixel colors are combined based on their spatial closeness and photometric similarity. In this paper, a particular class of fuzzy metrics is used to represent the spatial and photometric relations between the color pixels adapting the classical bilateral filtering. It is shown that the use of these fuzzy metrics is more appropriate than the classical measures used.

1 Introduction

Any image is systematically affected by the introduction of noise during its acquisition and transmission process. A fundamental problem in image processing is to effectively suppress noise while keeping intact the features of the image. Fortunately, two noise models can adequately represent most noise corrupting images: additive Gaussian noise and impulsive noise [12].

Additive Gaussian noise, which is usually introduced during the acquisition process, is characterized by adding to each image pixel channel a random value from a zero-mean Gaussian distribution. The variance of this distribution determines the intensity of the corrupting noise. The zero-mean property allows to remove such noise by locally averaging pixel channel values.

Ideally, removing Gaussian noise would involve to smooth the different areas of an image without degrading neither the sharpness of their edges nor their details. Classical linear filters, such as the Arithmetic Mean Filter (AMF) or the Gaussian Filter, smooth noise but blur edges significantly. Nonlinear methods have been used to approach this problem. The aim of the methods proposed in the literature is to detect edges by means of local measures and smooth them less than the rest of the image to better preserve their sharpness. A well-known method is the anisotropic diffusion introduced in [11]. In this technique, local image variation is measured at every point and pixels from neighborhoods whose size and shape depend on local variation are averaged. Diffusion methods are inherently iterative since the use of differential equations is involved. On the other hand, a non-iterative interesting method, which is the motivation of this work,

* The author acknowledges the support of Spanish Ministry of Education and Science under program "Becas de Formación de Profesorado Universitario FPU".

** The author acknowledges the support of Spanish Ministry of Science and Technology "Plan Nacional I+D+I", and FEDER, under Grant BFM2003-02302.

is the bilateral filter (BF) studied in [15]. The output of the BF at a particular location is a weighted mean of the pixels in its neighborhood where the weight of each pixel depends on the spatial closeness and photometric similarity respect to the pixel in substitution. The BF has been proved to perform effectively in Gaussian noise suppression and it has been the object of further studies [2,3,14].

In this paper, a certain class of fuzzy metrics is used to model the relations of spatial closeness and photometric similarity between image pixels used in the BF. Then, the BF structure is adapted and the, from now on called, *Fuzzy Bilateral Filter* (FBF) is proposed. The use of fuzzy metrics instead of the measures used in [15] makes the filter easier to use since the number of adjusting parameters is lower. Moreover, the performance of the proposed filter is improved respect to the other filters in the comparison in the sense that will be shown.

The paper is arranged as follows. The classical BF is described in Section 2. The use of fuzzy metrics to model the spatial and photometric relations is detailed in Section 3. Section 4 defines the *Fuzzy Bilateral Filter*. Experimental results and discussions are presented in Section 5 and conclusions are given in Section 6.

2 Bilateral Filtering

Let \mathbf{F} represent a multichannel image and let W be a sliding window of finite size $n \times n$. Consider the pixels in W represented in Cartesian Coordinates and so, denote by $\mathbf{i} = (i_1, i_2) \in Y^2$ the position of a pixel \mathbf{F}_i in W where $Y = \{0, 1, \dots, n - 1\}$ is endowed with the usual order. According to [15], the BF replaces the central pixel of each filtering window by a weighted average of its neighbor color pixels. The weighting function is designed to smooth in regions of similar colors while keeping edges intact by heavily weighting those pixels that are both spatially close and photometrically similar to the central pixel.

Denote by $\|\cdot\|_2$ the Euclidean norm and by \mathbf{F}_i the central pixel under consideration. Then the weight $\mathcal{W}(\mathbf{F}_i, \mathbf{F}_j)$ corresponding to any pixel \mathbf{F}_j respect to \mathbf{F}_i is the product of two components, one spatial and one photometrical

$$\mathcal{W}(\mathbf{F}_i, \mathbf{F}_j) = \mathcal{W}_s(\mathbf{F}_i, \mathbf{F}_j)\mathcal{W}_p(\mathbf{F}_i, \mathbf{F}_j) \tag{1}$$

where the spatial component $\mathcal{W}_s(\mathbf{F}_i, \mathbf{F}_j)$ is given by

$$\mathcal{W}_s(\mathbf{F}_i, \mathbf{F}_j) = e^{-\frac{\|\mathbf{i}-\mathbf{j}\|_2^2}{2\sigma_s^2}} \tag{2}$$

and the photometrical component $\mathcal{W}_p(\mathbf{F}_i, \mathbf{F}_j)$ is given by

$$\mathcal{W}_p(\mathbf{F}_i, \mathbf{F}_j) = e^{-\frac{\Delta E_{Lab}(\mathbf{F}_i, \mathbf{F}_j)^2}{2\sigma_p^2}} \tag{3}$$

where $\Delta E_{Lab} = [(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2]^{\frac{1}{2}}$ denotes the perceptual color error in the $L^*a^*b^*$ color space, and $\sigma_s, \sigma_p > 0$

The color vector output $\widehat{\mathbf{F}}_i$ of the filter is computed using the normalized weights and so it is given by

$$\widetilde{\mathbf{F}}_i = \frac{\sum_{\mathbf{F}_j \in W} \mathcal{W}(\mathbf{F}_i, \mathbf{F}_j) \mathbf{F}_j}{\sum_{\mathbf{F}_j \in W} \mathcal{W}(\mathbf{F}_i, \mathbf{F}_j)} \quad (4)$$

The \mathcal{W}_s weighting function decreases as the spatial distance in the image between \mathbf{i} and \mathbf{j} increases, and the \mathcal{W}_p weighting function decreases as the perceptual color difference between the color vectors increases. The spatial component decreases the influence of the furthest pixels reducing blurring while the photometric component reduces the influence of those pixels which are perceptually different respect to the one under processing. In this way, only perceptually similar areas of pixels are averaged together and the sharpness of edges is preserved.

The parameters σ_s and σ_p are used to adjust the influence of the spatial and the photometric components, respectively. They can be considered as rough thresholds for identifying pixels sufficiently close or similar to the central one. Note that when $\sigma_p \rightarrow \infty$ the BF approaches a Gaussian filter and when $\sigma_s \rightarrow \infty$ the filter approaches a range filter with no spatial notion. In the case when both $\sigma_p \rightarrow \infty$ and $\sigma_s \rightarrow \infty$ the BF behaves as the AMF.

3 Fuzzy Metric Approach

According to [4], a fuzzy metric space is an ordered triple $(X, M, *)$ such that X is a (nonempty) set, $*$ is a continuous t-norm and M is a fuzzy set of $X \times X \times]0, +\infty[$ satisfying the following conditions for all $x, y, z \in X, s, t > 0$:

- (FM1) $M(x, y, t) > 0$
- (FM2) $M(x, y, t) = 1$ if and only if $x = y$
- (FM3) $M(x, y, t) = M(y, x, t)$
- (FM4) $M(x, z, t + s) \geq M(x, y, t) * M(y, z, s)$
- (FM5) $M(x, y, \cdot) :]0, +\infty[\rightarrow]0, 1]$ is continuous.

$M(x, y, t)$ represents the degree of nearness of x and y with respect to t . If $(X, M, *)$ is a fuzzy metric space we will say that $(M, *)$ is a fuzzy metric on X . According to [5,6], the above definition can be considered an appropriate concept of fuzzy metric space. In the following, by a fuzzy metric we mean a fuzzy metric in the George and Veeramani's sense (from now on we will omit the mention to the continuous t-norm $*$ since in all the cases it will be the usual product in $[0, 1]$).

A fuzzy metric M on X is said to be stationary if M does not depend on t , i.e. for each $x, y \in X$ the function $M_{x,y}(t) = M(x, y, t)$ is constant [7]. In such case we write $M(x, y)$ instead of $M(x, y, t)$.

In this paper two fuzzy metrics, in a first step, will be used. The first one to measure the photometric fuzzy distance between color vectors and the second one to measure the spatial fuzzy distance between the pixels under comparison. In order to appropriately measure the photometric fuzzy distance between color vectors we will use the following fuzzy metric M :

Take $X = \{0, 1, 2, \dots, 255\}$ and let $K > 0$ fixed. Denote by $(F_i^1, F_i^2, F_i^3) \in X^3$ the color vector of a pixel \mathbf{F}_i . The function $M : X^3 \times X^3 \rightarrow]0, 1]$ defined by

$$M(\mathbf{F}_i, \mathbf{F}_j) = \prod_{s=1}^3 \frac{\min\{F_i^s, F_j^s\} + K}{\max\{F_i^s, F_j^s\} + K} \tag{5}$$

is, according to [9], a stationary fuzzy metric on X^3 . Previous works [9,10] have shown that a suitable value of the K parameter for standard RGB images is $K = 1024$, so, this value will be assumed from now on throughout the paper. Then, $M(\mathbf{F}_i, \mathbf{F}_j)$ will denote the photometric fuzzy distance between the color pixels \mathbf{F}_i and \mathbf{F}_j .

Now for the case of spatial fuzzy distance between pixels and using the terminology in Section 2, denote by $\mathbf{i} = (i_1, i_2) \in Y^2$ the position of the pixel \mathbf{F}_i in the window W . The function $M_{\|\cdot\|_2} : Y^2 \times]0, +\infty \rightarrow]0, 1]$ given by

$$M_{\|\cdot\|_2}(\mathbf{i}, \mathbf{j}, t) = \frac{t}{t + \|\mathbf{i} - \mathbf{j}\|_2} \tag{6}$$

is a fuzzy metric on Y^2 called the standard fuzzy metric deduced from the Euclidean norm $\|\cdot\|_2$ ([4] Example 2.9). Then, $M_{\|\cdot\|_2}(\mathbf{i}, \mathbf{j}, t)$ will denote the spatial fuzzy distance between the color pixels \mathbf{F}_i and \mathbf{F}_j with respect to t . Notice that $M_{\|\cdot\|_2}$ does not depend on the length of Y but on the relative position in the image of the pixels \mathbf{F}_i and \mathbf{F}_j under comparison. The t parameter may be interpreted as a parameter to adjust the importance given to the spatial closeness criterion.

4 Fuzzy Bilateral Filtering

In order to define the *Fuzzy Bilateral Filter* (FBF) it is just necessary to determine how the weight of each pixel in the filtering window is computed by using the fuzzy metrics in Section 3.

Since each pixel \mathbf{F}_i is characterized by its RGB color vector (F_i^1, F_i^2, F_i^3) and by its location (i_1, i_2) in the window, for our purpose, in a second step, it will be considered a fuzzy metric combining M (5) with $M_{\|\cdot\|_2}$ (6). So, to compute the weight of each pixel $\mathbf{F}_j, \mathbf{j} \in W$ it will be considered the following function

$$CFM(\mathbf{F}_i, \mathbf{F}_j, t) = M(\mathbf{F}_i, \mathbf{F}_j) \cdot M_{\|\cdot\|_2}(\mathbf{i}, \mathbf{j}, t) = \prod_{s=1}^3 \frac{\min\{F_i^s, F_j^s\} + K}{\max\{F_i^s, F_j^s\} + K} \cdot \frac{t}{t + \|\mathbf{i} - \mathbf{j}\|_2} \tag{7}$$

If we identify each pixel \mathbf{F}_i with $(F_i^1, F_i^2, F_i^3, i_1, i_2)$ then (from [13] Proposition 3.5) the above function CFM is a fuzzy metric on $X^3 \times Y^2$. In this way, the use of the above fuzzy metric is enough to simultaneously model the spatial closeness and photometric similarity criteria commented in Sections 1-2. The FBF output will be calculated as follows

$$\widetilde{\mathbf{F}}_i = \frac{\sum_{\mathbf{F}_j \in W} CFM(\mathbf{F}_i, \mathbf{F}_j, t) \mathbf{F}_j}{\sum_{\mathbf{F}_j \in W} CFM(\mathbf{F}_i, \mathbf{F}_j, t)} \tag{8}$$

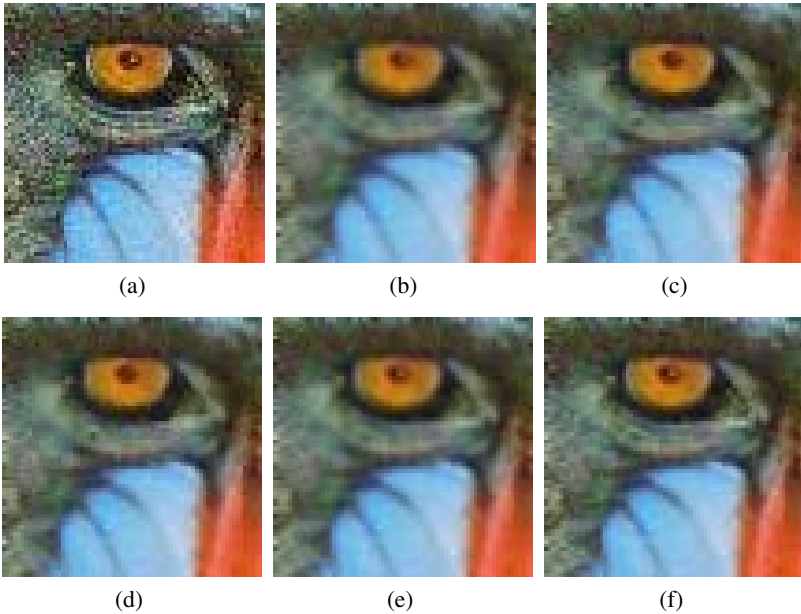
where the only t parameter is used to tune the importance of the spatial closeness criterion respect to the photometric criterion. Notice that, in a similar way to the BF, when $t \rightarrow \infty$ the FBF approaches a range filter without spatial notion. The reduction of the number of parameters respect to the BF makes the FBF easier to tune, however it can not behave as the Gaussian filter nor the AMF as in the case of the BF. The study of the t parameter is done in the following section.

Table 1. Comparison of the performance in terms of NCD (10^{-2}) using details of the Baboon, Peppers and Lenna images contaminated with different intensities of additive gaussian noise

| Filter | Detail of Baboon | | | Detail of Peppers | | | Detail of Lenna | | |
|--------|------------------|---------------|---------------|-------------------|---------------|---------------|-----------------|---------------|---------------|
| | $\sigma = 10$ | $\sigma = 20$ | $\sigma = 30$ | $\sigma = 5$ | $\sigma = 15$ | $\sigma = 30$ | $\sigma = 10$ | $\sigma = 20$ | $\sigma = 30$ |
| None | 10.08 | 20.14 | 29.90 | 3.86 | 11.57 | 22.85 | 9.11 | 17.49 | 25.75 |
| AMF | 7.42 | 9.58 | 11.81 | 3.74 | 5.43 | 8.38 | 5.08 | 7.28 | 9.73 |
| VMMF | 8.49 | 11.56 | 14.89 | 3.74 | 6.19 | 10.06 | 5.67 | 8.73 | 11.75 |
| VMF | 9.93 | 14.37 | 18.99 | 4.36 | 7.90 | 13.31 | 6.97 | 11.36 | 15.24 |
| EVMF | 8.55 | 11.25 | 13.96 | 3.90 | 6.20 | 9.50 | 5.68 | 8.50 | 11.29 |
| BVDF | 10.34 | 13.91 | 17.98 | 4.14 | 7.53 | 12.37 | 6.56 | 10.48 | 14.14 |
| DDF | 9.01 | 12.81 | 17.22 | 3.98 | 7.10 | 12.07 | 6.34 | 10.30 | 13.94 |
| BF | 6.91 | 9.38 | 11.70 | 3.39 | 5.25 | 8.37 | 4.79 | 7.21 | 9.70 |
| FBF | 6.42 | 9.24 | 11.61 | 3.43 | 5.29 | 8.33 | 4.74 | 7.12 | 9.64 |

5 Experimental Results

In order to study the performance of the proposed filter, some details of the well-known images Lenna, Peppers and Baboon have been contaminated with Gaussian noise following its classical model [12]. Performance comparison has been done using the *Normalized Color Difference* (NCD) objective quality measure since it approaches human perception and which is defined as

**Fig. 1.** Performance comparison: (a) Detail of Baboon image contaminated with Gaussian noise $\sigma = 10$, (b) AMF output, (c) VMMF output, (d) EVMF output, (e) BF output, (f) FBF output

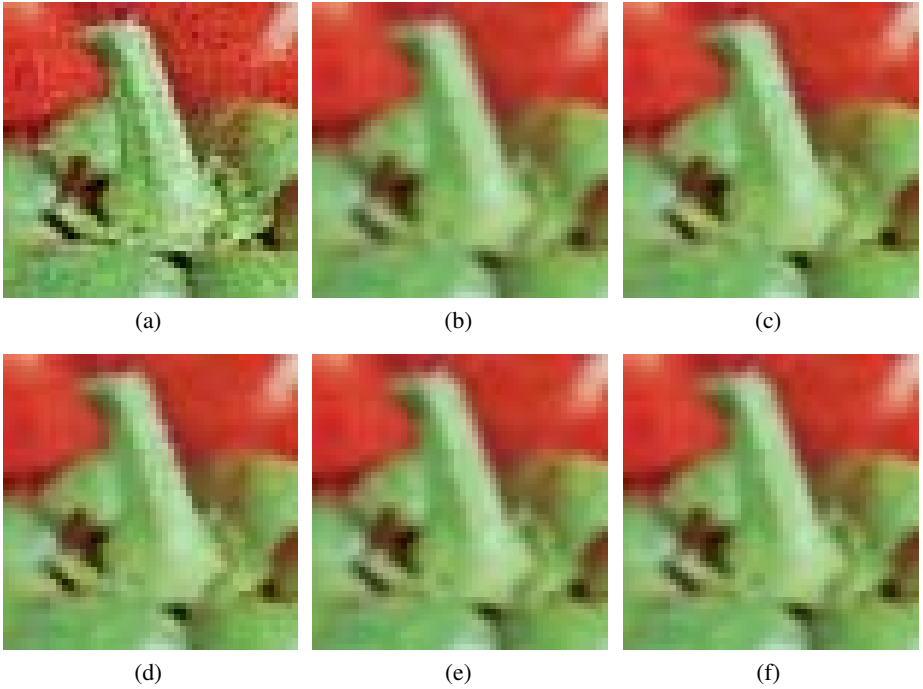


Fig. 2. Performance comparison: (a) Detail of Peppers image contaminated with Gaussian noise $\sigma = 15$, (b) AMF output, (c) VMMF output, (d) EVMF output, (e) BF output, (f) FBF output

$$NCD_{Lab} = \frac{\sum_{i=1}^N \sum_{j=1}^M \Delta E_{Lab}}{\sum_{i=1}^N \sum_{j=1}^M E_{Lab}^*} \quad (9)$$

where M, N are the image dimensions and $\Delta E_{Lab} = [(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2]^{\frac{1}{2}}$ denotes the perceptual color error and $E_{Lab}^* = [(L^*)^2 + (a^*)^2 + (b^*)^2]^{\frac{1}{2}}$ is the *norm* or *magnitude* of the original image color vector in the $L^*a^*b^*$ color space.

The proposed filter is assessed in front of the classical BF and other well-known vector filters: the Arithmetic Mean Filter (AMF), the Vector Marginal Median Filter (VMMF) [12], the Vector Median Filter (VMF) [1], the Extended Vector Median Filter (EMVF), the Basic Vector Directional Filter (BVDF) [16] and the Distance-Directional Filter (DDF) [8]. In all the cases it has been considered a 3×3 window size.

For both the classical BF and the proposed FBF extensive experiments have been carried out trying to find the optimal parameters that reach the best filter performance. The proposed filter has been much easier to adjust since only one parameter (t) is involved in the adjusting process in contrast to the BF where the presence of two parameters (σ_s and σ_p) makes this process more complex. Appropriate values of the t parameter are in the range $[1, 10]$ for Gaussian noise densities (σ) in $[0, 30]$. For higher values of t the smoothing performed is higher, as well, and the value of the t parameter in $[1, 10]$ can be easily set by simple proportionality respect to the noise density $\sigma \in [0, 30]$.

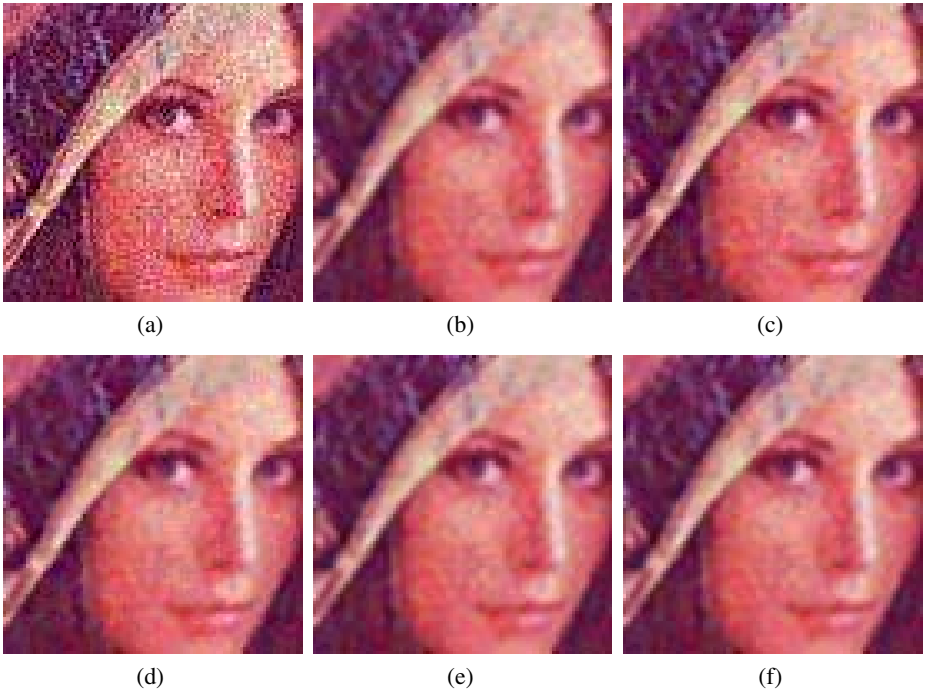


Fig. 3. Performance comparison: (a) Detail of Lenna image contaminated with Gaussian noise $\sigma = 30$, (b) AMF output, (c) VMMF output, (d) EVMF output, (e) BF output, (f) FBF output

From the results shown in Table 1 and Figures 1-3 it can be stated that the performance presented by the proposed filter is competitive respect to the BF and outperforms the rest of the techniques in comparison. The results seem to indicate that the FBF behaves better than the BF when dealing with highly textured images, as the Baboon image, and slightly worse in images with many homogeneous areas, as the Peppers image.

6 Conclusions

In this paper a certain class of fuzzy metrics has been used to simultaneously model a double relation between color pixels: spatial closeness and photometric similarity, using an only fuzzy metric. Notice that, as far as the authors know, this could not be done using a classical metric.

The proposed fuzzy metric has been used to adapt the classical BF, then the FBF has been proposed. The proposed filter is easier to use than its classical version since the filter adjusting process becomes simpler. The performance presented by the proposed filter is competitive respect to the BF outperforming it in many cases. These results indicate that this fuzzy metric may be considered more appropriate than the measures used in the classical BF.

The experiences of this paper constitute another proof of the appropriateness of the fuzzy metrics to model complex relations which motivates its further study.

References

1. J. Astola, P. Haavisto, Y. Neuvo, *Vector Median Filters*, Proc. IEEE. 78 4 (1990) 678-689.
2. M. Elad, *On the origin of bilateral filter and ways to improve it*, IEEE Transactions on Image Processing 11 10 (2002) 1141-1151.
3. R. Garnett, T. Huegerich, C. Chui, W. He, *A universal noise removal algorithm with an impulse detector*, IEEE Transactions on Image Processing 14 11 (2005) 1747-1754.
4. A. George, P. Veeramani, *On Some results in fuzzy metric spaces*, Fuzzy Sets and Systems 64 3 (1994) 395-399.
5. A. George, P. Veeramani, *Some theorems in fuzzy metric spaces*, J. Fuzzy Math. 3 (1995) 933-940.
6. V. Gregori, S. Romaguera, *Some properties of fuzzy metric spaces*, Fuzzy Sets and Systems 115 3 (2000) 477-483.
7. V. Gregori, S. Romaguera, *Characterizing completable fuzzy metric spaces*, Fuzzy Sets and Systems 144 3 (2004) 411-420.
8. D.G. Karakos, P.E. Trahanias, *Generalized multichannel image-filtering structures*, IEEE Transactions on Image Processing 6 7 (1997) 1038-1045.
9. S. Morillas, V. Gregori, G. Peris-Fajarnés, P. Latorre, *A new vector median filter based on fuzzy metrics*, ICIAR'05, Lecture Notes in Computer Science 3656 (2005) 81-90.
10. S. Morillas, V. Gregori, G. Peris-Fajarnés, P. Latorre, *A fast impulsive noise color image filter using fuzzy metrics*, Real-Time Imaging 11 5-6 (2005) 417-428.
11. P. Perona, J. Malik, *Scale-space and edge detection using anisotropic diffusion*, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 5 (1990) 629-639.
12. K.N. Plataniotis, A.N. Venetsanopoulos, *Color Image processing and applications*, Springer-Verlag, Berlin, 2000.
13. A. Sapena, *A contribution to the study of fuzzy metric spaces* Appl. Gen. Topology 2 1(2001) 63-76.
14. Y. Shen, K. Barner, *Fuzzy vector median-based surface smoothing*, IEEE Transactions on Visualization and Computer Graphics 10 3 (2004) 252-265.
15. C. Tomasi, R. Manduchi, *Bilateral filter for gray and color images*, Proc. IEEE International Conference Computer Vision, 1998, 839-846.
16. P.E. Trahanias, D. Karakos, A.N. Venetsanopoulos, *Vector Directional Filters: a new class of multichannel image processing filters*, IEEE Trans. Image Process. 2 4 (1993) 528-534.

MPEG Postprocessing System Using Edge Signal Variable Filter

Chan-Ho Han¹, Suk-Hwan Lee^{2,*}, Seong-Geun Kwon³,
Ki-Ryong Kwon⁴, and Dong Kyue Kim⁵

¹ School of Electrical Engineering & Computer Science, Kyungpook National University
chhan007@lycos.co.kr

² Dept. of Information Security, TongMyong University
skylee@tit.ac.kr

³ Mobile Communication Division, SAMSUNG electronics co.
seonggeunkwon@hanmail.net

⁴ Division of Electronic, Computer & Telecommunication Eng., Pukyong National Univ.
krkwon@pknu.ac.kr

⁵ Dept. of Electronics and Computer Engineering, Hanyang University
dqkim@hanyang.ac.kr

Abstract. We proposed the algorithm for the quantization noise reduction based on variable filter adaptive to edge signal in MPEG postprocessing system. In our algorithm, edge map and local modulus maxima in the decoded images are obtained by using 2D Mallat wavelet filter. And then, blocking artifacts in inter-block are reduced by Gaussian LPF that is variable to filtering region according to edge map. Ringing artifacts in intra-block are reduced by 2D SAF according to local modulus maxima. Experimental results show that the proposed algorithm was superior to the conventional algorithms as regards PSNR, which was improved by 0.04-0.20 dB, and the subjective image quality.

Keywords: MPEG, Postprocessing, Edge Signal Variable Filter.

1 Introduction

Recently, MPEG-2 has been adopted as the video standards for satellite, terrestrial, cable digital broadcasting besides the transmission of ATSC and DVB. In internet surroundings, there have been many multimedia standards such as H.263, WMV, QuickTime, and MPEG-4. DMB (Digital Multimedia Broadcasting) that is a digital transmission system for sending data, radio, TV to mobile devices can service various digital multimedia while preserving quality of service in mobile. T(terrestrial)-DMB use MPEG-4 AVC (H.264) for the video and MPEG-4 BSAC for the audio. But the transmitted image has the artifact because of high compression for the limit bandwidth. Especially, compression artifacts are often observed in block-based coding due to the quantization process that independently quantizes the DCT coefficients in each block. Among the artifacts caused by such quantization errors, there are blocking artifacts that are the most visible and ringing artifact. Blocking artifacts are divided into grid noise and staircase noise in general. Grid noise is easily noticeable as a slight

* Corresponding author.

change of intensity along a 8×8 block boundary in a monotone area, like a grid shape. Staircase noise occurs when a continuous edge is included in an inter-block, and this edge becomes discontinuous in the block-coded image. Ringing noise is the result of the truncation of high-frequency coefficients in the quantization process, producing a pseudo-edge around the edge, such as Gibb's phenomenon. JPEG-2000 that is a still image compression standard based on wavelet coding has not blocking artifacts but ringing artifact around edge.

The algorithms to reduce the quantization noise are categorized into the preprocessing and the postprocessing process. The preprocessing algorithm is to design the encoder that don't occur the quantization noise in compression process. But it has drawbacks that requires the additional information and is difficult to apply to the conventional standards. The postprocessing algorithm is to reduce the quantization noise after decoding the bit stream in the receiver. It can process real-time without the additional information considering HVS (Human visual system), which more sensitive to noise in monotone area than in complex areas. The post-filtering in recommended at MPEG-4 separates DC offset mode and default mode using the intensity difference of the neighborhood pixels. The first mode, corresponding to flat regions, involves the use of a strong filter inside the block as well as at the block boundaries. The second mode, corresponding to all other regions, then involves the use of a sophisticated smoothing filter, based on the frequency information around the block boundaries, to reduce blocking artifacts without introducing undesired blur. Although this algorithm is very simple and has a fast processing time, it cannot remove staircase.

Our paper proposed the variable filter adaptive to edge signal in MPEG postprocessing system to remove the quantization noise. The Mallat wavelet representation characterizes the local variations of a signal for multiscale edges preserving the same resolution. Mallat analyzed the characteristics of edge signal and singularity in multiscale edges. Firstly, an edge map and local modulus maxima are detected by using Mallat wavelet. The variable filter that the filter coefficients and the tap number can be controlled by the blocking strength and quantization parameter is performed to remove the blocking artifacts in the horizontal and vertical block boundary. The filtering region is determined by the edge map around the block boundary. 2D SAF (Signal Adaptive Filter) within intra-block is performed to remove ringing artifact using local modulus maxima. Experimental results confirmed that the proposed algorithm can reduce the presence of artifacts and preserve the edge details.

2 Proposed MPEG Processing

As shown in Fig. 1, the proposed algorithm detects the edge map and local modulus maxima using 2D Mallat wavelet filter in each frame in MPEG decoder system. And then, the filter with variable filtering region, tap coefficient, and tap number is performed at inter-block to remove blocking artifacts using the edge map. Furthermore, 2D SAF is performed within intra-block to remove ringing artifact using local modulus maxima. A brief description of Mallat wavelet is given in subsection 1, then the proposed post-filtering is presented closely in the following subsections.

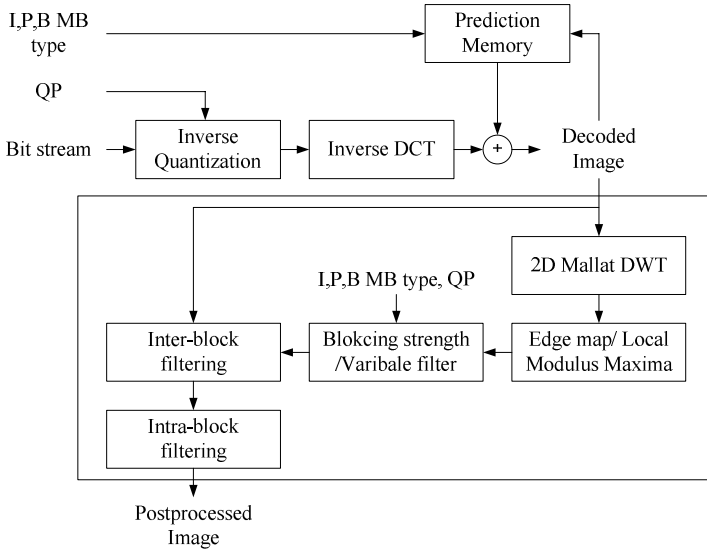


Fig. 1. The block-diagram of the proposed MPEG post-processing system for quantization noise reduction

2.1 Mallat Wavelet

Let a smoothing function be any real function $\theta(x)$, such that $\theta(x) = O(1/(1+x^2))$ where the integral is nonzero. Let $\theta_{2^j}(x) = (1/2^j)\theta(x/2^j)$ and $f(x)$ be a real function in $L^2(\mathbf{R})$. The edges at scale 2^j are defined as local sharp variation points of $f(x)$ smoothed by $\theta_{2^j}(x)$. Let $\psi(x)$ be the wavelets defined by $\psi(x) = \frac{d\theta(x)}{dx}$. The wavelet transforms defined with respect to these wavelets are given by $Wf(2^j, x) = f * \psi_{2^j}(x) = f * (2^j \frac{d\theta_{2^j}(x)}{dx})(x)$. The wavelet transform $Wf(2^j, x)$ is proportional respectively to the first derivative of $f(x)$ smoothed by $\theta_{2^j}(x)$. A local maximum is thus any point $(2^j, x_0)$ such that $|Wf(2^j, x)| < |Wf(2^j, x_0)|$ when x belongs to either the right or left neighborhood of x_0 , and $|Wf(2^j, x)| \leq |Wf(2^j, x_0)|$ when x belongs to the other side of the neighborhood of x_0 . A 2-D dyadic wavelet transform of $f(x, y)$ at the scale 2^j has two components defined by $W_{2^j}^h f(x, y) = f * \psi_{2^j}^h(x, y)$ and $W_{2^j}^v f(x, y) = f * \psi_{2^j}^v(x, y)$ with respect to the two wavelets $\psi_{2^j}^h = \frac{1}{2^{2j}} \psi^h(\frac{x}{2^j}, \frac{y}{2^j})$ and $\psi_{2^j}^v = \frac{1}{2^{2j}} \psi^v(\frac{x}{2^j}, \frac{y}{2^j})$, where $\psi^h(x, y) = \frac{\partial \theta(x, y)}{\partial x}$ and $\psi^v(x, y) = \frac{\partial \theta(x, y)}{\partial y}$. The smoothing operator S_{2^j} is defined by $S_{2^j} f(x, y) = f * \phi_{2^j}(x, y)$ with $\phi_{2^j}(x, y) = \frac{1}{2^j} \phi(\frac{x}{2^j}, \frac{y}{2^j})$

with respect to the smoothing function $\phi(x, y)$. The wavelet transform between the scales 1 and 2^J $\{W_{2^j}^h f(x, y), W_{2^j}^v f(x, y)\}_{1 \leq j \leq J}$ provides the details that are available in $S_1 f(x, y)$.

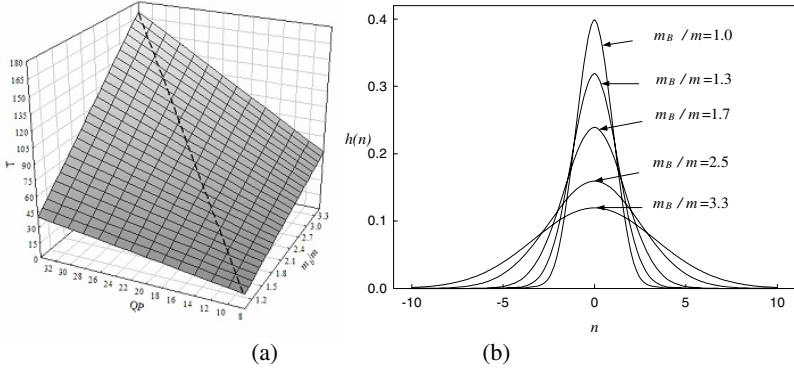


Fig. 2. (a) Threshold T according to quantization parameter QP and blocking strength m_B/m and (b) Gaussian filter according to blocking strength m_B/m

2.2 Edge Map Detection

The magnitude $M_{2^1}(x, y)$ and the phase $A_{2^1}(x, y)$ of 2D wavelet modulus in the first scale are respectively

$$M_{2^1} f(x, y)^2 = \sqrt{W_{2^1}^h f(x, y)^2 + W_{2^1}^v f(x, y)^2} \quad (1)$$

$$A_{2^1}(x, y) = \tan^{-1}(W_{2^1}^v f(x, y) / W_{2^1}^h f(x, y)). \quad (2)$$

Edge map $E(x, y)$ is IF $M_{2^1}(x, y) > T$, $E(x, y) = 1$, ELSE $E(x, y) = 0$ where T

$$T = QP \times \left(\left(\frac{m_B}{m} - 1 \right) \times A + 1 \right) \quad (3)$$

is determined by blocking strength m_B/m (≥ 1) and quantization parameter QP . T is a threshold value that distinguishes blocking artifacts and edge. Since blocking artifacts become visible differently according to compression ratio in block coded image, the post-filtering has to be performed adaptively to blocking strength and quantization parameter while preserving the edge. If QP and m_B/m are low, a threshold T has to be selected to low value because blocking artifacts become less visible than the edge. But if QP and m_B/m are high, a threshold T has to be selected to high value because blocking artifacts become visible remarkably. The parameter A is determined as 1.3 experimentally. The solid line in Fig. 2 (a) represents a threshold T in this paper. m_B and m are respectively.

$$m_B = \frac{S_B}{N_B}, \quad m = \left(\sum_{x=0}^{NV-1} \sum_{y=0}^{NH-1} M_{2^1}(x, y) - S_B \right) / (NV \times NH - N_B) \quad (4)$$

$$\text{where } S_B = \sum_{x=0}^{NV-1} \sum_{y=0}^{NH/8-2} M_{2^1}(x, 8y+7) + \sum_{x=0}^{NV/8-2} \sum_{y=0}^{NH-1} M_{2^1}(8x+7, y)$$

$$N_B = NV(NH/8-1) + NH(NV/8-1)$$

which m_B and m represent the average magnitude of $M_{2^1}(x, y)$ in block boundary and within block. NH and NV are the horizontal and vertical resolution.

2.3 Inter-block Filtering

In the proposed inter-block filtering, the variable filter that is adaptive to edge map around block boundary is performed to horizontal and vertical direction in inter-block to reduce blocking artifacts. The variable filter $h(n)$ is

$$h(n) = m/m_B \times e^{\frac{-n^2}{2 \times (c \times m/m_B)^2}} / \sum_{n=-N}^N (m/m_B \times e^{\frac{-n^2}{2 \times (c \times m/m_B)^2}}) \quad (5)$$

where $1.9 \leq c \leq 2.9$, $-N \leq n \leq N$

that is a Gaussian filter in proportion to the inverse of the blocking strength m_B/m . This filter is nearly mean filter if the m_B/m is low and also is nearly impulse filter if m_B/m is high as shown in Fig. 2 (b). The number of tap ($2N+1$) and the filtering region $[-d, d-1]$ are determined according to the edge map around block boundary. If edge map in 8 pixels around the block boundary are all 0, then $N=1$ and $d=1$. Otherwise, N, d are determined to the most near position in block boundary among the position in which edge map is 1. For example, in case of the pixels $f(8u+i, y)$, $-4 \leq i \leq 3$ for horizontal inter-block, if the position i is -4, 3 in which edge map $E(8u+i, y)$ is 1, then $N=3$ and $d=3$. Or if $i=-3, 2$, then $N=2$ and $d=2$. If $i=-1, 0$, the variable filter is not performed since $N=0$. This designed variable filter is performed at the block boundary

$$\hat{f}(8u+i, y) = \sum_{n=-N}^N w(n) \times h(n) \times f(8u+i+n, y) / \sum_{n=-N}^N w(n) \times h(n), \quad -d \leq i \leq d-1 \quad (6)$$

if $E(8u+i+n, y) = 0$, $-N \leq n < 0$,

then $w(n) = 0$ for $-N \leq n < 0$ and $w(n) = 1$ for $0 \leq n \leq N$

else if $E(8u+i+n, y) = 0$, $0 \leq n \leq N$,

then $w(n) = 0$ for $0 \leq n \leq N$ and $w(n) = 1$ for $-N \leq n < 0$

else $w(n) = 1$ for $-N \leq n \leq N$

by changing the weighting value $w(n)$ according to edge map while preserving the edge aside of the filtering region. Thus, when the pixel at n position within the support region of variable filter belongs to edge map, all weighting values behind n are 0 if $n < 0$, and also all weighting values beyond n are 0 if $n > 0$.

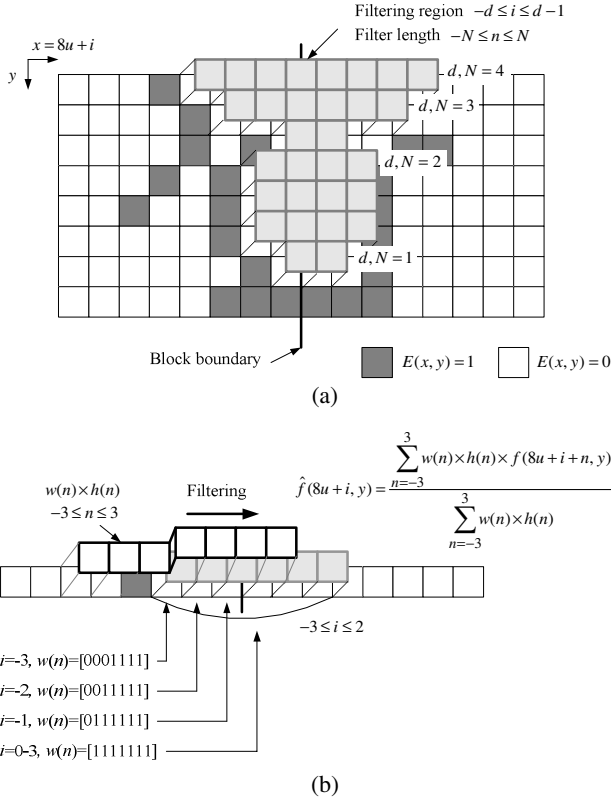


Fig. 3. (a) Filtering region and variable filter adaptive to edge map and (b) the filtering for 2th. line in (a) figure

2.4 Intra-block Filtering

The intensity difference of the neighborhood pixels and the angel with the intensity difference can be known from magnitude modulus $M_{2_1}(x, y)$ and phase modulus $A_{2_1}(x, y)$ at the first scale. If $M_{2_1}(x, y)$ at position (x, y) is higher than $M_{2_1}(x_1, y_1)$ and $M_{2_1}(x_2, y_2)$, $M_{2_1}(x, y)$ is the local modulus maxima $L(x, y)$. (x_1, y_1) and (x_2, y_2) are two positions that are adjacent to (x, y) to direction of phase modulus $A_{2_1}(x, y)$. Ringing artifact appears as the pseudo-noise around the abrupt edge. This pseudo-noise is detected by using local modulus maxima within edge block that include the abrupt edge. Edge block is a block including pixel that edge map is 1. The abrupt edge appears not at

only a block but at a series of blocks in general. If the number of edge block among 8 blocks $B_i(1 \leq i \leq 8)$ near the current edge block is above 2, this current block is set as block that must be performed to remove ringing artifact. And if $L(x, y)$ within this current block is above a threshold T , $L(x, y)$ is ringing artifact.

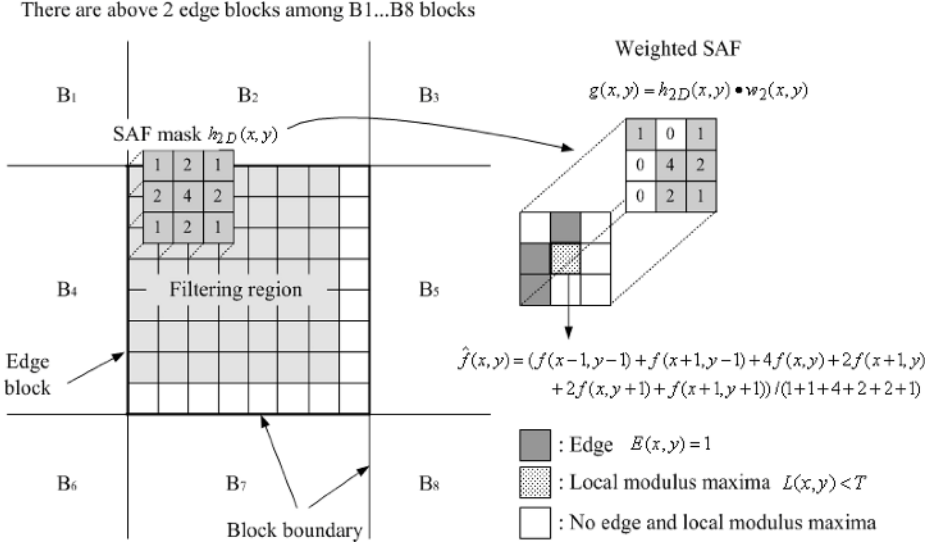


Fig. 4. Reduction for ringing artifact in intra-block using 3x3 SAF

Ringing artifact is reduced by using 3x3 SAF as shown in Fig. 4. Thus, the pixel detected as the pseudo-nose has to be filtered as follows;

$$\hat{f}(x, y) = \left(\sum_{i=-1}^1 \sum_{j=-1}^1 f(x+i, y+j) \times g(x+i, y+i) \right) / \sum_{i=-1}^1 \sum_{j=-1}^1 g(x+i, y+i) \quad (7)$$

where $x \neq 8u + 7, u = 0, 1, \dots, NV/8 - 1, y \neq 8v + 7, v = 0, 1, \dots, NH/8 - 1$

$g(x, y)$ is $h_{2D}(x, y) \times w_2(x, y)$ where 2D filter mask $h_{2D}(x, y)$ is $\{\{1, 2, 1\}, \{2, 4, 2\}, \{1, 2, 1\}\}$ and 2D weighting mask $w_2(x, y)$ is 0 if $M_{2,1}(x, y) > T_s$ and 1, otherwise. Since the weighting value of the abrupt edge within SAF support region is set to 0, ringing artifact can be reduced while preserving the edge. In case of the example in Fig. 4, weighting mask $w_2(x, y)$ is $\{\{1, 0, 1\}, \{0, 1, 1\}, \{0, 1, 1\}\}$ and $g(x, y)$ is $\{\{1, 0, 1\}, \{0, 4, 2\}, \{0, 2, 1\}\} / 11$.

3 Experimental Results

To demonstrate the performance of the proposed algorithm, a computer simulation was performed using MPEG-4 VM 18. DMB for mobile service provide generally the minimum quality of VCD degree in case of 5" LCD, which has 352x240 image size

and 15 frames per second. Therefore, the experiments used NEWS, MOTHER& DAUGHTER, FOREMAN, COAST GUARD, CONTAINER SHIP, and SILENT VOICE, which were CIF (352×288) and QCIF (176×144) in size. These moving pictures include 300 frames coded by MPEG-4 at various bit rates, frame rate, and QP. Fig. 5 shows an edge map and local modulus maxima of 17-th frame in FOREMAN coded to QP=30, 112kbps, and 15Hz. The grid noise in monotone cannot be detected as the edge so that the grid noise has to be removed, as shown in Fig. 5 (a). Furthermore, ringing artifact around cloth and building line can be detected as local modulus maxima as shown in Fig. 5 (b).

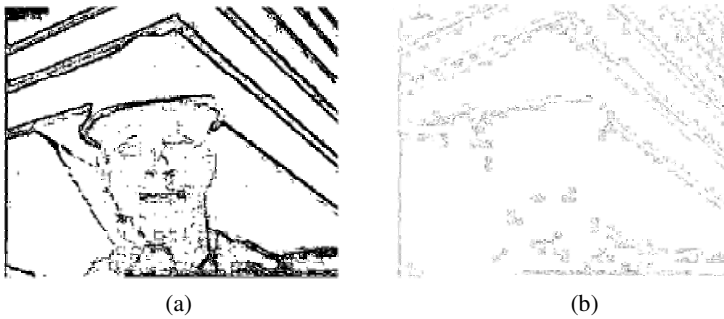


Fig. 5. (a) Edge map and (b) local modulus maxima of 17-th. frame in Foreman coded to QP=30, 112 kbps, 15 Hz

In the current study, the PSNR, one of the most popular criteria in image processing, was used to objectively evaluate the performance results, even though it is not an optimum criteria from the perspective of the human visual system. Plus, the images resulting from processing using the proposed algorithm and a conventional algorithm were also compared subjectively. A comparison of the PSNR results is shown in Table 1, in which various moving pictures were decoded in MPEG-4 and processed using MPEG-4 VM filtering, Kim's algorithm, and the proposed algorithm. This table shows that average PSNR of the proposed algorithm was higher 0.01dB – 0.20dB than that of the other conventional algorithms.

Fig. 6 shows the 17th. frame in FOREMAN sequence decoded to QP=30, 112kbps, and 15Hz by MPEG-4 and the postprocessed frames. The frames postprocessed by VM-18 post-filter in Fig. 6 (b) shows the edge blurred by the excessive low pass filtering and the staircase noise of the discontinuous edge. The frames postprocessed by Kim's algorithm in Fig. 6 (c) show still visible blocking artifacts around eye and hat in FOREMAN and also ringing artifact. However, the frames postprocessed by the proposed algorithm in Fig. 6 (d) show obvious improvement in terms of the removal of grid noise in flat region and ringing noise in edge block without blurring the edge and also the continuous of the staircase noise. Consequently, the performance of the proposed algorithm was demonstrated as superior to that of the other conventional algorithms based on both the PSNR and the subjective image quality.

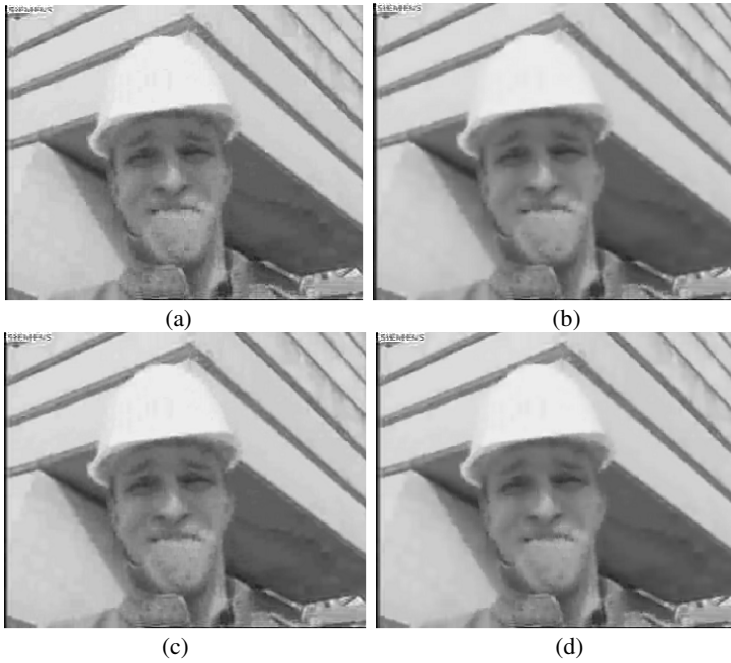


Fig. 6. (a) FOREMAN decoded with MPEG-4 by QP = 30, 112 kbps, and 15 Hz, and post-processed sequences by (b) VM-18 post filter method, (c) Kim’s method, (d) proposed method

Table 1. Experimental results for MPEG-4 decoded sequences

| Bit rate, size, and frame rate | Sequences | QP | Average PSNR [dB] | | | |
|--------------------------------|------------------|----|-------------------|-------|-------|----------|
| | | | MPEG-4 | VM-18 | Kim | Proposed |
| 10 kbps, QCIF, 7.5 Hz | CONTAINER SHIP | 17 | 29.53 | 29.64 | 29.83 | 29.84 |
| | MOTHER& DAUGHTER | 15 | 32.28 | 32.40 | 32.48 | 32.52 |
| 24 kbps, QCIF, 10 Hz | CONTAINER SHIP | 10 | 32.78 | 32.96 | 32.95 | 32.96 |
| | MOTHER& DAUGHTER | 8 | 35.32 | 35.38 | 35.40 | 35.46 |
| 48 kbps, QCIF, 10 Hz | FOREMAN | 13 | 30.96 | 31.10 | 31.12 | 31.14 |
| | COAST GUARD | 14 | 29.08 | 29.11 | 29.11 | 29.13 |
| | SILENT VOICE | 7 | 34.42 | 34.59 | 34.60 | 34.62 |
| 48 kbps, CIF, 7.5 Hz | MOTHER& DAUGHTER | 10 | 36.01 | 36.10 | 36.17 | 36.21 |
| | NEWS | 18 | 31.26 | 31.37 | 31.31 | 31.42 |
| | HALL MONITOR | 12 | 33.60 | 33.63 | 33.65 | 33.75 |
| 112 kbps, CIF, 15 Hz | FOREMAN | 30 | 28.43 | 28.53 | 28.57 | 28.59 |
| | COAST GUARD | 29 | 26.37 | 26.53 | 26.57 | 26.57 |
| | CONTAINER SHIP | 10 | 33.18 | 33.33 | 33.40 | 33.41 |

4 Conclusions

This paper presented the MPEG postprocessing system of quantization noise reduction based on the variable filter adaptive to edge. Firstly, the proposed algorithm obtains the edge map by thresholding the wavelet coefficient in multiscale edge. The variable filter is performed by at inter-block to remove blocking artifacts while changing the tap number and the filtering region according to the edge map. The coefficient of variable filter is determined by the blocking strength and quantization parameter. 2D SAF is performed within edge block to remove ringing artifact by using local modulus maxima. Experimental results confirmed that the proposed algorithm was superior to other conventional algorithms as regards the subjective image quality and PSNR, which was improved by 0.02 dB - 0.29 dB.

Acknowledgement

This research was supported by the Program for the Training of Graduate Students in Regional Innovation which was conducted by the Ministry of Commerce Industry and Energy of the Korean Government.

References

- [1] Motion Picture Experts Group, "MPEG test model 5 draft revision 2," ISO-IEC JTC1/SC29/WG11/602, Nov. 1993.
- [2] MPEG-4 Video Group, "MPEG-4 video verification model version 18.0," ISO-IEC JTC1/SC29/WG11, N3908, Jan. 2001.
- [3] G. K. Wallace, "The JPEG still picture compression standard," IEEE Trans. Consumer Electronics, vol. 38, no. 1, pp. xviii-xxxiv, Feb.1992.
- [4] M. D. Adams, "The JPEG-2000 Still Image Compression Standards," Tech. Rep. N2412, ISO/IEC JTC1/SC29/WG1, Sep. 2001.
- [5] S. D. Kim, J. Y. Yi, H. M. Kim, and J. B. Ra, "A deblocking filter with two separate modes in block- based video coding," IEEE Trans. Circuits Syst. Video Technol., vol. 9, no. 1, pp. 150-160, Feb. 1999.
- [6] Y. L Lee, H. C. Kim, and H. W. Park, "Blocking effect reduction of JPEG images by signal adaptive filtering," IEEE Trans. Image Processing, vol. 7, no. 2, pp.229-234, Feb. 1998.
- [7] G. A. Triantafyllidis, D. Tzovaras, and M. G Strintzis, "Blocking artifact detection and reduction in compressed data,"IEEE Trans. Circuits Syst. Video Technol., vol. 12, no. 10, pp. 877-890, Oct. 2002.
- [8] Y. Yang, N. Galatanos, and A. Katsaggelos, "Projection-based spatially adaptive reconstruction of block-transform compressed image," IEEE Trans. Image Processing, vol. 4, no. 7, pp. 896-908, July 1995.
- [9] N. C. Kim, I. H. Jang, D. H. Kim, and W. H. Hong, "Reduction of blocking artifact in block-coded images using wavelet transform," IEEE Trans. Circuits Syst. Video Technol., vol. 8, no. 3, pp. 253-257, June 1998.
- [10] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," IEEE Trans. PAMI, vol. 14, no. 7, pp. 710-732, July 1992.

Computational Framework for Family of Order Statistic Filters for Tensor Valued Data

Bogusław Cyganek

AGH – University of Science and Technology
Al. Mickiewicza 30, 30-059 Kraków, Poland
cyganek@uci.agh.edu.pl

Abstract. Nonlinear digital filters based on the order statistic belong to the very powerful methods of image restoration. The well known is the median filter operating on scalar valued images. However, the operation of the median filter can be extended into multi-valued pixels, such as colour images. It appears that we can go even further and define such filters for tensor valued data. Such a median filter for tensor valued images was originally proposed by Welk et.al. [10]. In this paper we present a different approach to this concept: We propose the family of nonlinear order statistic filters operating on tensor valued data and provide the computational framework for their non-numerical implementation.

Keywords: Order statistics, median filtering, weighted rank filters, tensor data.

1 Introduction

The order statistic based filtering of signals attracts much attention [2][7][8]. When applied to images this non-linear systems exhibit many desirable features that cannot be achieved with linear filters, such as edge preservation, removal of impulsive noise, etc. They operate on a fixed window of data for which the algebraic order of samples is used to find out the output value. However, there is a significant difference in finding order for scalar and non-scalar data, since for the latter there is no single way of ordering.

The most popular example of the order statistic filters is a median filter. For the grey valued images its operations is very simple: image intensity values found in a window are sorted and the middle sample is taken as an output. For example 5 is the median from the set $\{5,0,7,13,1\}$. However, for colour images, where pixels are composed of three scalars, there is not a uniform way of data ordering. Thus many solutions are possible, for instance a separate filtering of each channel, filtering of assigned scalars to colour pixels, etc. [9].

The concept of median filtering of vector data was further extended to the matrix and tensor valued data by Welk et.al. [10]. Filtering of tensor data is important for processing of the diffusion tensors of the magnetic resonance images, structural tensors (ST) [1][3], etc. Welk et.al. proposed a median filter of tensor data with the Frobenius norm used to measure their distance. This norm has an advantage of being rotation invariant which is a desirable property in case of tensors. The minimization

problem was solved with a modified gradient descent method. The median value in this case does not necessarily belong to the set of input data, however.

In this paper we propose a computational framework for the non-numerical realization of the family of order statistic filters operating on tensor valued data. It provides a uniform way of data processing for *different weighted* order statistics (not only median), as well as for different norms on tensors. On the other hand it is much easier approach when porting to the hardware accelerated platforms.

2 Order Statistic Filters on Tensor Data

As alluded to previously, realization of the order statistic for scalars, vectors, and tensor data differ in ordering scheme. For non-scalar values (multi-channel images) it should provide an equal importance to all components and use them all in data ordering. The usual way is to define a dissimilarity measure between each pair of data x_i and x_j , as follows [2][7][8][9]:

$$d_{(\cdot)}(x_i, x_j), \quad (1)$$

where it is assumed that $d_{(\cdot)}$ fulfils the distance properties (the subscript (\cdot) denotes a specific distance). Then, based on (1), an aggregated weighted distance $D_{(\cdot)}(x_i, \{x\}_K, \{c\}_K)$ of a sample x_i (for $i=1, 2, \dots, K$) with respect to the set $\{x\}_K$ of K other samples can be defined, as follows:

$$D_{(\cdot)}(x_i, \{x\}_K, \{c\}_K) = \sum_{j=1}^K c_j d_{(\cdot)}(x_i, x_j), \quad (2)$$

where $\{c\}_K$ is a set of K scalars (weights). Thus, choice of a distance $d_{(\cdot)}$ determines properties of the aggregated distance $D_{(\cdot)}$. The most popular metrics for vector data follow the Minkowski's metric $d_{(M)}$, defined for the two vectors \mathbf{a} and \mathbf{b} from the S dimensional space, as follows:

$$d_{(M, \alpha)}(\mathbf{a}, \mathbf{b}) = \left(\sum_{k=1}^S |a_k - b_k|^\alpha \right)^{1/\alpha}, \quad (3)$$

where a_k and b_k stand for the k -th component of the vectors \mathbf{a} and \mathbf{b} , respectively, and α is a parameter. To measure distance for the tensor valued data (or matrices) one of the matrix norms can be used. For completeness we remind the three most popular:

$$d_{(1)}(\mathbf{A}, \mathbf{B}) = \max_{1 \leq i \leq m} \sum_{j=1}^n |a_{ij} - b_{ij}|, \quad (4)$$

$$d_{(\infty)}(\mathbf{A}, \mathbf{B}) = \max_{1 \leq j \leq n} \sum_{i=1}^m |a_{ij} - b_{ij}|, \quad (5)$$

$$d_{(F)}(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij} - b_{ij}|^2}, \quad (6)$$

where \mathbf{A} and \mathbf{B} are $m \times n$ tensors with scalar elements a and b , respectively. The last $d_{(F)}$ is the Frobenius norm which, due to its desirable rotation invariance property, was used by Welk et.al. [10]. In the special case of the 2D tensor \mathbf{U} (such as the 2D structural tensors which are semi positive [3]) we have:

$$\mathbf{U} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} \text{ where } u_{21}=u_{12}, \tag{7}$$

and for two such tensors \mathbf{U} and \mathbf{V} , $d_{(F)}$ in (6) takes the following form:

$$d_{(F)}(\mathbf{U}, \mathbf{V}) = \sqrt{(u_{11} - v_{11})^2 + 2(u_{12} - v_{12})^2 + (u_{22} - v_{22})^2}. \tag{8}$$

Just recently, for the segmentation of diffusion tensor images, Luis-García et.al. proposed to use the Riemannian distance which is intrinsic to the manifold of the symmetric positive definite (SPD) matrices. It takes the following form [6]:

$$d_{(R)}(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{k=1}^N \ln^2(\lambda_k)}, \tag{9}$$

where \mathbf{A} and \mathbf{B} are two SPD matrices of rank N , λ_k are the positive eigenvalues of the $\mathbf{A}^{-1}\mathbf{B}$.

The order statistic filters are defined as follows: given a set of data $\{x\}_K$, for each $x_i \in \{x\}_K$ the scalar $D_{(\cdot)}(x_i)$ is computed according to (2) and chosen $d_{(\cdot)}$ in (1). Then these scalars are ranked in the order of their values, which process automatically orders their associated tensors x_i as well. The procedure is as follows (omitting subscript for clarity) [2] [7][8][9]:

$$\begin{aligned} D^{(1)} \leq D^{(2)} \leq \dots \leq D^{(p)} \leq \dots \leq D^{(K)}, \\ x^{(1)} \rightarrow x^{(2)} \rightarrow \dots \rightarrow x^{(p)} \rightarrow \dots \rightarrow x^{(K)}, \end{aligned} \tag{10}$$

where $D^{(p)} \in \{D(x_1), D(x_2), \dots, D(x_K)\}$ and $x^{(p)} \in \{x_1, x_2, \dots, x_K\}$ for $1 \leq p \leq K$. Having ordered data, one data $x^{(p)}$ is chosen as an output of an order statistic filter, depending on the desired characteristics of that filter. The special and very important case is the weighted median filter, defined generally as follows:

$$x_m = \arg \min_{x_i \in \{x\}_K} D_{(\cdot)}(x_i, \{x\}_K, \{c\}_K), \tag{11}$$

where x are tensors (or matrices, vectors, scalars) from a set $\{x\}_K$ of data with a chosen aggregated distance measure $D_{(\cdot)}$, $\{c\}_K$ is a set of scalar weights, and x_m is the median.

If in (2) $\forall i: c_i=1$, then (11) denotes the classical median filter. There are many mutations of (11) possible as well [2][7][8][9]. Welk et.al. [10] propose the numerical solution to (11) by a gradient descent method with a step size control to overcome the problem of missing information in the gradient of the sum on the distance to the minimum. However, such procedure leads to a solution which does not necessarily belong to the input set of data $\{x\}_K$. This is the main difference to this paper.

3 Algorithmic Considerations

Formula (10) with a chosen aggregated distance (2) constitute the base for implementation of the order statistic filters for tensor data. In this section we focus on the algorithmic aspects of data ordering in (10) and choice of an output value of a filter.

Fig. 1 depicts data structures for computations of the partial aggregated distances $D_{(.)}$ among nine elements, i.e. the filtering window is 3×3 for simplicity. The main structure is a square matrix \mathbf{M} that contains mutual distances $d_{(.)}$. Due to their metric properties \mathbf{M} is symmetrical with a zero diagonal. Thus, we need only to compute distances in an upper (or lower) triangle, since in general for K elements in the window we have $\binom{K}{2}$ different pairs of distances to be determined. The weights c_i are stored in a separate matrix \mathbf{C} . The aggregated distances $D_{(.)}$ are weighted sums of the consecutive rows, e.g. $D(x_7, \{x_i\}_{1 \leq i \leq 9}) = \sum_{1 \leq i \leq 9} c_i m[7, i]$. $D_{(.)}$ need to be computed only once, then they are rank ordered or otherwise processed to find the output value. For example, for the median filter we look for the minimal value of $D_{(.)}(x_p)$ and the corresponding x_p is the sought median value.

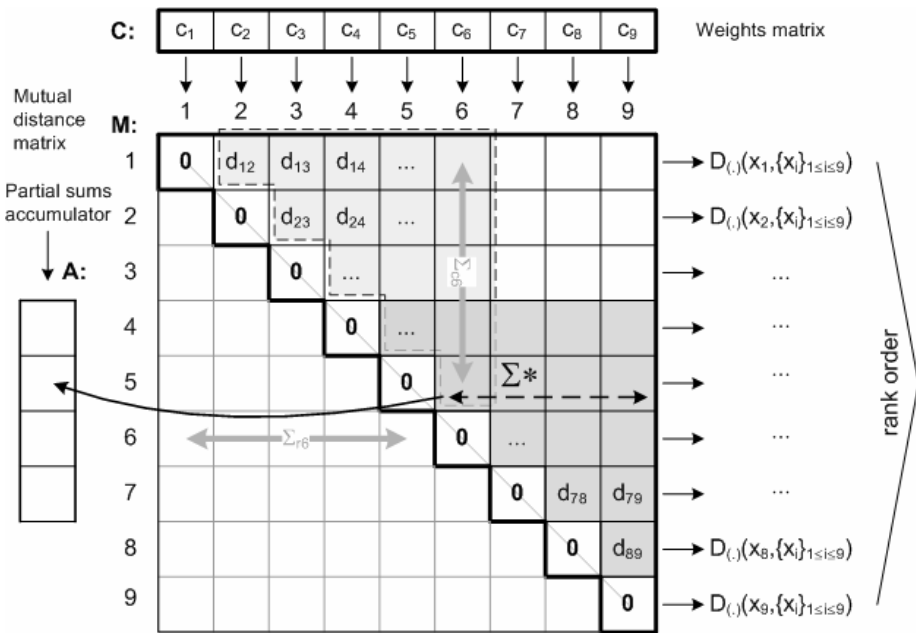


Fig. 1. Data structures for computation of the aggregated distances $D_{(.)}$

Fig. 2 illustrates filtering in a square odd size window (3×3 for simplicity) created around a central pixel. We notice that when moving filter to a new position W_{k+1} only one column (or a row for a vertical step) of the window is refilled with new data (pixels 1-3 in Fig. 2). This is a key observation leading to speed improvements of the algorithm. As a consequence in the matrix \mathbf{M} it is necessary to exchange *only* the rows associated with the changed data (i.e. the rows 7-9 in this case – see Fig. 1 and Fig. 2). The other distances can be reused (the grey area in Fig. 1).

The remaining distances (grey area in Fig. 1) have to be copied from the lower-right corner to the upper-left corner of \mathbf{M} . The rows with new data (i.e. the rows 7-9) have to be filled with the computed $d_{(\cdot)}$.

For the non-weighted versions of (2) (i.e. $\forall i: c_i=1$), the partial sums Σ^* of $D_{(\cdot)}$ can be stored in the partial-sums-accumulator \mathbf{A} in Fig. 1. It holds also that the partial column sums are the same as the corresponding partial row sums, i.e. Σ_{c6} equals Σ_{r6} in Fig. 1, so we don't need to store anything in the lower triangle of \mathbf{M} . This can speed computations and save memory especially for larger filtering windows.

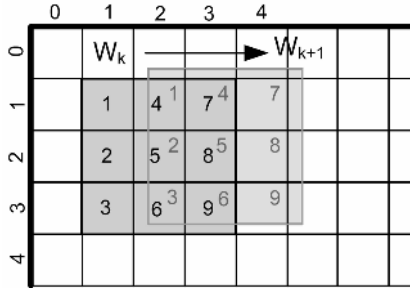


Fig. 2. The filtering window: moving one pixel right - only its last column needs to be refilled

We note also that the proposed data structures and algorithms are simpler for hardware implementation than e.g. direct numerical minimization of (11).

4 Experimental Results

The experimental results present rank filtering of the structural tensor \mathbf{U} in a form (7), which is further processed to exhibit local structures in images. The extension consists in a mapping of (7) into the tensor $\mathbf{s}=[s_1, s_2, s_3]$, where $s_1=u_{11}+u_{22}$ represents a magnitude, $s_2=\angle \mathbf{w}$ an angle, and $s_3=\rho$ a coherence of local structures in images. The mapping is given as follows [4][1][3]:

$$\mathbf{s} = \begin{bmatrix} u_{11} + u_{22} \\ \angle \mathbf{w} \\ \rho \end{bmatrix}, \quad \angle \mathbf{w} = \begin{cases} \arctan\left(\frac{2u_{12}}{u_{11} - u_{22}}\right), & u_{11} \neq u_{22} \\ \frac{\pi}{2}, & u_{11} = u_{22} \wedge u_{12} \geq 0 \\ -\frac{\pi}{2}, & u_{11} = u_{22} \wedge u_{12} < 0 \end{cases} \quad \mathbf{w} = \begin{bmatrix} u_{11} - u_{22} \\ 2u_{12} \end{bmatrix} \quad (12)$$

$$\rho = \frac{(\lambda_1 - \lambda_2)^2}{(\lambda_1 + \lambda_2)^2} = \frac{\|\mathbf{w}\|^2}{(\text{Tr}(\mathbf{U}))^2}$$

where λ_1 and λ_2 are eigenvalues of \mathbf{U} .

Fig. 3 depicts a grey test image ‘‘Street’’ (a) and the two colour representations of \mathbf{s} (magnitude, angle, and coherence represented as HSI channels), where first one is original (b), the second was created from \mathbf{s} filtered by the 7×7 non-weighted median filter (11). The latter shows more regular structures due to removal of the sparse data and noise.



Fig. 3. The “Street” grey-valued image (a), the colour representation of the ST (b), the colour representation of the structural tensor prior filtered with the 5×5 non-weighted median

Table 1. Speed of execution of the median filtering of the $512 \times 512 \times 3$ tensor data (floating point data representation)

| Filt. window size (# of dists. d_i) | 3×3 (36) | 5×5 (300) | 7×7 (1176) | 9×9 (3240) |
|--|-------------------|--------------------|---------------------|---------------------|
| Exec. time - Simple [s] | 0.812 | 3.75 | 12.8 | 34.3 |
| Exec. time - Optimized [s] | 0.44 | 1.51 | 4.72 | 12.29 |

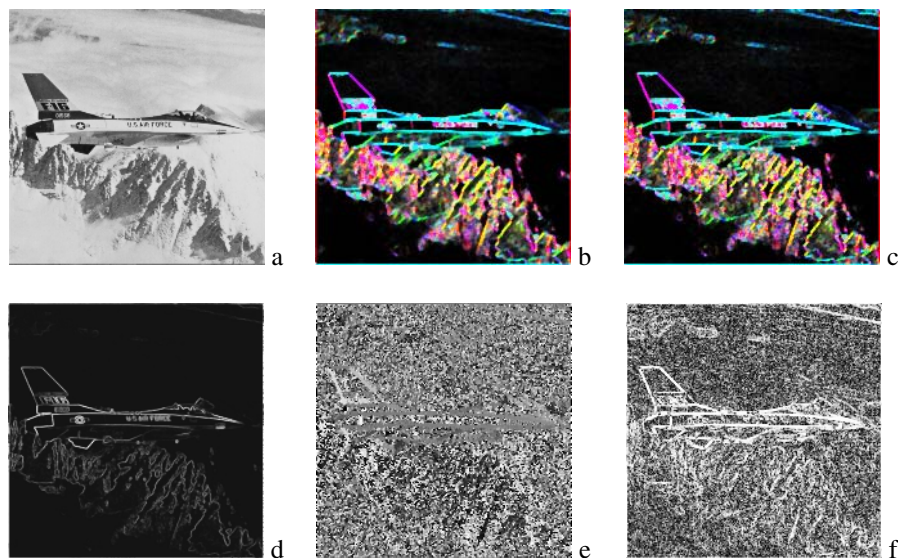


Fig. 4. The “Airplane” grey-valued image (a), the colour representation of the ST prior filtered with the 7×7 central-pixel-weighted-median with $c_m=2.2$ (b), and $c_m=10.0$ (c). The 3×3 median filtered channels s of the ST: magnitude (d), angle (e), coherence (f).

Fig. 4 presents the central-weighted-median-filter applied to the ST of the “Airplane” test image (a). The ST colour representation, filtered with the 7×7 central-pixel-weighted-median, with $c_i=1.0$ and the middle weight $c_m=2.2$ presents Fig. 4b, and for $c_m=10.0$ Fig. 4c; The mag-ang-coh of the ST 3×3 median filtered depict Fig. 4d-f. The Frobenius norm (8) was used in all experiments. Table 1 presents execution timings on Pentium 3.4 GHz with 2GB RAM. Software was implemented on the Microsoft Visual® C++ 6.0 platform. The speed-up factor for the optimized version (Sec.3) is about 2.5-2.9.

5 Conclusions

In this paper we propose the computational framework for the non-numerical realization of the family of the order statistic filters operating on tensor valued data. It is a different approach to the concept proposed by Welk et.al. [10]. The main novelty is a uniform treatment of the whole family of the rank ordered weighted filters, not only the median. The second difference from [10] is the non-numerical solution of the minimization problem for medians. Thanks to this feature the found median value always belong to the set of input data. Last but not least, the presented algorithms and data structures allow fast execution in software and simpler hardware implementation.

References

1. Bigün, J., Granlund, G.H., Wiklund, J.: Multidimensional Orientation Estimation with Applications to Texture Analysis and Optical Flow. *IEEE PAMI* **13**(8), (1991) 775-790
2. Dougherty, E.R., Astola, J.T.: *Nonlinear filters for image processing*. IEEE Press (1999)
3. Granlund, G.H., Knutsson, H.: *Signal Processing for Computer Vision*. Kluwer (1995)
4. Jähne, B.: *Digital Image Processing*, Springer-Verlag (1997)
5. Klette, R., Zamperoni, P.: *Handbook of Image Processing Operators*. Wiley (1996)
6. de Luis-García, R., Deriche, R., Rousson, M., Alberola-López, C. : *Tensor Processing for Texture and Colour Segmentation*. LNCS 3540 (2005) 1117-1127
7. Mitra, S.K., Sicuranza, G.L.: *Nonlinear image processing*. Academic Press (2001)
8. Pitas, I., Venetssanopoulos, A.N.: *Nonlinear Digital Filters*. Kluwer (1990)
9. Smolka, B.: *Nonlinear techniques of noise reduction in digital color images*. Silesian University (2004)
10. Welk, M., Feddern, C., Burgeth, B., Weickert, J.: Median Filtering of Tensor-Valued Images. LNCS 2781, Springer (2003) 17-24

Gaussian Noise Removal by Color Morphology and Polar Color Models

Francisco Ortiz

Image Technological Group
Dept. Physics, Systems Engineering and Signal Theory
University of Alicante
Ap. Correos 99, 03080 Alicante, Spain
fortiz@ua.es

Abstract. This paper deals with the use of morphological filters by reconstruction of the mathematical morphology for Gaussian noise removal in color images. These new vector connected have the property of suppressing details preserving the contours of the objects. For the extension of the mathematical morphology to color images we chose a new polar color space, the *l1-norme*. This color model guarantees the formation of the complete lattice necessary in mathematical morphology avoiding the drawbacks of others polar spaces. Finally, after having defined the vectorial geodesic operators, the opening and closing by reconstruction are then employed for the Gaussian noise elimination.

1 Introduction

In image processing, one of the most important tasks is the improving of the visual presentation or the reduction of noise. The main problem in noise removal is to get a clean image, which is without noise, but keeping all attributes of the original image as could be the shape, size, color, disposition and edges, among others. Some methods have been proposed so far for the elimination of noise. In [1], a directional vectorial filter is defined, which minimises the angle between vectors to eliminate the noisy pixel. In [2], the so-called vectorial median filters are an extension of the scalar median filter. In such a case, the ordering of the vectors is done with a Euclidean distance value. In [3], a combination of morphological alternate filters is used with RGB to reduce the impulsive noise.

In this paper, we show the utility of using the vectorial geodesic reconstruction in combination with a mean filter for eliminating Gaussian noise from the color images. Section 2 presents the color space chosen for mathematical processing, the *l1-norme*. In Section 3, we extend the geodesic operations to color images. In Section 4, we apply the vector-connected filters for eliminating the Gaussian noise in color images. Finally, our conclusions are outlined in the final section.

2 Color Mathematical Morphology

Mathematical morphology is a non-linear image processing approach which is based on the application of lattice theory to spatial structures [4]. The definition of

morphological operators needs a totally ordered complete lattice structure [5]. The application of mathematical morphology to color images is difficult, due to the vectorial nature of the color data (RGB, CMY, HLS, YIQ...). Many works have been carried out on the application of mathematical morphology to color images [6,7,8,9,10]. The most commonly adopted approach is based on the use of a lexicographical order which imposes a total order on the vectors. This way, we avoid the false colors in an individual filtering of signals. Let $\mathbf{x}=(x_1, x_2, \dots, x_n)$ and $\mathbf{y}=(y_1, y_2, \dots, y_n)$ be two arbitrary vectors ($\mathbf{x}, \mathbf{y} \in Z^n$). An example of lexicographical order o_{lex} , will be:

$$\mathbf{x} < \mathbf{y} \text{ if } \begin{cases} x_1 < y_1 & \text{or} \\ x_1 < y_1 & \text{and } x_2 < y_2 & \text{or} \\ x_1 < y_1 & \text{and } x_2 < y_2 \dots \text{ and } \dots x_n < y_n \end{cases} \quad (1)$$

On the other hand, it is important to define the color space in which operations are to be made. The intuitive systems (HSI, HLS, HSV,...) are used in color vision because they represent the information in a similar way to the human brain. The preference or disposition of the components of the HSI in the lexicographical ordering depends on the application and the properties of the image. Ordering with luminance (intensity) in the first position is the best way of preserving the contours of the objects in the image (lattice influenced by intensity). In situations in which the objects of interest are highly colored or in which only objects of a specific color are of interest, the operations with hue in the first position are the best (lattice influenced by hue).

2.1 The New Color Space for Processing: *LI-Norme*

There are a great number of variations for the transformation of RGB to polar color spaces (HSI, HLS HSV...). Some transformations cause that these spaces present incoherences that prevent the use of these color representations in some image processing. These chromatic models are inadequate for the quantitative treatment of the images. For example, some instability arises in saturation of these spaces for small variations of RGB values. In addition, the saturation of the primary colors is not visually agreeable. An advisable representation must be based in distances or norms for the vectors and to provide independence between the chromatic and achromatic signals.

In order to avoid the inconveniences of HSI, HLS or HSV color models, we use, in our investigation, the new Serra's *LI-norme* [11]. Figure 1 shows the MS diagram from Serra's *LI-norme* as a positive projection of all the corners of the RGB cube in a normalization of the achromatic line to the *m* called signal.

This new chromatic representation has been very useful in brightness elimination of color images [12,13]. The intensity (achromatic signal) *m* and saturation signal *s* in the *li-norme* are calculated from *r*, *g* and *b* values of RGB, where the *m* signal calculated is a normalization ($0 \leq m \leq 255$) of the achromatic axes of the RGB cube, and the *s* values are visually more agreeable with respect to the saturation of HLS or HSV spaces:

$$s = \begin{cases} m = \frac{1}{3}(r + g + b) \\ \frac{1}{2}(2r - g - b) = \frac{3}{2}(r - m) & \text{if } (b + r) \geq 2g \\ \frac{1}{2}(r + g - 2b) = \frac{3}{2}(m - b) & \text{if } (b + r) < 2g \end{cases} \quad (2)$$

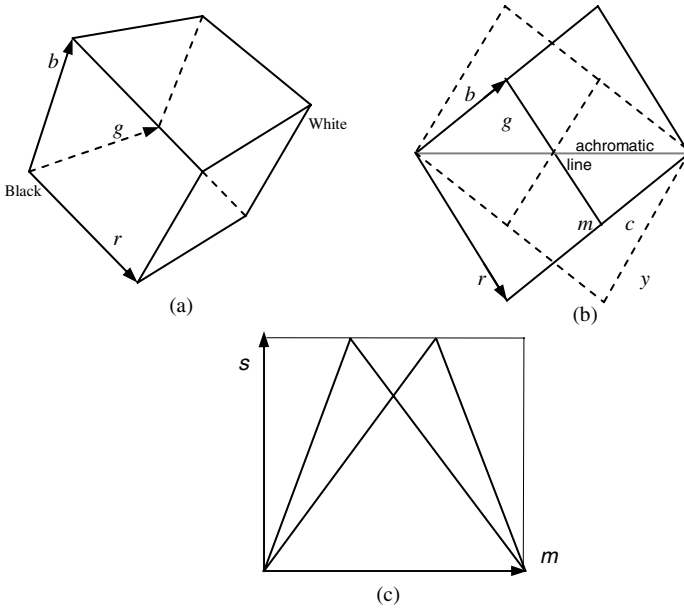


Fig. 1. RGB cube and its transformation in MS diagram. (a) 3D projection. (b) 2D opposite projections. (c) Shape and limits of MS diagram.

3 Vector Connected Filters

Morphological filters by reconstruction have the property of suppressing details, preserving the contours of the remaining objects [14,15]. The use of these filters in color images requires an ordering relationship among the pixels of the image. For the vectorial morphological processing the lexicographical ordering previously defined o_{lex} will be used. As such, the infimum (\wedge_v) and supremum (\vee_v) will be vectorial operators, and they will select pixels according to their order o_{lex} in the l_1 -norme.

Once the orders have been defined, the morphological operators of reconstruction for color images can be generated and applied. A elementary geodesic operation is the geodesic dilation. Let \mathbf{g} denote a marker color image and \mathbf{f} a mask color image (if $o_{lex}(\mathbf{g}) \leq o_{lex}(\mathbf{f})$, then $\mathbf{g} \wedge_v \mathbf{f} = \mathbf{g}$). The vectorial geodesic dilation of size 1 of the marker image \mathbf{g} with respect to the mask \mathbf{f} can be defined as:

$$\delta_{\mathbf{v}\mathbf{f}}^{(1)}(\mathbf{g}) = \delta_{\mathbf{v}}^{(1)}(\mathbf{g}) \wedge_{\mathbf{v}} \mathbf{f} \quad (3)$$

where it is very important that the infimum operation is done with the same lexicographical ordering as the vectorial dilation. This way, there are not false colors in the result.

The vectorial geodesic dilation of size n of a marker color image \mathbf{g} with respect to a mask color image \mathbf{f} is obtained by performing n successive geodesic dilations of \mathbf{g} with respect to \mathbf{f} :

$$\delta_{\mathbf{v}\mathbf{f}}^{(n)}(\mathbf{g}) = \delta_{\mathbf{v}\mathbf{f}}^{(1)} \left[\delta_{\mathbf{v}\mathbf{f}}^{(n-1)}(\mathbf{g}) \right] \quad (4)$$

with $\delta_{\mathbf{v}\mathbf{f}}^{(0)}(\mathbf{g}) = \mathbf{f}$

Geodesic transformations of bounded images always converge after a finite number of iterations [13]. The propagation of the marker image is impeded by the mask image. Morphological reconstruction of a mask image is based on this principle.

The vectorial reconstruction by dilation of a mask color image \mathbf{f} from a marker color image \mathbf{g} , (both with the same dominion and $o_{lex}(\mathbf{g}) \leq o_{lex}(\mathbf{f})$) can be defined as:

$$R_{\mathbf{v}\mathbf{f}}(\mathbf{g}) = \delta_{\mathbf{v}\mathbf{f}}^{(n)}(\mathbf{g}) \quad (5)$$

where n is such that $\delta_{\mathbf{v}\mathbf{f}}^{(n)}(\mathbf{g}) = \delta_{\mathbf{v}\mathbf{f}}^{(n+1)}(\mathbf{g})$.

The vectorial reconstruction is an algebraic opening only if all the operations between pixels respect the lexicographical order.

4 Application: Gaussian Noise Removal

Vector-connected filters will be used to eliminate Gaussian noise from color images. Chromatic Gaussian noise, in contrast to impulsive noise, it changes the entire definition of the image. Its effect is most obvious in the homogenous regions, which become spotty. With vector-connected filters we can reduce the Gaussian noise by merging the flat zones in the image. As such, the noisy image is simplified and the contours of the objects are preserved.

The images of our study are the color images of “Parrots” (Fig. 2.a), “Vases” (Fig. 2.c), “Lenna” (Fig. 2.e) and “Peppers” (Fig. 2.g) which has been corrupted by a Gaussian noise with a variance between 15 and 20.

We apply a vectorial opening by reconstruction (VOR) in the noisy image. For the morphological processing, the lexicographical ordering $o_{lex}=m \rightarrow saturation \rightarrow hue$, will be used. VOR is defined as follows:

$$\gamma_{\mathbf{v}\mathbf{f}}^{(n)} = \delta_{\mathbf{v}\mathbf{f}}^{(n)}(\mathcal{E}_{\mathbf{v}}^{(s)}(\mathbf{f})) \quad (6)$$

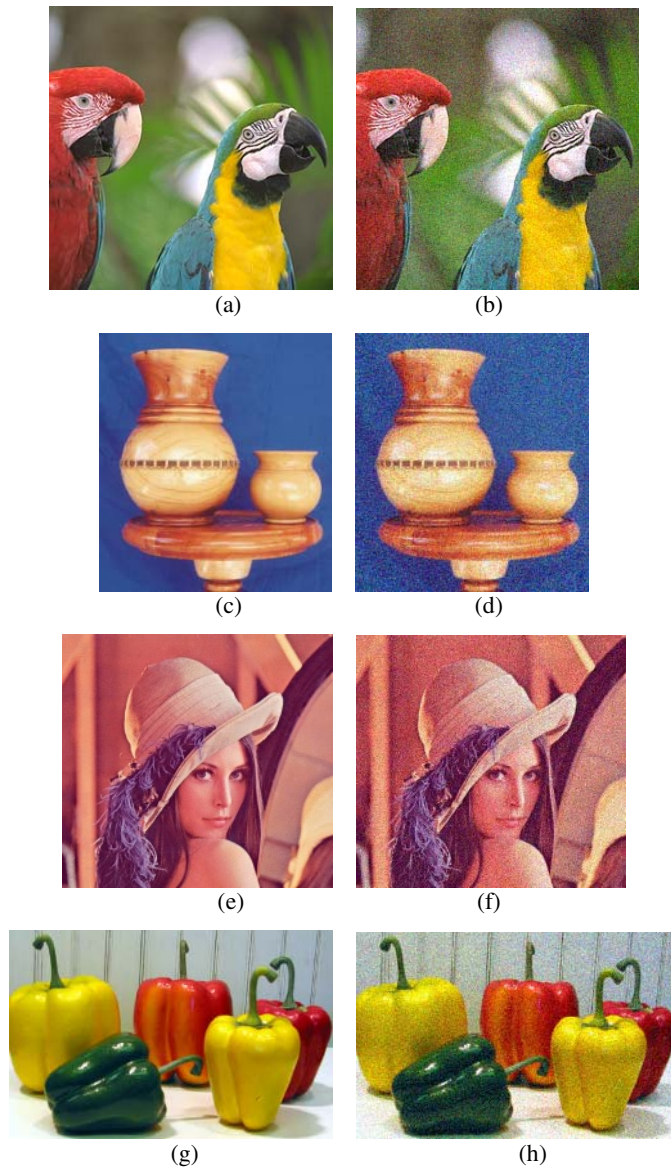


Fig. 2. Original and noisy images. (a-b) “Parrots” $\sigma = 20$, (c-d) “Vases” $\sigma = 15$, (e-f) “Lenna” $\sigma = 18$ and (g-h) “Peppers” $\sigma = 17$.

The vectorial erosion of the opening by reconstruction is made with increasing sizes (s) of the structuring element, from 5×5 to 11×11 . In Figure 3, the results of the connected filter for the “Parrots” image are observed. In accordance with the increase of the structuring element, the noise is reduced, but the image is darkened (VOR is an antie-extensive operation).

We now apply a vectorial closing by reconstruction (VCR) to the original image and we obtain the results which are presented in Figure 4. In this case, the image is clarified as the size of the structuring element of the vectorial dilation is increased. This is due to the extensive operation of the VCR. VCR is defined as:

$$\phi_{V\mathbf{f}}^{(n)} = \varepsilon_{V\mathbf{f}}^{(n)}(\delta_V^{(s)}(\mathbf{f})) \quad (7)$$

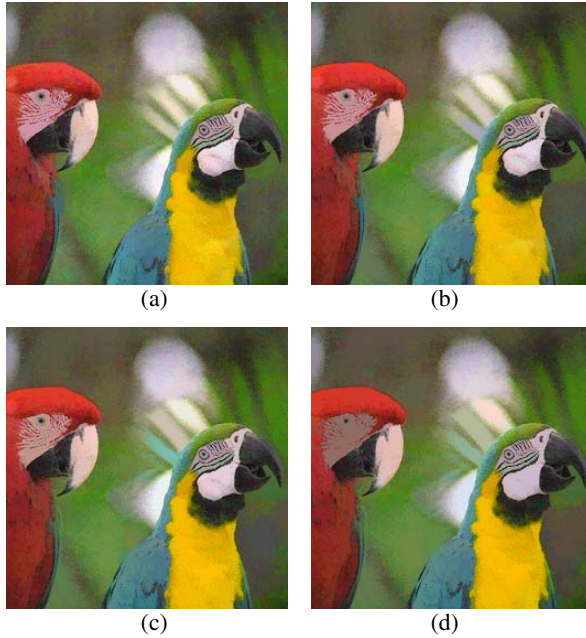


Fig. 3. Vectorial openings by reconstruction (VOR) from: 5x5-eroded image (a), 7x7-eroded image (b), 9x9-eroded image (c), 11x11-eroded image (d)

In Figure 5, the effect of reducing flat zones within the noisy image is shown. In particular, the upper left corner of the images is detailed. Figure 5.a shows the original noisy section. In Figures 5.b and 5.c, we can observe the progressive reduction of flat zones as the size of the structuring element (s) increases in the operation of erosion of the vectorial opening by reconstruction. For our purpose of noise removal we chose a structuring element of 7x7 as the best for preserving contours and details in the images.

In order to attenuate the dark and light effects of the previous filters, we calculate the vectorial mean of both filters. In Figure 6 we can see the art flow of the algorithm for Gaussian noise removal. The visual result of our algorithm and the final images are shown in Figure 7.

We use the normalised mean squared error (NMSE) to assess the performance of the different means of the filters. The NMSE test for color images is calculated from the RGB tri-stimulus values [3], where \mathbf{i} is the original color image and \mathbf{f} is the processed image:

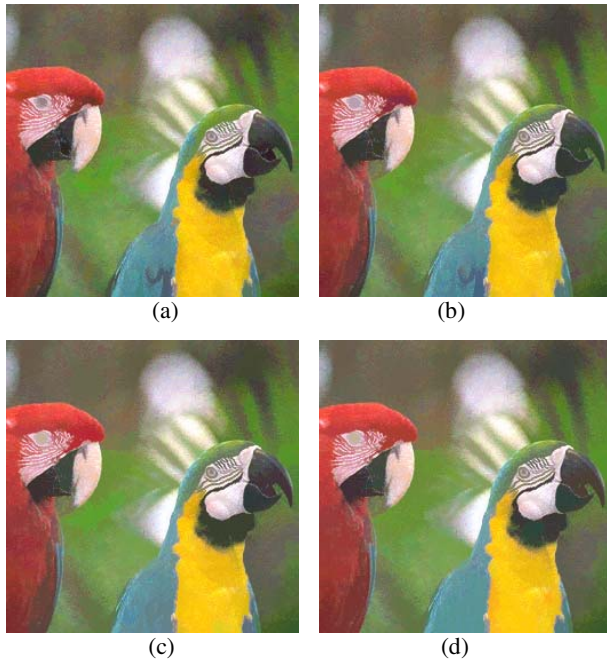


Fig. 4. Vectorial closings by reconstruction (VCR) from: 5x5-dilated image (a), 7x7-dilated image (b), 9x9-dilated image (c), 11x11-dilated image (d)

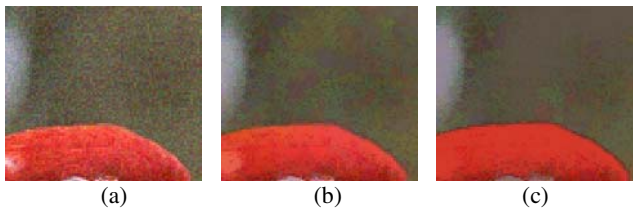


Fig. 5. Detail of simplification of the image. (a) noisy section. (b) VOR-7x7. (c) VOR-11x11.

$$NMSE = \frac{\sum \sum \left[(i_r(x, y) - f_r(x, y))^2 + (i_g(x, y) - f_g(x, y))^2 + (i_b(x, y) - f_b(x, y))^2 \right]}{\sum \sum (i_r^2(x, y) + i_g^2(x, y) + i_b^2(x, y))} \quad (8)$$

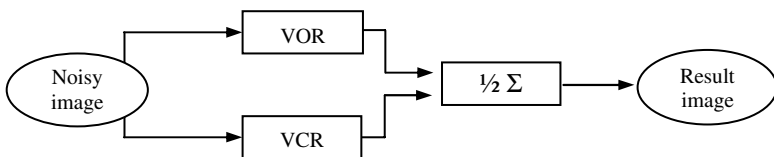


Fig. 6. The Algorithm for Gaussian noise attenuation in color images by morphological reconstruction. Structuring element (s) 7x7. $o_{lex}=m \rightarrow saturation \rightarrow hue$.

In addition to the NMSE test, three subjective criteria can be used:

- Visual attenuation of noise.
- Contour preservation.
- Detail preservation.

The evaluation of the quality with subjective criteria is divided into four categories: i.e., excellent (4), good (3), regular (2) and bad (1). The result of the filter on the noise-polluted images is illustrated in Table 1. The value of the NSME test for the original image and the noisy one are 0.0952 for “Parrots”, 0.0892 for “Vases”, 0.102 for “Lenna” and 0.919 for “Peppers”.

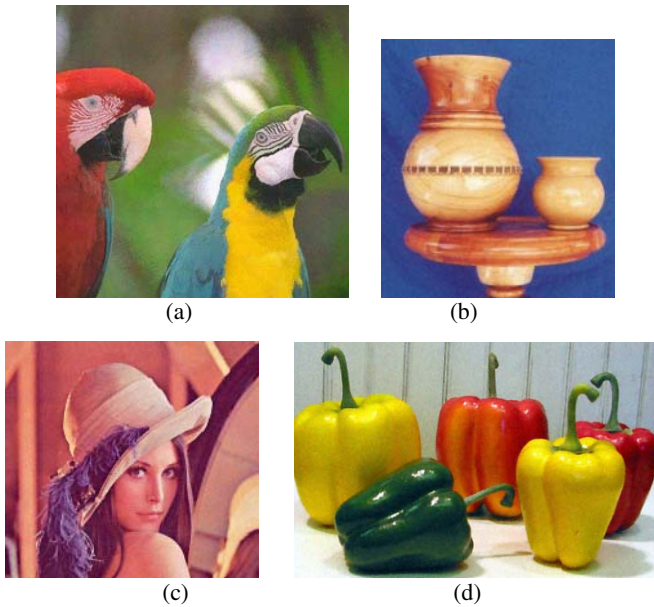


Fig. 7. Final results. Vectorial Mean of VOR-VCR in (a) “Parrots”, (b) “Vases”, (c) “Lenna” and (d) “Peppers”.

Table 1. Color NSME and subjective criteria. The “NSME original/noisy” is the normalised mean squared error calculated between the original image and the noisy image. The NSME “M-(VOR,VCR)” is the normalised mean squared error calculated between the original image and the filtered image.

| | NMSE Original/ noisy | NMSE Original/ M-(VORVCR) | Visual attenuation of noise | Contour preservation | Detail preservation |
|----------------|----------------------------|---------------------------------|-----------------------------------|-------------------------|------------------------|
| Parrots | 0.0952 | 0.0775 | 3 | 4 | 3 |
| Vases | 0.0892 | 0.0688 | 4 | 4 | 4 |
| Lenna | 0.102 | 0.0717 | 3 | 3 | 3 |
| Peppers | 0.0919 | 0.0597 | 4 | 4 | 4 |

Regarding the values from the NMSE test, we should emphasize that they are all located below the value of the noisy image. The most optimal result, according to the NSME test, is the presented in the image of "Vases". Regarding the subjective evaluation of the quality of the filter, the best conservation of contours and details are already presented in the "Vases" image (noise $\sigma = 15$). Nevertheless, visual attenuation of noise (subjective criteria) is achieved in all the images.

5 Conclusions

In this paper, we have presented an algorithm for eliminating Gaussian noise in color images. The geodesic method for reducing noise has been shown to be efficient for a Gaussian noise of a variance of 20. In the experiments carried out, better results were obtained for lower variances, specifically 15.

A best elimination of noise can be obtained if the morphological elemental operations of the connected filters are made with a large structuring element, but the best conservation of structures and details in the image are achieved with a smaller structuring element. For this reason we have used a structuring element of maximum size of 7×7 .

Based on the success shown by these results, we are now working on an improvement of our method for noisy removal in color images. We work in multi-processor configurations for color geodesic operations in order to reduce the processing time these operations.

References

1. Trahanias, P., Venetsanopoulos, A.: Vector directional filters: A new class of multichannel image processing filters. *IEEE Transactions on Image Processing*. Vol.2, I.4 (1993) 528-534.
2. Astola, J., Haavisto, P., Neuvo, Y.: Vector median filters. *Proceedings of the IEEE*, Vol. 78, I. 4, (1990) 678-689.
3. Deng-Wong, P., Cheng, F., Venetsanopoulos, A.: Adaptive Morphological Filters for Color Image Enhancement. *Journal of Intelligent and Robotic System*, Vol. 15, (1996) 181-207.
4. Serra, J.: *Image analysis and Mathematical Morphology*. Vol I, and *Image Analysis and Mathematical Morphology*. Vol II: Theoretical Advances, Academic Press, London (1982) and (1988).
5. Serra, J.: Anamorphoses and Function Lattices (Multivalued Morphology). In: E. Douguerty (Ed.): *Mathematical Morphology in Image Processing*, Marcel-Dekker, (1992) 483-523.
6. Hanbury, A., Serra, J.: Morphological operators on the unit circle. *IEEE Transactions on Image Processing*, Vol. 10, I. 12, (2001) 1842-1850.
7. Ortiz, F., Torres, F., De Juan, E., Cuenca, N.: Color Mathematical Morphology For Neural Image Análisis. *Real-Time Imaging*, Vol. 8 (2002) 455-465.
8. Ortiz, F.: *Procesamiento morfológico de imágenes en color. Aplicación a la reconstrucción geodésica*. PhD Thesis, University of Alicante (2002).
9. Ortiz, F., Torres, F., Angulo, J., Puente, S.: Comparative study of vectorial morphological operations in different color spaces. *Proceedings SPIE: Algorithms, Techniques and Active Vision*, vol. 4572, (2001) 259-268.

10. Peters II, A.: Mathematical morphology for angle-valued images. Proceedings of SPIE, Non-Linear Image Processing VIII , Vol. 3026, (1997) 84-94.
11. Serra, J.: Espaces couleur et traitement d'images. Tech. Report N-34/02/MM. Centre de Morphologie Mathématique, École des Mines de Paris (2002).
12. 12. Ortiz, F., Torres, F.: Real-time elimination of specular reflectance in colour images by 2D-histogram and mathematical morphology. Journal of Imaging Science and Technology. Special issue on Real-Time Imaging. Vol. 49 (2005) 220- 229.
13. Ortiz, F., Torres, F.: Vectorial Morphological Reconstruction for Brightness Elimination in Colour Images. Real Time Imaging. Vol. 10 (2004) 379-387.
14. Salembier, P., Serra, J.: Flat zones filtering, connected operators, and filters by reconstruction. IEEE Transactions on Image Processing, Vol.4, I.8, (1995) 1153-1160.
15. Soille, P. Morphological Image Analysis. Principles and Applications. Springer-Verlag, Berlin (1999).

A Shape-Based Approach to Robust Image Segmentation

Samuel Dambreville, Yogesh Rathi, and Allen Tannenbaum

Georgia Institute of Technology, 30322 Atlanta Georgia USA

Abstract. We propose a novel segmentation approach for introducing shape priors in the geometric active contour framework. Following the work of Leventon, we propose to revisit the use of linear principal component analysis (PCA) to introduce prior knowledge about shapes in a more robust manner. Our contribution in this paper is twofold. First, we demonstrate that building a space of familiar shapes by applying PCA on binary images (instead of signed distance functions) enables one to constrain the contour evolution in a way that is more faithful to the elements of a training set. Secondly, we present a novel region-based segmentation framework, able to separate regions of different intensities in an image. Shape knowledge and image information are encoded into two energy functionals entirely described in terms of shapes. This consistent description allows for the simultaneous encoding of multiple types of shapes and leads to promising segmentation results. In particular, our shape-driven segmentation technique offers a convincing level of robustness with respect to noise, clutter, partial occlusions, and blurring.

1 Introduction

Segmentation consists of extracting an object from an image, a ubiquitous task in computer vision applications. Such applications range from finding special features in medical images to tracking deformable objects; see [1,2,3,4,5] and the references therein. The active contour framework [6], which utilizes image information to evolve a segmenting curve, has proven to be quite valuable for performing this task. However, the use of image information alone often leads to poor segmentation results in the presence of noise, clutter or occlusion. The introduction of shapes priors in the contour evolution process has proven to be an effective way to circumvent this issue, leading to more robust segmentation performances.

Many different algorithms have been proposed for incorporating shape priors in the active contour framework. For example, various approaches for utilizing shape priors in parameterized representations of contours were proposed by Cootes *et al.* [7], Wang and Staib [8], and Cremers *et al.* [9]. Moreover, Cremers *et al.* [10], recently presented a statistical approach using kernel methods [11], for building shape models. Using this method for parameterized contours, the authors were able to construct shape priors involving various objects, and to obtain convincing segmentation results.

The geometric active contour framework (GAC) (see [12] and the references therein) involves a parameter free representation of contours: contours are represented implicitly by level-set functions (such as signed distance function [13]). Within this framework, Leventon *et al.* [1], proposed an algorithm in which principal component analysis (PCA) was performed on a training set of signed distance functions (SDFs) and the shape statistics thus obtained were used to drive the segmentation process. This statistical approach was shown to be able to convincingly capture the variability in shape of a particular object. This approach inspired other segmentation schemes described in [3,14], notably, where SDFs were used to learn the shape of an object.

In this paper, we propose to revisit the use of linear PCA to introduce prior knowledge about shapes into the geometric active contour framework. To this end, we present a novel variational approach totally described in terms of shapes. Experimental results are presented to illustrate the robustness of our method: We first demonstrate the ability of our algorithm to constrain the contour evolution in a way that is more faithful to the training sets of shapes than prior work involving linear PCA. Then, we show the possibility, within our framework, to simultaneously encode knowledge about the shape of different objects, while capturing the variability in shape of each particular object.

2 Shape-Based Segmentation with Level-Sets

Level-set representations were introduced in [13] in the field of computational physics and became a popular tool for image segmentation. The idea consists of representing the segmenting contour by the zero level-set of a smooth and continuous function. The standard choice is to use a signed distance function for embedding the contour. During the segmentation process, the contour is propagated implicitly by evolving the embedding function. Implicit representations present the advantage of avoiding to deal with complex re-sampling schemes of control points. Moreover, the contour represented implicitly can naturally undergo topological changes such as splitting and merging.

In what follows, we consider the problem of segmenting a gray-level given image $I : \Omega \rightarrow \mathbf{R}$, where Ω is a subdomain of the plane. The term *segmentation* in this context will be taken to refer to the process of extracting an object of interest from the background and potential clutter in I . Let Φ denote the corresponding signed distance function to be used for segmentation [12]. We denote the curve corresponding to the zero level-set of Φ as γ . The curve γ separates the image domain Ω into two disjoint regions, γ_1 and γ_2 . We assume that for all $(x, y) \in \gamma_1$, we have that $\Phi(x, y) \geq 0$. We denote by $H\Phi$ the Heaviside function defined such as, $H\Phi(x, y) = 1$ if $\Phi(x, y) \geq 0$ and $H\Phi(x, y) = 0$, otherwise. $H\Phi$ is a binary map and will be interpreted as the *shape* associated to the signed distance function Φ , in what follows.

Segmentation using shape priors can be carried using an energy of the form (i.e.: [9])

$$E(\Phi, I) = \beta_1 E_{shape}(\Phi) + \beta_2 E_{image}(\Phi, I). \quad (1)$$

In this expression, E_{shape} is an energy functional embedding shape information, and E_{image} is an energy functional encoding image information available at time t . The minimization of $E(\Phi, I)$ with respect to Φ can be accomplished via the standard gradient descent approach:

$$\frac{d\Phi}{dt} = -\nabla_{\Phi} E(\Phi, I) \quad \text{i.e.,} \quad \Phi(t + dt) = \Phi(t) - dt \cdot \nabla_{\Phi} E(\Phi, I) \quad (2)$$

In the rest of this section, we present a method for the introduction of shape priors into level-set framework for purpose of segmentation. We first discuss our approach for building a space of shapes and compare our method with prior work. We then introduce an energy function encoding our knowledge of the shape of the object of interest. Finally, we propose an energy functional involving a shape model that aims at exploiting intensity information in the image.

2.1 Space of Shapes

In what follows, we will assume that we have a training set τ of N binary images $\{\mathbf{I}_1, \mathbf{I}_2, \dots, \mathbf{I}_N\}$ (the \mathbf{I}_i 's are described by $m \times n$ matrices). The \mathbf{I}_i 's represent the possible shapes of objects of interest. Each \mathbf{I}_i has values of 1 inside the object and 0 outside. Such training sets are presented Figure 1. The shapes in τ are supposed to be aligned using an appropriate registration scheme (see, e.g., [3]) in order to discard differences between them due to similarity transformations.

In this paper, we propose to perform PCA directly on the binary images of the training set τ , instead of applying PCA on signed distance functions as advocated in [3,1]. The method is succinctly presented in what follows. First, the mean shape μ is computed by taking the mean of the training shapes, \mathbf{I}_i 's, $\mu = \frac{1}{N} \sum_{i=1}^N \mathbf{I}_i$. Then, the covariance matrix \mathbf{C} , representing the variations in shapes, can be computed as follows. The mean shape μ is subtracted from each \mathbf{I}_i to create a mean-offset map $\tilde{\mathbf{I}}_i$. Each such map, $\tilde{\mathbf{I}}_i$ is written as a column vector $\tilde{\mathbf{I}}_i^c$ (The n columns of $\tilde{\mathbf{I}}_i$ are stacked on top of one another to form a large mn -dimensional column vector). Each $\tilde{\mathbf{I}}_i^c$ is then placed as the i^{th} column of a matrix \mathbf{M} , resulting in a $(mn) \times N$ -dimensional matrix $\mathbf{M} = [\tilde{\mathbf{I}}_1^c, \tilde{\mathbf{I}}_2^c, \dots, \tilde{\mathbf{I}}_N^c]$. The covariance matrix is then $\mathbf{C} = \frac{1}{N} \mathbf{M} \mathbf{M}^T$. Finally, using singular value decomposition \mathbf{C} is decomposed as $\mathbf{C} = \mathbf{U} \mathbf{\Sigma} \mathbf{U}^T$ where \mathbf{U} is a matrix whose column vectors represent the set of orthogonal modes of shape variation (principal components) and $\mathbf{\Sigma}$ is a diagonal matrix of corresponding eigenvalues. Each column \mathbf{u}_i of \mathbf{U} can be rearranged as an $m \times n$ matrix by inverting the stacking process involved in the construction of the $\tilde{\mathbf{I}}_i^c$'s. The rearranged column \mathbf{u}_i , forms the eigen-shape modes \mathbf{S}_i (corresponding to the i^{th} eigenvalue of $\mathbf{\Sigma}$).

Let \mathbf{S} be any binary map, representing an arbitrary shape. The coordinates α^k of the projection of \mathbf{S} onto the first k components of the space of shapes can be computed as

$$\alpha^k = \mathbf{U}_k^T (\mathbf{S}^c - \mu^c) \quad (3)$$

where \mathbf{U}_k is a matrix consisting of the first k columns of \mathbf{U} , \mathbf{S}^c and μ^c are the column vectors obtained by stacking the columns of \mathbf{S} and μ , respectively. Given the coordinates $\alpha^k = [\alpha_1^k, \alpha_2^k, \dots, \alpha_k^k]^T$, the projection of \mathbf{S} , denoted by $P^k(\mathbf{S})$, can be obtained as $P^k(\mathbf{S}) = \sum_{i=1}^k \alpha_i^k \cdot \mathbf{S}_i + \mu$.

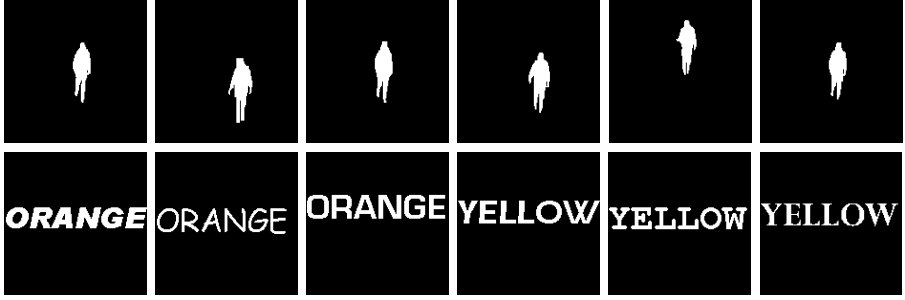


Fig. 1. Two Training sets (before alignment). First row, training set of objects of the same type: human silhouettes. Second row, training set of objects of different aspects: 2 words "ORANGE" and "YELLOW" (3 of the 20 binary maps for each).

2.2 Shape Priors

Shape Energy: To include prior knowledge on the shape, we propose to use $P^k(H\Phi)$ as a model and to minimize the following energy:

$$E_{shape}(\Phi) := \|H\Phi - P^k(H\Phi)\|^2 = \int_{\Omega} [H\Phi - P^k(H\Phi)]^2 dx dy. \quad (4)$$

Note that $E_{shape}(\Phi)$ is the squared L_2 -distance between $H\Phi$ and the projection of $H\Phi$ onto the shape space. Minimizing E_{shape} amounts to drive the shape $H\Phi$ towards the space of learnt shapes represented by binary maps.

Comparison with Past Work: We believe that the main advantage of using binary maps over SDFs resides in the fact that binary maps have limited support (0 everywhere but inside the object, where it is 1), whereas SDFs can take a wide range of values on the whole domain Ω . As a consequence, linear combinations of SDFs can lead to shapes that are very different from the learned shape. This phenomenon is illustrated Figure 2. Familiar spaces of shapes were constructed for the training sets presented in Figure 1, using either SDFs or binary maps. One of the shapes of the training set was slightly modified to form a new shape \mathbf{S} , see Figure 2(b). The projections of \mathbf{S} on both spaces (SDFs and binary maps) are presented in Figure 2(c) and (d). For each of the two cases presented, the shape obtained from the projection on the space derived from the SDFs (Figure 2(c)) bears little resemblance with the learnt shapes. In contrast, the projection obtained from the space constructed from binary maps, Figure 2(d), is clearly more faithful to the learned shapes. Hence, it can be expected that shape spaces based on binary maps afford more robustness to clutter than shape spaces built from SDFs. This point will be reinforced in the sequel.

2.3 Shape-Based Approach to Region-Based Segmentation

Different models [15,16], which incorporate geometric and/or photometric (color, texture, intensity) information, have been proposed to perform region-based

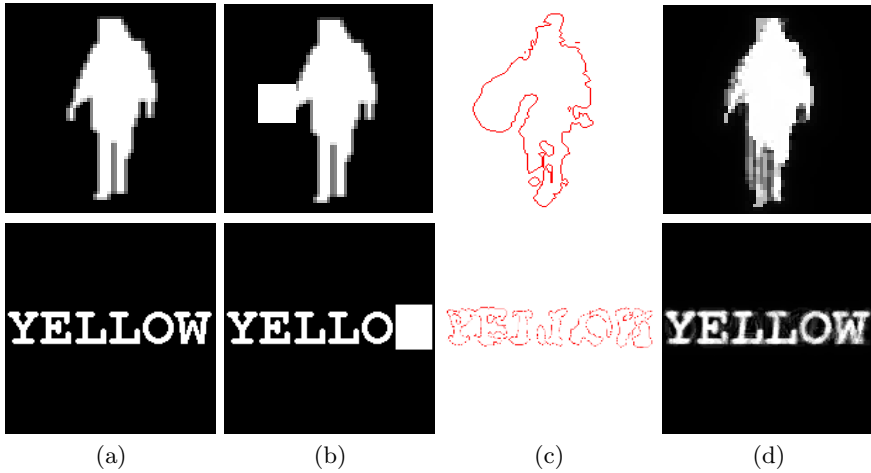


Fig. 2. Comparison of shape priors built from SDFs and binary maps. (a): Original training shape. (b): Slight modification of the training shape. (c): Projection on space of Shape built from SDFs. (d): Projection on space of Shape built from binary maps. Projections obtained using binary maps are more faithful to the learnt shapes.

segmentation using level-sets. In what follows, we present an region-based based segmentation framework, with a strong shape interpretation. As in Section (2.2), at each step t of the evolution process we build a model of shape to drive segmenting contour towards it. Here, the model of shape is extracted from the image.

Following [16], we assume that the image I is formed by two regions of different mean intensity. The first region can be interpreted as the background, whereas the second region can be interpreted as the object of interest. As presented in Section (2) the zero level-set of Φ separates Ω into two regions γ_1 and γ_2 . We compute the mean c_1 and c_2 of the intensity corresponding to pixels located in γ_1 and γ_2 , respectively: $c_1 := \frac{\int I(x,y)H(\Phi)dx dy}{\int H(\Phi) dx dy}$ and $c_2 := \frac{\int I(x,y)(1-H(\Phi)) dx dy}{\int (1-H(\Phi)) dx dy}$. In these expressions, c_1 can be interpreted as the best estimate of the mean intensity of the object of interest, while c_2 can be interpreted as the best estimate of the intensity of the background, at time t . We now build the image shape model $G_{[I,\Phi]}$ by thresholding I in the following manner:

$$\begin{aligned}
 \text{if } c_1 > c_2; \quad G_{[I,\Phi]}(x, y) &= 1 & \text{if } I(x, y) \geq \frac{c_1+c_2}{2} \\
 &= 0 & \text{otherwise;} \\
 \text{if } c_1 \leq c_2; \quad G_{[I,\Phi]}(x, y) &= 0 & \text{if } I(x, y) \geq \frac{c_1+c_2}{2} \\
 &= 1 & \text{otherwise.}
 \end{aligned} \tag{5}$$

This thresholding insures that $G_{[I,\Phi]}$ is a binary map representing a certain shape. The manner in which the cases are defined makes the pixels in the image whose intensity is closer to c_1 to be set to 1 in the model, while the pixels closer to c_2 are set to 0. $G_{[I,\Phi]}$ can be interpreted as the most likely shape present in the image, knowing Φ .

We next minimize the following energy in order to drive the contour evolution towards the model of shape obtained from thresholding the image I :

$$E_{image}(\Phi, I) := \|H\phi - G_{[I, \Phi]}\|^2 = \int_{\Omega} (H\phi - G_{[I, \Phi]})^2 \, dx dy. \tag{6}$$

The energy functional amounts again to measuring the L^2 distance between two shapes, represented by $H\Phi$ and $G_{[I, \Phi]}$. Results of this image segmentation approach are presented Figure 3(b) and Figure 5(b) (where contour was evolved minimizing E_{image} only). The consistent description of E_{image} and E_{shape} in terms of binary maps allows for efficient and intuitive equilibration between image cues and shape knowledge.

2.4 Numerical Algorithm

Approximation of Functions: In our implementation of the above framework, we used the following $C^\infty(\bar{\Omega})$ regularizations:

$$H_{\varepsilon_1}\Phi(x, y) := \left(\frac{1}{2} + \frac{1}{\pi} \arctan \frac{\Phi(x, y)}{\varepsilon_1} \right) \quad \text{and} \quad \delta_{\varepsilon_1}\Phi(x, y) := \frac{1}{\pi} \left(\frac{\varepsilon_1}{\Phi^2(x, y) + \varepsilon_1^2} \right) \tag{7}$$

where ε_1 , is a parameter such that $H_{\varepsilon_1} \rightarrow H$ and $\delta_{\varepsilon_1} \rightarrow \delta$ as $\varepsilon_1 \rightarrow 0$ (with $\delta = H'$). The function $G_{[I, \Phi]}$ in (5) is regularized as follows:

$$\begin{aligned} \text{if } c_1 > c_2; \quad & G_{[I, \Phi, \varepsilon_2]}(x, y) = \frac{1}{2} + \frac{1}{\pi} \arctan \left(\frac{I(x, y) - \frac{c_1 + c_2}{2}}{\varepsilon_2} \right) \\ \text{if } c_1 \leq c_2; \quad & G_{[I, \Phi, \varepsilon_2]}(x, y) = \frac{1}{2} - \frac{1}{\pi} \arctan \left(\frac{I(x, y) - \frac{c_1 + c_2}{2}}{\varepsilon_2} \right), \end{aligned} \tag{8}$$

where ε_2 , is a parameter such that $G_{[I, \Phi, \varepsilon_2]} \rightarrow G_{[I, \Phi]}$ as $\varepsilon_2 \rightarrow 0$.

Invariance to Similarity Transformations: Let $\mathbf{p} = [a, b, \theta, \rho] = [p_1, p_2, p_3, p_4]$ be the vector of parameters corresponding to an affine transformation: a and b correspond to translation according to x and y -axis, θ is the rotation angle and ρ the scale parameter. Let us denote by $\hat{I}(\hat{x}, \hat{y})$ the image of I by the affine transformation of parameter \mathbf{p} : $\hat{I}(\hat{x}, \hat{y}) = I(\rho(x \cos \theta - y \sin \theta + a), \rho(x \sin \theta + y \cos \theta + b))$. As mentioned above the elements of the training sets are aligned prior to the construction of the space of shapes. Let us suppose that the object of interest in I differs from the registered elements of the training set by an affine transformation. This transformation can be recovered by minimizing $E(\Phi, \hat{I})$ with respect to the p_i 's. During evolution, the following gradient descent scheme can be performed:

$$\frac{dp_i}{dt} = -\nabla_{p_i} E(\Phi, \hat{I}) = -\nabla_{p_i} E_{image}(\Phi, \hat{I}) \quad \text{for } i \in [1, 4].$$

Level-set Evolution: Computing the gradients corresponding to E_{shape} (equation (4)) and E_{image} (equation (6)) and accounting for possible affine transformations of the object of interest in I , equation (2) can be written as

$$\frac{d\Phi}{dt} = 2\delta_{\varepsilon_1}\Phi[\beta_1 P^k(H_{\varepsilon_1}\Phi) + \beta_2 G_{[j,\Phi,\varepsilon_2]}] - (\beta_1 + \beta_2)H_{\varepsilon_1}\Phi. \quad (9)$$

3 Segmentation Results

In this section we present experimental results aimed at testing the segmentation performances of our framework on challenging images.

3.1 Shape Prior Involving Objects of the Same Type: “Swedish Couple”

The Swedish Couple sequence was used as a base of test for many tracking algorithms using active contours [6]. One of the difficulties of performing tracking in this video resides in maintaining the identity of each person: Throughout the sequence, the two people often touch each other and the segmenting contour can leak from one person to the other, leading to a loss of track.

A training set of 7 silhouettes was obtained by manually selecting the contour for the person on the right on 7 different images from the sequence (the corresponding binary maps or SDFs were then completed).

Figure 3(c) presents a segmentation result obtained using shape priors built from applying PCA to SDFs (e.g., [3]). In the image presented, the two persons touch each other. The shape prior is not sufficiently constraining to prevent the contour from leaking from the person on the right (man) to the person on the left (woman). Hence, if no further control is applied to bound the coordinates of the contour in (3), leakage can occur leading to a shape that is very different from the ones in the training set. This phenomenon does not occur when using shape priors built from applying PCA on binary maps as presented in Figure 3(d). Here, the shape prior prevents leakage leading to a satisfying segmentation.

In fact, using shape priors built from the 7 binary maps obtained from the man, we were able to track the entire sequence for both persons (while maintaining their identity). Figure 4 presents the results obtained for a few images from the video. Despite the small number of shapes used, the general posture of each person was convincingly captured in each image. In addition, the final contours are faithful to the training set: No leakage occurred from one person to the other and the bag held by the woman is discarded. Tracking was performed using a very simple scheme. The same initial contour (small square) was used for each image and initially positioned wherever the final contour was in the preceding image. The parameters were set in the following manner: $\beta_1 = \beta_2$ in (1) and $\varepsilon_1 = \varepsilon_2 = .1$ in (7) and (8), respectively. Convincing results were obtained without involved considerations about the system dynamics.

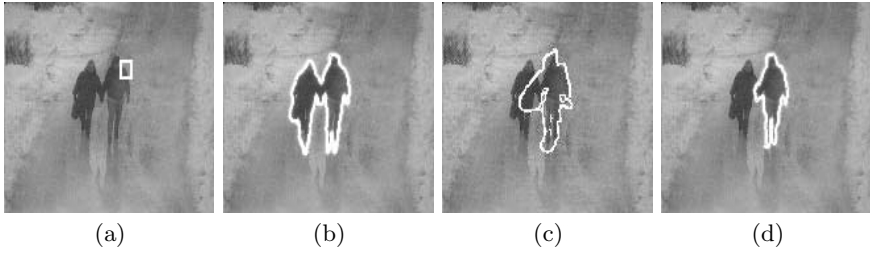


Fig. 3. Comparison of results. (a): base image and initial contour. (b): Segmentation result without shape prior. (c): Result with shape prior obtained from applying PCA on SDFs. (d): Result with shape prior obtained from applying PCA on Binary Maps.



Fig. 4. Tracking results for the Swedish Couple Sequence, using a training set of 7 silhouettes of the man. First row: Tracking the man. Second row: Tracking the woman.

3.2 Shape Priors Involving Objects of Different Types: “Yellow” and “Orange”

The goal of this section is to investigate the ability of our method to deal with objects of different shapes. To this end, we built a training set consisting of two words, “orange” and “yellow”, each written using twenty different fonts. The size of the fonts was chosen to lead to words of roughly the same length. The obtained words (binary maps, see Figure 1) were then registered according to their centroid. No further effort such as matching the letters of the different words was pursued. The method presented in Section (2.1) was used to build the corresponding space of shapes for the registered binary maps.

We tested our framework on images where a corrupted version of either the word “orange” or “yellow” was present (Figure 5(a)). Word recognition is a very challenging task and using geometric active contours to address it may not be a panacea. However, the ability of the level-set representation to naturally handle topological changes was found to be useful for this purpose: In the experiments

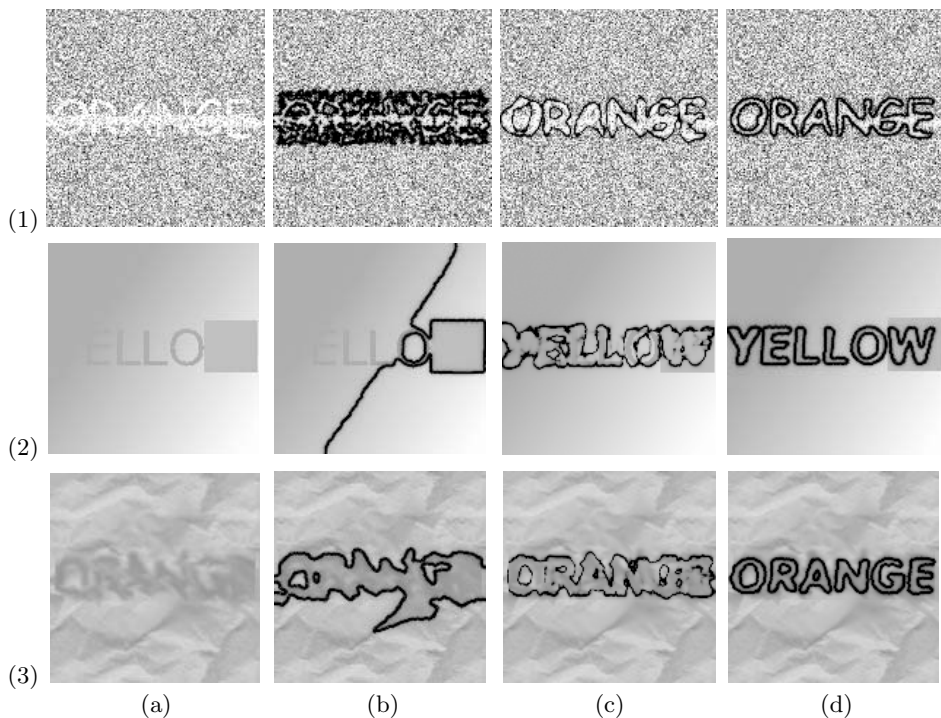


Fig. 5. Each row presents the segmentation results obtained for experiment 1, 2 and 3 respectively. (a): Initial image. (b): Segmentation result using E_{image} only (no shape prior). (c): Typical segmentation result with shape prior built from SDFs. (d): Segmentation result with shape prior built from binary map.

presented below, evolution led the contour to split and merge a certain number of times to segment the disconnected letters of the different words.

In all the following experiments, we have chosen $\beta_1 = \beta_2$ in (1) and $\varepsilon_1 = \varepsilon_2 = .1$ in (7) and (8), respectively. The same initial contour was used for all the test images. Starting from the same initial contour our framework was able to accurately detect which word was present in the image. This highlights the ability of our method to gather image information throughout evolution and to distinguish between objects of different classes (“yellow” and “orange”).

Experiment 1: In this experiment, one of the elements of the training set was used (Figure 1(2b)). A thick line (occlusion) was drawn on the word and a fair amount of gaussian noise was added to the resulting image (Figure 5(1a)). The result of applying our method is presented Figure 5(1d). Despite the noise and the occlusion, a reasonable segmentation is obtained. In particular, the correct font is detected and the thick line almost completely removed. In addition, the final result is smooth as compared to the result obtained without shape prior; see Figure 5(1b). Hence, using binary maps to represent shape priors can have valuable smoothing effects, when dealing with noisy images.

Experiment 2: In this second test, the word “yellow” was written using a *different* font from the ones used to build the training set (visual check was performed to ensure that the length of the word was comparable to the length of the words in the training set). In addition, a “linear shadowing” was used in the background, making the first letter “y” completely hidden. The letter “w” was also replaced by a grey square (Figure 5(2a)). The result of applying our framework is presented in Figure 5(2d). The word “yellow” is correctly segmented. In particular, the letters “y” and “w”, were completely reconstructed. Comparing to the results obtained in Figure 5(2b) and (2c) obtained without prior knowledge of the shape or with shape prior built from SDFs, one can notice the effect of our shape prior model in constraining the contour evolution.

Experiment 3: In this experiment, the word “orange” was *handwritten* in capital letters roughly matching the size of letters of the components of the training set. The intensity of the letters was chosen to be rather close to some parts of the background. In addition, the word was blurred and smeared in a way that made its letters barely recognizable (Figure 5 (3a)). This type of blurring effect is often observed in medical images due to patient motion. This image is particularly difficult to segment, since the spacing between letters and the letters themselves are very irregular due to the combined effects of handwriting and blurring. Hence, mixing between classes (confusion between either “yellow” or “orange”) can be expected in the final result. The final result of the segmentation process is presented Figure 5(3d). The word “orange” is not only recognized (no mixing) but satisfyingly recovered; in particular, a thick font was obtained to model the thick letters of the word.

4 Conclusion

Our contributions in this paper are twofold. First, we demonstrate that building a space of shapes by applying PCA to binary images (instead of signed distance functions) enables one to constrain the contour evolution in a way that is more faithful to the training set. Secondly, we present a novel region-based segmentation framework, able to separate regions of different intensities in an image. Shape knowledge and image information were encoded into two energies entirely described in terms of shapes. This consistent description allows for intuitive and simple equilibration between both image cues and shape prior.

The method presented allows for the simultaneous encoding of multiple types of shapes and seems to lead to robust segmentation results. In particular, our shape driven segmentation technique was able to cope with noise, clutter, partial occlusion, change in aspect and blurring, in a convincing manner.

In our future work, we aim at comparing segmentation performances obtained with shape priors built from linear and kernel PCA methods. Kernel PCA was proven to be a powerful method to describe data sets. The consistent shape approach, characteristic of our framework, is envisioned to be particularly suitable to deal with the “exotic” norms involved in kernel PCA methods.

References

1. Leventon, M., Grimson, E., Faugeras, O.: Statistical shape influence in geodesic active contours. In: Proc. CVPR, IEEE (2000) 1316–1324
2. Paragios, N., Deriche, R.: Geodesic active contours and level sets for the detection and tracking of moving objects. In: Transactions on Pattern analysis and Machine Intelligence. Volume 22. (2000) 266–280
3. Tsai, A., Yezzi, T., et al., W.W.: A shape-based approach to the segmentation of medical imagery using level sets. *IEEE Trans. on Medical Imaging* **22** (2003) 137–153
4. Yezzi, A., Kichenassamy, S., et al., A.K.: A geometric snake model for segmentation of medical imagery. In: *IEEE Trans. Medical Imag.* Volume 16. (1997) 199–209
5. A. Yezzi, S.S.: Deformation: Deforming motion, shape average and the joint registration and approximation of structures in images. In: *International Journal of Computer Vision*. Volume 53. (2003) 153–167
6. Blake, A., Isard, M., eds.: *Active Contours*. Springer (1998)
7. Cootes, T., Taylor, C., et al., D.C.: Active shape models-their training and application. In: *Comput.Vis. Image Understanding*. Volume 61. (1995) 38–59
8. Wang, Y., Staib, L.: Boundary finding with correspondance using statistical shape models. In: *IEEE Conf. Computer Vision and Pattern Recognition*. (1998) 338–345
9. Cremers, D., Kohlberger, T., Schnoerr, C.: Diffusion snakes: introducing statistical shape knowledge into the mumford-shah functional. In: *International journal of computer vision*. (Volume 50.)
10. Cremers, D., Kohlberger, T., Schnoerr, C.: Shape statistics in kernel space for variational image segmentation. In: *Pattern Recognition*. Volume 36. (2003) 1292–1943
11. Mika, S., Scholkopf, B., et al., A.S.: Kernel pca and de-noising in feature spaces. In: *Advances in neural information processing systems*. Volume 11. (1998)
12. Sapiro, G.: *Geometric Partial Differential Equations and Image Analysis*. Cambridge University Press (2001)
13. Osher, S., Sethian, J.: Fronts propagation with curvature dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics* **79** (1988) 12–49
14. Rousson, M., Paragios, N.: Shape priors for level set representations. In: *Proceedings of European Conference on Computer Vision*. (2002) 78–92
15. Yezzi, A., Tsai, A., Willsky, A.: A statistical approach to snakes for bimodal and trimodal imagery. In: *Proc. Int. Conf. Computer Vision*. Volume 2. (1999) 898–903
16. Chan, T., Vese, L.: Active contours without edges. In: *IEEE Trans. Image Processing*. Volume 10. (2001) 266–277

Novel Statistical Approaches to the Quantitative Combination of Multiple Edge Detectors

Stamatia Giannarou and Tania Stathaki

Communications and Signal Processing Group
Imperial College London
Exhibition Road, SW7 2AZ, London, UK
`stamatia.giannarou@imperial.ac.uk`

Abstract. This paper aims at describing a new framework which allows for the quantitative combination of different edge detectors based on the correspondence between the outcomes of a preselected set of operators. This is inspired from the problem that despite the enormous amount of literature on edge detection techniques, there is no single one that performs well in every possible image context. The so called Kappa Statistics are employed in a novel fashion to enable a sound performance evaluation of the edge maps emerged from different parameter specifications. The proposed method is unique in the sense that the balance between the false detections (False Positives and False Negatives) is explicitly assessed in advanced and incorporated in the estimation of the optimum threshold. Results of this technique are demonstrated and compared to individual edge detection methods.

1 Introduction

Edge detection is by far the most common and direct approach for detecting discontinuities that could highlight the boundaries of an object captured in a digital image. Despite our familiarity with the concept of an edge in an image, there is no rigorous definition for it. Edges can be viewed as regions in an image where the intensity values undergo a sharp variation. Furthermore, changes in some physical and surface properties, such as illumination (e.g. shadows), geometry (orientation or depth) and reflectance can also signal the existence of an edge in the scene.

The ubiquitous interest in edge detection stems from the fact that not only is it in the forefront of image processing for object detection, but it is a fundamental operation in computer vision too. Among the various applications that can benefit from an edge image representation are object recognition, motion analysis and image compression. Edge detection must be efficient and reliable since it is crucial in determining how successful subsequent processing stages will be.

Although the human visual system carries out the edge detection process effortlessly, it remains to be a complex task for a computer algorithm to achieve the equivalent edge detection functions due to problems caused by noise, quantization and blurring effects. In order to fulfill the reliability requirement of edge

detection, a great diversity of operators have been devised with differences in their mathematical and algorithmic properties. Some of the earliest methods such as Sobel [14], are based on the "Enhancement and Thresholding" approach [1]. According to that method, the image is convolved with small kernels and the result is thresholded to identify the edge points.

Since then, more sophisticated operators have been developed. Marr and Hildreth [13] were the first to introduce the Gaussian smoothing as a pre-processing step in feature extraction. Their method detects edges by locating the zero-crossings of the Laplacian (second derivative) of Gaussian of an image. Canny [3] developed another Gaussian edge detector based on optimizing three criteria. He employed Gaussian smoothing to reduce noise and the first derivative of the Gaussian to detect edges. Deriche [4] extended Canny's work to derive a recursively implemented edge detector. Rothwell [17] designed an operator which is able to recover reliable topological information.

A different approach to edge detection is the multiresolution one. In such a representation framework, the image is convolved with Gaussian filters of different sizes to produce a set of images at different resolutions. These images are integrated to produce a complete final edge map. Typical algorithms of this approach have been produced by Bergholm [2], Lacroix [12] and Schunck [18]. Parametric fitting is another approach used in edge detection. This involves fitting the image with a parametric edge model and then finding the parameters that minimize the fitting error. A detector that belong to the above category is proposed by Nalwa-Binford [15]. Furthermore, the idea of replicating the human vision performance using mathematical models gave way to the development of feature detection algorithms based on the human visual system. A typical example is the edge detector developed by Peli [16]. Another interesting category of edge detectors is the Logical/Linear operators [8] which combine aspects of linear operators' theory and Boolean algebra.

In spite of the aforementioned work, an ideal scheme able to detect and localize with precision edges in many different contexts, has not yet been produced. This is getting even more difficult because of the absence of an evident ground truth, on which the performance of an edge detector is evaluated. Subjective evaluation achieved by visual judgment, is inaccurate and it can not be used to measure the performance of detectors but only to confirm their failure. Thus, objective evaluation has been introduced based on quantitative measures [5].

The aim of this work is to describe a method for combining multiple edge detectors. Our major goal is to produce a composite final edge image. This is achieved by applying a correspondence test on the edge maps emerged by a preselected set of detectors and seeking the optimum correspondence threshold. The approach proposed for this purpose is the Kappa Statistics. In particular, the Weighted Kappa Coefficient is employed assigning different costs to the false positive and false negative detection qualities.

The paper is organized as follows. In the next section we present an approach for automatic edge detection and discuss its scope. The results emerged by the selected edge detectors using the proposed statistical technique are compared

in Section 3. We discuss the performance of the method and conclude with Section 4.

2 Automatic Edge Detection

In this paper, we intend to throw light on the uncertainty associated with the parametric edge detection performance. The statistical approach described here attempts to automatically form an optimum edge map, by combining edge images emerged from different detectors.

We begin with the assumption that N different edge detectors will be combined. The first step of the algorithm comprises the correspondence test of the edge images, E_i for $i = 1 \dots N$. A correspondence value is assigned to each pixel and is then stored in a separate array, V , of the same size as the initial image. The correspondence value is indicative of the frequency of identifying a pixel as an edge by the set of detectors. Intuitively, the higher the correspondence associated with a pixel, the greater the possibility for that pixel to be a true edge. Hence, the above correspondence can be used as a reliable measure to distinguish between true and false edges.

However, these data require specific statistical methods to assess accuracy of the resulted edge images- accuracy here being the extent to which detected edges agree with 'true' edges. Correspondence values ranging from 0 to N produce $N+1$ thresholds which correspond to edge detections with different combinations of true positive and false positive rates. The threshold that corresponds to correspondence value 0 is ignored. So, the main goal of the method is to estimate the correspondence threshold CT (from the set CT_i where $i = 1 \dots N$) which results in an accurate edge map that gives the finest fit to all edge images E_i .

2.1 Weighted Kappa Coefficient

In edge detection, it is prudent to consider the relative seriousness of each possible disagreement between true and detected edges when performing accuracy evaluation. This section is confined to the examination of an accuracy measure which is based on the acknowledgement that in detecting edges, the consequences of a false positive may be quite different from the consequences of a false negative. For this purpose, the *Weighted Kappa Coefficient* [6], [11] is introduced for the estimation of the correspondence threshold that results in an optimum final edge map.

In our case, the classification task is a binary one including the actual classes $\{e, ne\}$, which stand for the *edge* and *non-edge* event, respectively and the predictive classes, *predicted edge* and *predicted non-edge*, denoted by $\{E, NE\}$. Traditionally, the data obtained by an edge detector are displayed graphically in a 2×2 matrix, the *confusion matrix*, with the notation indicated in Table 1.

For the accuracy assessment the metrics of sensitivity and specificity [9] have a pivotal role. Both these measures describe the edge detector's ability to correctly identify true edges while it negates the false alarms. Sensitivity (SE) corresponds to the probability of identifying a true edge as edge pixel. The term specificity

Table 1. Confusion Matrix

| | | |
|----|-----------------|-----------------|
| | e | ne |
| E | True Positives | False Positives |
| NE | False Negatives | True Negatives |

(*SP*) refers to the probability of identifying an actual non-edge as non-edge pixel. They are defined as:

$$SE = TP / (TP + FN) \tag{1}$$

$$SP = TN / (TN + FP) \tag{2}$$

where the sum $TP + FN$ indicates the *prevalence*, P , which refers to the occurrence of true edge pixels in the image, whereas the sum $TP + FP$ represents the *level*, Q , of the detection which corresponds to the occurrence of pixels detected as edges.

Although there is a great variety of accuracy measures in edge detection [20], few of them take into consideration the agreement between true and detected edges expected merely by chance i.e when there is no correlation between the probability of a pixel to be a true edge and the probability of a pixel to be detected as edge. Therefore, in order to evaluate edge detection accuracy properly, the *kappa coefficient* [19] is introduced. It is defined as the ratio of achieved beyond-chance agreement to maximum possible beyond-chance agreement and is mathematically defined as:

$$k = \frac{A_0 - A_c}{A_a - A_c} \tag{3}$$

where A_0 and A_c are the observed and chance-expected proportions of agreement, respectively, while A_a represents the proportion of complete agreement.

A generalization of the above coefficient can be made to incorporate the relative cost of false positives and false negatives into our accuracy measure. We assume that weights $w_{u,v}$, for $u = 1, 2$ and $v = 1, 2$, are assigned to the four possible outcomes of the edge detection process displayed in the confusion matrix. The observed weighted proportion of agreement is given as:

$$D_{0_w} = \sum_{u=1}^2 \sum_{v=1}^2 w_{u,v} d_{u,v} \tag{4}$$

where $d_{u,v}$ indicates the probabilities in the confusion matrix. Similarly, the chance-expected weighted proportion of agreement has the form:

$$D_{c_w} = \sum_{u=1}^2 \sum_{v=1}^2 w_{u,v} c_{u,v} \tag{5}$$

where $c_{u,v}$ refers to the above four probabilities but in the case of random edge detection i.e, the edges are identified purely by chance. Both these proportions

Table 2.

| | Legitimate Edge Detection | Random Edge Detection |
|----|---------------------------|-------------------------|
| TP | $d_{1,1} = P \cdot SE$ | $c_{1,1} = P \cdot Q$ |
| FP | $d_{1,2} = P' \cdot SP'$ | $c_{1,2} = P' \cdot Q$ |
| FN | $d_{2,1} = P \cdot SE'$ | $c_{2,1} = P \cdot Q'$ |
| TN | $d_{2,2} = P' \cdot SP$ | $c_{2,2} = P' \cdot Q'$ |

are arrayed and calculated as shown in Table 2. Based on the definition of kappa coefficient described previously, weighted kappa is then given by:

$$k_w = \frac{D_{0_w} - D_{c_w}}{\max(D_{0_w} - D_{c_w})} \tag{6}$$

Substituting (4)-(5) in (6) gives:

$$k_w = \frac{w_{1,1}P \cdot Q' \cdot k(1,0) + w_{1,2}P'(SP' - Q) + w_{2,1}P(SE' - Q') + w_{2,2}P' \cdot Q \cdot k(0,0)}{\max(D_{0_w} - D_{c_w})} \tag{7}$$

where P' , Q' are the complements of P and Q , respectively. $k(1,0)$ and $k(0,0)$ are the quality indices of sensitivity and specificity, respectively, defined as:

$$k(1,0) = \frac{SE - Q}{Q'} \quad \text{and} \quad k(0,0) = \frac{SP - Q'}{Q}$$

The major source of confusion in statistical methods related to the Weighted Kappa Coefficient is the assignment of weights. In the method analyzed here the weights indicate gain or cost. The weights lie in the interval $0 \leq |w_{u,v}| \leq 1$. From (7) it can be deduced that the total cost, W_1 , for true edges being properly identified as edges or not, is equal to:

$$W_1 = |w_{1,1}| + |w_{2,1}|$$

Similarly, the total cost, W_2 , for the non-edge pixel is defined as:

$$W_2 = |w_{1,2}| + |w_{2,2}|$$

We propose that true detections should be assigned positive weights representing gain whereas, the weights for false detections should be negative, representing loss. It can be proved that no matter how the split of these total costs is made between true and false outcomes, the result of the method is not affected [10]. Hence, for the sake of convenience the total costs are split evenly. As a result, we end up with two different weights instead of four:

$$k_w = \frac{\frac{W_1}{2}P \cdot Q' \cdot k(1,0) + (-\frac{W_2}{2})P'(SP' - Q) + (-\frac{W_1}{2})P(SE' - Q') + \frac{W_2}{2}P' \cdot Q \cdot k(0,0)}{\max(D_{0_w} - D_{c_w})}$$

A further simplification leads to:

$$k_w = \frac{W_1 \cdot P \cdot Q' \cdot k(1,0) + W_2 \cdot P' \cdot Q \cdot k(0,0)}{\max(D_{0_w} - D_{c_w})} \tag{8}$$

Taking into account the fact that the maximum value of the quality indices $k(1, 0)$ and $k(0, 0)$ is equal to 1, the denominator in (8) takes the form: $W_1 \cdot P \cdot Q' + W_2 \cdot P' \cdot Q$. Dividing both numerator and denominator by $W_1 + W_2$, the final expression of the Weighted Kappa Coefficient, in accordance with the quality indices of sensitivity and specificity, becomes:

$$k(r, 0) = \frac{r \cdot P \cdot Q' \cdot k(1, 0) + r' \cdot P' \cdot Q \cdot k(0, 0)}{r \cdot P \cdot Q' + r' \cdot P' \cdot Q} \tag{9}$$

where

$$r = \frac{W_1}{W_1 + W_2} \tag{10}$$

and r' is the complement of r . The Weighted Kappa Coefficient $k(r, 0)$ indicates the quality of the detection as a function of r . It is unique in the sense that the balance between the two types of errors (FP and FN) is determined in advance and then is incorporated in the measure.

The index r is indicative of the relative importance of false negatives to false positives. Its value is dictated by which error carries the greatest importance and ranges from 0 to 1. If we focus on the elimination of false positives in edge detection, W_2 will predominate in (8) and consequently r will be close to 0 as it can be seen from (10). On the other hand, a choice of r close to 1 signifies our interest in avoiding false negatives since W_1 will predominate in (8). A value of $r = 1/2$ reflects the idea that both false positives and false negatives are equally unwanted.

However, no standard choice of r can be regarded as optimum. The balance between the two errors shifts according to the application. Thus, the estimation of the optimum edge map emerged from different edge detectors is performed by choosing the correspondence threshold that maximizes the weighted kappa coefficient for a selected value of r .

In order to calculate the accuracy for each threshold value, we apply each correspondence threshold CT_i on the correspondence test outcome, i.e, the matrix V mentioned above. This means the pixels are classified as edges and non-edges according to whether their correspondence value exceeds a CT_i or not. Thus, we end up with a set of possible best edge maps M_j , for $j = 1 \dots N$, corresponding to each CT_i . Every M_j is compared to the set of the initial edge images, E_i , with the aim of estimating the accuracy associated with each of them.

Initially, the sensitivity, SE_j , and the specificity, SP_j , measures are calculated for each edge map M_j according to Equations (1)-(2) as:

$$SE_j = \frac{\overline{TP}_j}{\overline{TP}_j + \overline{FN}_j} \tag{11}$$

$$SP_j = \frac{\overline{TN}_j}{\overline{FP}_j + \overline{TN}_j} \tag{12}$$

where, $\overline{TP}_j + \overline{FN}_j$ is the *prevalence*, P , representing the average number of true edges in M_j . Similarly, the *level*, Q , is defined as: $\overline{TP}_j + \overline{FP}_j$. Averaging in

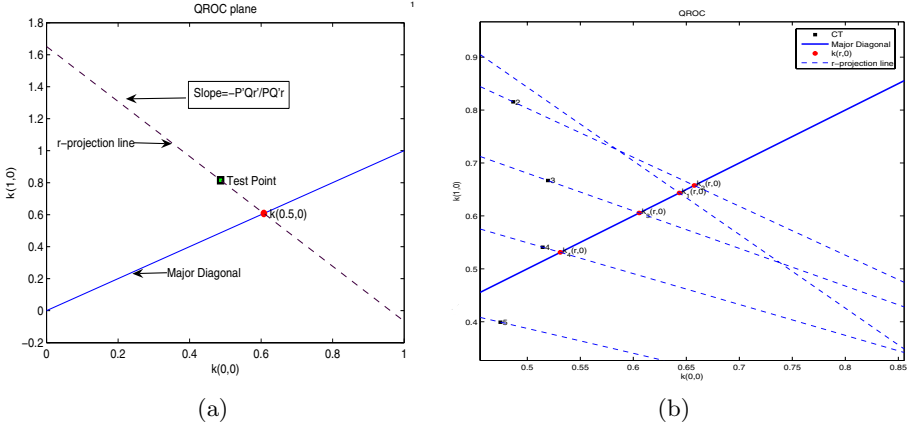


Fig. 1. (a) Calculation of $k(0.5, 0)$ using a graphical approach on the $QROC$ plane (b) Graphical estimation of the optimum CT for the Cameraman image

(11)-(12) refers to the joint use of multiple edge detectors. The corresponding expressions for the average probabilities are given by the authors of this paper in [7]. It is important to stress out that based on these expressions it is proved that the value of the *prevalence*, P , is the same for every j , as expected.

Finally, for a selected value of r , the Weighted Kappa Coefficient $k_j(r, 0)$ is calculated for each edge map as it is given in (9). The optimum CT is the one that maximizes the Weighted Kappa Coefficient. The emerged value of the CT determines how detailed the final edge image will be.

2.2 Geometric Approach

The estimation of the Weighted Kappa Coefficient $k_j(r, 0)$ can also be done geometrically. Every edge map M_j , for $j = 1 \dots N$, can be located on a graph by its quality indices $k_j(1, 0)$ versus $k_j(0, 0)$ constructing the Quality Receiver Operating Characteristic curve ($QROC$). A great deal of information is available from visual examination of such a geometric presentation.

The Weighted Kappa Coefficient $k_j(r, 0)$ that corresponds to each edge map M_j can be estimated by means of the r -projection line.

The slope of the r -projection line is defined as [10]:

$$s = \frac{k_j(r, 0) - k_j(1, 0)}{k_j(r, 0) - k_j(0, 0)} = -\frac{P' \cdot Q \cdot r'}{P \cdot Q' \cdot r} \tag{13}$$

This means the Weighted Kappa Coefficient $k_j(r, 0)$ can be calculated graphically by drawing a line, for any value r of interest, through the correspondence threshold point $(k_j(0, 0), k_j(1, 0))$ with slope given by (13). The intersection point, $(k_j(r, 0), k_j(r, 0))$, of this line with the major diagonal in the $QROC$ plane is clearly indicative of the $k_j(r, 0)$ value. Figure 1(a) presents an example for the

calculation of the Weighted Kappa Coefficient for a test point for $r = 0.5$. The procedure is repeated for every CT_i to generate N different intersection points. The closer the intersection point to the upper right corner (ideal point), the higher the value of the Weighted Kappa Coefficient. Hence, the optimum correspondence threshold is the one that produces an intersection point closer to the point $(1, 1)$ in the $QROC$ plane.

3 Experimental Results and Conclusion

Using the framework developed, six edge detectors, proposed by Canny, Deriche, Bergholm, Lacroix, Schunck and Rothwell were combined to produce the optimum edge maps of a given image. The selection of the edge detectors relies on the fact that they basically follow the same mathematical approach.

Specifying the value of the input parameters for the operators was a crucial step. In fact, the parameter evaluation depends on the specific application and intends to maximize the quality of the detection. In our work, we were consistent with the parameters proposed by the authors of the selected detectors. The Bergholm algorithm was slightly modified by adding hysteresis thresholding to allow a more detailed result. In Lacroix technique we applied non-maximal suppression by keeping the size, $k \times 1$, of the window fixed at 3×1 . For simplicity, in the case of Schunck edge detection the non-maximal suppression method we used is the one proposed by Canny in [3] and hysteresis thresholding was applied for a more efficient thresholding.

In the case of the "Cameraman" image, the standard deviation of the Gaussian (*sigma*) in Canny's algorithm was set to $sigma = 1$, whereas, the *low* and *high thresholds* were automatically calculated by the image histogram. In Deriche's technique, the parameters' values were set to $a = 2$ and $w = 1.5$. The Bergholm parameter set was a combination of *starting sigma*, *ending sigma* and *low* and *high threshold*, where $starting\ sigma = 3.5$, $ending\ sigma = 0.7$ and the thresholds were automatically determined as previously. For the *Primary Raster* in Lacroix's method, the coarsest resolution was set to $\sigma_2 = 2$ and the finest one to $\sigma_0 = 0.7$. The intermediate scale σ_1 was computed according to the expression proposed in [12]. The gradient and homogeneity thresholds were estimated by the histogram of the gradient and homogeneity images, respectively. For the Schunck edge detector, the number of resolution scales was arbitrary set to three as: $\sigma_1 = 0.7$, $\sigma_2 = 1.2$, $\sigma_3 = 1.7$. The difference between two consecutive scales was selected not to be greater than 0.5 in order to avoid edge pixel displacement in the resulted edge maps. The values for the *low* and *high thresholds* were calculated by the histogram of the gradient magnitude image. In the case of Rothwell method, the *alpha* parameter was set to 0.9, the *low threshold* was estimated by the image histogram again and the value of the smoothing parameter, *sigma*, was equal to 1. The results of the above detection methods are presented in Figure 2. It is important to stress out that the selected values for all of the above parameters fall within ranges proposed in the literature by the authors of the individual detectors.



Fig. 2. (a) Canny detection (b) Deriche detection (c) Bergholm detection (d) Lacroix detection (e) Schunck detection (f) Rothwell detection forming the sample set for the Cameraman image



Fig. 3. (a) Original "Cameraman" image and (b) Final edge map when applying the "Weighted Kappa Coefficient" approach with $r = 0.65$

The approach described in this paper for the estimation of the optimum correspondence threshold is based on the maximization of the Weighted Kappa Coefficient. The cost, r , is initially determined according to the particular quality of the detection (FP or FN) that is chosen to be optimized. For example as far as target object detection in military applications is concerned, missing existing targets in the image (misdetections) is less desirable rather than falsely detecting non-existing ones (false alarms). This is as well the scenario we assume in this piece of work, namely, we are primarily concerned with the elimination of FN at the expense of increasing the number of FP. Therefore, according to the analysis in the previous section, the cost value should range from 0.5 to 1. Moreover, a trade-off between the increase in information and the decrease in noise in the final edge image is necessary when selecting the value of r .

The experimental results demonstrated in this paper correspond to a value of r equal to 0.65. For this value of r , the calculation of the Weighted Kappa Coefficients yields $k_j(r, 0) = [0.64346, 0.65754, 0.60561, 0.53111, 0.42329, 0.27007]$. Observing these results, it is clear that the Weighted Kappa Coefficient takes its maximum value at $k_2(r, 0)$. Thus, the optimum CT for the "Cameraman" image is equal to 2. The selection of such an optimum correspondence threshold indicates that the essential information is kept at the edge features that occur at 33.34% of the initial sample set. The graphical estimation of $k_j(r, 0)$ for each CT is illustrated in Figure 1(b).

The final edge map in Figure 3(b) verify the method's success in combining high accuracy with good noise reduction. The detected edges are well defined regarding their shape and contour as it can be seen on the background buildings in the image. A considerable amount of insignificant information is cleared, while the one preserved allows for easy, quick and accurate object recognition. This is mainly noticeable in the area of the cameraman's face and on the ground, when

comparing the final result with several edge maps of the sample set in Figure 2. Furthermore, objects that some of the initial edge detectors failed to detect, such as the cameraman's hand and buildings on the background of the image, are included in the final result. A different selection of the cost r would result in a different trade-off between preserved information and reduced noise in the final edge map.

The computational cost of the proposed method is higher compared to that of applying each edge detector individually. However, this is acceptable since the goal is to form a more complete final edge image by combining these edge detectors in order to take advantage of their strengths while overcoming their weaknesses.

The above conclusions arise after a large number of experimental results involving different types of images.

4 Discussion

In this paper we proposed a technique for automatic statistical analysis of the correspondence of edge images emerged from different operators. The Weighted Kappa Coefficient was employed to calculate the CT which produces a composite final edge map. This approach was proved to be a good choice as a performance evaluation measure. As it was observed, applying the Weighted Kappa Coefficient for $r > 0.5$ led to quite noisy results but with good detection quality, regarding the number and the shape of detected objects. Hence, the selection of the cost r should be done according to the trade-off between the information we want to capture and the noise to suppress in a particular application. The major benefit of using the Weighted Kappa Coefficient is that it explicitly requires the relative importance of FP and FN to be specified and incorporated in the performance evaluation measure *a priori*. A possible application of the above scheme is to obtain clean edge images in which only the significant information is preserved.

Acknowledgements

The work reported in this paper was funded by the Systems Engineering for Autonomous Systems (SEAS) Defence Technology Centre established by the UK Ministry of Defence.

References

1. I. E. Abdou and W. K. Pratt. Quantitative design and evaluation enhancement/thresholding edge detectors. In *Proc. IEEE*, volume 67, pages 753–763, 1979.
2. F. Bergholm. Edge focusing. *IEEE Trans. Pattern Anal. Machine Intell.*, 9(6):726–741, Nov 1995.
3. J. F. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Machine Intell.*, 8(6):679–698, Nov 1986.

4. R. Deriche. Using canny's criteria to derive a recursive implemented optimal edge detector. *The International Journal of Computer Vision*, 1(2):167–187, May 1987.
5. J. R. Farm and E. W. Deutsch. On the quantitative evaluation of edge detection schemes and their comparison with human performance. *IEEE Trans. Comput.*, 24(6):616–628, June 1975.
6. J.L. Fleiss. *Statistical Methods For Rates and Proportions*. Wiley series in probability and mathematical statistics, 1981.
7. S. Giannarou and T. Stathaki. Edge detection using quantitative combination of multiple operators. SiPS conference, IEEE Workshop on Signal Processing Systems, November 2005.
8. L. A. Iverson and S. W. Zucker. Logical/linear operators for image curves. *IEEE Trans. Pattern Anal. Machine Intell.*, 17(10):982–996, Oct 1995.
9. B.R. Kirkwood and Jonathan A.C. Sterne. *Essential medical statistics*. Oxford : Blackwell Science, 2003.
10. H.C. Kraemer. *Evaluating Medical Test: Objective and Quantitative Guidelines*. Saga Publications, Newbury Park, Calif., 1992.
11. H.C. Kraemer, V.S. Periyakoil, and A. Noda. Tutorial in biostatistics: Kappa coefficients in medical research. *Statistics in Medicine*, 21(14):2109–2129, Jul 2002.
12. V. Lacroix. The primary raster: a multiresolution image description. In *Proceedings. 10th International Conference on Pattern Recognition*, volume 1, pages 903–907, Jun 1990.
13. D. Marr and E. C. Hildreth. Theory of edge detection. In *Proceedings of the Royal Society of London*, volume B207, pages 187–217, 1980.
14. James Matthews. An introduction to edge detection: The sobel edge detector. Available at <http://www.generation5.org/content/2002/im01.asp>, 2002.
15. V. S. Nalwa and T. O. Binford. On detecting edges. *IEEE Trans. Pattern Anal. Machine Intell.*, 8(6):699–714, Nov 1986.
16. Eli Peli. Feature detection algorithm based on visual system models. In *Proc. IEEE*, volume 90, pages 78–93, Jan 2002.
17. C. A. Rothwell, J. L. Mundy, W. Hoffman, and V. D. Nguyen. Driving vision by topology. *Int'l Symp. Computer Vision*, pages 395–400, Nov 1995. Coral Gables Fla.
18. B.G. Schunck. Edge detection with gaussian filters at multiple scales. In *Proc. IEEE Computer Society Workshop on Computer Vision*, pages 208–210, 1987.
19. J. Sim and C.C. Wright. The kappa statistic in reliability studies: Use, interpretation, and sample size requirements. *Journal of the American Physical Therapy*, 85(3):257–268, March 2005.
20. M. Ulrich and C. Steger. Empirical performance evaluation of object recognition methods. In H. I. Christensen and P. J. Phillips, editors, *Empirical Evaluation Methods in Computer Vision*, pages 62–76. IEEE Computer Society Press, 2001.

Bio-inspired Motion-Based Object Segmentation

Sonia Mota¹, Eduardo Ros², Javier Díaz², Rodrigo Agis², and Francisco de Toro³

¹ Departamento de Informática y Análisis Numérico, Universidad de Córdoba, Campus de Rabanales, Edificio Albert Einstein, 14071 Córdoba, Spain

² Departamento de Arquitectura y Tecnología de Computadores, Universidad de Granada, Periodista Daniel Saucedo Aranda s/n, 18071 Granada, Spain

³ Departamento de Teoría de la Señal, Telemática y Comunicaciones, Universidad de Granada, Periodista Daniel Saucedo Aranda s/n, 18071 Granada, Spain
smota@uco.es, {eros, jdiaz, ragis}@atc.ugr.es, ftoro@ugr.es

Abstract. Although motion extraction requires high computational resources and normally produces very noisy patterns in real sequences, it provides useful cues to achieve an efficient segmentation of independent moving objects. Our goal is to employ basic knowledge about biological vision systems to address this problem. We use the Reichardt motion detectors as first extraction primitive to characterize the motion in scene. The saliency map is noisy, therefore we use a neural structure that takes full advantage of the neural population coding, and extracts the structure of motion by means of local competition. This scheme is used to efficiently segment independent moving objects. In order to evaluate the model, we apply it to a real-life case of an automatic watch-up system for car-overtaking situations seen from the rear-view mirror. We describe how a simple, competitive, neural processing scheme can take full advantage of this motion structure for segmenting overtaking-cars.

1 Introduction

Alive beings have multimodal systems based on motion, colour, texture, size, shape, etc., which efficiently segment objects.

Could a mono-modal system work as well as biological systems do? Motion, by itself, provides useful cues to achieve an efficient segmentation of independent moving objects. Machine vision systems are adapting strategies from biological systems that represent highly efficient computing schemes. These artificial vision systems still need deal with two main problems: current bio-inspired vision models (based on vertebrates visual systems) are limited and require high computational cost therefore real-time applications are seldom addressed.

On the other hand, simpler models based on insects' motion detection are being developed as valid alternatives for specific tasks; and although they are much more limited require less computing resources. Insects process visual motion information in a local, hierarchical manner. Flies are capable of exploiting optical flow by calculating the local image motion through Elementary Motion Detectors (EMDs) described by Reichardt [1], that emulate the dynamics of early visual stages in insects.

This paper describes the software implementation of an algorithm, based on EMDs. The system proposed follows different stages [2, 3]. The original sequence is pre-processed. The algorithm extracts the edges in each frame. As a result, the number of pixels to be processed in successive stages becomes smaller and therefore the computing requirements of the system are reduced. The next stage uses the Reichardt motion detector to extract sideways moving features. The resulting saliency map is a noisy pattern and it is difficult to extract useful information to segment objects. We use a neural structure that further process the saliency map allowing the emergence rigid bodies characterized as the independent moving objects in the scene (Velocity Channel, VCh).

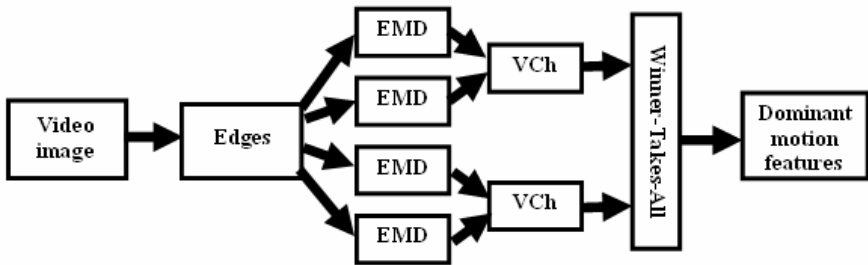


Fig. 1. Structure of the model

2 Functional Description of the Proposed Scheme

2.1 Edges Extraction

In flies, information from photoreceptors converges onto tangential cells. A subset of these neurons has been found to respond primarily to horizontal motion. While another class of neurons respond to vertical motion [4].

Hence, we start by obtaining spatial edges of the original image. We use a conventional Sobel edge extraction procedure with a 3x3 mask [5]. This is easy to be implemented through specific hardware [3], and this detector is able to provide acceptable real-time outputs to Reichardt detectors. A threshold allows us to select the main scene features among the whole extracted edges to be processed in the next stage. This resultant sparse map is composed of features whose strength depends on the local contrast (zero if the local contrast is under the threshold). In this way we dramatically reduce the number of points to be processed and also the computational requirements of the system.

2.2 Velocity Tuning

The system described is based on the Reichardt correlator model [1] and, in particular, on the fly motion detection model, that is based on a multiplicative correlation detector. Behavioural and electrophysiological data from many animals are consistent with a correlation-type mechanism for the detection of local motion.

When a motion pattern is detected, it is seen as a stimulus that reaches the two detector inputs of the EMD with a certain delay. In this process, the output of the first cell precedes the output of the second one by Δt . An EMD is sensitive to a concrete direction movement, i.e. it compensates by an intrinsic delay d the temporal separation (Δt) between the cell signals if an edge moves in the preferred detector direction. In this way the output of both channels will be simultaneous when the delay compensates Δt , and the two stimuli will be perfectly correlated (the detector response will be maximum). Otherwise, the delay increases the temporal separation between the two channel outputs and the output signals are less correlated, therefore the detector response will be weak.

A single EMD gives us information about the motion direction of the detected features, and a set of EMDs, each of them tuned to a different velocity, can also provide information about the velocity module. The EMD that maximizes the correlation is the one that detects the correct velocity (module and direction) of the covered area [6, 7].

The obtained saliency map is a cloud of points where each pixel is associated to a different velocity module and direction (rightward or leftward direction). The output is very noisy, mainly because the response of the EMDs is highly contrast-dependent and it can oscillate from frame to frame. In order to reduce the noise and effectively segment the independent moving object we can use some clustering criteria based on the responses of the specific “*velocity channels*”.

2.3 Coherent Motion Based on “Velocity Channels”

Many studies suggest that the integration of local information permits the discrimination of objects in a noisy background [8, 9, 10, 11]. The mechanism of this integration in biological systems is still an open topic.

The neural structure presented here can take full advantage of population coding for a specific application such as the segmentation of independent moving objects. The main goal of the proposed scheme is the improvement of rigid-body motion (RBM) detection by the constructive integration of local information into global patterns. If we detect a population of pixels labelled with the same speed, and all of them are in a limited area of the image, they are good candidates of a segmented rigid body.

In applications where exists perspective distortion, points in the distant part of a rigid body seem to move more slowly than closer points. Because of this, it becomes necessary to consider different velocities to coherently match the whole rigid body. We need to cluster the velocities of the pixels into a range of velocities. Hence, we introduce a new concept, “*velocity channels*” (VCh in Fig. 1), this allows us to apply the RBM rule when perspective is a problem.

In the proposed scheme, the neurons at this stage receive excitatory, convergent, many-to-one connections from the EMDs layer and integrate the local motion information in a region. The layer has different integrative neurons in the same area. Each integrative neuron is sensitive to a set of velocities $V \pm \Delta V$ (“*a velocity channel*”) from EMDs outputs, where ΔV represents slight variations in module from preferred values, i.e. each neuron integrates the activity of a number of EMDs into a spatial neighbourhood that tunes the characteristic velocity of this integrative neuron. Integrative neuron produces an output if it gathers a high activity from its input EMDs.

The layer is configured as a self-competitive one: the neuron that receives the maximum contribution in its area of spatial influence inhibits the others (tuned to different velocity channels) and dominates (winner-takes-all). This function works as a filter that enhances the presence of rigid bodies and neglects noise patterns.

The configuration of this layer can reduce the perspective deformations of motion patterns. Particularly, the sensitivity of each neuron to a set of characteristic speeds (a velocity channel) rather than a single one reduces the effect of this perspective problem.

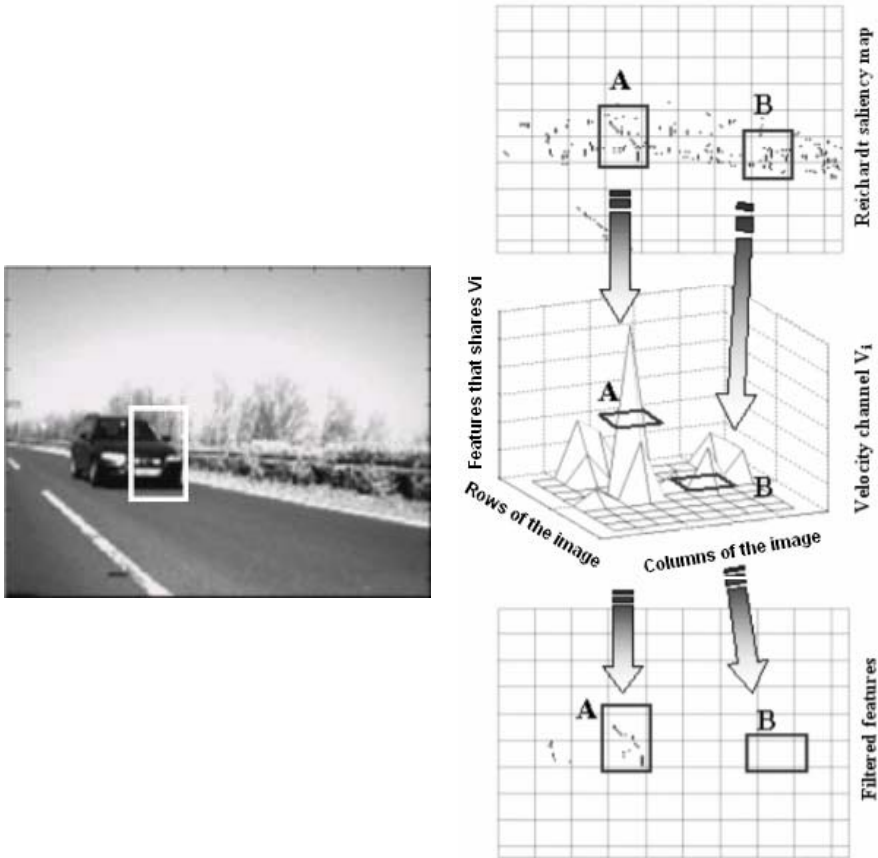


Fig. 2. The top plot of the part of figure shows rightward motion features detected at the Reichardt stage; the velocity channel V_i (in the middle plot of the figure) acts as a coherency filter to segment the overtaking vehicle. It maintains active only the filtered rightward motion features in the bottom plot of the figure.

Fig. 2 illustrates the way the integration is done by the neural layer in a concrete example: the segmentation of a car approaching us. We divide the image into a grid that constitutes the receptive field of each integrative neuron. An integrative neuron that responds to a velocity channel V_i integrates the number of points tuned with

velocities around V_i for each square of the grid (receptive field). We will have as many velocity channels (or integrative neurons) as velocity sub-sets we consider.

The top plot of the Fig. 2 represents rightward motion features detected at the Reichardt stage. The central plot of Fig. 2 represents the activity of a velocity channel V_i . This 3D figure shows the grid in x-y plane (receptive integration areas), and the vertical dimension represents the number of points that match this velocity V_i for all the receptive fields in the grid. These velocity channels work as band pass dynamic filters. Only the points of the saliency map tuned to the cluster velocities that produce the local maximum in each receptive field of the velocity channel are maintained active. The maximum corresponds to points where a rigid body motion induces coherent feature motion. Hence, the points tuned to V_i in receptive field **A** reach the final output layer (the bottom plot of Fig. 2), while the receptive field **B** (in the top plot of the Fig. 2) does not produce any output, i.e., the points belonging to this receptive field are discarded because their contribution is poor and the integrative neuron has low activity.

Finally, we obtain the results in Fig. 2 when the full set of velocity channels are applied to the input map (i.e. the EMDs output).

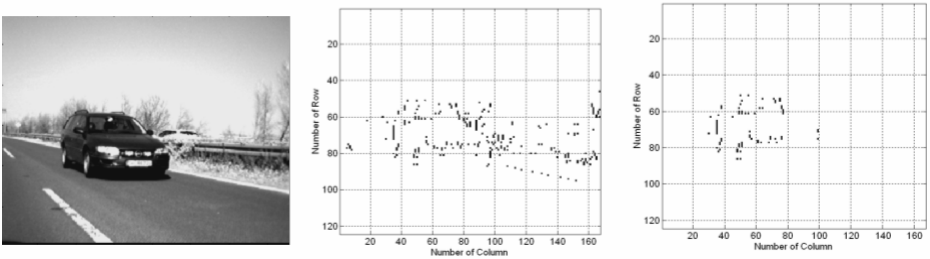


Fig. 3. Saliency map of the Reichardt stage. **(a)** Original image; **(b)** Rightward motion features before the newer filtering process; **(c)** Rightward motion features after the newer filtering process.

Figure 3 shows an example of the moving features extracted at Reichardt stage. The scene shows an overtaking vehicle approaching to the vehicle in which the camera is placed on. This example is a complex scenario, due to the ego-motion of the camera all the points are moving through the scene, and the system task is to segment the overtaking vehicle. The output is very noisy (there are points labelled as moving to the left that belong to the overtaking car, and others are labelled as moving to the right and, in fact, they belong to the landscape) and it would be difficult to automatically segment the overtaking vehicle, however after applying this neural layer segmentation became easier.

It is a difficult task to evaluate the segmentation quality of the algorithm using real images since pixels are not labeled according to the object they belong to. Therefore, we manually mark the goal objects labeling each pixel belonging to them, and then, we will compare the results of the automatic motion-driven segmentation with the marked labels.

We define the segmentation rate as the ratio between the well classified features in an object and the total number of moving features segmented that belong to this

object. Figure 4 shows the results of segmentation rate in a few frames from a sequence. They are always above 85% when rigid body motion filtering is applied, these results improve the segmentation without filtering stage, which are below 85% when the rightward motion features are considered and below 80% when leftward motion features are taken into account. This difference between both directions of motion is due to the perspective distortion that produces a non-symmetrical image to motion concerns. Velocity channels allow us to minimize the noisy patterns and to segment the overtaking car in the sequence.

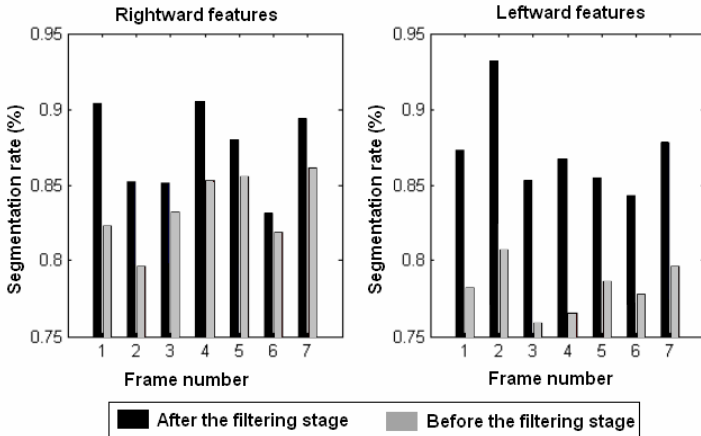


Fig. 4. Segmentation rate

3 Application to an Overtaking Warning System

The next section evaluates the previous model in a concrete real-world application.

Each year thousands of drivers are killed and millions injured in motor vehicle accidents around world. The accident rates associated with the different crash situations shall motivate the development of technologies towards the reduction of the number of accidents. Most deaths in two-car crashes occur in front-to-front crashes. However, side impacts have now become more important, probably reflecting improvements in protecting occupants in head-on collisions [12]. One of the most dangerous operations in driving is to overtake other vehicle. The driver's attention is on his way, and sometimes does not use the rear-view mirror or it is unhelpful because of the blind spot.

The automobile industry is very interested in introducing systems applied to drive assistance, [13, 14]. Radar detection devices warning of potential threats have long been employed in aircraft and ships. Their installation on the craft themselves as well as on land near areas of congestion, have allowed conflicts between ships to be detected at distances that permit courses to be changed in time to avoid collisions. Due to the traffic, applying such collision avoidance technology to the highway [15, 16] is more challenging than in airways or waterways. Detecting relatively small position

changes that represent possible threats, against the background of other forms of absolute or relative motion, and doing it reliably enough to discriminate changes in speed or direction seem beyond today technology. However, it may be possible at least to provide a warning alert to a driver. In some recent works research is focussed on systems that use both jointly radar and vision approaches for this purpose [17, 18].

This section is concerned with the use of the previous model to watch up overtaking scenarios: it detects and tracks the vehicle behind us, discriminates whether it is approaching or not and alerts us about its presence if necessary.

We have used real overtaking sequences that have been taken with a camera placed onto the driver's rear-view mirror. The sequences are composed by at least 50 frames with a resolution of 640x480 pixels per frame and 256 grey levels.

In previous works [2, 3] we have described and tested the proposed algorithm under different weather conditions. In this contribution, we present some key improvements with respect to the previous works and we evaluate the algorithm under different overtaking speeds, which is important to consider as well, since a useful system shall work for wide range of overtaking relative speeds.

These sequences include different velocities of the overtaking processes that have been measured using an instrumented car with a laser-based system. This variety of speeds is difficult to process by other bio-inspired vision models that give good responses to slow or rapid motions but have difficulties to cover the different ranges.

We have marked the overtaking car with a rectangle frame in each image. This process has been done manually, so we know the limits of the rectangle in each frame. We take this centre to plot the overtaking car speed in four sequences where the approaching speed is between 1-3 m/s, 5-10 m/s, 15-20 m/s and 25-30 m/s.

In an overtaking scenario the motion structure of the approaching vehicle shows a high contrast respect to the landmarks moving in the opposite direction due to the ego-motion of the host car. Therefore, in each image, we will label the motion features moving rightward as belonging to the overtaking vehicle and discard the rest. In a second stage we will calculate the performance of the segmentation (P_s) as the ratio between the features moving right inside the marked rectangle surrounding the car ($N_{mr(in)}$ correctly labelled features) and all the features detected with rightward motion ($N_{mr(in)}+N_{mr(out)}$) according to Equation (1). The accuracy of the segmentation will depend on the relative speed and size of the overtaking car. We have distinguished different cases depending on the size of the car (due to the distance from the host vehicle). The considered car sizes, estimated in area (S) as the number of pixels contained in the rectangle, are:

A ($315 < S \leq 450$), B ($450 < S \leq 875$), C ($875 < S \leq 2220$), D ($2220 < S \leq 6000$) and E ($S > 6000$).

$$P_s = \frac{N_{mr(in)}}{N_{mr(in)} + N_{mr(out)}} \quad (1)$$

Fig. 5 summarizes the results. It can be seen that performance increases as the overtaking vehicle approaches. In other words, the nearer the overtaking car is, the better the segmentation and based on less noisy patterns.

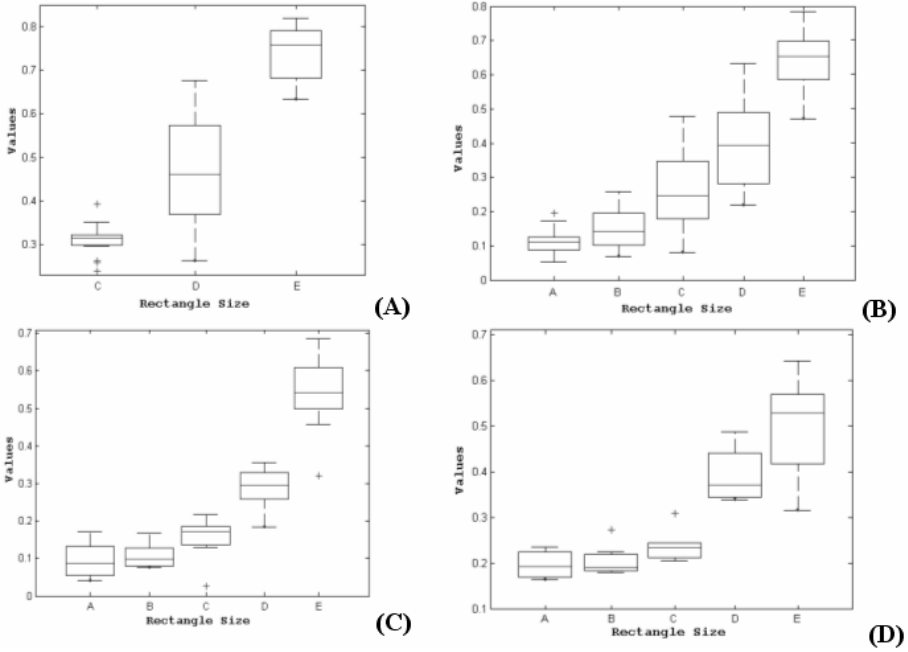


Fig. 5. Performance of different overtaking sequences: (A) 1-3 m/s, (B) 5-10 m/s, (C) 15-20 m/s and (D) 25-30 m/s

To track the overtaking vehicle we calculate the centre of mass of all the points that are moving rightward in the filtered output. Fig. 6 shows an example of the tracking process along a sequence.



Fig. 6. Tracking results along a sequence

4 Conclusions

The present contribution describes a motion processing system able to segment independent moving objects using sparse saliency maps. We have evaluated the model as a warning device for overtaking scenarios. The front-end of the system are Reichardt motion detectors. We define dynamic filters based on motion patterns of the image

that seem to correspond to independent moving objects. These dynamic filters effectively clean noisy patterns and help to segment independent moving objects. This filtering technique is a robust scheme because it is only based on a rigid body motion rule. It detects areas within a population of features moving coherently (with the same velocity and direction), being good candidates for a moving rigid body, and these motion patterns compete locally with opposite direction motion features. In this way, the moving features are processed in a competitive manner; only patterns that activate a whole population of detectors with a similar velocity become salient and remain active after the dynamic filter stage.

A correlation based analysis and the rigid body motion filtering is enough to segment two objects moving with different velocities. Although such a monomodal system based on velocity detection is not able to identify two vehicles with synchronized motion, for instance two cars running in parallel and at the same speed. Even so, the application (a real-life case of an automatic watch-up system for car overtaking situations seen from the rear-view mirror) is not limited by this because we only need to know the location of the dangerous agent, i.e. the nearest part of the nearest moving object, which is obtained by the system.

The system has been tested on real overtaking sequences in a different speed ranges. A simple “centre of mass” calculation has been tested to validate the obtained features for the tracking application.

Acknowledgements

This work has been supported by the National Project DEPROVI (DPI2004-07032) and the EU Project DRIVSCO (IST-016276-2). We thank Hella (Dept. of predevelopment EE-11, Hella KG Hueck & Co., Germany) for providing us with the overtaking sequences.

References

1. Reichardt, W.: Autocorrelation, a principle for the evaluation of sensory information by central nervous system. in Rosenblith W. A. *Sensory Communication*, (1961) 303-317.
2. Mota, S., Ros, E., Ortigosa, E.M., Pelayo, F. J.: Bio-Inspired motion detection for blind spot overtaking monitor. *International Journal of Robotics and Automation* 19 (4) (2004)
3. Mota, S., Ros, E., Díaz, J., Ortigosa, E.M., Agís, R., Carrillo, R.: Real-time visual motion detection of overtaking cars for driving assistance using FPGAs. *Lecture Notes in Computer Science*, Vol. 3203. Springer-Verlag, Berlin Heidelberg New York (2004) 1158-1161
4. Haag, J., Borst, A.: Encoding of visual motion information and reliability in spiking and graded potential neurons. *Journal of Neuroscience*, Vol. 17. (1997) 4809-4819.
5. Gonzalez, R. & Woods, R.: *Digital Image Processing*. Addison-Wesley (1992)
6. Clifford, C. W. G., Ibbotson, M. R., Langley, K.: An adaptive Reichardt detector model of motion adaptation in insects and mammals. *Visual Neuroscience*, Vol. 14. (1997) 741-749.
7. Ros, E., Pelayo, F. J., Palomar, D., Rojas, I., Bernier, J. L., Prieto, A.: Stimulus correlation and adaptive local motion detection using spiking neurons, *International Journal of Neural Systems* Vol. 9(5). (1999) 485-490.

8. Barlow, H. B.: The efficiency of detecting changes of intensity in random dot patterns. *Vision Research* Vol. 18 (6). (1978) 637-650.
9. Field, D. J., Hayes, A., Hess, R. F.: Contour integration by the human visual system: evidence for local "association field". *Vision Research*, Vol. 33 (2). (1993) 173-193.
10. Saarinen, J., Levi, D. M., Shen, B.: Integration of local pattern elements into a global shape in human vision. In *Proceeding of the National Academic of Sciences USA*, Vol. 94. (1997) 8267-8271.
11. Gilbert, C. D., Wiesel, T. N.: Intrinsic connectivity and receptive field properties in visual cortex, *Vision Research*, Vol. 25 (3). (1985) 365-374.
12. Ross, M., Wenzel, T.: Losing weight to save lives: a review of the role of automobile weight and size in traffic fatalities. Report from the American Council for an Energy-Efficient Economy, ACEEE-T013 (2001)
13. Franke, U. et al.: From door to door- Principles and Application on Computer Vision for driver assistant systems. In *Proceeding of Intelligent Vehicle Technologies: Theory and Applications* (2000)
14. Handmann U. et al.: Computer Vision for Driver Assistance Systems. In *Proceeding of SPIE*, Vol. 3364. (1998) 136-147.
15. Schneider, R., Wenger, J.: High resolution radar for automobile applications. *Advances in Radio Science*, Vol. 1. (2003) 105-111
16. Ewald, A., Willhoeft, V.: Laser Scanners for Obstacle Detection in Automotive Applications. In *Proceedings of the IEEE Intelligent Vehicle Symposium*. (2000) 682-687.
17. Heisele, B., Neef, N., Ritter, W., Schneider, R., Wanielik, G.: Object Detection in Traffic Scenes by a Colour Video and Radar Data Fusion Approach. *First Australian Data Fusion Symposium*. (1996) 48-52.
18. Fang, Y., Masaki, I., Horn, B.: Depth-Based Target Segmentation for Intelligent Vehicles: Fusion of Radar and Binocular Stereo. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 3 (3). (2002)

An Effective and Fast Scene Change Detection Algorithm for MPEG Compressed Videos

Z. Li¹, J. Jiang^{1,2}, G. Xiao¹, and H. Fang²

¹ Faculty of Informatics & Computing, Southwest China University, Chongqing, China
j.jiang1@bradford.ac.uk, g.xiao@swcu.edu.cn

² Department of EIMC, University of Bradford, UK

Abstract. In this paper, we propose an effective and fast scene change detection algorithm directly in MPEG compressed domain. The proposed scene change detection exploits the MPEG motion estimation and compensation scheme by examining the prediction status for each macro-block inside B frames and P frames. As a result, locating both abrupt and dissolved scene changes is operated by a sequence of comparison tests, and no feature extraction or histogram differentiation is needed. Therefore, the proposed algorithm can operate in compressed domain, and suitable for real-time implementations. Extensive experiments illustrate that the proposed algorithm achieves up to 94% precision for abrupt scene change detection and 100% for gradual scene change detection. In comparison with similar existing techniques, the proposed algorithm achieves superiority measured by recall and precision rates.

1 Introduction

Detection of scene changes plays important roles in video processing with many applications ranging from video indexing and video summarization to object tracking and video content management. Over the last three decades, scene change detection has been widely studied and researched. As a result, many scene change detection techniques have been proposed and published in the literature. For our convenience of surveying existing research in this subject area, all these algorithms and techniques can be broadly classified as operating on decompressed data (pixel domain), or working directly on the compressed data (compressed domain).

In pixel domain, the major techniques are based on pixels, histogram comparison and edge difference examinations [1],[2],[15],[16],[17],[18]. Seung Hoon Han [17] proposed an algorithm combining Bayesian and structural information to detect scene changes. JungHwan et. Al. [16] developed a content-based scene change detection algorithm, which computes background difference between frames, and use background tracking to handle various camera motions.

Recent trends focus on developing scene change detection algorithms directly in compressed domain, especially corresponding to MPEG compressed videos[1, 3-14]. Scene change detection algorithms operating on compressed data may use MB and motion type information for detecting transitions. Smeaton et. Al [4] developed such a technique to support content-based navigation and browsing through digital video

archives. To reduce the decoding operation to its minimum, Lelescu and Schonfeld [5] have proposed a real-time approach for detecting scene changes based on statistical sequential analysis directly on compressed video bitstreams. Dulaverakis et al [7] have proposed an approach for the segmentation of MPEG compressed video, which relies on the analysis and combination of various types of video features derived from motion information in compressed domain. Edmundo et al [9] have proposed an edges and luminance based algorithm to detect scene changes. At present, MPEG compression schemes have been widely studied, and its motion estimation and compensation has been exploited to detect scene changes by using MB type information and motion vectors [1,12~14]. Fernando et al [12] proposed that abrupt scene change could be detected by computing the number of interpolated macroblocks and the number of backward macroblocks inside B frames. In an approach that can detect both abrupt and gradual scene changes with MB type information[1], the authors explore the number of intra macroblocks and backward macroblocks to detect abrupt scene changes, and the number of interpolated macroblocks to detect gradual scene changes. Johngho et al [14] explored each macroblock in a B-frame is compared with the type of corresponding macroblock (i.e. the macroblock in the same position) of the previous B-frame to detect abrupt scene changes.

In this paper, via focusing on MB type information and coding modes, we propose a fast scene change detection method, which features that threshold selection and comparisons are made adaptive to the input video content. By using the number of intra-coded macroblocks inside P frame and the number of interpolated macroblocks inside B frames, the abrupt scene changes and gradual changes can be automatically detected. In addition, the number of backward-predicted macroblock and forward-predicted macroblocks inside B frames are also used as an enhancement to improve the precision.

The rest of the paper is structured as follows. While section 2 describes the algorithm design, section 3 reports some experiments with real video films to derive empirically decided parameters, and section 4 reports experimental results in comparison with the existing algorithms, and section V provides conclusions and discussion on possible future work.

2 The Proposed Algorithm Design

To achieve effective and efficient motion estimation and compensation inside the video compression scheme, MPEG arranged video sequences into group of pictures (GoP), for which the structure of such arrangement can be illustrated in Figure-1. As seen, the MPEG video structure has the feature that there exist two B-frames between every pair of I-P frames or P-P frames. To make it more convenient in describing the proposed algorithm, we follow the work reported in [1] to label the first B-frame as B_f (front-B), and the second B-frame as B_r (rear-B). As a result, each GoP can be further divided into two sub-groups, which are IB_fB_r and PB_fB_r , and thus the proposed algorithm can also be designed in terms of the two sub-groups.

Essentially, the proposed algorithm follows the principle that the MPEG properties and its embedded motion estimation and compensation scheme should be further exploited for scene change detection[1,2], where the status of such motion estimation

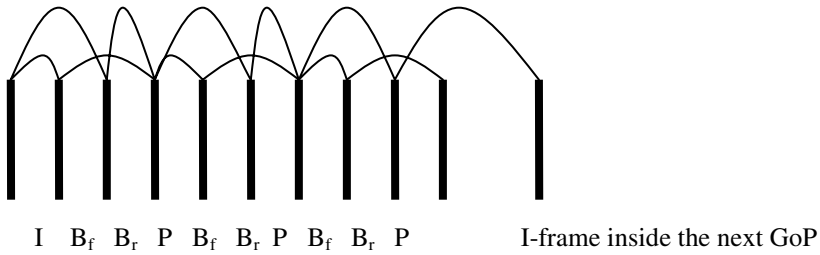


Fig. 1. Illustration of MPEG video structure for motion estimation & compensation

and compensation for each macro-block can be recorded to design such scene change detection algorithm. To this end, Pei and Chou [1] proposed a simple MB-type based scene change detection algorithm for both abrupt scene changes and gradual scene changes. The principle they applied is to monitor the number of MBs in intra-coding mode inside P-frames. Whenever the number of intra-coded MBs is above a pre-defined threshold, two separate operations for scene change detection are activated for abrupt changes and gradual changes respectively. For abrupt changes, their detection is based on one of the three motion estimation and compensation forms in MPEG videos as illustrated in Figure-2. For gradual changes, their detection is based on conditions. One is that a significant number of MBs inside P-frames are intra-coded, indicating significant change of content, and the other is that a dominant number of MBs inside B-frames are interpolative motion compensated. Detailed analysis reveals that there exist a range of weaknesses in such an algorithm, which include: (i) the scene change detection is dependent on four fixed and pre-defined thresholds; (ii) abrupt change detection and gradual scene change detection are two separate operation procedures; and (iii) the performance is low in terms of precision in detecting scene changes. To this end, we propose to: (i) introduce an adaptive mechanism in selecting all the thresholds, and make them adaptive to the input video content; (ii) combine abrupt change detection and gradual change detection into an integrated algorithm; and (iii) add a consecutive detection window to improve the gradual change detection.

Given the fact that specific motion estimation and compensation status for each macro-block is indicated by the number of bi-directionally predicted blocks for B-frames, and intra-coded macroblocks for P-frames, we define a range of essential variables as follows:

N_i : the number of intra-coded MBs inside P-frames

N : the total number of macroblocks inside each video frame;

N_{fb} : the number of both forward and backward predicted macroblocks inside each B-frame;

N_b : the number of backward predicted macroblocks inside each B-frame, indicating that this macroblock is only predicted in backward mode;

N_f the number of forward predicted macroblocks inside each B-frame.

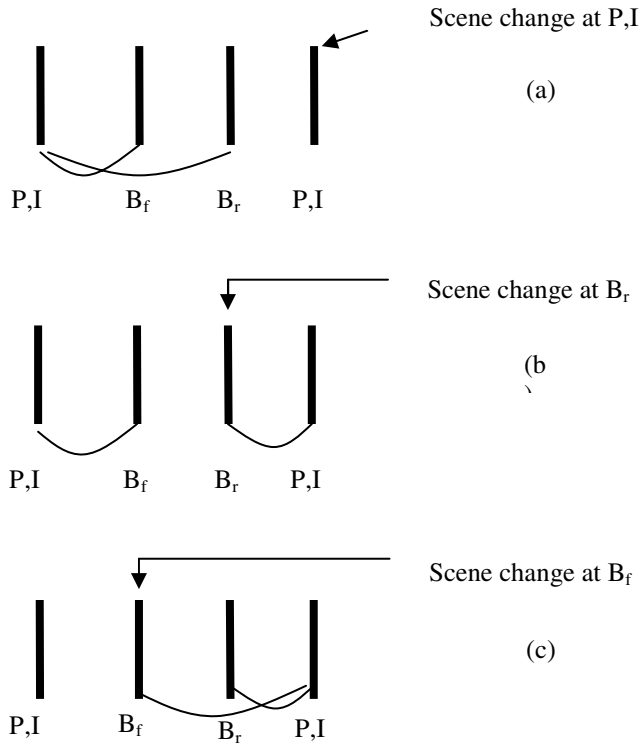


Fig. 2. Illustration of abrupt scene change detection

To decide whether the scene change detection should be activated or not, the following test is conducted for every P-frame, where the proportion of intra-coded MBs is examined with respect to a predefined threshold:

$$\frac{N_i}{N} > T_1 \quad (1)$$

Where T_1 stands for a threshold, which is to be determined empirically.

From (1), a number of observations and hence analysis can be made. Firstly, the essence of the ratio between N_i and N reflect the proportion of predicted macroblocks inside this P-frame. When the ratio is smaller than the threshold, it indicates that most of the macroblocks can be motion compensated by its reference frame, and hence a significant extent of correlation between this P-frame and its reference frame can be established. Therefore, it is not likely that there exists any scene change around this P-frame. As a result, we should carry on examining the next P-frame. Secondly, if the condition represented by (1) is satisfied, it should indicate that most of the macroblocks are not well compensated by the reference frame. Therefore, it is likely that there may exist a scene change in the neighbourhood of this P-frame. As a result, further confirmation of such scene change needs to be confirmed by examining the

B-frames to find out: (i) whether such scene change is abrupt or gradual; (ii) its exact location of such scene changes.

As illustrated in Figure-1, we have front-B and rear-B to be examined at this stage. For all B-frames, there are three possibilities for their specific process of motion estimation and compensation, which include: forward prediction, backward prediction, and bi-directional prediction. To detect the situation given in Figure-2, we propose a

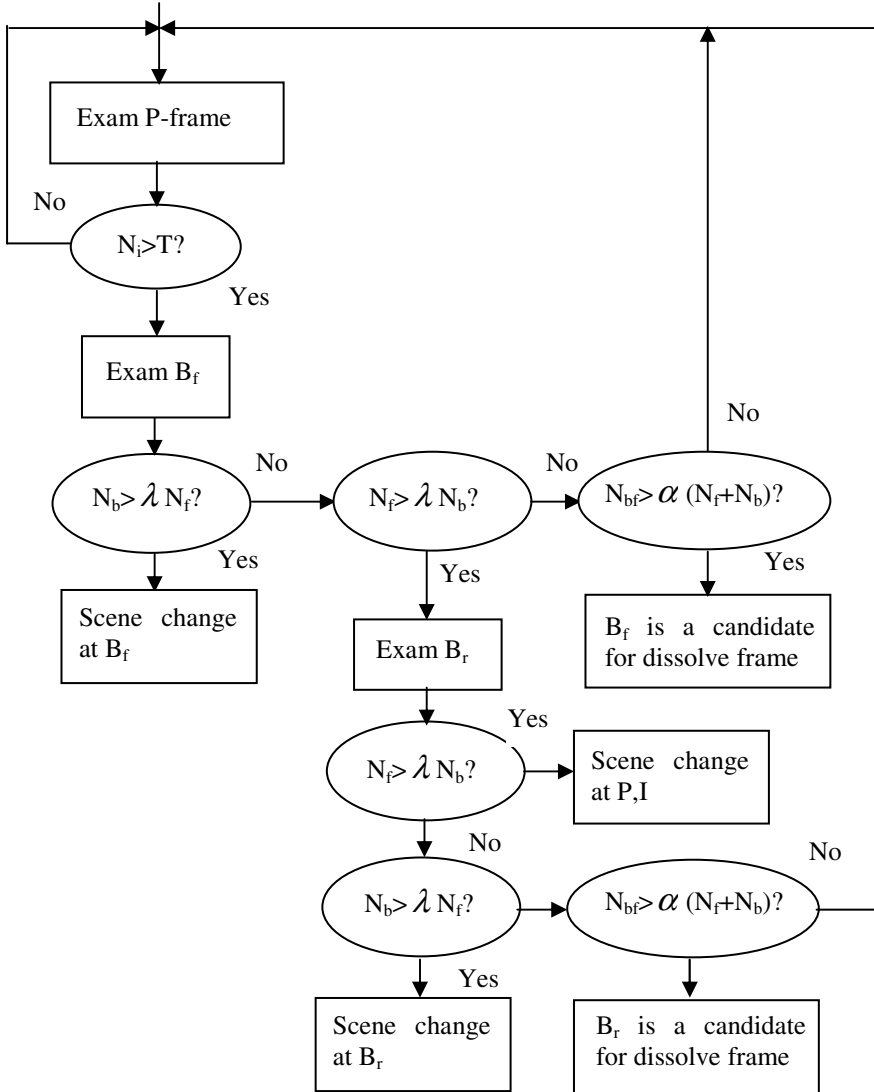


Fig. 3. Overview of the proposed scene change detection algorithm

test, where its threshold is adaptive to the input video content and its motion estimation and compensation process. Therefore, to examine the front-B frame, we have:

$$N_b > \lambda N_f \quad (2)$$

Where λ is a parameter controlling the balance between the backward prediction and forward prediction.

Satisfaction of (2) indicates that backward predicted MBs overwhelm the forward predicted MBs. As a result, the possible scene change can be further confirmed, and located at B_f as shown at the bottom of Figure-2. If the test is negative, we need to find out whether forward predicted MBs overwhelm the backward predicted MBs, in order to verify whether any of the two forms given in (a) and (b) of Figure-2 can be detected. Such condition can be tested by similar equation as illustrated below:

$$N_f > \lambda N_b \quad (3)$$

Satisfaction of the above test confirms either (a) or (b) in Figure-2 depending on the result of examining the rear-B frame. In this circumstance, we firstly repeat the test given in (3) on the rear-B frame. Satisfaction of such test will confirm the case of (a) in Figure-2 and hence a scene change can be detected at the next P or I frame. If the test is negative, we check if the backward predicted MBs overwhelm the forward predicted MBs inside the rear-B frame by (2). Positive test indicates that scene change is at the B_r frame as given in (b) of Figure-2. If not, we need to check if B_r frame is a candidate for dissolve frame (gradual change) by the following test:

$$N_{bf} > \alpha(N_f + N_b) \quad (4)$$

Where α is another parameter indicating the dominance of bi-directional prediction of MBs inside B-frames.

Similarly, negative test of (3) on the front-B frame will prompt the test (4) being applied to B_f to see if it is a candidate for a dissolve frame.

In summary, the proposed algorithm can be overviewed in Figure-3.

3 Algorithm Evaluations

To evaluate the performance of the proposed scene change detection algorithm, a database of video sequences has been obtained from recording of TV programmes, including documents, news, sports, and films. The database represents a total of 114 abrupt scene changes and 22 dissolves in 15 clips of videos. The selected sequences are complex with extensive graphical effects. Videos were captured at a rate of 30 frames/s, 640×480 pixel resolution, and stored in PPM format. Details of the test video clips are described in Table-1.

For benchmarking purposes, we selected the algorithm reported in [1] as a representation of the existing techniques to provide a comparison for the evaluation of the proposed algorithm. Performance measurements are implemented with the well-known recall and precision figures [16], which are defined as follows.

$$\text{Recall} = \frac{\text{detects}}{(\text{detects} + MD)} \quad (5)$$

$$\text{Precision} = \frac{\text{detects}}{(\text{detects} + FA)} \quad (6)$$

Table 1. Description of video clip test set

| Video sequences | Number of abrupt scene changes |
|-----------------|--------------------------------|
| Clip-1 | 24 |
| Clip-2 | 15 |
| Clip-3 | 13 |
| Clip-4 | 21 |
| Clip-5 | 19 |
| Clip-6 | 22 |
| Clip-7 | 86~107;160~179 |
| Clip-8 | 135~165;540~574 |
| Clip-9 | 80~115;230~258 |
| Clip-10 | 6~101,128~144;315~325 |
| Clip-11 | 5~16;209~248;358~365 |
| Clip-12 | 27~42;67~80;140~181 |
| Clip-13 | 70~88;115~130;190~289 |
| Clip-14 | 132~175;224~278 |
| Clip-15 | 23~44;158~189 |
| total | Abrupt=136; dissolved=22 |

Where *detects* stands for the correctly detected boundaries, while MD and FA denote missed detections and false alarms, respectively. In other terms, at fixed parameters, recall measures the ratio between right detected scene changes and total scene changes in a video, while precision measures the ratio between the right detected scene changes and the total scene changes detected by algorithm.

The final results of the evaluations are given in Tables 2.

From Table 2, it can be seen that the proposed method achieves on average a 90.38% recall rate with a 75.49% precision rate for both abrupt and gradual scene changes. In comparison with the existing counterparts [1] given in Table-2 as the benchmark, the proposed method achieves superior performances in terms of both recall and precision rates. Additional advantages with the proposed method can be highlighted as: (i) integrated algorithm for both abrupt scene change and dissolve detection; (ii) directly operates in compressed domain and thus suitable for real-time implementation; and (iii) only one threshold and two parameters are required for scene change detection and yet the detection mechanism is made adaptive to the input video content, or the performance of motion estimation and compensation embedded inside MPEG techniques.

Table 2. Summary of experimental results

| Clips | The proposed algorithm | | The benchmark | |
|---------|------------------------|-----------|---------------|-----------|
| | Recall | Precision | Recall | Precision |
| 1 | 87 | 90 | 56 | 85 |
| 2 | 92 | 92 | 72 | 80 |
| 3 | 96 | 86 | 76 | 78 |
| 4 | 89 | 85 | 65 | 75 |
| 5 | 90 | 94 | 84 | 81 |
| 6 | 85 | 92 | 80 | 89 |
| 7 | 100 | 100 | 100 | 66.7 |
| 8 | 100 | 66.7 | 100 | 50 |
| 9 | 50 | 50 | 50 | 33.3 |
| 10 | 100 | 75 | 66.7 | 50 |
| 11 | 100 | 60 | 100 | 43 |
| 12 | 66.7 | 75 | 100 | 43 |
| 13 | 100 | 50 | 66.7 | 50 |
| 14 | 100 | 66.7 | 100 | 40 |
| 15 | 100 | 50 | 100 | 50 |
| Average | 90.38 | 75.49 | 81.09 | 60.93 |

5 Conclusions

In this paper, we described an integrated algorithm, which is capable of directly and effectively detect scene changes in MPEG compressed videos. The proposed algorithm works in compressed domain exploiting existing MPEG motion estimation and compensation mechanisms. Therefore, it achieves significant advantages in terms of processing speed and algorithm complexity, and thus suitable for fast and real-time implementations. While the proposed algorithm can save a lot of computation cost, extensive experiments support that the proposed algorithm also achieves superior performances over the existing counterparts.

Finally, the authors wish to acknowledge the financial support under EU IST FP-6 Research Programme as funded for the integrated project: LIVE (Contract No. IST-4-027312).

References:

1. Aidan Totterdell, "An algorithm for detecting and classifying scene breaks in MPEG video bit streams", Technical report , No.5, September 1998;
2. Ramin Zabih, Justin Miller and Kevin Mai, "A feature-based algorithm for detecting and classifying production effects", *Multimedia Systems*, Vol.6, No.2, 1999 , pp.119-128;
3. Georgios Akrivas, Nikolaos.D.Doulamis, Anastasios.D.Doulamis and Stefanos.D.Kollias, " Scene detection methods for MPEG-encoded video signals", *Proceeding of the MELECON 2000 Mediterranean Electrotechnical Conference*, Nicosia, Cyprus, May 2000;

4. A. Smeaton et al, "An evaluation of alternative techniques for automatic detection of shot boundaries in digital video", in Proc. Irish Machine Vision and Image Processing Conference (IMVIP'99), Dublin, Ireland, 8-9 Sep. 1999, pp.45-62;
5. Dan Lelescu and Dan Schonfeld, "Statistical sequential analysis for real-time video scene change detection on compressed multimedia bitstream", *Multimedia, IEEE Transactions on*, Vol.5, No.1, March 2003, pp.106-117;
6. Jung-Rim Kim, Sungjoo Suh, and Sanghoon Sull, "Fast scene change detection for personal video recorder", *IEEE Transactions on Consumer Electronics*, Vol.49, No.3, August 2003, pp.683-688;
7. C.Dulaverakis, S.Vagionitis, M.Zervakis and E.Petrakis, "Adaptive methods for motion characterization and segmentation of MPEG compressed frame sequences", *International Conference on Image Analysis and Recognition (ICIAR 2004)*, Porto, Portugal, September 29-October 1, 2004, pp.310-317;
8. S.Porter, M.Mirmehdi, B.Thosma, "Temporal video segmentation and classification of edit effects", *Image and Vision Computing*, Vol.21, No.13-14, December 2003, pp.1098-1106;
9. Edmundo Saez, Jose I.Benavides and Nicolas Guil, "Reliable time scene change detection in MPEG compressed video", *ICME 2004*, 27-30 June 2004, Taipei, pp.567-570;
10. Jesus Bescos, "Real-time shot change detection over online MPEG-2 video", *Circuits and Systems for Video Technology, IEEE Transactions on*, Vol.14, No.4, April 2004, pp.475-484;
11. Serkan Kiranyaz, Kerem Caglar, Bogdan Cramariuc and Moncef Gabbouj, "Unsupervised scene change detection techniques in features domain via clustering and elimination", *Proc. of Tyrrhenian International Workshop on Digital Communications*, Capri, Italy, September 8th - 11th, 2002;
12. W.A.C.Fernando, C.N.Canagarajah and D.R.Bull, "Scene change detection algorithms for content-based video indexing and retrieval", *Electronics & communication engineering journal*, June 2001, pp.117-126;
13. Soo-Chang Pei and Yu-Zuon Chou, "Novel error concealment method with adaptive prediction to the abrupt and gradual scene changes", *IEEE transactions on multimedia*, Vol. 6, No.1, February 2004, pp.158-173.
14. Johngho Nang, Seungwook Hong and Youngin Ihm, "An efficient video segmentation scheme for MPEG video stream using macroblock information", *Proceedings of the 7th ACM International Conference on Multimedia '99*, October 30 - November 5, 1999, Orlando, FL, USA. ACM, 1999, Vol.1, pp.23-26;
15. Shu-Ching Chen, Mei-Ling Shyu, Cheng-Cui Zhang and R.L.Kashyap, "Video scene change detection method using unsuper segmentation and object tracking", *IEEE International Conference on Multimedia and Expo (ICME)*, 2001, pp.57-60;
16. JungHwan Oh**, Kien A.Hua**, Ning Liang **, "A content-based scene change detection and classification technique using background tracking", *Proc. of IS&T/SPIE Conference on Multimedia Computing and Networking 2000*, San Jose CA, January 2000, pp.254-265;
17. Seung Hoon Han, "Shot detection combining bayesian and structural information", *Storage and Retrieval for Media Databases 2001*, Vol. 4315, December 2001, pp509-516;
18. H.B.Lu and Y.J.Zhang, "Detecting abrupt scene change using neural network", *Visual Information and Information Systems, Third International Conference, VISUAL '99*, Amsterdam, The Netherlands, June 2-4, 1999, pp.291-298.

Automatic Segmentation Based on AdaBoost Learning and Graph-Cuts*

Dongfeng Han, Wenhui Li, Xiaosuo Lu, Tianzhu Wang, and Yi Wang

College of Computer Science and Technology, Key Laboratory of Symbol Computation and Knowledge Engineering of the Ministry of Education, Jilin University, Changchun, 130012, P.R. China
handongfeng@gmail.com

Abstract. An automatic segmentation algorithm based on AdaBoost learning and iterative Graph-Cuts are shown in this paper. In order to find the approximate location of the object, AdaBoost learning method is used to automatically find the object by the trained classifier. Some details on AdaBoost are described. Then the nodes aggregation method and the iterative Graph-Cuts method are used to model the automatic segmentation problem. Compared to previous methods based on Graph-Cuts, our method is automatic. This is a main feature of the proposed algorithm. Experiments and comparisons show the efficiency of the proposed method.

1 Introduction

Image segmentation is a process of grouping together neighboring pixel whose properties are coherent. It is an integral part of image processing applications such as accident disposal, medical images analysis and photo editing. Many researchers are focus on this subject. However, even to this day, many of the computational issues of perceptual grouping have remained unresolved. In the analysis of the object in images it is essential that we can distinguish between the object of interest and the rest. This latter group is also referred to as the background. The techniques that are used to find the object of interest are usually referred to as segmentation techniques: segmenting the foreground from background. Semi-automatic segmentation techniques that allow solving moderate and hard segmentation tasks by modest effort on the part of the user are becoming more and more popular. However in some situation, we need automatic segmentation for certain object. Previous automatic segmentation methods have two major drawbacks:

1. The final segmentation results are far from users' expectations.
2. The running time is so slow that it can't meet the real-time demands.

In order to overcome the disadvantages of automatic and semi-automatic segmentation algorithms, an AdaBoost learning and iterative Graph-Cuts segmentation algorithm is proposed. This paper's contribution is twofold.

First, we introduce the AdaBoost learning to the image segmentation. It is a powerful tool for detecting certain object.

* This work has been supported by NSFC Project 60573182, 69883004 and 50338030.

Second, we describe a novel automation segmentation scheme based on AdaBoost learning and iterative Graph-Cuts, which has several favorable properties:

1. Capable of solving automation segmentation problem with high precision.
2. Performs multi-label image segmentation (the computation time does not depend on the number of labels).
3. Fast enough for practical application using multi-scale resolution decomposing. We have applied our method to region-based image retrieval system outputting a good performance.
4. Is extensible, allows constructing new families of segmentation algorithms with specific properties.

First we begin with a review of the related work. In section 3 we give the details of our algorithm. Experiences are given in section 4. In section 5 we give the conclusions.

2 Relative Work

In this section we briefly outline the main features of current state-of-the-art segmentation techniques. We loosely divide these methods into two families: semi-automatic image segmentation and automatic image segmentation.

2.1 Semi-automatic Method

1. Magic Wand is a common selection tool for almost any image editor nowadays. It gathers color statistics from the user specified image point (or region) and segments (connected) image region with pixels, which color properties fall within some given tolerance of the gathered statistics.
2. Intelligent Paint is region-based interactive segmentation technique, based on hierarchical image segmentation by tobogganing [1]. It uses a connect-and-collect strategy to define an object region. This strategy uses a hierarchical tobogganing algorithm to automatically connect image regions that naturally ordered expansion interface to interactively collect those regions which constitute the object of interest. This strategy coordinates human-computer interaction to extract regions of interest from complex backgrounds using paint strokes with a mouse.
3. Intelligent Scissors is a boundary-based method, which computes minimum-cost path between user-specified boundary points [2]. It treats each pixel as a graph node and uses shortest-path graph algorithms for boundary calculation. A faster variant of region-based intelligent scissors uses tobogganing for image over-segmentation and then treats homogenous regions as graph nodes.
4. Graph-Cuts is a powerful semi-automatic image segmentation technique. In [3] a new technique is proposed for a general purpose interactive segmentation of N-dimensional images. The users make certain pixels as foreground and background. Graph-Cuts is used to find the globally optimal segmentation. Given user-specified object and background seed pixels, the rest of the pixels are labelled automatically. Because this procedure is interactive, we call this technique as semi-automatic method. This method can exactly extract the object and has a good time saving property. However the segmentation result is influenced by the users.

5. GrabCuts [4] extends Graph-Cuts by introducing iterative segmentation scheme, that uses Graph-Cuts for intermediate steps. The user draws rectangle around the object of interest-this gives the first approximation of the final object/ background labelling. Then, each iteration step gathers color statistics according to current segmentation, re-weights the image graph and applies graph-cut to compute new re-fined segmentation. After the iterations stop the segmentation results can be re-fined by specifying additional seeds, similar to original Graph-Cuts.

2.2 Automatic Method

1. Watershed transform treats the image as a surface with the relief specified by the pixel brightness, or by absolute value of the image gradient. The valleys of the resulting “landscape” are filled with water, until it reaches the “mountains”. Markers placed flow together, and a user-guided, cumulative cost-in the image specify the initial labels that should be segmented from each other.
2. Clustering methods such as K-mean and C-mean methods are always used. These methods often follow a region tracking and region merge procedure. Such methods are also called region-based segmentation.
3. Mean Shift procedure-based image segmentation is a straightforward extension of the discontinuity preserving smoothing algorithm. Each pixel is associated with a significant mode of the joint domain density located in its neighborhood, after nearby modes were pruned as in the generic feature space analysis technique [5]. There are some parameters can be dynamic defined by users and can give different results.
4. Normalized Cuts is a classical automatic image segmentation technique. It is first proposed by [6] and generalized in [7]. It optimizes a global measure, a normalized-cut type function to evaluate the saliency of a segment. The minimization of this function can be formulated as a generalized eigenvalue problem. The results of this method are not good enough in practical applications. However it gives a direction for the future relative work from the point of graph based image segmentation.

3 The Proposed Algorithm

Our segmentation algorithm includes four stages: AdaBoost learning for determining object location; expanding location for segmentation; multi-scale nodes aggregation; iterative Graph-Cuts segmentation for final results. Below we will describe them in detail.

3.1 Using AdaBoost Learning to the Segmentation Problem

AdaBoost is a recently developed learning algorithm [8]. It boosts the classification performance by combining a collection of weak classifiers to form a stronger classifier. In each step of AdaBoost, the classifier with the best performance is selected and a higher weight is put on the miss-classified training data. In this way, the classifier will gradually focus on the difficult examples to be classified correctly. The formal guarantees provided by the AdaBoost learning procedure are quite strong. In theory, it

is proved that AdaBoost could minimize the margin between positive and negative examples. The conventional AdaBoost procedure can be easily interpreted as a greedy feature selection process. Consider the general problem of boosting, in which a large set of classification functions are combined using a weighted majority vote. The challenge is to associate a large weight with each good classification function and a smaller weight with poor functions. AdaBoost is an aggressive mechanism for selecting a small set of good classification functions which nevertheless have significant variety. Drawing an analogy between weak classifiers and features, AdaBoost is an effective procedure for searching out a small number of good “features” which nevertheless have significant variety.

A variant of AdaBoost algorithm for aggressive feature selection can be described as follows.

- Given example images $(x_1, y_1), \dots, (x_n, y_n)$ where $y_i = 0, 1$ for negative and positive examples respectively.
- Initialize weights $w_{1,i} = 1/(2m), 1/(2l)$ for training example i , where m and l are the number of negatives and positives respectively.

For $t = 1 \dots T$

- 1) Normalize weights so that w_t is a distribution
- 2) For each feature j train a classifier h_j and evaluate its error ϵ_j with respect to w_t .
- 3) Chose the classifier h_j with lowest error.
- 4) Update weights according to:

$$w_{t+1,i} = w_{t,i} \beta_t^{1-\epsilon_i}$$

Where $\epsilon_i = 0$ is x_i is classified correctly, 1 otherwise, and

$$\beta_t = \frac{\epsilon_t}{1 - \epsilon_t}$$

- The final strong classifier is:

$$h(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & \text{otherwise} \end{cases}$$

Where $\alpha_t = \log\left(\frac{1}{\beta_t}\right)$.

The overall form of the detection process is that of a degenerate decision tree, what we call a “cascade” which achieves increased detection performance while radically reducing computation time. The word "cascade" in the classifier name means that the resultant classifier consists of several simpler classifiers (stages) that are applied subsequently to a region of interest until at some stage the candidate is rejected or all the stages are passed. A positive result from the first classifier triggers the evaluation of a second classifier which has also been adjusted to achieve very high detection rates. A positive result from the second classifier triggers a third classifier, and so on. A negative outcome at any point leads to the immediate rejection of the sub-window. As shown in Fig. 1, a series of classifiers are applied to every sub-window. The initial classifier eliminates a large number of negative examples with very little processing.

Subsequent layers eliminate additional negatives but require additional computation. After several stages of processing, the number of sub-windows has been reduced radically. Further processing can take any form such as additional stages of the cascade or an alternative detection system. Stages in the cascade are constructed by training classifiers using AdaBoost and then adjusting the threshold to minimize false negatives. The whole procedure is similar with the face detection method.

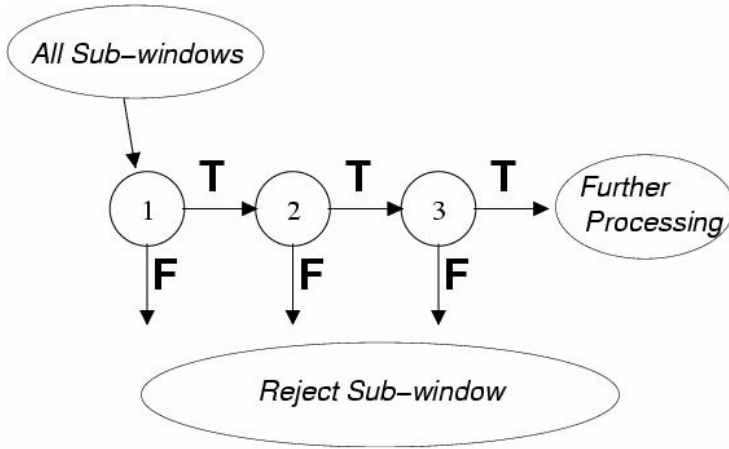


Fig. 1. The description of a detection cascade

The cascade design process is driven from a set of detection and performance goals. Given a trained cascade of classifiers, the false positive rate of the cascade is $F = \prod_{i=1}^K f_i$, where F is the false positive rate of the cascaded classifier, K is the number of classifiers, and f_i is the false positive rate of the i th classifier on the examples that get through to it. The detection rate is $D = \prod_{i=1}^K d_i$, where D is the detection rate of the cascaded classifier, K is the number of classifiers, and d_i is the detection rate of the i th classifier on the examples that get through to it. To achieve efficient cascade for a given false positive rate F and detection rate D we would like to minimize the expected number of features evaluated N :

$$N = n_0 + \sum_{i=1}^K (n_i \prod_{j<i} p_j) \quad (1)$$

Since this optimization is extremely difficult, the usual framework is to choose a minimal acceptable false positive and detection rate per layer.

We can describe the algorithm for training a cascade of classifiers as follows:

- User selects values for f , the maximum acceptable false positive rate per layer and d , the minimum acceptable detection rate per layer.

- User selects target overall false positive rate F_{target} .
- P = set of positive examples
- N = set of negative examples
- $F_0 = 1.0; D_0 = 1.0; i = 0$
 While $F_i > F_{target}$
 $i++; n_i = 0; F_i = F_{i-1}$
 While $F_i > f \times F_{i-1}$
 n_i++
 Use P and N to train a classifier with n_i features using AdaBoost
 Evaluate current cascaded classifier on validation set to determine F_i and D_i
 Decrease threshold for the i th classifier until the current cascaded classifier has a detection rate of at least $dx D_{i-1}$ (this also affects F_i)
 $N = \emptyset$
 If $F_i > F_{target}$ then evaluate the current cascaded detector on the set of non-object images and put any false detections into the set N .

To search for the object in the whole image one can move the search window across the image and check every location using the cascade classifier. The classifier is designed so that it can be easily "resized" in order to be able to find the objects of interest at different sizes, which is more efficient than resizing the image itself. So, to find an object of an unknown size in the image the scan procedure should be done several times at different scales.

3.2 Iterative Segmentation by Graph-Cuts

Definition 1. S-T graph: A S - T graph is defined as $G = (V, E, C)$. It has such properties,

- (i) A source node and a sink node.
- (ii) Directed edge (Arc) $\langle v_i, v_j \rangle$ from node i to node j .
- (iii) Each arc $\langle v_i, v_j \rangle$ has a nonnegative capacity $cap(v_i, v_j)$.
- (iv) $cap(v_i, v_j) = 0$ for non-exist arcs.

Definition 2. Flow in S-T graph: Flow is a real value function f that assign a real value $f(v_i, v_j)$ to each arc $\langle v_i, v_j \rangle$ under,

- (i) Capacity constraint: $f(v_i, v_j) \leq cap(v_i, v_j)$.
- (ii) Mass balance constraint,

$$\sum_{\langle v_i, v_j \rangle \in E} f(v_i, v_j) - \sum_{\langle v_k, v_i \rangle \in E} f(v_k, v_i) = \begin{cases} 0 & v_i \in V - \{v_s, v_t\} \\ |f| & v_i = v_s \\ -|f| & v_i = v_t \end{cases}$$

Definition 3. S-T cut: A cut is a partition of node set V which has two subsets S and T . Additional, a cut is a S - T cut if and only if $s \in S, t \in T$.

Definition 4. Capacity of S - T cut (cost): The cost of S - T cut is defined as,

$$cap([S, T]) = \sum_{\langle v_i, v_j \rangle \in E, v_i \in S, v_j \in T} cap(v_i, v_j)$$

Definition 5. Flow in S - T cut: Flow in S - T cut is defined as,

$$f([S, T]) = \sum_{\langle v_i, v_j \rangle \in E, v_i \in S, v_j \in T} f(v_i, v_j) - \sum_{\langle v_j, v_i \rangle \in E, v_i \in S, v_j \in T} f(v_j, v_i)$$

The minimum cut is the S - T cut whose capacity is minimum among all possible S - T cuts.

For above definitions, there are some properties.

Property 1. For any S - T cut and flow, the capacity of S - T cut is the upper-bound of the flow across the S - T cut.

Proof

$$\begin{aligned} & cap([S, T]) \\ &= \sum_{v_i \in S, \langle v_i, v_j \rangle \in E, v_j \in T} cap(v_i, v_j) \geq \sum_{v_i \in S, \langle v_i, v_j \rangle \in E, v_j \in T} f(v_i, v_j) \\ &\geq \sum_{v_i \in S, \langle v_i, v_j \rangle \in E, v_j \in T} f(v_i, v_j) - \sum_{v_i \in S, \langle v_j, v_i \rangle \in E, v_j \in T} f(v_j, v_i) \\ &= f([S, T]) \end{aligned}$$

• End of proof

Property 2. For any S - T cut, the value of flow across the cut equals the value of the flow.

Proof

$$\begin{aligned} |f| &= \sum_{v_i \in S} [\sum_{\langle v_i, v_j \rangle \in E} f(v_i, v_j) - \sum_{\langle v_j, v_i \rangle \in E} f(v_j, v_i)] \\ &= \sum_{v_i \in S} [\sum_{\langle v_i, v_j \rangle \in E, v_j \in S} f(v_i, v_j) + \sum_{\langle v_i, v_j \rangle \in E, v_j \in T} f(v_i, v_j) - \sum_{\langle v_j, v_i \rangle \in E, v_j \in S} f(v_j, v_i) - \sum_{\langle v_j, v_i \rangle \in E, v_j \in T} f(v_j, v_i)] \\ &= \sum_{v_i \in S, \langle v_i, v_j \rangle \in E, v_j \in S} f(v_i, v_j) + \sum_{v_i \in S, \langle v_i, v_j \rangle \in E, v_j \in T} f(v_i, v_j) - \sum_{v_i \in S, \langle v_j, v_i \rangle \in E, v_j \in S} f(v_j, v_i) - \sum_{v_i \in S, \langle v_j, v_i \rangle \in E, v_j \in T} f(v_j, v_i) \\ &= \sum_{v_i \in S, \langle v_i, v_j \rangle \in E, v_j \in T} f(v_i, v_j) - \sum_{v_i \in S, \langle v_j, v_i \rangle \in E, v_j \in T} f(v_j, v_i) \\ &= f([S, T]) \end{aligned}$$

• End of proof

Property 3. For any S - T cut and any flow, the capacity of S - T cut is the upper-bound of value of the flow.

Proof

$$\therefore f([S, T]) \leq cap([S, T])$$

$$\begin{aligned} \therefore f([S, T]) &= |f| \\ \therefore |f| &\leq \text{cap}([S, T]) \end{aligned}$$

• End of proof

Then we can draw the following lemma.

Lemma 1 (Equivalence of maximum flow and minimum cut). If f is a flow function of a S - T graph, then the follow statements are equivalent,

- A. f is a maximum flow.
- B. There is a S - T cut that it's capacity equals to the value of f .
- C. The residual graph contains no directed path from source to sink.

Proof

- If B: there is a S - T cut that its capacity equals to the value of f .
- Then A: f is a maximum flow.

The capacity of any S - T cut is upper bound of the value of any flow. So if there is a S - T cut that its capacity equals to the value of flow f , then the value of f is also the upper bound of any flow. That means f is a maximum flow.

- If A: f is a maximum flow.
- Then C: The residual graph contains no directed path from source to sink.

If has a path from source to sink, we augment the flow of this path by an amount equal to the minimum residual capacity of the edges along this path. So the original flow f is not a maximum flow. So if f is a maximum flow, then no augmenting path can exist.

- If C: The residual graph contains no directed path from source to sink.
- Then B: there is a S - T cut that its capacity equals to the value of f .

From the source, the residual graph can be separate into reachable S and non-reachable T , it must be a S - T cut. And there is no arc from S to T in the residual graph. So $f(v_i, v_j) = \text{cap}(v_i, v_j)$ for any arc from S to T and $f(v_j, v_i) = 0$ for arc from T to S . In this case, $\text{cap}([S, T]) = f([S, T])$, as we know $f([S, T]) = |f|$, so $\text{cap}([S, T]) = |f|$.

• End of proof

We briefly introduce some of the basic terminology used throughout the paper. An image that contains $N = n \times n$ pixels, we construct a graph $G = (V, E, W)$ in which each node $v_i \in V$ represents a pixel and every two nodes v_i, v_j representing neighboring pixels are connected by an edge $e_{i,j} \in E$. Each edge has a weight $w_{i,j} \in W$ reflecting the contrast in the corresponding location in the image. We can connect each node to the four or eight neighbors of the respective pixel, producing a graph. Because only background pixels can be determined by AdaBoost, the graph can be partitioned into three disjoint node sets, “ U ”, “ B ” and “ O ”, where $U \cup B \cup O = V$, $U \cap B \cap O = \emptyset$. U means uncertain pixels and B and O mean background and object pixels. Also at first $O = \emptyset$. Finding the most likely labeling translates to optimizing an energy function. In vision and image processing, these labels often tend to vary smoothly within the image, except at boundaries. Because a pixel always has the

similar value with its neighbors, we can model the optimization problem as a MRF. In [3], the authors find the most likely labeling for some given data is equivalent to seeking the MAP (maximum a posteriori) estimate. A graph is constructed and the Potts Energy Model (2) is used as the minimization target.

$$E(G) = E_{data}(v_i) + E_{smooth}(v_i, v_j) \quad (2)$$

$v_i \in V$ $\{v_i, v_j\} \in N$

The graph G contains two kinds of vertices: p -vertices (pixels which are the sites in the associated MRF) and l -vertices (which coincide with the labels and will be terminals in the graph cut problem). All the edges present in the neighborhood system N are edges in G . These edges are called n -links. Edges between the p -vertices and the l -vertices called t -links are added to the graph. t -links are assigned weights based on the data term (first term in Equations 1 reflecting individual label-preferences of pixels based on observed intensity and pre-specified likelihood function) while n -links are assigned weights based on the interaction term (second term in Equation 1 encouraging spatial coherence by penalizing discontinuities between neighboring pixels). While n -links are bi-directional, t -links are un-directional, leaving the source and entering the sink.

For a large image, it will be slow to segment the whole graph straightly. We use a multi-scale nodes aggregation method to construct a pyramid structure over the image. Each procedure produces a coarser graph with about half size.

Algorithm proposed in [3] uses a Graph-Cuts based optimization approach to extract foregrounds from images according to a small amount of user input, such as a few strokes. Previous natural image segmentation approaches heavily rely on the user specified trimap. In our situation, only background trimap can be initialized. We use an iterative Graph-Cuts procedure for image matting inspired by [4].

The object of segmentation is to minimize the energy function (2). The first term reflects individual label-preferences of pixels based on observed intensity and pre-specified likelihood function. The second term encourages spatial coherence by penalizing discontinuities between neighboring pixels. So our goal is to minimize the energy function and make it adapt to human vision system. In [3] authors give the construction of the graph in detail. In order to finish the automatic segmentation, a different way is used to construct the graph in this paper.

We use the pixels inside and outside the rectangle specified by AdaBoost to build two Gaussian mixture models (GMM), one for object and the other for background, which are similar with the method described in [3], [4] and [9]. Each GMM is taken to be a full-covariance Gaussian mixture with $K=5$ components. The Potts Energy Model (2) is equivalent with the Gibbs energy (3)

$$E = \sum_i D(v_i, k_i, \theta, \beta_i) + \sum_{\{v_i, v_j\} \in N} V(v_i, v_j, \beta) \quad (3)$$

$$D(v_i, k_i, \theta, \beta_i) = -\log \pi(k_i, \beta_i) + \frac{1}{2} \log \det \Sigma(k_i, \beta_i) + \frac{1}{2} [v_i - \mu(k_i, \beta_i)]^T \Sigma(k_i, \beta_i)^{-1} [v_i - \mu(k_i, \beta_i)] \quad (4)$$

Where $\pi(\cdot)$, $\mu(\cdot)$, and $\Sigma(\cdot)$ are the mixture weighting coefficients, means and covariance of the 2K Gaussian components for the object and background pixels distributions. So the parameter θ has the form as

$$\theta = \{ \pi(k, \beta), \mu(k, \beta), \Sigma(k, \beta) \} \quad k = 1 \dots K = 5; \beta = 1 \text{ or } 0 \quad (5)$$

The second term of Gibbs energy is

$$V_{\{v_i, v_j\} \in N} (v_i, v_j) = 1 - \exp(-(v_i - v_j)^2 / \sigma^2) \quad (6)$$

We set σ empirically to be 0.2 in our system.

We use a similar way with [4] to minimize the Gibbs energy which can guarantee the convergence. A more detail can be found in [4].

4 Experiments

The detection results and the iterative segmentation results are given in Fig. 2. In Fig .2 (a) is the car segmentation result. Fig .2 (b), (c) are the segmentation

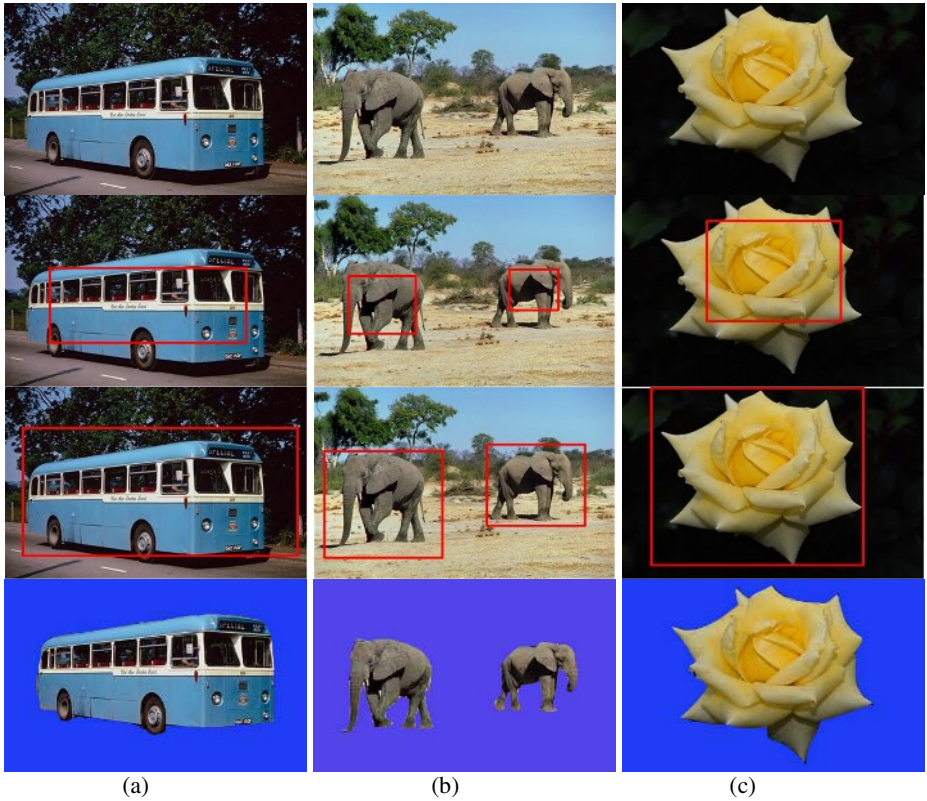


Fig. 2. The detection results using AdaBoost and segmentation results by iterative Graph-Cuts respectively. (a) is the bus result. (b) is the elephant result. (c) is the flower result.

experiments for nature color object images. The segmentation results are good enough for some practical applications and the running time is fast enough to meet the real-time demands because we use multi-scale method to reduce the number of nodes. There are some methods for nodes aggregation. We use a simply but efficient method. For more accurate result we can develop new methods for nodes aggregation. The average ratio of running time of node aggregation method is about 20 times faster than non-nodes aggregation. In addition, we also apply our segmentation method to interest region-based image retrieval system and get a satisfying performance.

5 Conclusions

This paper proposes a machine learning based image segmentation method. Compared to previous methods, our scheme is automatic and accurate. Also the AdaBoost learning concept is introduced to segmentation problems. In the future, we plan to develop new method to reduce the training examples.

Acknowledgments

This work has been supported by NSFC Project 60573182, 69883004 and 50338030.

References

1. Reese, L.: Intelligent Paint: Region-Based Interactive Image Segmentation. Master's thesis. Department of Computer Science, Brigham Young University, Provo, UT, (1999)
2. Mortensen, E. N., AND Barrett, W. A.: Interactive segmentation with intelligent scissors. *Graphical Models and Image Processing*, 60, 5 (1998) 349–384
3. Boykov, Y, And Jolly, M.: Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In *Proceedings of ICCV (2001)* 105-112
4. Rother, C, Blake, A, And Kolmogorov, V.: Grabcut-interactive foreground extraction using iterated graph cuts. In *Proceedings of ACM SIGGRAPH (2004)*
5. D. Comanicu, P. Meer.: Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Machine Intell*, May (2002)
6. J. Shi and J. Malik.: Normalized Cuts and Image Segmentation. *Proc. IEEE Conf. Computer Vision and Pattern Recognition (1997)* 731-737
7. J. Shi and J. Malik.: Normalized cuts and image segmentation. *PAMI*, 22(8) (2000) 888–905
8. P. Viola and M. Jones.: Rapid object detection using a boosted cascade of simple features. *Proc. CVPR*, vol. 1 (2001) 511-518
9. Y. Li, J. Sun, C.K. Tang and H.Y. Shum.: Lazy snapping. *Proc. of ACM SIGGRAPH (2004)* 303-308

Accurate Contour Detection Based on Snakes for Objects with Boundary Concavities

Shin-Hyoung Kim, Ashraf Alattar, and Jong Whan Jang

PaiChai University, Daejeon, The Republic of Korea, 302-435
{zeros, ashraf, jangjw}@pcu.ac.kr

Abstract. In this paper we present a snake-based scheme for efficiently detecting contours of objects with boundary concavities. The proposed method is composed of two steps. First, the object's boundary is detected using the proposed snake model. Second, snake points are optimized by inserting new points and deleting unnecessary points to better describe the object's boundary. The proposed algorithm can successfully extract objects with boundary concavities. Experimental results have shown that our algorithm produces more accurate segmentation results than the conventional algorithm.

1 Introduction

Object segmentation is the basis for many important applications such as machine vision, video games, 3D interactive TV, interactive multimedia systems, and image and video coding [1]. In fact, the recently developed MPEG-4 standard [2] is an object-based image and video coding algorithm that requires object segmentation in the first stage.

Over the past two decades, various object segmentation schemes have been developed for extracting an object from an ordinary image. One of the approaches that received significant research attention, and has been extensively used is the active contour (snake) algorithm [3]-[6]. A snake is an energy-minimizing spline guided by internal forces that preserve its characteristics, and external forces that pull it toward image features such as lines and edges. The original snake model suffers from challenges such as the difficulty in progressing into boundary concavities. Xu *et al.* proposed the GVF (gradient vector flow) snake [5] to handle the concavities problem. The GVF method uses a spatial diffusion of the gradient of the edge map of the image instead of using the edge map directly as an external force. Although the method has the advantages of insensitivity to initialization and the ability to move into boundary concavities, it can not handle gourd-shaped concavities. This is due to the concentration of GVF energy in the neck of the gourd.

In this paper we present a snake-based method for object segmentation addressing the above challenges through a modified snake model with optimized points. We propose new modifications to the energy formulation of the snake model to handle the concavity problem. Optimization of snake points is addressed by managing the insertion of new points and deletion of unnecessary points to better describe the object's boundary.

This paper is organized as follows. Section 2 covers a background on the snake model. Our proposed method for extracting objects with boundary concavities using an optimized snake is presented in Section 3. In Section 4 we show results of the simulation for evaluating the performance of our method, and the conclusions are given in Section 5.

2 The Snake Model

The snake algorithm was first introduced by Kass *et al.* [3]. A snake is the energy-minimizing spline guided by internal constraint forces and influenced by external image forces that pull it toward features such as lines and edges.

In the discrete formulation of a snake, the contour is represented as a set of snake points $v_i = (x_i, y_i)$ for $i=0, \dots, M-1$ where x_i and y_i are the x and y coordinates of a snake point, respectively and M is the total number of snake points. The total energy of each point is typically expressed as a sum of individual energy terms:

$$E_{snake}(v_i) = E_{int}(v_i) + E_{ext}(v_i) \quad (1)$$

The function contains two energy components. The first component is called the *internal energy* and it concerns contour properties such as curvature and discontinuity. The second component is the *external energy* which is typically defined by the gradient of the image at a snake point.

3 Proposed Algorithm

Our proposed algorithm modifies the conventional snake model, and introduces a new mechanism for the insertion and/or deletion of snake points as necessary.

3.1 Proposed New Snake Energy Function

In this section, we describe our special snake energy function defined for a 2D image. Internal energy is typically expressed in terms of two terms: *continuity energy*, and *curvature energy*. Our modifications to these two terms and to the external energy term will be explained in the following subsections

Internal Energy Terms. The first internal energy term is the continuity energy term. The main role of the conventional continuity energy term is to make even spacing between the snake points by minimizing the difference between the average distance and the distance between neighboring snake points. In this arrangement point spacing is globally and uniformly constrained by the average distance. This is not suitable for objects with concavities because the distance between snake points should be relaxed in boundary concavities. Therefore, in our method, point spacing in boundary concavities is allowed to differ from other parts of the snake. We define the normalized continuity energy as follows:

$$E_{con}(v_i) = \frac{\| \|v_{i-1} - v_i\| - \|v_i - v_{i+1}\| \|}{con_{max}} \tag{2}$$

where con_{max} is the maximum value in the search neighborhood. The proposed continuity energy can make even spacing between neighboring snake points in concave parts of the object’s boundary.

For curvature energy, the goal is to control the smoothness of the contour between neighboring snake points. The way in which this term is formulated affects the ability of snake points to progress into boundary concavities, and conventional snake algorithms show poor performance in this aspect. In this paper we present a new curvature energy solving the problem based on the Frenet formula. As depicted in Fig. 1(a), let $\Psi : I \rightarrow \mathfrak{R}^3$ be a unit-speed curve. Then, $T = \Psi'$ is the unit tangent vector field on Ψ , and $N = T' / \|T'\|$ is the principal normal vector field of Ψ ; where $\|T'\|$ is the length of the curvature. The vector field $B = T \times N$ on Ψ is the binormal vector field of Ψ , and the sign of the z-dimension of B is positive if B is upward and is negative if it is downward. Therefore, we consider the sign of the binormal vector. In 2D the sign of the binormal vector $B(v_i)$ can be obtained using the cross product of the two vectors $T(v_i)$ and $N(v_i)$ as follows:

$$B(v_i) = T(v_i) \times N(v_i) = \begin{vmatrix} x_i^T & y_i^T \\ x_i^N & y_i^N \end{vmatrix} = (x_i^T y_i^N - y_i^T x_i^N) \bar{e}_z \tag{3}$$

where (x_i^T, y_i^T) and (x_i^N, y_i^N) are the x and y coordinates of the tangent vector $T(v_i)$ and normal vector $N(v_i)$ at the current point v_i , respectively. \bar{e}_z is a unit vector of the z-component. In the discrete formulation of a snake point, $T(v_i)$ and $N(v_i)$ are defined as follows :

$$T(v_i) \approx v_{i+1} - v_i \tag{4}$$

$$N(v_i) \approx v_{i-1} - 2v_i + v_{i+1} \tag{5}$$

In the case that a snake point is outside the object (as in Fig. 1(b, c, d)), the movement of a snake point can be described as follows:

- i) When $B(v_i)$ is negative, as the case in Fig. 1(b), v_i moves in the opposite direction of $N(v_i)$ seeking a location with maximum value of $\|N(v_i)\|$.
- ii) When $B(v_i)$ is positive, as the case in Fig. 1(c), the snake point v_i must take the direction of $N(v_i)$ to minimize curvature and to converge on the boundary of the object.

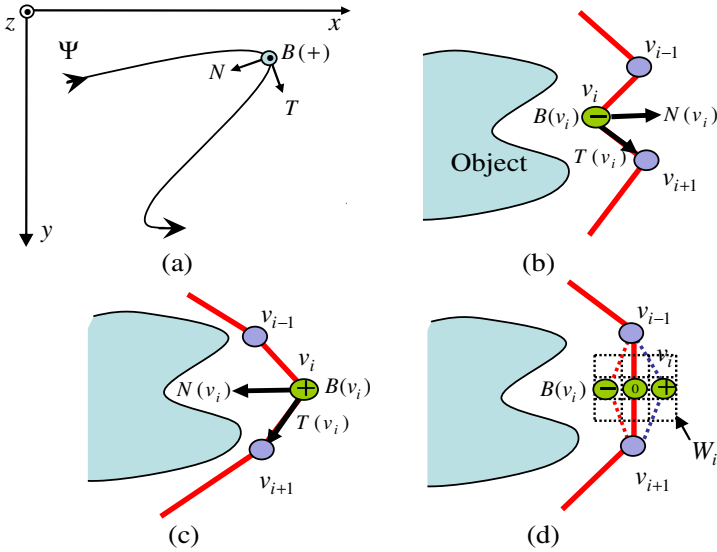


Fig. 1. Movement of snake points based on binormal vector

iii) In the case of Fig. 1(d), the value of $\|N(v_i)\|$ at v_i is zero because the point is on a straight line, and, therefore, $B(v_i)$ has a zero vector at v_i . In this case, we examine neighboring points located within the window W_i . A candidate point with a negative $B(v_i)$, such as the point to the left of v_i , maximizes the $\|N(v_i)\|$ and thus v_i must move to that location.

The normalized curvature energy is defined as Eq. (6) [6]. It is further weighted by the constant λ , as given in Eq. (7).

$$E_c(v_i) = \left\| \frac{T(v_i)}{\|T(v_i)\|} - \frac{T(v_{i-1})}{\|T(v_{i-1})\|} \right\| \tag{6}$$

$$E_{cur}(v_i) = \lambda E_c(v_i) \tag{7}$$

The constant λ can be set for different applications, relative to the sign of $B(v_i)$ on the contour. In our work, λ is set to +1 when the sign of $B(v_i)$ is positive and -1 otherwise.

External Energy Terms. The external energy is defined as $-\|\nabla[G_\sigma(v_i) * f(v_i)]\|/e_{\max}$, where $G_\sigma(v_i)$ is a two-dimensional Gaussian function with standard deviation σ and ∇ is the gradient operator. $f(v_i)$ is the image intensity function, and e_{\max} is the maximum value in the search neighborhood.

Total Snake Energy. The proposed snake energy function is defined as follows:

$$E_{snake}(v) = \sum_{i=0}^{M-1} (\alpha E_{con}(v_i) + \beta E_{cur}(v_i) + \gamma E_{ext}(v_i)) \tag{8}$$

The parameters α , β and γ are set to 1.0, 1.0 and 1.2 respectively.

3.2 Optimizing the Number of Snake Points

Following the convergence of snake points in the first step to the boundary of the object, additional points are inserted and unnecessary points are deleted to better describe the boundary concavities. This step is explained as follows.

Point Insertion and Deletion. The decision to insert a new point between two points v_i and v_{i+1} depends on the $\|N(v_i)\|$ and $\|T(v_i)\|$ at v_i . If $\|N(v_i)\|$ is above a threshold th_N and $\|T(v_i)\|$ is above a threshold th_T , an additional point is inserted. This condition can be expressed by the Boolean expression:

$$\|N(v_i)\| \geq th_N \text{ AND } \|T(v_i)\| \geq th_T \tag{9}$$

and the newly inserted point is defined as $c_i = (v_i + v_{i+1})/2$.

On the other hand, when the $\|N(v_i)\|$ at a point v_i is small enough to fall below th_N , the point is likely to be on a straight line. In such case, v_i can be safely removed because the boundary can be described well enough by the two snake points adjacent on each side.

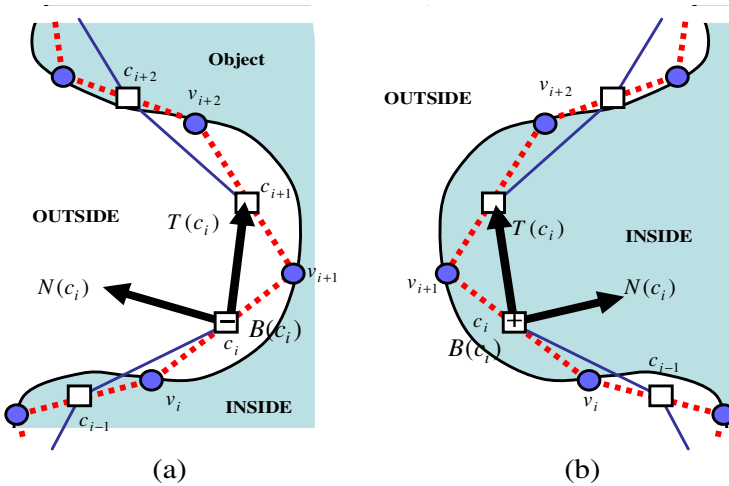


Fig. 2. Movement of inserted points in convex/concave segments of the object's boundary

Movement of Inserted Points. Fig. 2 illustrates the decision controlling the movement of inserted points taking into consideration whether the inserted point is inside or outside the object. As shown in Fig. 2, the sign of $B(c_i)$ is negative in concave segments and positive in convex segments. To decide the movement of an inserted point, we first check the sign of its binormal vector $B(c_i)$ relative to previous and next inserted points. If $B(c_i)$ is negative, then the point is outside, and in that case we apply exactly the same scheme explained in section 3.1.1 above. On the other hand, if the inserted point turns out to be inside (i.e., $B(c_i)$ is positive), we apply the same scheme except that we reverse the signs for λ .

4 Experimental Results

To verify the performance of our algorithm a set of experiments has been performed. The algorithm was coded in Visual C++ 6.0, and the experiments were performed on a Pentium IV machine with 1GByte of memory running at 3GHz clock speed. We used binary and color real images with 320×240 image size.

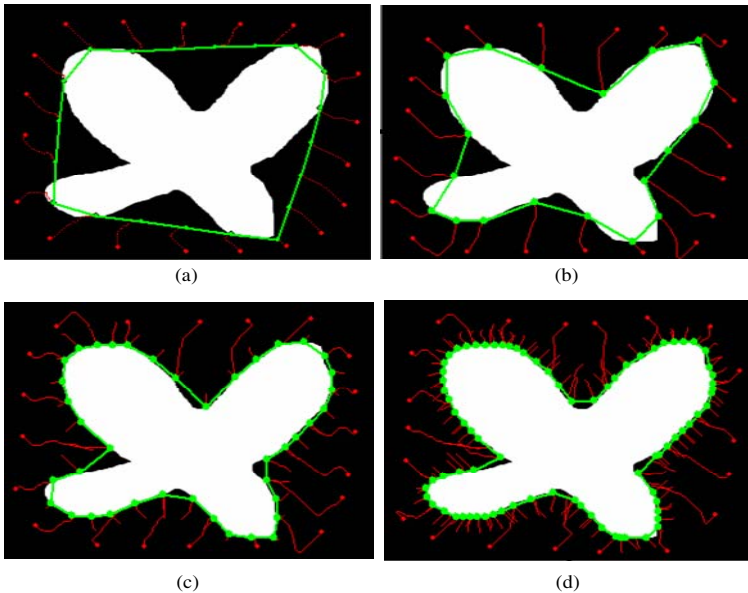


Fig. 3. Results of conventional algorithm and the proposed algorithm. (a) Greedy snake [3]. (b)-(d) Proposed algorithm.

The criteria used to verify the accuracy of the estimated snake region R_{est} , compared with the original object region R_{ori} is *Relative Shape Distortion*, $RSD(R)$, which is defined as:

$$RSD(R) = \left(\sum_{(x,y) \in f} R_{ori}(x,y) \oplus R_{est}(x,y) \right) / \sum_{(x,y) \in f} R_{ori}(x,y) \quad (10)$$

where the \oplus sign is the binary XOR operation. The simulation process and its results are illustrated in Figs. 3 and 4, and the performance comparison in terms of RSD is listed in Table 1.

Fig. 3 shows results of an experiment on a binary image for an object with boundary concavities. Points of the greedy snake failed to proceed into the concave parts of the object's contour, Fig. 3 (a), but in our proposed algorithm the snake points converged better onto boundary concavities, Fig. 3 (b)-(d). Fig. 3(b) shows the result of the first iteration of point insertion and convergence. Results of the second and third iteration are shown in Fig. 3 (c) and (d), respectively. Convergence results improve as more points are inserted.

Fig. 4 shows results of an experiment on a binary image for an object with a gourd-shaped concavity. The greedy snake could not detect the boundary of the object, Fig. 4 (a), and the GVF snake points failed to proceed beyond the neck of the gourd, Fig. 4 (b). With our proposed algorithm the snake points converged better onto the boundary concavity in only four iterations, Fig. 4 (c). Fig. 4 (d) shows the results for our proposed algorithm on an ordinary image. We performed an additional test involving our algorithm and the GVF snake on an MR image of the left ventricle of a human heart as shown in Fig. 5. Both snakes were initialized inside of the object (ventricle). Our algorithm performed a complete segmentation, Fig. 5 (b), while the GVF segmented the left side only. The GVF snake can segment the whole ventricle only if it were initialized in the space between the papillary muscles (the bumps that protrude into the cavity).

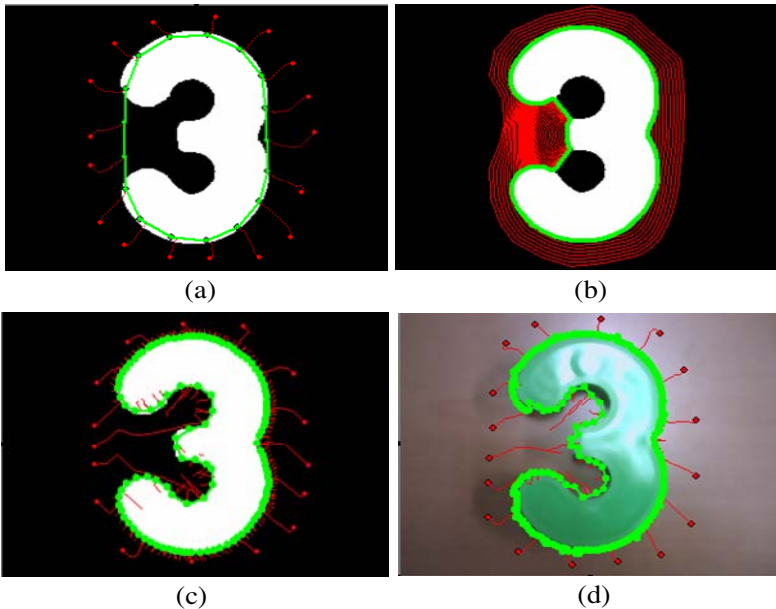


Fig. 4. Results of experiment on the gourd-shaped object. (a) Greedy snake, (b) GVF snake, (c) and (d) Proposed algorithm.

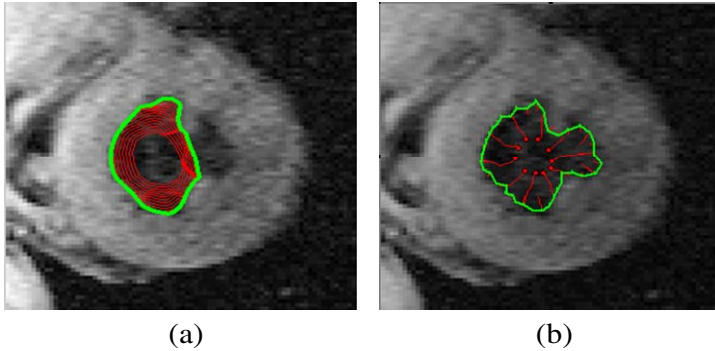


Fig. 5. Results of experiment on the image of human heart ($\sigma=2.5$)

Fig. 6 shows the relationship between the number of initial snake points and the number of iterations. When the snake is initialized with a sufficient number of points, convergence is achieved with less iterations and point insertion. However, a snake that is initialized with a few number of points, requires an increased number of iterations with point insertions. The conventional methods in [3] and [4] do not optimize the number of snake points, and thus can not handle boundary concavities which require more snake points.

Performance comparison between the proposed algorithm and the conventional algorithm in terms of *RSD* is summarized in Table 1. Table 2 summarizes the results shown in Fig. 6, and gives performance measures in terms of *RSD*.

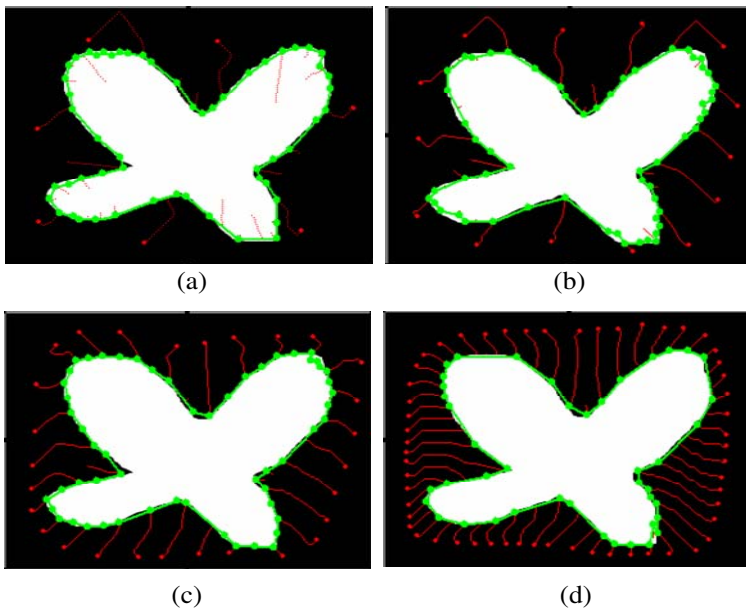


Fig. 6. Results of experiment on the relations between the number of initial snake points and the number of iterations

Table 1. Performace comparison in terms of *RSD*

| | Results of Fig. 4 | | |
|------------|-----------------------|-----------------------|------------------------|
| | (a) Greedy snake | (b) GVF snake | (c) Proposed algorithm |
| <i>RSD</i> | (5839/15143) 0.385 | (3880/15143) 0.256 | (128/15143) 0.008 |

Table 2. Performance comparison in terms of *RSD*

| | Number of initial snake points | Number of iterations (insertion & deletion) | Final number of snake points | <i>RSD</i> |
|-----|--------------------------------|---|------------------------------|----------------------|
| (a) | 7 | 4 | 55 | (459/25907) 0.017 |
| (b) | 14 | 3 | 58 | (473/25907) 0.018 |
| (c) | 28 | 2 | 56 | (311/25907) 0.012 |
| (d) | 56 | 1 | 43 | (288/25907) 0.011 |

5 Conclusions

In this paper, we have presented a new snake-based segmentation algorithm for objects with boundary concavities. Our algorithm extends and improves the conventional snake model. The developed algorithm was tested and showed successful results in extracting objects with boundary concavities.

In the proposed new snake algorithm, the movement of snake points is determined using the sign of the cross product of the tangent and normal vectors. In addition, we proposed a method for optimizing the number of snake points to better describe the object's boundary. In consequence, we can solve the problem which occurs with gourd-shaped boundary concavities. Performance has been evaluated by running a set of experiments using 2D image data having objects with varying degrees of boundary concavity. When compared with conventional snake algorithms, our method has shown a superior object segmentation capability in terms of accuracy. Further research work is being considered to follow up from object segmentation to object tracking in video sequences.

References

1. M. Bais, J. Cosmas, C. Dosch, A. Engelsberg, A. Erk, P. S. Hansen, P. Healey, G.K. Klungsoeyr, R. Mies, J.R. Ohm, Y. Paker, A. Pearmain, L. Pedersen, A. Sandvancd, R. Schafer, P. Schoonjans, and P. Stammnitz : Customized Television: Standards Compliant Advanced Digital Television, IEEE, Trans. Broad, Vol. 48, No.2, (2002) 151-158

2. ISO/IEC JTC/SC29/WG11/W4350: Information Technology - Coding of Audio-Visual Objects Part2: Visual, ISO/IEC 14496-2, (2001)
3. M. Kass, A. Witkin, and D. Terzopoulos : Snake: Active Contour Models, Int'l J. Computer Vision, Vol. 1, No. 4, (1987) 321-331
4. D. J. Williams and M. Shah : A Fast Algorithm for Active Contours And Curvature Estimation, Computer Vision, Graphics, and Image Processing, Vol. 55, (1992) 14-26
5. C. Xu and J. L. Prince : Snakes, Shapes, and Gradient Vector Flow, IEEE Trans. Image Processing, Vol. 7, No. 3, (1998) 359-369
6. S. H. Kim and J. W. Jang : Object Contour Tracking Using Snakes in Stereo Image Sequences, International Conference on Image Processing 2005, Vol. 2, (2005) 414-417

Graph-Based Spatio-temporal Region Extraction

Eric Galmar and Benoit Huet

Institut Eurécom, Département multimédia, Sophia-Antipolis, France
{galmar, huet}@eurecom.fr

Abstract. Motion-based segmentation is traditionally used for video object extraction. Objects are detected as groups of significant moving regions and tracked through the sequence. However, this approach presents difficulties for video shots that contain both static and dynamic moments, and detection is prone to fail in absence of motion. In addition, retrieval of static contents is needed for high-level descriptions.

In this paper, we present a new graph-based approach to extract spatio-temporal regions. The method performs iteratively on pairs of frames through a hierarchical merging process. Spatial merging is first performed to build spatial atomic regions, based on color similarities. Then, we propose a new matching procedure for the temporal grouping of both static and moving regions. A feature point tracking stage allows to create dynamic temporal edges between frames and group strongly connected regions. Space-time constraints are then applied to merge the main static regions and a region graph matching stage completes the procedure to reach high temporal coherence. Finally, we show the potential of our method for the segmentation of real moving video sequences.

1 Introduction

Multimedia technologies are becoming important in many aspects of our now-day lives. Processing of huge amount of raw data requires efficient methods to extract video contents. Achieving content-based functionalities, such as search and manipulation of objects, semantic description of scenes, detection of unusual events, and recognition of objects has driven intensive research over the past years. To exploit video contents, shots must be decomposed into meaningful objects which are composed of space time regions. This process is called video *indexing*.

Unsupervised extraction of video objects is generally based intensively on motion information. Two strategies are generally adopted. The first one searches for homogeneous colored or textured regions, and then groups the regions that undergo similar motion [1]. The second strategy performs motion estimation to yield coherent moving regions, then groups adjacent regions basing on color cues [2]. Sophisticated methods use robust motion estimation to deal with multiple objects and motion. However, tracking becomes difficult in case of non-rigid or fast motion, and the apparition and disappearance of new object models cannot be

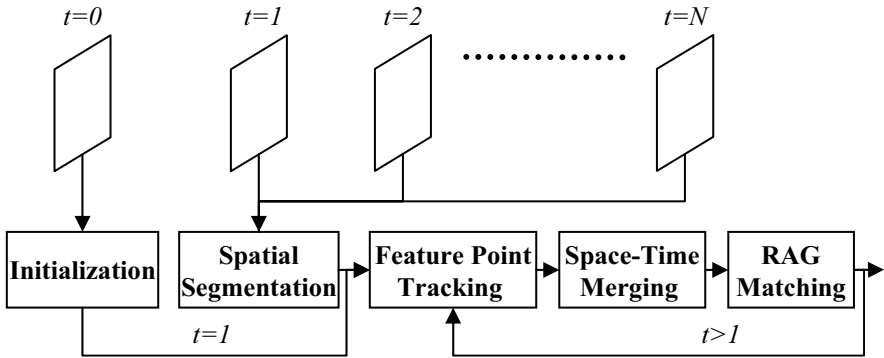


Fig. 1. Scheme of the overall segmentation process

integrated easily. To overcome these problems, two alternative approaches have been proposed, spatio-temporal segmentation and graph-based region merging.

The first category searches for meaningful volumes inside a block of frames to improve temporal coherence. A feature clustering approach is described in [3]. Elementary objects are represented as color patches with linear motion, called video strands. Space-time features describing color, position, and dynamics are extracted for each pixel. Therefore, video shot can be mapped to a 7D feature space representative of the strands. A hierarchical mean-shift technique is then employed to cluster pixels and build object hierarchy jointly. A probabilistic representation scheme is proposed in [4]. The video sequence is modeled by a succession of spatial gaussian mixture models (GMM). GMMs are initialized via EM algorithm in the first frame, then are updated on subsequent frames. Appearance of new objects is handled by thresholding a likelihood map and creating new models from unlabeled connected pixels. This allows the method to track coherent regions with complex motion patterns.

These spatio-temporal approaches are robust at the expense of memory bandwidth and computational cost when the shot duration becomes important. In region merging approaches, the segmentation is first initialized on each frame from an image segmentation technique. Popular algorithms are derived from watersheds [5] or color quantization [6] and yield to segments with small color variations. Spatial and temporal merging are then achieved by labeling or matching.

Unlike pixel-based approaches, region-based graphs use more reliable region information and allow to represent various relationships between regions. In [7], a set of spatial region adjacency graphs (RAG) is built from a shot section, and then the optimal partition of the whole graph is found according to a global cut criterion. However, the method suffers from the instability of image segmentation on different frames. To make matching easier, Gomila et al. [8] reduce the difference between consecutive RAGs by a region splitting process. For each frame, a hierarchy of segmentations is generated through a multiscale image segmentation method. Closer RAGs are then built by checking if missing regions

are edited in the decomposition. Then, the graphs are iteratively merged using a relaxation technique.

The proposed approach is closely related to both space-time and graph-based region-merging. It aims at decomposing video shots into spatio-temporal regions. Unlike other methods, we give particular attention to the stability of the projected spatial segmentation for both static and moving regions, in prospect of object detection and region-based shot representation. This paper is organized as follows. Section 2 provides an overview of the proposed algorithm and motivations for our approach. Section 3 introduces the efficient graph-based merging algorithm used at different stages of the process. In section 4, we describe the temporal merging procedure. Finally, experimental results illustrate the application of our algorithm to real video sequences in section 5.

2 Overview of the Proposed Approach

Extraction of space-time regions can be difficult when video objects show strong variations in color, texture or motion. Unfortunately, these features are common in real video sequences. In this work, we design an incremental scheme to reduce the complexity of region grouping and matching tasks.

A block diagram of our system is shown figure 1. The segmentation is initialized on the first frame of the shot from coherent spatial regions and defines the spatial level of details of the segmentation. A graph-based segmentation algorithm is used for this purpose. Then, the method iteratively processes frame pairs. The regions are grouped temporally in three steps. The first stage builds slightly oversegmented spatial regions in the new frame, so that these new regions corresponds to a partition of the previous segmentation. Instead of using motion compensation, we track a population of feature points to create dynamic temporal edges between regions. This allows us to group with high confidence static and moving regions that are strongly connected. We then complete the temporal linkage of static regions using local edges, under space-time merging constraints. At this stage, the segmentation maps become close and region neighborhoods can be compared. Finally, we test the validity of new regions by comparing locally RAGs between frame pairs.

With this design, we achieve incremental merging with strong rules, reaching progressively temporal coherence for various region types.

3 Spatial Merging

In this section, we present the efficient graph segmentation algorithm introduced in [9]. Then we describe how we apply it to initialize regions and how we adapt it for partial segmentation of new frames.

3.1 Efficient Graph Based Segmentation

Let $G = \{V, E\}$ be a weighted undirected graph. Each vertex is a pixel. The algorithm aims to decompose G into a partition $S = \{C_1, C_2, \dots, C_k\}$ of G , where

each component is a minimum spanning tree (MST). The procedure is similar to Kruskal’s algorithm, with addition to a merging criterion to limit the grouping of components. At each step, two components are merged if the minimum edge connecting them is weaker than the maximum edges of the components plus a tolerance depending on the component size. Therefore, fewer merge are done when the region size increases. Thanks to the adaptive rule, the algorithm is sensitive in areas of low variability whereas it remains stable in areas of high variability preserving both local and global properties.

We apply this algorithm to segment the first image by building the graph on a pixel grid, so that the algorithm is fast and subgraphs correspond to spatially connected regions. In the experiments, the weights are built using color distance.

3.2 Edge Constrained Segmentation

Using directly the procedure described in 3.1 to initialize regions in any frame does not work, since the segmentation may differ substantially from one frame to another. To avoid resegmentation, we adapt the method so that S_t is over-segmented compared with S_{t-1} . To this aim, we use an edge detection map \mathcal{C}_t to discard possible region boundaries from the merge. Thus, the propagation is done in areas of low-variability, resulting in more homogeneous components.

Edge-constrained segmentation and original image segmentation can be compared in figure 2. We can see that the constrained method (c) results in a decomposition, or oversegmentation of the unconstrained one (b). In addition, since we use Canny detection, edges with local intensity variations are also pruned so that the components are more homogeneous.



Fig. 2. (a) Input Image. (b) Unconstrained image segmentation used as initialisation. (c) Edge-constrained initialisation of the new regions.

4 Temporal Grouping

In this section, we first describe the temporal grouping of regions based on dense feature points and space-time constraints. Then, we show how RAGS are employed to check efficiently the stability of the regions.

4.1 Feature Point Matching

The regions in the previous segmentation S_{t-1} have various shape, size and possibly non-rigid motion. In addition, regions might be partially occluded in

the new frame, so that one region can have several matches in the next frame. In this case, traditional motion compensation cannot be used. Our solution is to group new oversegmented regions by spreading a population of feature point trackers \mathbf{P}_f . In this way, no hypothesis is made on motion models and we avoid optical flow computation on full regions.

Feature point trackers have been proposed by Tomasi et al. [10]. Good feature points are extracted from corners or textured regions. However, these points are likely to correspond to region borders, thus hampering the matching between regions. Therefore, we rather consider flat points that we can expect to lie reliably inside regions, at the expense of motion precision. Feature points are then tracked using a block matching algorithm. Figure 3 shows typical feature point detection and tracking. We can see that feature points are concentrated in homogeneous areas (fig. 3a). Even if some tracked points are inaccurate (fig. 3b), they can be considered as outliers in the statistical distribution of the points. We explain how we use these points for region grouping in the next section.

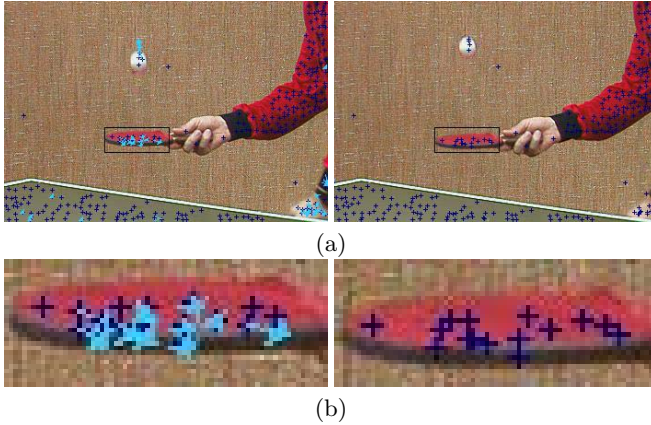


Fig. 3. (a) Distribution of feature point matches. (b) Feature points inside the racket. Arrows represent the estimated displacement.

4.2 Region Grouping with Feature Points

Feature points matches described in the previous section can be viewed as potential inter-frame edges between pair of regions. We construct a 3D graph $G_T = \{V_T, E_T\}$ between two consecutive frames. The node set E_T contains two subsets of regions A and B generated from S_{t-1} and S_t . The edge set contains inter-frame arcs generated from feature point pairs. Due to possible high variations (section 4.1), grouping based on single linkage will no be relevant. We consider instead robust grouping analysing statistical properties of connections between subsets A and B .

The procedure (fig.4) is based on a sequence of tests. We first simplify the graph and prune weak connections between A and B with the M_{strong} test.

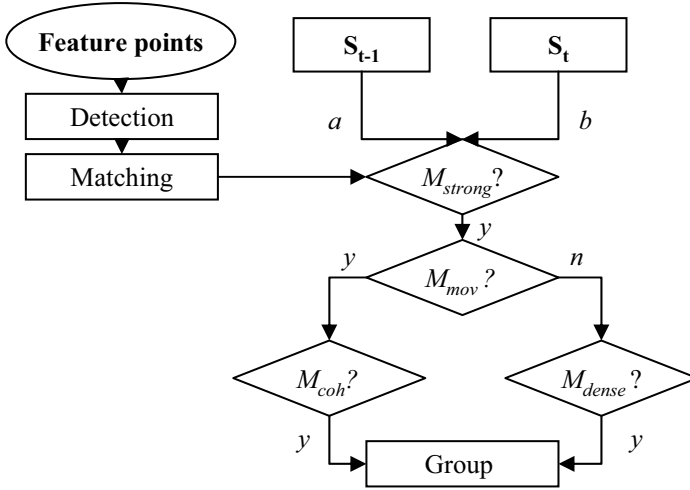


Fig. 4. The temporal region grouping scheme. Feature points are first detected and matched. Then, the temporal merging of strongly connected regions is performed using a hierarchy of tests.

Second and third tests (M_{mov} and M_{coh}) verify if region couples undergo significant and similar motion. This helps to detect potential splitting regions. Finally, we further check the homogeneity of region couples (M_{dense}) for static regions. Denote by $a \in A$ and $b \in B$ two candidate regions for grouping.

M_{strong} : Two regions a and b are strongly connected if there is a significant proportion of arcs linking a to b . Formally, we compare the cut between a and b to the degrees of a and b . The test is accepted if :

$$cut(a, b) > \alpha \min deg(a), deg(b) \quad (1)$$

α is fixed to $\alpha = 0.5$ in our experiments. In other words, if edges are given equal weights, the test is verified when at least half edges of either a or b connects a to b . Once all regions have been tested, weak edges that do not satisfy the condition are pruned.

M_{mov} : From the displacement of feature points, we deduce information on region motion. For this purpose, we map the points to a velocity space $\mathcal{D} = [d_n, \beta d_\theta]$ where d_n is the displacement norm and d_θ is the motion orientation. β controls the influence of orientation information with respect to motion speed. In case that there is substantial background or camera motion, the displacements are compensated with the mean velocity of the complete set of points. The test separates moving regions from static regions. The moving condition is given by

$$d_n(a) > d_{mov} \quad (2)$$

where d_{mov} is a minimum substantial displacement. Default value is $d_{mov} = 3$ in all our experiments.

M_{coh}: If a and b are moving regions, they must undergo coherent motion to be grouped. A simple measure is to compare the variance of the velocity distributions of a , b and $a \cup b$. The test $M_{coh}(a, b)$ is given by

$$tr(C_{a \cup b}) < \gamma(tr(C_a) + tr(C_b)) \quad (3)$$

where C_a denotes the covariance matrix of the velocity points of a . The test favors the creation of new moving regions in S_t when one region in S_{t-1} is matched to ones with different motions. In this way, we handle apparition of new moving regions.

M_{dense}: When either region has no motion, we further check if they have comparable homogeneity. We characterise this feature by the density of feature points between regions, since each point corresponds to a local maximum of homogeneity. The density η_a of one region a is estimated by

$$f_a = \frac{card(a \times V_T)}{size(a)} \quad (4)$$

As the density is variable over the regions, we use a statistical proportion test for that purpose. Let's consider two parent populations P_a and P_b representing space-time regions and their final proportion of points p_a and p_b . a and b are samples drawn from P_i and P_j . f_a and f_b are estimations of p_a and p_b .

We consider the following hypotheses

$$\begin{aligned} H_0 : p_a &= p_b \\ H_1 : p_a &\neq p_b \end{aligned} \quad (5)$$

Assuming normal laws for P_a and P_b , it is possible to check if we can accept H_0 with a significance level α [11].

At the end of the process, temporal grouping has been performed reliably on homogeneous moving regions. To group more textured areas on the sequence, the population of seed points will be increased inside regions finally created in S_t , i.e. if they have not been matched in S_{t-1} . In this way, the tracked points will focus progressively on the regions of interest.

4.3 Grid-Based Space-Time Merging

We complete the segmentation S_t by a space-time merging technique applied on the unmatched regions. The method is an adaptation of the efficient graph algorithm discussed in section 3.2 for grouping components spatially and temporally. We construct a space-time pixel grid on a 3D volume bounded by two successive frames. As in [9] each component C_i is characterized by its internal variation, which represents a p -quantile of the weight distribution of the edges inside C_i . However, this turns out to be too complex in practice and we use the

mean weight μ_i of C_i as a measurement. When comparing two components C_i and C_j , a new space-time merging rule is applied to examine both local and global properties of the grouping:

$$\|\mu_i - \mu_j\| < \tau_G \text{ and } \max(W_L) < \tau_L \tag{6}$$

where

$$\tau_G = \max(T_G, p_g \min(\mu_i, \mu_j)) \tag{7}$$

$$\tau_L = \min(T_L, \mu_i) \tag{8}$$

τ_L and τ_G are local and global adaptive thresholds. Default parameters are $T_G = 10$, $p_g = 0.3$, $T_L = 5$ in all experiments. For local properties, we define a four edge neighborhood W_L (fig. 5a). The neighborhood is considered as homogeneous if the maximum weight is weak compared to the variability μ_i and T_L . Small values of T_L limit grouping in inhomogeneous areas. In this way, we do not merge component from edges with high variability. For global properties, we check if the components have similar homogeneity. For regions with strong homogeneity, we consider directly the distance between μ_i and μ_j . For more variable components, a tolerance p_g is accepted on the relative error between μ_i and μ_j . Small values of T_G and p_g limit the temporal variation of the components.

Thus, by combining these two aspects, the merging occurs in space-time areas of low local variability on globally coherent components.

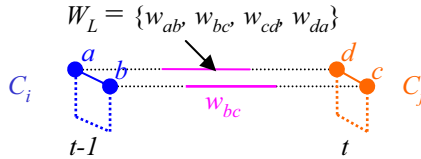


Fig. 5. Space-time grid based merging and local neighborhood W_L

4.4 Subgraph Matching

The last step in the process is to confirm the creation of new regions by analysing region neighborhoods at time $t - 1$ and t . Thanks to the previous merging steps, segmentations S_t and S_{t-1} are sufficiently close to be compared. We consider, as in section 4.2, a 3D graph on a volume bounded by two successive frames. The graph contains region adjacency graphs (RAG) R_{t-1} from S_{t-1} and R_t from S_t . It also includes inter-frame edges corresponding to the temporal grouping of regions. For each node v in R_t , we define its neighborhood subgraph $G_t^N(v)$ as the smallest subgraph containing all its adjacent nodes $u \in R_t$. Let v_n be a node from a new region in R_t and $u \in G_t^N(v_n)$ connected to a node $u' \in R_{t-1}$. Let consider a distance measure $d(u, v)$ between two nodes. We denote by u'' a node in $G_t^N(u')$. u'' and v_n are matched temporally if

$$d(u'', v_n) < \min_{z \in G_{t-1}^N(u'')} d(u'', z) \tag{9}$$

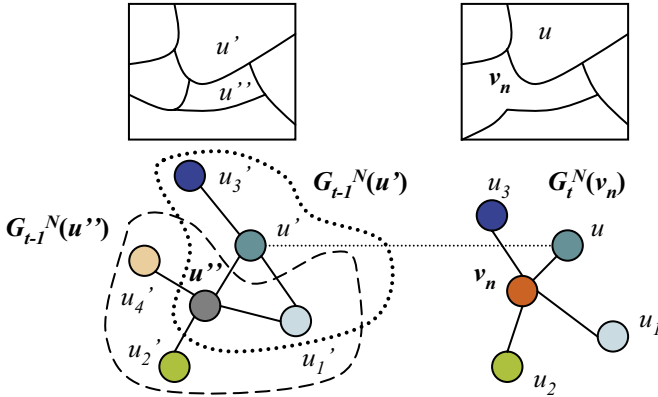


Fig. 6. Neighborhood subgraphs for matching new nodes v_n . For each node $u \in G_t^N(v_n)$, the neighborhood of u in RAG_{t-1} , $G_{t-1}^N(u')$ is examined. Lost nodes u'' are then retrieved by comparing v_n to adjacent nodes of u'' in $G_{t-1}^N(u'')$.

Equation 9 checks if an untracked node in R_{t-1} can be matched with a new node in R_t in the proximate neighborhood (fig. 6). In this way, lost objects can be recovered in case of fast motion or homogeneity changes. For the distance measure, the node attributes represent dominant color (c) and size (s) of the regions. For two nodes u and v , the distance is given by

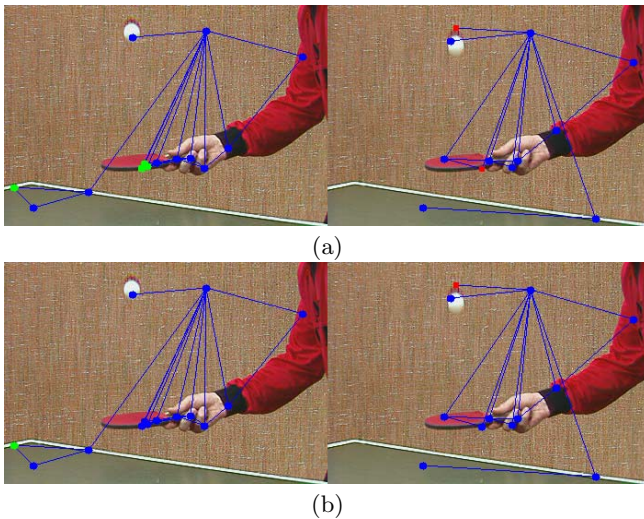


Fig. 7. Subgraph matching. Untracked nodes are shown as green (clear) rounds, tracked nodes as dark (blue) rounds and new nodes as (red) squares. (a) RAGs before matching. (b) RAGs after matching.

$$d(u, v) = |c_u - c_v|^2 \frac{s_u s_v}{s_u + s_v} \quad (10)$$

Thus, we favor the grouping of smaller regions with similar attributes.

An example of matching is shown figure 7 on the *tennis* sequence. Before matching (fig. 7a), untracked regions are located in the racket and the table left corner. The new regions are located above the ball and inside the racket border. After matching (fig. 7b), the nodes at the racket border have been grouped as they have close similarity, whereas the table left corner is not linked to any new node and thus cannot be reliably tracked.

5 Experimental Results

In this section, we test the proposed method on various real video sequences. We analyse the segmentation results on the *akiyo*, *tennis*, and *walking* sequences (CIF format). The processing time is about 1s per frame on a 2.8GHz PC with unoptimized code.

Figure 8 shows the final spatio-temporal segmentation, i.e. when all the frames have been processed. In figure 8a, the video is composed of stationary background and slow head motion. We see that the main regions are the woman

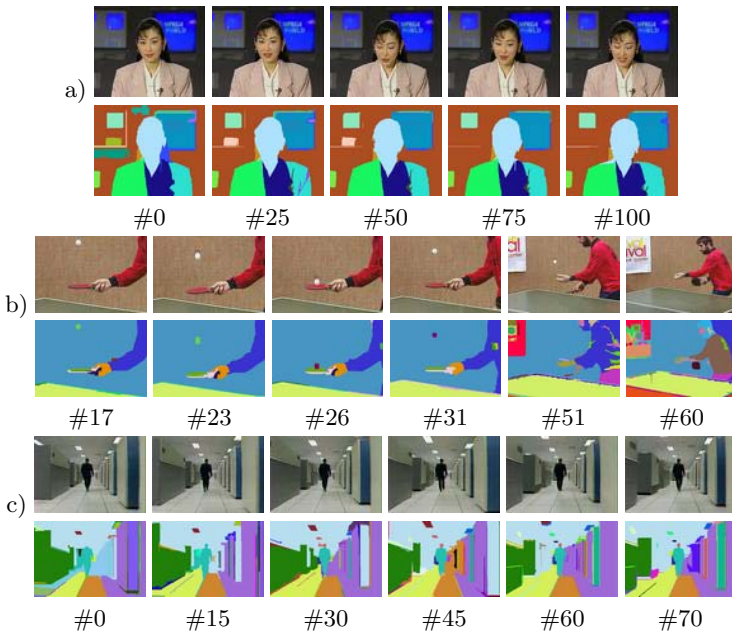


Fig. 8. Segmentation results. a) *akiyo* sequence. b) *tennis* sequence. c) *walking* sequence.

and the TV screens which have smooth spatial variations whereas tiny varying components such as face elements are not kept. In the next images, face moving elements are detected, but they are too tiny to be extracted from the sequence. In consequence, these elements are incorporated into the face.

In figure 8b, the video is composed of several motions. The ball and the racket undergo rigid motion whereas the player undergoes non rigid-motion. Besides these motions, the camera is zooming out during the entire sequence. We see that the ball region remains until it hits the racket in frame #26. As the ball was speeding up in previous frames, the ball and its shadow were splitted into two adjacent regions. The similarity between these regions is lower than their temporal similarity with the new ball region, so that a new region is created for the ball. The ball is tracked successfully until frame #31. From this moment on, the camera quickly zooms out and the ball becomes smaller and less homogeneous. As a result, the ball sometimes does not appear after the spatial merging stage. However, the other regions, which are larger and more stable, such as the table, the racket and the hand are correctly segmented during the whole sequence. Finally, we can see that a strong scale change happens gradually between frame #31 and frame #60. While the player is appearing progressively at the left of the image, the corresponding regions are splitted until fitting the body of the player. In this way, the segmentation follows the temporal changes of the video.

In the last sequence (8c), the camera is tracking the walking man so that the walls surrounding him are moving towards the foreground and exiting the frame progressively. In the first frame #0, the regions are composed of the man, the tiled floor, the walls, the ceiling and the lamps. The man region remains consistent along the sequence, just as the different parts of the walls and the lights until they exit the frame. We can further notice that apparent static regions such as the floor and the ceiling are coherent in the entire sequence.

These results illustrate the potential of the method to extract coherent volumes from video shots. Given a level of details, both moving and static elements can be tracked thanks to our hierarchical matching stage. Besides, we handle dynamic temporal changes by favoring the creation of new regions when some regions cannot be reliably matched between frame pairs. In this way, we achieve good compromise between the span and the consistency of the regions. Therefore, the method can help higher level grouping tasks considerably.

6 Conclusion

We have proposed a new method for extracting meaningful regions from videos. Graphs appear as an efficient solution to build space-time relationships at different levels. We have used both pixel-based graphs to build low-level regions and RAGs to enforce consistency of the regions. We have proposed a temporal grouping method exploiting feature points to handle both static and dynamic regions. Finally, encouraging results show that the method is promising as a preliminary step for object-based video indexing and retrieval.

Acknowledgments

This research was supported by the European Commission under contract FP6-027026-K-SPACE.

References

1. D. Zhong and S. Chang. Long-term moving object segmentation and tracking using spatio-temporal consistency. In *ICIP'01*, volume 2, pages 57–60, Thessaloniki, Greece, Oct. 2001.
2. H. Xu, A. Younis, and M. Kabuka. Automatic moving object extraction for content-based applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(6):796–812, June 2004.
3. D. DeMenthon and D. Doermann. Video retrieval using spatio-temporal descriptors. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 508–517, Berkeley, CA, USA, Nov. 2003.
4. H. Greenspan, J. Goldberger, and A. Mayer. Probabilistic space-time video modeling via piecewise gmm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):384–396, Mar. 2004.
5. L. Vincent and P. Soille. Watersheds in digital space: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598, Aug. 1991.
6. Y. Deng and B. Manjunath. Unsupervised segmentation of color-texture regions in images and video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):800–810, Aug. 2001.
7. C. Yuan, Y. F. Ma, and H. J. Zhang. A graph theoretic approach to video object segmentation in 2d+t space. Technical report, MSR, 2003.
8. C. Gomila and F. Meyer. Graph-based object tracking. In *ICIP'03*, volume 2, pages 41–44, 2003.
9. P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, Sept. 2004.
10. J. Shi and C. Tomasi. Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 593–600, Seattle, WA, USA, 1994.
11. J. S. Milton and J. Arnold. *Introduction to Probability and Statistics*. McGraw-Hill, 2002.

Performance Evaluation of Image Segmentation

Fernando C. Monteiro^{1,2} and Aurlio C. Campilho^{1,3}

¹ INEB - Instituto de Engenharia Biomédica

² Escola Superior de Tecnologia e de Gestão de Bragança
Campus de Santa Apolónia, Apartado 134, 5301-857 Bragança, Portugal
monteiro@ipb.pt

³ FEUP - Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
campilho@fe.up.pt

Abstract. In spite of significant advances in image segmentation techniques, evaluation of these methods thus far has been largely subjective. Typically, the effectiveness of a new algorithm is demonstrated only by the presentation of a few segmented images that are evaluated by some method, or it is otherwise left to subjective evaluation by the reader. We propose a new approach for evaluation of segmentation that takes into account not only the accuracy of the boundary localization of the created segments but also the under-segmentation and over-segmentation effects, regardless to the number of regions in each partition. In addition, it takes into account the way humans perceive visual information. This new metric can be applied both to automatically provide a ranking among different segmentation algorithms and to find an optimal set of input parameters of a given algorithm.

1 Introduction

In a conventional sense, image segmentation is the partitioning of an image into regions, where parts within a region are similar according to some uniformity predicate, and dissimilar between neighbouring regions. Due to its importance, many segmentation algorithms have been proposed, and a number of evaluation criteria have also been proposed. In spite of this, very few comparative studies on the methods used for evaluation have been published [14].

Typically, researchers show their segmentation results on a few images and point out why the results 'look good'. We never know from such studies if the results are good or typical examples. Since none of the proposed segmentation algorithms are generally applicable to all images, and different algorithms are not equally suitable for a particular application, there needs to be a way of comparing them, so that the better ones can be selected. The majority of studies proposing and comparing segmentation methods evaluate the results only with one evaluation method. However, results vary significantly between different evaluators, because each evaluator may have distinct standards for measuring the quality of the segmentation.

Only a few evaluation methods actually explore the segments obtained from the segmentation process. Most measures are best suited to evaluate edge detection [12], working directly on the binary image of the regions' boundaries [3]. Although we can always treat segmentation as a boundary map, the problem is in the simplified use of the edge map, as simply counting the misclassified pixels, on an edge/non-edge basis. Pixels on different sides of an edge are different in the sense that they belong to different regions - that is why it may be more reasonable to use the segmentation partition itself.

Evaluation of image segmentation differs considerably from the binary foreground/background segmentation evaluation problem examined in [3,13], in that the correctness of the two class boundary localization is not the only quantity to be measured. This derives from the presence of an arbitrary number of regions in both the reference segmentation and the segmentation to be evaluated.

An evaluation metric is desired to take into account the following effects:

- **Over-segmentation.** A region of the reference is represented by two or more regions in the examined segmentation.
- **Under-segmentation.** Two or more regions of the reference are represented by a single region in the examined segmentation.
- **Inaccurate boundary localization.** Ground truth is usually produced by humans that segment at different granularities.
- **Different number of segments.** We need to be able to compare two segmentations when they have different numbers of segments.

Under-segmentation is considered to be as a much more serious problem as it is easier to recover true segments through a merging process after over-segmentation rather than trying to split an heterogeneous region. One desirable property of a good evaluation measure is to accommodate refinement only in regions that human segmenters could find ambiguous and to penalize differences in refinements elsewhere. In addition to being tolerant to refinement, any evaluation measure should also be robust to noise along region boundaries and tolerant to different number of segments in each partition.

This work will focus on discrepancy evaluation methods, that consist in comparing the results of a segmentation algorithm with a reference and measuring the differences or discrepancy between them. We introduce a new approach for segmentation evaluation that takes into account, using a single metric, not only the accuracy of the boundary localization of the created segments but also the under-segmentation and over-segmentation effects, regardless to the number of regions in each partition. In addition, it takes into account the way humans perceive visual information, given different weights to false positive and false negative pixels. In order to test the accuracy of the proposed measure we compared it with a set of key methods used for the evaluation of image segmentation using real and synthetic images.

The remainder of this paper is organized as follows: in Section 2, previous segmentation evaluation methods are presented. In Sections 3 and 4, we present region-based and boundary-based evaluation methods currently in literature.

The proposed metric for evaluation is presented in Section 5. In Section 6, experimental evaluation is analysed and discussed, and, finally, conclusions are drawn in Section 7.

2 Previous Work

A review on evaluation of image segmentation is presented by Zhang in [14], who classifies the methods into three categories: *analytical*, where performance is judged not on the output of the segmentation method but on the basis of their properties, principles, complexity, requirements and so forth, without reference to a concrete implementation of the algorithm or test data. While in domains such as edge detection this may be useful, in general the lack of a general theory of image segmentation limits these methods; *empirical goodness methods*, which compute some manner of 'goodness' metric such as uniformity within regions [3], contrast between regions [4], shape of segmented regions [12]; and finally, *empirical discrepancy methods*, which evaluate segmentation algorithms by comparing the segmented image against a manually-segmented reference image, which is often referred to as ground truth, and computes error measures.

As stated by Zhang [14], the major difficulty in applying analytical methods is the lack of general theory for image segmentation. The analytical methods may only be useful for simple algorithms or straightforward segmentation problems, where the researchers have to be confident in the models on which these algorithms are based.

Empirical goodness methods have the advantage that they do not require manually segmented images to be supplied as ground truth data. The great disadvantage is that the goodness metrics are at best heuristics, and may exhibit strong bias towards a particular algorithm. For example the intra-region grey-level uniformity metric will assume that a well-segmented image region should have low variance of grey-level. This will cause that any segmentation algorithm which forms regions of uniform texture to be evaluated poorly. Although these evaluation methods can be very useful in some applications, their results do not necessarily coincide with the human perception of the goodness of segmentation. For this reason, when a reference image is available or can be generated, empirical discrepancy methods are preferred.

Empirical discrepancy methods, which compare segmentation output with ground truth segmentation of the test data and quantify the levels of agreement and/or disagreement, have the benefit that the direct comparison between a segmented image and a reference image is believed to provide a finer resolution of evaluation, and as such, they are the most commonly used methods of segmentation evaluation.

Zhang [14] has proposed a discrepancy evaluation method based on misclassified pixels. Yasnoff *et al.* [13], in one of the earliest attempts, have shown that measuring the discrepancy based only on the number of misclassified pixels does not consider the pixel position error. Their solution is based on the number of misclassified pixels and their distance to the nearest correctly segmented pixels.

They only applied it to foreground/background segmentation. Other discrepancy measures calculate the distances between wrong segmented pixels and the nearest correctly segmented pixels [8], thus introducing a spatial component to the measure, or are based on differences between feature values measured from regions of the correctly segmented and output images. Huang and Dom [3] introduced the concept of distance distribution signatures.

3 Region-Based Evaluation

The region-based scheme evaluates the segmentation accuracy in the number of regions, the locations and the sizes. A region-based evaluation between two segmented images can be defined as the total amount of differences between corresponding regions.

3.1 Hamming Distance

Huang and Dom [3] introduced the concept of directional Hamming distance between two segmentations, denoted by $D_H(S_1 \Rightarrow S_2)$. Let S and R be two segmentations. They began by establishing the correspondence between each region of S with a region of R such that $s_i \cap r_j$ is maximized. The directional Hamming distance from S to R is defined as:

$$D_H(S \Rightarrow R) = \sum_{r_i \in R} \sum_{s_k \neq s_j, s_k \cap r_i \neq \emptyset} |r_i \cap s_k|, \quad (1)$$

where $|\cdot|$ denote the size of a set. Therefore, $D_H(S \Rightarrow R)$ is the total area under the intersections between all $r_i \in R$ and their non-maximal intersected regions from S . A region-based evaluation measure based on normalized Hamming distance is defined as

$$p = 1 - \frac{D_H(S \Rightarrow R) + D_H(R \Rightarrow S)}{2 \times |S|}, \quad (2)$$

where $|S|$ is the image size and $p \in [0, 1]$. The smaller the degree of mismatch, the closer the p is to one.

3.2 Local Consistency Error

To compensate for the difference in granularity while comparing segmentations, many measures allow label refinement uniformly through the image. D. Martin's thesis [6] proposed an error measure to quantify the consistency between image segmentations of differing granularities - *Local Consistency Error* (LCE) that allows labelling refinement between segmentation and ground truth.

$$LCE(S, R, p_i) = \frac{1}{N} \sum_i \min \{E(S, R, p_i), E(R, S, p_i)\}, \quad (3)$$

where $E(S, R, p)$ measures the degree to which two segmentations agree at pixel p , and N is the size of region where pixel p belongs.

Note that the LCE is an error measure, with a score 0 meaning no error and a score 1 meaning maximum error. Since LCE is tolerant to refinement, it is only meaningful if the two segmentations have similar number of segments. As observed by Martin in [6], there are two segmentations that give zero error for LCE - one pixel per segment, and one segment for the whole image.

3.3 Bidirectional Consistency Error

To overcome the problem of degenerate segmentations, Martin adapted the LCE formula and proposed a measure that penalizes dissimilarity between segmentations proportional to the degree of region overlap. If we replace the pixelwise minimum with a maximum, we get a measure that does not tolerate refinement at all. The *Bidirectional Consistency Error* (BCE) is defined as:

$$BCE(S, R, p_i) = \frac{1}{N} \sum_i \max \{E(S, R, p_i), E(R, S, p_i)\} . \quad (4)$$

3.4 Partition Distance Measure

Cardoso and Corte-Real [1] proposed a new discrepancy measure - *partition distance* (d_{sym}) defined as: "given two partitions P and Q of S , the partition distance is the minimum number of elements that must be deleted from S , so that the two induced partitions (P and Q restricted to the remaining elements) are identical". $d_{sym}(Q, P) = 0$ means that no points need to be removed from S to make the partitions equal, i.e., when $Q = P$.

4 Boundary-Based Evaluation

Boundary-based approach evaluates segmentation in terms of both localization and shape accuracy of extracted regions boundaries.

4.1 Distance Distribution Signatures

Huang and Dom in [3] presented a boundary performance evaluation scheme based on the distance between distribution signatures that represent boundary points of two segmentation masks.

Let B_S represent the boundary point set derived from the segmentation and B_R the boundary ground truth. A distance distribution signature from the set B_S to the set B_R of boundary points, denoted $D_B(B_S, B_R)$, is a discrete function whose distribution characterizes the discrepancy, measure in distance, from B_S to B_R . Define the distance from x in set B_S to B_R as the minimum absolute distance from all the points in B_R , $d(x, B_R) = \min \{d_E(x, y)\}, \forall y \in B_R$, where d_E denotes the Euclidean distance between points x and y .

The discrepancy between B_S and B_R is described by the shape of the signature, which is commonly measured by its mean and standard deviation. As a rule, $D_B(B_S, B_R)$ with a near-zero mean and a small standard deviation indicates

high between segmentation masks. Since these measures are not normalized, we cannot determine which segmentation is the most desirable.

We introduce a modification to the distance distribution signature of Huang and Dom, in order to normalize the result between 0 and 1. Doing $d(x, B_R) = \min\{d_E(x, y), c\}$, where the c value sets an upper limit for the error, the two boundary distances could be combined in a framework similar to the one presented in Eq. (2):

$$b = 1 - \frac{D_B(B_S, B_R) + D_B(B_R, B_S)}{c \times (|R| + |S|)}, \quad (5)$$

where $|R|$ and $|S|$ are the number of boundary points in reference mask and segmented mask, respectively.

4.2 Precision-Recall Measures

Martin in his thesis [6], propose the use of *precision* and *recall* values to characterize the agreement between the oriented boundary edge elements (termed *edgels*) of region boundaries of two segmentations. Given two segmentations, S and R , where S is the result of segmentation and R is the ground truth, precision is proportional to the fraction of edgels from S that matches with the ground truth R , and recall is proportional to the fraction of edgels from R for which a suitable match was found in S . Precision measure is defined as follows:

$$Precision = \frac{Matched(S, R)}{|S|} \quad Recall = \frac{Matched(R, S)}{|R|}, \quad (6)$$

where $|S|$ and $|R|$ are the total amount of boundary pixels. In probabilistic terms, precision is the probability that the result is valid, and recall is the probability that the ground truth data was detected.

A low recall value is typically the result of under-segmentation and indicates failure to capture salient image structure. Precision is low when there is significant over-segmentation, or when a large number of boundary pixels have greater localization errors than some threshold δ_{\max} .

Precision and recall measures have been used in the information retrieval systems for a long time [10]. However, the interpretation of the precision and recall for evaluation of segmentation are a little different from the evaluation of retrieval systems. In retrieval, the aim is to get a high precision for all values of recall. However in image segmentation, the aim is to get both high precision and high recall. The two statistics may be distilled into a single figure of merit:

$$F = \frac{PR}{\alpha R + (1 - \alpha)P}, \quad (7)$$

where α determines the relative importance of each term. Following [6], α is selected as 0.5, expressing no preference for either.

The main advantage of using precision and recall for the evaluation of segmentation results is that we can compare not only the segmentations produced

by different algorithms, but also the results produced by the same algorithm using different input parameters. However, since these measures are not tolerant to refinement, it is possible for two segmentations that are perfect mutual refinements of each other to have very low precision and recall scores.

4.3 Earth Mover's Distance

The concept of using the Earth Mover's Distance (EMD) to measure perceptual similarity between images was first explored by Peleg *et al.* in [9] for the purpose of measuring distance between two grey-scale images. More recently EMD has been used for image retrieval [11].

EMD evaluates dissimilarity between two distributions or *signatures* in some feature space where a distance measure between single features is given. The EMD between two distributions is given by the minimal sum of costs incurred to move all the individual points between the signatures.

Let $P = \{(p_1, w_{p_1}), \dots, (p_m, w_{p_m})\}$ be the first signature with m pixels, where p_i is the pixel representative and w_{p_i} is the weight of the pixel; the second signature with n pixels is represented by $Q = \{(q_1, w_{q_1}), \dots, (q_n, w_{q_n})\}$; and $D = [d_{ij}]$ the distance matrix where d_{ij} is the distance between two contour points' image coordinates p_i and q_j . The flow f_{ij} is the amount of weight moved from p_i to q_j . The EMD is defined as the work normalized by the total flow f_{ij} , that minimizes the overall cost:

$$EMD(P, Q) = \frac{\sum_i \sum_j f_{ij} d_{ij}}{\sum_i \sum_j f_{ij}}, \quad (8)$$

As pointed by Rubner *et al* [11], if two weighted point sets have unequal total weights, EMD is not a true metric. It is desirable for robust matching to allow point sets with varying total weights and cardinalities. In order to embed two sets of contour features with different total weights, we simulate equal weights by adding the appropriate number of points, to the lower weight set, with a penalty of maximal distance. Since normalizing signatures, with the same total weight do not affect their EMD, we made $\sum_{i,j} f_{ij} = 1$. Equation (8) becomes,

$$EMD(P, Q) = \sum_i \sum_j f_{ij} d_{ij}, \quad (9)$$

subject to the following constraints: $f_{ij} \geq 0$, $\sum_j f_{ij} = w_{p_i}$ and $\sum_i f_{ij} = w_{q_j}$.

As a measure of distance for the EMD ground distance we use

$$d_{ij} = 1 - e^{-\frac{\|p_i - q_j\|}{\alpha}}, \quad (10)$$

where $\|p_i - q_j\|$ is the Euclidean distance between p_i and q_j and α is used in order to accept some deformation resulted from manual segmentation of ground truth. The exponential map limits the effect of large distances, which otherwise dominate the result.

5 New Discrepancy Measure

In the context of image segmentation, the reference mask is generally produced by humans. There is an agreement that interpretations of images by human subjects differ in granularity of label assignments, but they are consistent if refinements of segments are admissible [6]. One desirable property of a good evaluation measure is to accommodate refinement only in regions that human segmenters could find ambiguous and to penalize differences in refinements elsewhere. In addition to being tolerant to refinement, any evaluation measure should also be robust to noise along region boundaries and tolerant to different number of segments in each partition. The section introduces a new evaluation measure that addresses the above concerns.

For the purpose of evaluating image segmentation results, a correspondence between the examined segmentation mask, S , and the reference mask, R , has initially be established, indicating which region of S better represents each reference region. This is performed by associating each region r_i of mask R with a different region s_j of mask S on the basis of region overlapping, i.e. s_j is chosen so that $r_i \cap s_j$ is maximized. The set of pixels assigned to s_j but not belonging to r_i are false positives, F_p , that can be expressed as $F_p = s_j \cap \bar{r}_i$, where \bar{r}_i denotes the complement of r_i . The pixels belonging to r_i but not assigned to s_j are false negatives, F_n , and can be expressed as $F_n = \bar{s}_j \cap r_i$.

The minimum required overlap between r_i and s_j is 50% of the reference region. Pixels belonging to regions where this ratio is not achieved are considered as false pixels. These measure quantify the errors due to under and over segmentation. Clearly, more visually significant regions that were missed in the segmented mask are assigned a significantly higher error.

The normalized sum of false detections is an objective discrepancy measure that quantifies the deviation of the results of segmentation from the ground truth and can be expressed as:

$$\varepsilon_F = \frac{F_p + F_n}{2N} , \quad (11)$$

where N is the set of all pixels in the image. The value of ε_F is proportional to the total amount of errors and indicates the accuracy of region boundaries localization. The quality of the segmentation is inversely proportional to the amount of deviation between the two masks.

In applications where the final evaluator of quality is the human being, it is fundamental to consider human perception to deal with the fact that different kind of errors are not visually significant to the same degree. To accommodate human perception, the different error contributions are weighted according to their visual relevance. Gelasca *et al.* [2], present a psychophysical experiment to assess the different perceptual importance of errors. They conclude that a false positive pixel contributes differently to the quality than a false negative. False negatives are more significant, and the larger the distance, the larger the error.

We use the weighted functions w_p and w_n to deal with that fact. They are normalized by the diagonal distance, D . Let d_p be the distance of a false positive pixel from the boundary of the reference region, and d_n be the distance of a false

negative pixel. The weight function for each false pixel is defined by Eq. (12) and represented in Fig. 1.

$$w_p = \frac{\alpha_p \log(1 + d_p)}{D} \quad w_n = \frac{\alpha_n d_p}{D} . \quad (12)$$

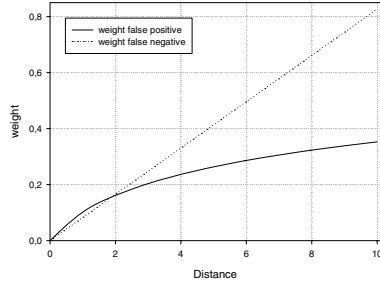


Fig. 1. Weight functions for false negative and false positive pixels

The weights for false negative pixels increase linearly and are larger than those for false positive pixels at the same distance from border. As we move away from the border of an object, missing parts are more important than added background, e.g., in medical imaging, it may be enough that the segmented region overlaps with the true region, so the tumour can be located. But if there are missing parts of the tumour the segmentation results will be poor.

To obtain a measure between $[0, 1]$, we normalize the total amount of weight by the image size. The discrepancy measure of weighted distance, ε_w , becomes:

$$\varepsilon_w = \frac{1}{N} \left(\sum_{f_n} w_n + \sum_{f_p} w_p \right) , \quad (13)$$

where f_n and f_p represent the false pixels. We define a measure of similarity as $s_w = 1 - \varepsilon_w$. The value of $s_w = 1$ indicates a perfect match between the segmentation and the reference mask.

6 Analysis on Evaluation Methods

To achieve comparative results about different evaluation methods, two strategies can be followed: the first one consists in applying the evaluation methods to segmented images obtained from different segmentation approaches. The second one consists in simulating results of segmentation processes. To exempt the influence of segmentation algorithms, the latter has been adopted and a set of images obtained from manual segmentation available in [5] was used (Fig. 2).

A good evaluation measure has to give large similarity values for images (b) to (g) and has to strongly penalize other images. Figure 3.a) shows the comparative

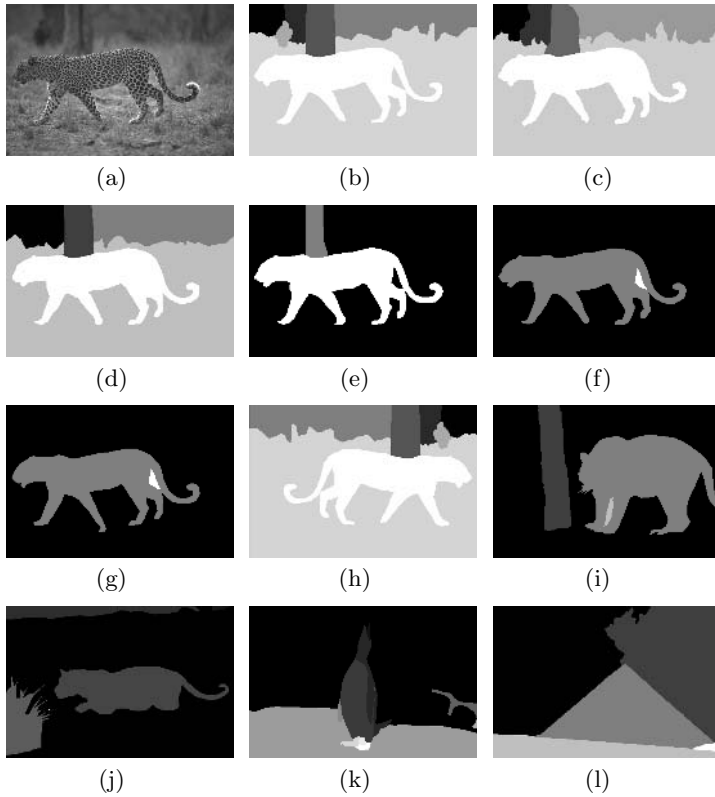


Fig. 2. The image and its ground truth are shown in (a) and (b), respectively. From (c) to (g) we have different segmentations of image (a). (h) is the reflected image of ground truth. Images (i) to (l) are segmentations of other images.

study between the proposed method and the methods presented in Section 3, expressed in terms of region-based evaluation.

Due to its tolerance to refinement, LCE gives low error (high similarity) scores, even when the segmentation is very different from the ground truth. Measure p has a similar behaviour. BCE and d_{sym} give good results for images ((h)-(l)), however, since they are not tolerant to refinement, the results are poor for other images. Note that the proposed measure is tolerant to refinement and at the same time strongly penalizes images ((h)-(l)).

Results of boundary-based evaluation on the same set of images of Fig. 2 are reported in Fig. 3.b). On comparing the results of the boundary-based measures, it is made evident that they are well correlated. EMD tolerates well some amount of deformations that normally happens in the manual segmentation process. However, when the number of pixels in ground truth differs a lot from the number of pixels in the segmented image, EMD gives poor results. Despite its success, the EMD method still needs to be refined to address the limitation in the complexity

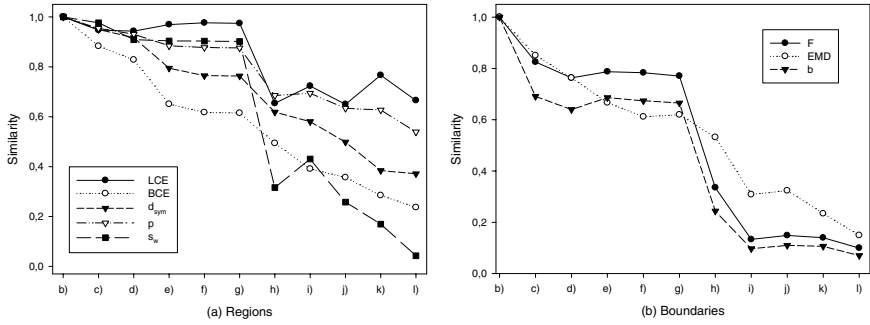


Fig. 3. Evaluation of segmentation, in terms of similarity, from Fig. 2



Fig. 4. Synthetically generated set of segmentations, where (a) is the reference

Table 1. Numerical evaluation of segmentations from Fig. 4

| images | LCE | BCE | d_{sym} | p | s_w |
|--------|---------|---------|-----------|---------|---------|
| (b) | 0.99380 | 0.98088 | 0.99349 | 0.99349 | 0.99741 |
| (c) | 0.99380 | 0.98088 | 0.99349 | 0.99349 | 0.99612 |
| (d) | 0.99380 | 0.98088 | 0.99349 | 0.99349 | 0.99159 |

of algorithm that require to be further reduced. The b-measure gives results similar with F-measure, but is even more intolerant to refinement.

Table 1 presents the evaluation results obtained from a set of trivial synthetically generated segmentations presented in Fig. 4, where we make constant the number of false detections in each segmentation.

Since LCE, BCE, d_{sym} and p , are just proportional to the total amount of false detections, different position of those pixels do not affect the similarity. This makes those methods unreliable for applications where the results will be presented to humans. Note that s_w produces results that agree with the visual relevance of errors.

7 Conclusion

In this paper, we introduce a new approach for segmentation evaluation that takes into account, using a single metric, not only the accuracy of the bound-

ary localization of the created segments but also the under-segmentation and over-segmentation effects according to the ambiguity of the regions, regardless to the number of segments in each partition. We experimentally demonstrated the efficiency of this new measure against well known methods. This new metric can be applied both to automatically provide a ranking among different segmentation algorithms and to find an optimal set of input parameters of a given algorithm according with their results of segmentation evaluation. Moreover, this paper introduces a modification to the distance distribution signature of Huang and Dom, b -measure; it applies the concept of Earth Mover's Distance to the evaluation of image segmentation.

References

1. Cardoso, J.S., Corte-Real, L., Toward a generic evaluation of image segmentation, *IEEE Transactions on Image Processing*, 14(11):1773-1782, (2005).
2. Gelasca, E.D., Ebrahimi, T., Farias, M.C.Q., Carli, M., Mitra, S.K., Towards perceptually driven segmentation evaluation metrics, in *Proc. IEEE Computer Vision and Pattern Recognition Workshop*, Vol. 4, page 52, (2004).
3. Huang, Q., Dom, Byron, Quantitative methods of evaluating image segmentation, in *Proc. IEEE International Conference on Image Processing*, Vol. III:53-56, (1995).
4. Levine, M.D., Nazif, A.M., Dynamic measurement of computer generated image segmentations, *Trans. Pattern Analysis and Machine Intelligence* 7(2):155-164, (1985).
5. Martin, D., Fowlkes, C., The Berkeley segmentation database and benchmark, online at <http://www.cs.berkeley.edu/projects/vision/grouping/segbench/>.
6. Martin, D., *An empirical approach to grouping and segmentation*, PhD dissertation, University of California, Berkeley (2002).
7. Mezaris, V., Kompatsiaris, I., Strintzis, M.G., Still image objective segmentation evaluation using ground truth, in *Proc. of 5th COST 276 Workshop*, pp: 9-14, (2003).
8. Odet, C., Belaroussi, B., Cattin, H.B., Scalable discrepancy measures for segmentation evaluation, in *Proc. Intern. Conf. on Image Processing*, Vol. I:785-788, (2002).
9. Peleg, S., Werman, M., Rom, H., A unified approach to the change of resolution: Space and gray-level, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11:739-742, (1989).
10. Raghavan, V., Bollmann, P., Jung, G., A critical investigation of recall and precision as measures of retrieval system performance, *ACM Transactions on Information Systems*, 7(3):205-229, (1989).
11. Rubner, Y., Tomasi, C., Guibas, L.J., The Earth Mover's Distance as a metric for image retrieval, *International Journal of Computer Vision*, 40(2):99-121, (2000).
12. Sahoo, P.K., Soltani, S., Wang, A.K.C., A survey of thresholding techniques, *Computer Vision, Graphics and Image Processing*, 41(2):233-260, (1988).
13. Yasnoff, W.A., Mui, J.K., Bacus, J.W., Error measures in scene segmentation, *Pattern Recognition*, 9(4):217-231, (1977).
14. Zhang, Y.J., A survey on evaluation methods for image segmentation, *Pattern Recognition*, 29(8):1335-1346, (1996).

Improvement of Image Transform Calculation Based on a Weighted Primitive

María Teresa Signes Pont, Juan Manuel García Chamizo,
Higinio Mora Mora, and Gregorio de Miguel Casado

Departamento de Tecnología Informática y Computación
University of Alicante, 03690, San Vicente del Raspeig, Alicante, Spain
{teresa, juanma, higinio, demiguel}@dtic.ua.es
<http://www.ua.es/i2rc/index2-eng.html>

Abstract. This paper presents a method to improve the calculation of functions that demand a great amount of computing resources. The fundamentals argue for an increase of the computing power of the primitive level in order to decrease the number of computing levels required to carry out calculations. A weighted primitive substitutes the usual primitives sum and multiplication and calculates the function values by successive iterations. The parametric architecture associated to the weighted primitive is particularly suitable in the case of combined trigonometric functions sine and cosine involved in the calculation of image transforms. The Hough Transform (HT) and the Fourier Transform (FT) are analyzed under this scope, obtaining a good performance and trade-off between speed and area requirements when comparing with other well-known proposals.

1 Introduction

The Hough Transform (HT) has become a widely used technique in image segmentation: plane curve detection [1], object recognition [2], air picture vectorization [3], etc. The HT is very suitable because of its robustness, although the great amount of temporary and spatial resources required has moved it away from real time applications. This way, the investigation efforts in HT have dealt with the design of fast algorithms and parallel or ad-hoc architectures. There are also implementations of the CORDIC algorithm for applications that demand high speed and precision, such as digital signal and image processing and algebra [4], [5]. However, their drawback is the lower degree of regularity and parallelism capabilities when comparing with the traditional algorithm.

The Fourier Transform (FT) is a reference tool in image filtering and reconstruction [6], [7]. A Fast Fourier Transform (FFT) scheme has been used in OFDM modulation (Orthogonal Frequency Division Multiplexing) and has shown to be a valuable tool in the scope of communications [8], [9]. The most relevant algorithm for FFT calculation was developed by Cooley and Tukey in 1965 [10]. It is based on successive folding scheme and its main contribution is the computational complexity reduction that decreases from $O(N^2)$ to $O(N \cdot \log_2 N)$. The variants of FFT algorithm follow

different ways to realize the calculations and to store the intervening results [11]. These differences give rise to different improvements such as memory saving, high speed or architecture regularity. The features of the different algorithms point to different hardware trends [12], [13].

This paper approaches the improvement of the calculation performance under an architectural scope. Usually, the CPUs implement the primitive operations sum and multiplication sequentially within the electronic circuits. This physical basis supports a hierarchical organization of computing. The choice of more powerful operations for the first level in the hierarchy, instead of sums and shifts, would contribute to avoid the scalability problem tied to the growing number of levels. As consequence, modeling, formalization and calculations are expected to become easier. The proposed primitive has the structure of a weighted sum and as an estimation of its computing power the calculation of the usual convolution is carried out at the primitive level. Digital image processing is an interesting application field because the calculation of the functions usually involve sine and cosine functions which can be improved by the new weighted primitive.

The paper is structured in five parts: following the introduction, section 2 presents the formalization of the weighted primitive and an estimation of its computing capabilities is provided by means of the calculation of the usual convolution. Section 3 analyzes parametric architecture associated to the weighted primitive. Section 4 develops the calculation of the HT and the FT and develops the comparison with other proposals with respect to area and delay time estimations. Section 5 summarizes results and presents concluding remarks.

2 Recursive Operation Based on a Weighted Primitive

This section introduces the formalization of a weighted primitive and defines a recursive operation based on the successive iterations of the primitive. As a measure of its computing capabilities the convolution of two functions is carried out by means of successive iterations of the mentioned primitive.

2.1 Formalization and Examples

The proposed primitive combines the usual primitives sum and multiplication under the form of a weighted sum \oplus (Equation (1)):

$$\forall(A, B) \in \mathbb{R}, A \oplus B = \alpha A + \beta B, \quad (1)$$

where α and β are real parameters that characterize the primitive.

The defined weighted sum is equivalent to the usual sum when $\alpha=\beta=1$.

A recursive operation can be defined in the set of real numbers by an initial value and a recursive formula (see Equation (2)).

$$\begin{aligned} &F_0, \\ &F_{i+1} = \alpha F_i + \beta G_i, \\ &i \in \mathbb{N}, \quad F_i, G_i \in \mathbb{R}, \quad (\alpha, \beta) \in \mathbb{R} \end{aligned} \quad (2)$$

It can be noticed that the recursive operation is built by running successive iterations of the weighted primitive, with no more than doing A and B equal to the operands F_i, G_i respectively. So, if F_0 and $\{G_i\}$ are given, $\{F_i\}$ can be generated when (α, β) is known. $\{G_i\}$ and $\{F_i\}$ are sets of real values.

When the G_i values in the set $\{G_i\}$ and the parameters α and β are particularized, the recursive operation outputs particular behavioural results. As an example, the following cases have been considered in Fig.1a-e. In any case $F_0=1$ and for any $i, G_i=1$. The horizontal axis represents i values, that is to say the number of the current iteration and the vertical axis represents the corresponding F_i values.

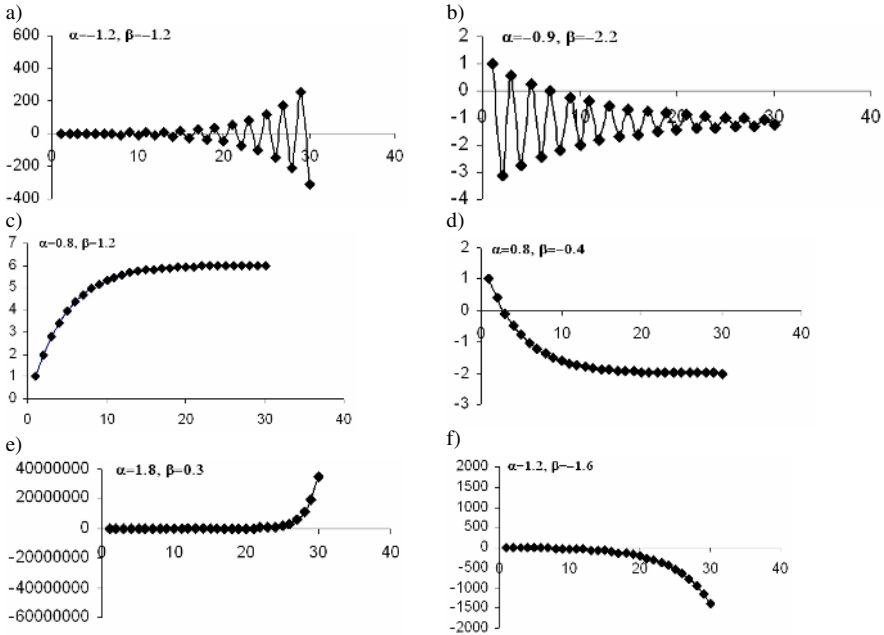


Fig. 1. a) Oscillatory increasing pattern $\alpha \in]-\infty, -1], \forall \beta \in \mathbb{R}$; b) Oscillatory decreasing pattern $\alpha \in]-1, 0], \forall \beta \in \mathbb{R}$; c) Increasing pattern with horizontal asymptote $\alpha \in]0, 1], \alpha + \beta > 1$; d) Decreasing pattern with horizontal asymptote $\alpha \in]0, 1], \alpha + \beta < 1$; e) Increasing pattern with vertical asymptote $\alpha \in]1, +\infty], \alpha + \beta > 1$; f) Decreasing pattern with vertical asymptote $\alpha \in]1, +\infty], \alpha + \beta < 1$

Other values for the different parameters would carry out more different behavioural patterns.

An important feature of the primitive capabilities is that little changes in the parameters α, β and G trigger important changes in the results of the recursive operation. So, this primitive level seems to have the same features as a non trivial level of derivation in a current CPU.

2.2 Estimation of the Computing Power of the Weighted Primitive

In order to have a more accurate estimation of the computing power of the weighted primitive level, an equivalence have been established between the recursive operation and the usual convolution in the sense that both are able to generate a set of real values. The equivalence is based on the conditions that have to be fulfilled in order to make equivalent the generated sets.

The convolution is an operation between two functions that is relevant to many different applications as digital signal and image processing [14], control engineering [15], mathematical morphology [16] or pattern analysis [17]. All these applications must calculate spatial or temporal integral transforms that are carried out by means of convolution. Equation (3) shows the expression of convolution in the case of two discrete real-valued functions f and g in the interval $[x, \infty[$, h is a real number which stands for the discretization step and k is a natural number which can reach any value $0 < k < \infty$. The calculation of the convolution has $O(N^2)$ computational complexity.

$$f * g(x + kh) = \sum_{p=0}^{p=k} f(x + ph)g(x + (k - p)h). \quad (3)$$

In order to establish the equivalence between the recursive operation and the convolution, equation (3) can be transformed into equation (4) considering that the discrete convolution can achieve its own recursive formulation too.

$$\begin{aligned} f * g(x + kh) &= f * g(x + (k - 1)h) + hg(x) \frac{f(x + kh) - f(x + (k - 1)h)}{h} + \dots \\ &\dots + hg(x + (k - 1)h) \frac{f(x + h) - f(x)}{h} + f(x)g(x + kh). \end{aligned} \quad (4)$$

The equivalence between the sets of values generated by (4) and (2) can be understood under the following normalization scheme: the initial point x has to be equalized to the first iteration number, that is to say $x \equiv 0$; the discretization step $h \equiv 1$, so the number of the successive iterations agrees with the number of points, $k \equiv i$.

It can be highlight that this equivalence suits for any x and for any h , and enables to transform a convolution to a recursive operation.

Reciprocally, it can be demonstrated that any discrete recursive operation represented by a weighted sum of two terms like those of equation (2) can be split up into two convolving discrete functions f and g .

Let equation (5) be a discrete recursive operation, defined as follows:

$$\begin{aligned} Y_0, \\ Y_{i+1} &= \alpha Y_i + \beta Z_i, \\ i \in \mathbb{N}, \quad Y_i, Z_i &\in \mathbb{R}, \quad (\alpha, \beta) \in \mathbb{R}. \end{aligned} \quad (5)$$

If $\alpha \neq 0$ and $\beta \neq 0$, $f(k) = \alpha^k$ and $g(k) = \beta Z_i$.

If $\alpha \neq 0$ and $\beta = 0$, $f(k) = \delta(k - i)$ and $g(k) = \alpha^k g(0)$ ($\delta(k - i)$ is the Dirac impulse).

If $\alpha = 0$ and $\beta \neq 0$, $f(k) = \delta(k - i)$ and $g(k) = \beta Z_i$.

The case $\alpha=0$ and $\beta=0$ is trivial and does not lead to any solution.

The normalization obtained outlines the fact that there is no need to discern between the sets $\{f * g(k)\}$ and $\{F_k\}$.

The computational complexity of the recursive calculation of the values in $\{F_k\}$ is $O(N)$ if the set $\{G_k\}$ can be obtained with the same computational complexity or if these values are known. Compared with the $O(N^2)$ complexity of the usual calculation of convolution, based on the usual primitives sum and multiplication, the weighted primitive presents an attractive issue to improve the calculations involved in many engineering applications.

3 Parametric Architecture

The recursive operation (2) leads to the architecture shown in Fig. 2a, where the calculation of each F_i value involves two multiplications and one sum. The implementation could be performed using two multipliers and one adder or, alternatively, avoiding multiplications by substituting multipliers by look-up tables (LUTs) so that to access and capture partial products. On every cycle the LUT is addressed by two input operators A and B (A, B are few bits portions of F and G respectively) and the partial product $\alpha A + \beta B$ is taken out from the memory cell (the partial product $\alpha A + \beta B$ is conceptually a weighted primitive). A new F_i value is carried out by adding all the shifted partial products corresponding to all inputs A, B . In the next iteration, both, the new F_i value and a new G_i value access the LUT in order to repeat the extraction process. Figure 2b shows the extraction of the partial products which provide the pursued result. Table 1 shows the structure of the LUT when A, B are portions of 1 bit.

Table 1. LUT Structure

| AB | $F > 0$ and $G > 0$ | $F > 0$ and $G < 0$ | $F < 0$ and $G > 0$ | $F < 0$ and $G < 0$ |
|------|---------------------|---------------------|---------------------|---------------------|
| 00 | 0 | 0 | 0 | 0 |
| 01 | β | $-\beta$ | β | $-\beta$ |
| 10 | α | α | $-\alpha$ | $-\alpha$ |
| 11 | $\alpha + \beta$ | $\alpha - \beta$ | $-\alpha + \beta$ | $-\alpha - \beta$ |

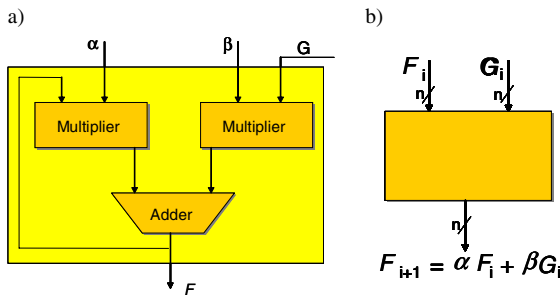


Fig. 2. a) Parametric architecture; b) Addressed partial product (weighted primitive)

In order to have the capability to realize a comparison of computing resources, an estimation of the area and time delay of the proposed architecture is presented here. The model that we use for the estimations is taken from [18] and [19]. The unit τ_a represents the area of a complex gate, which is defined as the pair (AND, XOR) as a meaningful unit because these two gates implement the most basic computing device: the one bit full-adder; the unit τ_t is the delay of this complex gate. This model is very useful because it provides a direct way to compare different architectures, irrespectively from their implementation. As an example the area cost and time delay for 16 bits data are estimated as shown in Table 2.

Table 2. Parametric architecture estimations of area cost and time delay for 16 bits data

| Parametric architecture | Occupied area | Time delay |
|----------------------------------|--|--|
| Multiplexer | $0,25 \cdot 2 \cdot 16 \tau_a = 8 \tau_a$ | $0,5 \tau_t$ |
| Shift register | $0,5 \cdot 16 \tau_a = 8 \tau_a$ | $15 \cdot 0,5 \tau_t = 7,5 \tau_t$ |
| LUT | $40 \tau_a / \text{Kbit} \cdot 16 \text{ bits} \cdot 16 \text{ cells} = 10 \tau_a$ | $3,5 \tau_t \cdot 16 \text{ accesses} = 56 \tau_t$ |
| Register | $0,5 \cdot 16 \tau_a = 8 \tau_a$ | $1 \tau_t$ |
| Reduction structure 4:2+adder | $4 \tau_a + 16 \tau_a = 20 \tau_a$ | $3 \text{ reductions} \cdot 3 \tau_t + \lg 16 \tau_t$ $= 13 \tau_t$ |
| Overall | $54 \tau_a$ | $78 \tau_t$ |

The proposed architecture has been tested in the XS4010XL-PC84 FPGA; $\tau_t \approx 1$ ns is assumed, so a time delay estimation in usual time units can also be provided.

4 Applications

In this section two digital image applications are presented: the calculation of the HT and the FT based on the recursive operation. These are then compared with the proposals presented by Deng [20] and by Chien [21] in what concerns to the area cost and the time delay.

4.1 The HT Calculation Under the Scope of the Recursive Operation

In the case of a straight line as the geometric primitive to be detected, the HT transforms each point $P(x, y)$ in the Cartesian domain in a point (ρ, θ) in the Hough domain. The parametric space is discretized in N_θ levels, from 0 to Π and N_ρ levels, from ρ_{min} to ρ_{max} . The HT calculates the ρ values for all the angles in $[0, \Pi]$ and for every pixel in the image. The direct calculation has $O(N^2)$ complexity and the global amount of operations is $N^2 \cdot N_\theta$. If $[0, \Pi]$ is considered as $[0, \Pi/2] \cup [\Pi/2, \Pi]$, the HT for every pixel (x_i, y_j) in the image can be written as:

$$\begin{aligned} (\rho_I)_k &= x_i \cdot \cos \theta_k + y_j \cdot \sin \theta_k, & 0 \leq \theta_k < \Pi/2, \\ (\rho_{II})_k &= y_j \cdot \cos \theta_k - x_i \cdot \sin \theta_k, & \Pi/2 \leq \theta_k < \Pi. \end{aligned} \quad (6)$$

If $\Delta\theta$ is the discretization step, any angle can be obtained by adding the step to the former angle, that is, $\theta_{i+1} = \theta_i + \Delta\theta$. If $\cos \Delta\theta = \alpha$ and $\sin \Delta\theta = \beta$, equation (6) leads to equation (7).

$$\begin{aligned}
 (\rho_I)_k &= \alpha \cdot (\rho_I)_{k-1} + \beta (\rho_{II})_{k-1}, \\
 (\rho_{II})_k &= \alpha \cdot (\rho_{II})_{k-1} - \beta \cdot (\rho_I)_{k-1}, \\
 \text{with } \alpha^2 + \beta^2 &= 1.
 \end{aligned}
 \tag{7}$$

Comparing (7) and (2) it can be deduced that HT is performed by means of two crossed recursive operations that provide the ρ values for all the angles in $[0, \Pi[$ and for every pixel in the image under a restricting condition on the parameters α, β . The initial values $(\rho_I)_0$ and $(\rho_{II})_0$ are the values of the coordinates of each pixel in the image.

Comparison

This implementation has been compared to Deng’s proposal [20] which consists of a pipelined CORDIC algorithm application for the HT calculation. The CORDIC algorithm (COordinate Rotation DIgital Computer) is an approximation method that performs rotations in the plane in three coordinates systems: linear, circular and hyperbolic [22], [23]. The only operations needed to perform the calculations are additions, subtractions and shifts. The proposal calculates the HT using 16-bit fixed-point arithmetic. The 12-iteration CORDIC is implemented using a Xilinx XS4010XL-PC84 FPGA for fast prototyping. This device has medium capacity and speed; it consists on 400 CLBs (Logic Block) arranged in a 20x20 matrix, which is approximately equivalent to 10.000 gates. The present calculation uses 333 CLBs. For the Xilinx device it has been considered that a CLB consists of one LUT-3, two LUT-4 and two latches. The areas for the CORDIC implementation are estimated in τ_a and τ units (see Table 3), as it was done for the parametric architecture, following [18] and [19] (see Table 2).

This implementation can be clocked at more than 40 MHz with a computational complexity of $O(N^2)$ for $N \times N$ image. At this frequency, a 128x128 binary image with 128 discrete angles ($\Delta\theta=1.40625^\circ$) takes 0.0262 seconds to be transformed. It can be assumed that $\tau \approx 1$ ns.

Table 3. Parametric architecture estimations of area cost and time delay for 16 bits data

| Pipelined CORDIC | CLBs=333 | Area |
|------------------|-----------|---|
| LUT-3 | 1·333=333 | $333 \cdot 2^3 \cdot 2^4 \cdot 40 \tau_a / \text{Kbit} = 1665 \tau_a$ |
| LUT-4 | 2·333=666 | $2 \cdot 333 \cdot 2^4 \cdot 2^4 \cdot 40 \tau_a / \text{Kbit} = 6660 \tau_a$ |
| Latches | 2·333=666 | $2 \cdot 333 \cdot 0.5 \cdot 2^4 \cdot \tau_a = 5328 \tau_a$ |
| Overall | | 13653 τ_a |

According to the estimations shown in Table 2, the area occupied for HT calculation is $2 \cdot 54 \tau_a = 108 \tau_a$ considering that the processing element (PE) needed consists of two modules to carry out the cross evaluation. The time delay is $78 \tau \cdot 64 \cdot 128 \cdot 128 = 0,081788928$ seconds for calculating a number of points equal to the number of pixels (128x128) multiplied by the number of the angles (64), assuming $\tau \approx 1$ ns for the device.

It can be highlighted that the amount of hardware resources needed by the pipelined CORDIC is much more important than for the parametric architecture (13653 τ_a versus $108 \tau_a$). Nevertheless, its time delay is almost four times the CORDIC one (81,78 ms versus 26,2 ms). The HT evaluation by the parametric architecture may be

better balanced between the speed and the area cost by means of parallelization. The parallelization can be performed on the pixels by multiplying by a factor k the number of PEs involved in order to divide by k the time delay of the overall calculation. So, each PE calculates only the k^{th} part of the overall points of the transform. A satisfying trade-off between area cost and time delay can easily be found for a large range of k values, $4 < k < 125$.

4.2 The Fourier Transform Calculation Under the Scope of the Recursive Operation

Equation (8) is the expression of a discretized real function of one-dimensional discrete Fourier Transform. Let's have $N=2M=2^n$.

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x)W_{2M}^{ux},$$

$$W_N = \exp \frac{-2j\pi}{N}.$$
(8)

Cooley and Tukey algorithm segregates the FT in even and odd fragments in order to perform the successive folding scheme, as shown in (9):

$$F(u) = \frac{1}{2} (F_{\text{even}}(u) + F_{\text{odd}}(u)W_{2M}^u),$$

$$F(u+M) = \frac{1}{2} (F_{\text{even}}(u) - F_{\text{odd}}(u)W_{2M}^u),$$

$$F_{\text{even}}(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(2x)W_M^{ux}, \quad F_{\text{odd}}(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(2x+1)W_M^{ux}.$$
(9)

The process starts for any $u, u \in [0, M]$ by setting the M two-point initial transforms. In the second step $M/2$ four-point transforms are carried out by combining the former transforms and so on till to the n^{th} step where one M -point transform is finally obtained. For values of $u, u \in [M, N]$, no more extra calculations are required considering that the corresponding transforms can be obtained by changing the sign, as shown in the second expression in Equation (9). Our method enhances this process by adding a new segregation held by both, real (cosine) and imaginary (sine), parts in order to run the crossed evaluation. Equations (10), (11) and (12) show the first, the second and the n^{th} stages of this process, respectively. The u argument has been omitted in (11) and (12) in order to make clear the expansion.

$$R_{(0,0)EVEN}(u) = f(0) + \alpha_1(u)f(2^{n-1}),$$

$$R_{(0,1)ODD}(u) = f(2^{n-2}) + \alpha_1(u)f(2^{n-2} + 2^{n-1}),$$

...

$$R_{(0,M-1)ODD}(u) = f(2 + 2^2 \dots + 2^{n-2}) + \alpha_1(u)f(2 + 2^2 \dots + 2^{n-2} + 2^{n-1}),$$

$$I_{(0,0)EVEN}(u) = -\beta_1(u)f(2^{n-1}),$$

$$I_{(0,1)ODD}(u) = -\beta_1(u)f(2^{n-2} + 2^{n-1}),$$

...

$$I_{(0,M-1)ODD}(u) = -\beta_1(u)f(2 + 2^2 + \dots 2^{n-2} + 2^{n-1}).$$
(10)

$$\begin{aligned}
R_{(1,0)EVEN} &= R_{(0,0)EVEN} + \alpha_2 R_{(0,0)ODD} - \beta_2 I_{(0,1)ODD}, \\
I_{(1,0)EVEN} &= I_{(0,0)EVEN} + \beta_2 R_{(0,1)ODD} - \alpha_2 I_{(0,3)ODD}, \\
R_{(1,1)ODD} &= R_{(0,2)EVEN} + \alpha_2 R_{(0,3)ODD} - \beta_2 I_{(0,3)ODD}, \\
I_{(1,1)ODD} &= I_{(0,2)EVEN} + \beta_2 R_{(0,3)ODD} + \alpha_2 I_{(0,3)ODD}, \\
&\dots \\
R_{(1, \frac{M}{2})ODD} &= R_{(0, \frac{M}{2})EVEN} + \alpha_2 R_{(0, \frac{M}{2}+1)ODD} - \beta_2 I_{(0, \frac{M}{2}+1)ODD}, \\
I_{(1, \frac{M}{2})ODD} &= I_{(0, \frac{M}{2})EVEN} + \beta_2 R_{(0, \frac{M}{2}+1)ODD} - \alpha_2 I_{(0, \frac{M}{2}+1)ODD}.
\end{aligned} \tag{11}$$

$$\begin{aligned}
u &\in [0, M], \\
R &= R_{(n-1,0)} = R_{(n-2,0)EVEN} + \alpha_n R_{(n-2,1)ODD} - \beta_n I_{(n-2,1)ODD}, \\
I &= I_{(n-1,0)} = I_{(n-2,0)EVEN} + \beta_n R_{(n-2,1)ODD} + \alpha_n I_{(n-2,1)ODD}.
\end{aligned} \tag{12}$$

$$\begin{aligned}
u &\in]M, N], \\
R &= R_{(n-1,0)} = R_{(n-2,0)EVEN} - \alpha_n R_{(n-2,1)ODD} + \beta_n I_{(n-2,1)ODD}, \\
I &= I_{(n-1,0)} = I_{(n-2,0)EVEN} - \beta_n R_{(n-2,1)ODD} - \alpha_n I_{(n-2,1)ODD}.
\end{aligned}$$

It can be seen that after the second stage, any pair R, I is obtained using a recursive operation. Equations (11) and (12) show that each stage involves the results R and I obtained in the two previous stages. This way the number of operations is halved in each stage.

Comparison

The BDA proposal presented by Chien [21] carries out the Fourier transform of variable length by controlling the architecture. The single processing element follows the Cooley and Tukey algorithm radix-4 and calculates 16/32/64 points transform. When the number of points N grows, it can be split out into a product of two factors $N_1 \cdot N_2$ in order to process the transform in a row-column structure. Formally, the four terms of the butterfly are set as a cyclic convolution that allows performing the calculations by means of distributed arithmetic based on blocks. The memory is partitioned in blocks that store the set of coefficients involved in the multiplications of the butterfly. A rotator is added to control the sequence of use of the blocks. The processing column consists on an input buffer, a CORDIC processor that runs the complex multiplications followed by a parallel-serial register and a rotator. Four RAM memories and sixteen accumulators implement the distributed arithmetic. At last, four buffers are needed to reorder the partial products that are involved in the basic four points operation. The algorithmic complexity is $O(N_l/4M \cdot W_L)$ where N_l is the length of the transform, $M=4$ in the design and W_L is the data length, when the transform is longer than 64 points. Table 4 shows the results obtained by the Synopsis implementation of the circuit that has been described in Verilog HDL.

In order to compare the performance of our proposal and that of BDA, estimations of the occupied area and time delay are provided. The devices for both implementations are listed in Table 5 and evaluated in terms of τ_a and τ_t in Table 6. For the

parametric architecture, two PEs are needed because of the two segregations (even/odd and real/imaginary); 64 cells LUTs are assumed, considering that the parameter set is $(\alpha, \beta, 1)$. Data is 16 bits long for any proposal. In Table 5, neither the rotator nor the CORDIC processor has been considered in the BDA implementation considering that the reference does not facilitate any detail upon their structure. The estimations of the time delay are based on the author’s indications and presented in terms of τ_a and τ_t units. The average computation time is indicated as $(N/4 \cdot W_L)(T_{ROM} + 2 \cdot T_{ADD} + T_{LATCH})$.

Table 4. Critical path of the basic calculation module in the BDA architecture

| | Preprocessor | P/SRAM | Adder+Acc | Postprocessor | 4-point DFT | Overall |
|------------------------|--------------|----------|-----------|---------------|-------------|----------|
| Time per column | 13,71 ns | 12,45 ns | 14,06 ns | 17,7 ns | 10,35 ns | 68,27 ns |
| Critical path | 17,7 ns | 17,7 ns | 17,7 ns | 17,7 ns | 17,7 ns | 88,5 ns |

Table 5. Hardware resource comparison between the BDA and the parametric architecture implementations

| <i>N</i> | Devices for Implementing the DBA Architecture | Devices for Implementing the Parametric Architecture |
|----------|--|--|
| 16 | 5 buffers, 1 CORDIC processor, P/S-R, 1 rotator, 4 (4x16) bits RAMs, 16 MAC | 4 MUX, 4 S-R, 2 (64 x16) bits LUTs 4 registers, 4 reduction structures and 4 adders |
| 64 | 5 buffers, 1 CORDIC processor, P/S-R, 1 rotator, 4 (16x16) bits RAMs, 16 MAC | |
| 512 | 9 buffers, 1 CORDIC processor, 2 P/S-R, 1 rotator, 8 (8x16) bits RAMs, 32 MAC 1 transposition memory | |
| 4096 | 9 buffers, 1 CORDIC processor, 2 P/S-R, 1 rotator, 8 (16x16) bits RAMs, 32 MAC 1 transposition memory | |

Table 6. Comparison between the BDA and the parametric architecture implementations

| <i>N</i> | DBA Architecture | | Parametric Architecture | |
|----------|------------------|---------------------------|-------------------------|-----------------------------|
| | Area | Time delay | Area | Time delay |
| 16 | $314 \tau_a$ | $3,3 \cdot 10^3 \tau_t$ | $336 \tau_a$ | $1,248 \cdot 10^3 \tau_t$ |
| 64 | $344 \tau_a$ | $13,2 \cdot 10^3 \tau_t$ | | $4,992 \cdot 10^3 \tau_t$ |
| 512 | $632 \tau_a$ | $105,6 \cdot 10^3 \tau_t$ | | $39,936 \cdot 10^3 \tau_t$ |
| 4096 | $672 \tau_a$ | $844,8 \cdot 10^3 \tau_t$ | | $119,808 \cdot 10^3 \tau_t$ |

It can be seen that BDA architecture is much more expensive than the parametric one with respect to the occupied area because the hardware requirements have necessarily to be increased stepwise when the number of points of the transforms increases. The time delay is lower for the parametric architecture for the values of *N* that have been considered and will remain lower for any *N*, because it achieves linearly growing for both BDA and parametric architectures.

5 Conclusions

This paper has presented an approach to contain the exploding needs of computing resources based on an increase of the computing power of the primitive level. The parametric architecture associated to the weighted primitive guarantees computing improvements for the HT and FT calculation because of the simplicity in hardware requirements due to the compactness of the primitive. When comparing with other well-known proposals, it has been confirmed that our approach provides memory and hardware resource saving as well as satisfying trade-off between area and speed.

References

1. Muamar, H.K and Nixon, M.: Tristage Hough Transform for multiple ellipse extraction. *IEEE Proc. Computer and Digital Techniques*, Vol. 138, 1 (1991) 27–35
2. Haule, D.D and Malowany, A.S.: Object Recognition using fast adaptative HT. *IEEE Comp. Pacific Conf. On Communication, Compiler and Signal Processing* (1989) 91–94
3. da Silva, I.: Vectorization from aerial photographs applying the HT method. *Proc. SPIE*, Vol. 1395, 2 (1990) 956–963
4. Hu, Y.H.: CORDIC-based VLSI architectures for Digital Signal Processing. *IEEE Signal Processing Magazine*, 7 (1992) 16–35
5. Villalba, J.: Diseño de Arquitecturas CORDIC multidimensionales. Tesis Doctoral Dept. de Arquitectura de Computadores. Universidad de Málaga (1995)
6. Chamberlain, R. and Lord, E.: Real-time 2D floating-point fast Fourier transforms for seeker simulation. *Proceedings SPIE*, Vol. 4717, (2002) 15–23
7. Yan, P. and Liu, H.: Image restoration based on the discrete fraction Fourier transform. *Proceedings SPIE*, Vol. 4552, (2001) 280–285
8. Chang, C.H., Wang, C.L. and Chang, C.H.: Efficient VLSI architectures for fast computation of the discrete Fourier transform and its Inverse. *IEEE Transactions on Signal Processing*, Vol. 48, 11 (2000) 3206–3216
9. Hsiao, S.F.; Shiue, W.R.: Design of low-cost and high throughput linear arrays for DFT computations: algorithms, architectures and implementations. *IEEE Transactions on Circuits and Systems II*, Vol. 47, 11 (2000) 1188–1203
10. Cooley, J.W. and Tukey, J. W.: An algorithm for the machine calculation of complex Fourier series. *Math. Comput.* 19, (1965) 297–301
11. Swartztrauber, E.E.: Multiprocessor FFTs. *P.N. Parallel Computing*, 5 (1987) 197–210
12. Chang, T.S., Guo, J.T and Jen, C.W.: Hardware Efficient DFT Designs with Cyclic Convolution and Subexpression Sharing. *IEEE Transactions on CAS II*, Vol. 47, 9 (2000) 886–892
13. Fang, W.H. and Wu M.L.: An efficient unified systolic architecture for the computation of discrete trigonometric transform. *Proceedings ISCAS*, (1997) 2092–2095
14. González, R.C.; Woods, R.E.: *Digital Image Processing*, Prentice Hall (2002)
15. Katsuhiko, O.: *Modern Control Engineering*, Prentice Hall (2001)
16. Serra, J.: *Image Analysis and Mathematical Morphology*. Academic Press, London (1982)
17. Kittler, J., Alkoot, F. M.: Sum versus Vote Fusion in Multiple Classifier Systems, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, 1 (2003) 110–115
18. Ercegovac, M. and Lang, T.: *Division and Square root: Digit-Recurrence, Algorithms and Implementations*, Kluwer Academic Pub. (1994)

19. Piñeiro, J.A., Bruguera, J.D. and Muller. J.M.: Faithful powering computation using table look-up and a fused accumulation tree. Proc. of the 15th International Symposium of Computer Arithmetic (ARITH'15) (2001)
20. Deng, Dixon, D.S and El Gindy H.: High speed Parametrizable HT using reconfigurable hardware. Pan-Sydney Area Workshop on Visual Information Processing (VIP) (2001)
21. Chien-Chang L., Chih-Da, Ch.: A parametrized hardware design for the variable length discrete Fourier transform. 15th International Conference on VLSI design (2002)
22. Volder, J.E.: The CORDIC trigonometric computing technique. IRE Trans. Elect. Comput., EC- 8 (1957) 330–334
23. Andraka, R.: A survey of CORDIC algorithms for FPGA. Proceedings of the ACM/SIGDA sixth international symposium of Field Programmable Gate Arrays (1998) 191–2000

Topological Active Nets Optimization Using Genetic Algorithms

O. Ibáñez, N. Barreira, J. Santos, and M.G. Penedo

Computer Science Department, University of A Coruña, Spain
{nbarreira, santos}@udc.es

Abstract. The Topological Active Net (TAN) model is a deformable model used for image segmentation. It integrates features of region-based and edge-based segmentation techniques. This way, the model is able to fit the edges of the objects and model their inner topology. The model consists of a two dimensional mesh controlled by energy functions. The minimization of these energy functions leads to the TAN adjustment.

This paper presents a new approach to the energy minimization process based on genetic algorithms (GA), that defines several suitable genetic operators for the optimization task. The results of the new GA approach are compared to the results of a greedy algorithm developed for the same task.

Keywords: Active Nets, Topological Active Nets, Genetic Algorithms.

1 Introduction

Deformable models are well-known tools for image segmentation. They were proposed by Kass et al. [1] in 1988 applied in several fields such as segmentation, mapping or tracking and motion analysis. The active nets model was proposed by Tsumiyama et al. [2] as a variant of the deformable models that integrates features of region-based and boundary-based segmentation techniques. To this end, active nets distinguish two kind of nodes: internal nodes, related to the region-based information, and external nodes, related to the boundary-based information. The former models the inner topology of the objects whereas the latter fits the edges of the objects. The Topological Active Net (TAN) model was developed as an extension of the original active net model [3,4]. It solves some intrinsic problems to the deformable models such as the initialization problem. It also has a dynamic behavior that allows topological local changes in order to perform accurate adjustments and find all the objects of interest in the scene. The model deformation is controlled by energy functions in such a way that the mesh energy has a minimum when the model is over the objects of the scene. This way, the segmentation process turns into a minimization task.

On one hand, it is known that the local search methods, and specially those that use a greedy strategy, have a certain probability to fall in local minima or maxima. On the other hand, it will be interesting to use global search methods to minimize that probability. Evolutionary methods, with a parallel search based

on the points defined by their populations of individuals, fall in this category, and Genetic Algorithms (GA) [5,6] are the most used among the different evolutionary methods. With little previous work of their use in the optimization of deformable models, mainly in edge or surface extraction [7,8,9], we will use this alternative in this work, with the attempt to overcome the difficulties found by a local greedy algorithm. We must define and adapt the classical genetic operators to deal with our problem, with emphasis in genetic operators that produce TANs with no crossings in their definition nodes. Finally, the experiments must prove the superiority of a GA in the minimization of the probability of getting stuck in local minima.

This paper is organized as follows. Section 2 explains the basis of the Topological Active Nets model. Section 3 introduces the GA approach proposed in this paper. Section 4 shows some results of the new model. Finally, section 5 expounds the conclusions and future work in this field.

2 Topological Active Nets

A Topological Active Net (TAN) is a discrete implementation of an elastic two dimensional mesh with interrelated nodes [3]. The structure of a small TAN is depicted in Fig. 1. As this figure shows, the model has two kinds of nodes: internal and external. Each kind of node represents different features of the objects: the external nodes fit the edges of the objects whereas the internal nodes model the internal topology of the object. Therefore, this model allows the integration of information based on discontinuities and information based on regions in the segmentation process. The former is associated to external nodes and the latter to internal nodes.

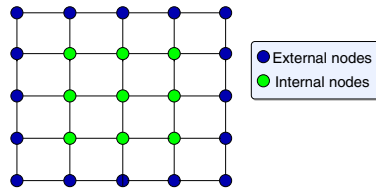


Fig. 1. A 5 × 5 mesh

A Topological Active Net is defined parametrically as $v(r, s) = (x(r, s), y(r, s))$ where $(r, s) \in ([0, 1] \times [0, 1])$. The mesh deformations are controlled by an energy function defined as follows:

$$E(v(r, s)) = \int_0^1 \int_0^1 E_{int}(v(r, s)) + E_{ext}(v(r, s)) dr ds \tag{1}$$

where E_{int} and E_{ext} are the internal and the external energy of the TAN, respectively. The internal energy controls the shape and the structure of the mesh

whereas the external energy represents the external forces which govern the adjustment process.

The internal energy depends on first and second order derivatives which control contraction and bending, respectively. The internal energy term is defined by the following equation:

$$E_{int}(v(r, s)) = \alpha(|v_r(r, s)|^2 + |v_s(r, s)|^2) + \beta(|v_{rr}(r, s)|^2 + |v_{rs}(r, s)|^2 + |v_{ss}(r, s)|^2) \tag{2}$$

where subscripts represents partial derivatives. α and β are coefficients that control the first and second order smoothness of the net. In order to calculate the energy, the parameter domain $[0, 1] \times [0, 1]$ is discretized as a regular grid defined by the internode spacing (k, l) and the first and second derivatives are estimated using the finite differences technique.

The external energy represents the features of the scene that guide the adjustment process. It is defined by the following equation:

$$E_{ext}(v(r, s)) = \omega f[I(v(r, s))] + \frac{\rho}{|\mathfrak{N}(r, s)|} \sum_{p \in \mathfrak{N}(r, s)} \frac{1}{\|v(r, s) - v(p)\|} f[I(v(p))] \tag{3}$$

where ω and ρ are weights, $I(v(r, s))$ is the intensity value of the original image in the position $v(r, s)$, $\mathfrak{N}(r, s)$ is the neighborhood of the node (r, s) and f is a function, which is different for both types of nodes since the external nodes fit the edges whereas the internal nodes model the inner features of the objects.

If the objects to detect are dark and the background is bright, the energy of an internal node will be minimum when it is on a point with a low grey level. On the other hand, the energy of an external node will be minimum when it is on a discontinuity and on a light point outside the object. In this situation, function f is defined as:

$$f[I(v(r, s))] = \begin{cases} h[\overline{I(v(r, s))}_n] & \text{for internal nodes} \\ h[I_{max} - \overline{I(v(r, s))}_n + \xi(G_{max} - G(v(r, s)))] & \text{for external nodes} \end{cases} \tag{4}$$

where ξ and δ are weighting terms, I_{max} and G_{max} are the maximum intensity values of image I and the gradient image G , respectively, $I(v(r, s))$ and $G(v(r, s))$ are the intensity values of the original image and the gradient image in node position $v(r, s)$, $\overline{I(v(r, s))}_n$ is the mean intensity in a $n \times n$ square and h is an appropriate scaling function.

As a broad outline, the adjustment process consists in minimizing these energy functions. First, the mesh is placed over the whole image and, then, the energy of each node is minimized using a greedy algorithm. In each step of the algorithm, the energy of each node is computed in its current position and in its nearest neighborhood. The position with the lowest energy value is selected as the new position of the node. The algorithm stops when there is no node in the mesh that can move to a position with lower energy.

The greedy algorithm gets good results in most cases since it takes the best local adjustment. Nevertheless, this local adjustment may not be the best global

one. Moreover, the greedy algorithm does not consider any possible alternatives. This way, if the model reaches a wrong segmentation, it gets stuck in it. These are the main drawbacks of using a greedy algorithm to minimize the energy in a deformable model.

3 Genetic Algorithms for TAN Optimization

In this paper, we have used a standard GA [6] with new defined operators. A TAN chromosome has two genes for each TAN node, one for its x coordinate and another for its y coordinate, both encoded as integer values. A tournament selection was used as selection method, with different tournament window sizes to control the selection pressure. But the key issues are the features of the genetic operators developed.

3.1 Crossover Operator

The classical crossover operator, with one-point, n-point or uniform crossover, is not very useful because a great number of genotypes, this is, the results of an interchange of genes from two selected parent chromosomes, are not correct TANs, as they present a high number of crossings in their definition nodes. These incorrect chromosomes could be eliminated of the genetic population due to their greater energy with respect to similar correct ones. Nevertheless, a great number of incorrect TANs in the population can complicate and slow down the evolutionary process. There are other alternatives that reduce the probability of incorrect TANs, such as the one used by Sakaue in [10], with a crossover operator that interchanges sub-rectangles of the two parents from an identical definition of those sub-areas in both parents. This operator restricts the crossings, but does not eliminate them completely. This way, our results with similar operators show that a high number of incorrect genotypes are kept in the population and even an incorrect TAN can be the best final genotype of the evolutionary process.

Due to this, we make emphasis in operators that take into account the crossing restriction, this is, that produce correct TANs. One alternative is a crossover operator that defines the new genes as a mean between the corresponding values in the two parent chromosomes, this is, an *arithmetical crossover* that considers a linear combination of the parents. So, with the arithmetical crossover the new value for the x gene coordinate in the new chromosome will be:

$$x_{new} = \alpha x_1 + (1 - \alpha)x_2 \quad (5)$$

where x_1 and x_2 are the values of the corresponding parent genes. If α is in the interval $[0, 1]$, the result TANs are correct, this is, with no crossings. The new TANs can be more or less similar to the parents and the distances among the nodes will be proportional to the distances of the correct parents.

3.2 Mutation Operator

As with the previous operator, the aim is to obtain TANs without crossings. We have tested two alternatives to this end.

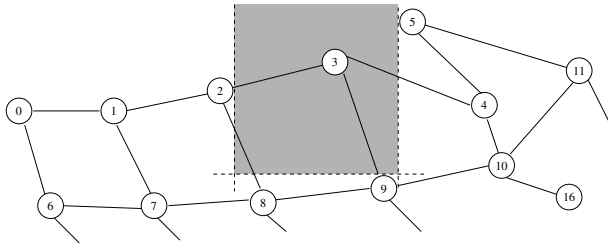


Fig. 2. First mutation alternative. The shadowed area represents the mutation boundaries for node number 3.

The first one mutates the node coordinates with boundary restrictions that limit the possible node positions. The mutation boundaries of a node are set from the coordinates of its neighboring nodes. So, for a given node, if one of its coordinates is chosen to mutate in a direction according to the mutation probability, the new coordinate cannot exceed the nearest coordinate of its 3 neighboring nodes in that direction. For an external node, if it has not 3 neighboring nodes, we consider some additional nodes. For example, if there are only two nodes in the direction of movement, we consider another node, a “neighbor of the neighbor” in that direction, as Fig. 2 shows. If node number 3 mutates towards positive values in the x coordinate, nodes 10, 4, and 5 set the boundary. If the node mutates towards positive values in the y coordinate, we impose a predetermined limit.

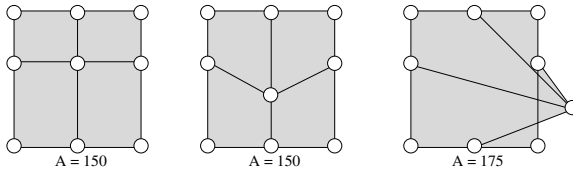


Fig. 3. Second mutation alternative. It uses the area of the polygons formed by the 8 neighboring nodes and the central node that mutates. The left figure shows the area of the 4 polygons before the mutation. The central figure shows a correct mutation. In this case, the addition of the areas remains unchanged. The right figure is an example of an incorrect mutation.

The second alternative allows that a node can mutate to all the possible positions without any crossing in the TAN. The basic idea is to compute the area of the 4 polygons formed by the 8 neighboring nodes and the central node that mutates, as Fig. 3 shows. If the addition of the 4 sub-areas is the same before and after the mutation, the mutation is correct as it will not produce any crossing. Although this method has a higher computational cost than the first one, its computational cost is lower than other methods, such as checking if there are crossings after a mutation.

3.3 *Ad hoc* Operators

We have introduced other operators, the spread operator and the group mutation. These operators were specially designed in order to avoid the difficulties found in the application of the GA to our problem and to improve the evolutionary optimization process.

Spread Operator. One problem of the proposed crossover operator, that calculates the mean value between the parent genes, is that the new TANs tend to have less size than their parents. With the aim of maintaining the diversity of sizes in the population, we have implemented the spread operator. The main idea is to stretch the TAN in a given direction. For example, if we stretch the TAN towards positive values of x , we maintain invariable the y positions of all the nodes, but the distance between two neighboring nodes in the x direction is increased with the same random factor since the nodes change their x coordinates, except for the nodes in the first left column.

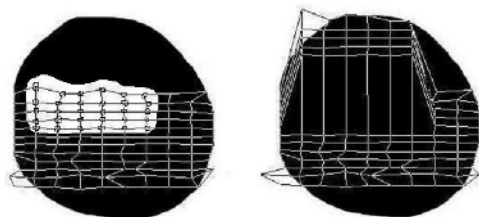


Fig. 4. Group mutation. The right figure shows the results of a group mutation of the nodes in the white sub-area of the left figure. The left TAN is the best individual of the 80th generation whereas the right TAN is the best individual of the 100th generation after the group mutation. This mutated individual is the best of its population since the group mutation minimizes the total energy.

Group Mutation. In a great number of cases, the simple mutation of one node coordinate can help the TAN to be closer to the ideal image segmentation. However, the minimization of the internal energy tends to search for the equidistance among the nodes so that one mutation could not improve the TAN energy. Due to this, it can be useful to mutate simultaneously a group of neighboring nodes, in the same direction and with the same value. This group of nodes are delimited by two rows and two columns randomly chosen.

This idea is depicted in Fig. 4, where the group of nodes in the white sub-area is mutated in the same direction, producing a TAN with less total energy.

3.4 Additional Considerations

In addition to the energy terms used in the greedy algorithm (eq. 4), we include a new term in the external energy of the external nodes, the gradient distance.

The new equation is defined as follows:

$$f[I(v(r, s))] = \begin{cases} h[\overline{I(v(r, s))}_n] & \text{for internal nodes} \\ h[I_{max} - \overline{I(v(r, s))}_n + \xi(G_{max} - G(v(r, s)))] & \text{for external nodes} \\ + \delta GD(v(r, s)) & \text{nodes} \end{cases} \quad (6)$$

where $GD(v(r, s))$ is the distance from the position $v(r, s)$ to the nearest edge and δ is a coefficient that weights the gradient distance. This term introduces a continuous range in the external energy since its value diminishes as the node gets closer to an edge. This way, the gradient distance facilitates that the external nodes fit the object edges.

With the intention that the population maps could cover the image in the first generations, we have used two steps in the evolutionary process. In the first step, the energy parameters allow the nodes to be outside the image without a high penalization. Moreover, TANs initialized inside the object in the first generation population cannot survive in next generations. In the second step, the parameter values are changed in order to search for a more homogeneous distribution of the internal nodes, reduce the size of the map, and adapt it to the image.

4 Results

The genetic algorithm optimization has been tested with several 256 grey level images and the results were compared to the results of the greedy minimization. In all the examples, the same image was used as the external energy for the internal and external nodes.

In the results shown in this section, the GA used a tournament selection with a window size of 3% of the population. The probability of crossover was 0.5, whereas the probability of mutation was 0.0005, using only the 2nd alternative of

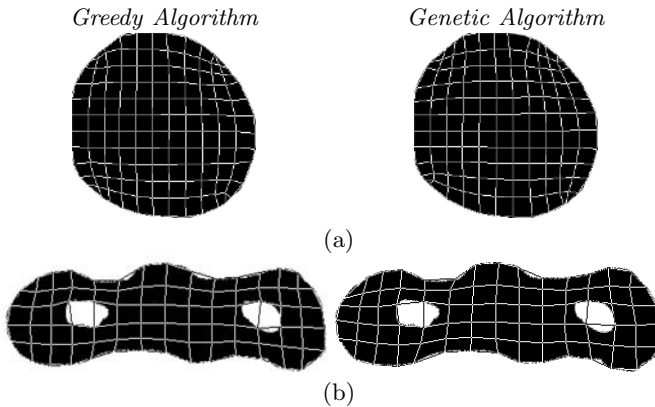


Fig. 5. Results obtained from simple objects

Table 1. TAN parameters used in the segmentation processes of Fig. 5. δ was used only in the GA segmentation process.

| <i>Example</i> | <i>Size</i> | α | β | ω | ρ | ξ | δ |
|----------------|----------------|----------|---------|----------|--------|-------|----------|
| 5(a) | 12×12 | 2 | 0.0005 | 2 | 1.5 | 3.5 | 6 |
| 5(b) | 15×6 | 2.5 | 0.0001 | 2 | 4 | 3 | 6 |

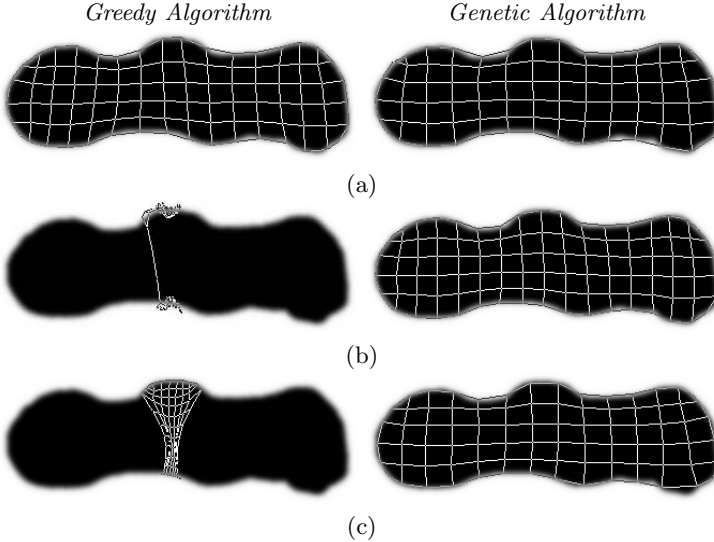


Fig. 6. Results obtained with different mesh sizes and parameter sets. (a) Reference segmentation. (b) Results after increasing the mesh size. (c) Results after changing the parameter set. The genetic algorithm is less sensitive to changes in the mesh size or parameter set.

Table 2. TAN parameter sets in the segmentation processes of Fig. 6. δ was used only in the GA segmentation process.

| <i>Example</i> | <i>Size</i> | α | β | ω | ρ | ξ | δ |
|----------------|---------------|----------|---------|----------|--------|-------|----------|
| 6(a) | 14×6 | 1 | 0.00001 | 4 | 1 | 4 | 5 |
| 6(b) | 17×6 | 1 | 0.00001 | 4 | 1 | 4 | 5 |
| 6(c) | 14×6 | 2 | 0.00001 | 2 | 1.5 | 3.5 | 5 |

mutation which involves the calculation of areas. Regarding the *ad hoc* operators, in the set of experiments carried out, we have experimentally set a good interval for the probability of the spread operator as $[0.001,0.0005]$ and $[0.005,0.0001]$ for the group mutation. The different examples in this section have used values in that intervals for both operators. Finally, the number of generations of the first evolutionary step is in the interval $[150,200]$.

Figure 5 shows several examples of TAN segmentation using greedy and GA approaches. The segmented objects are very simple so both methods produce

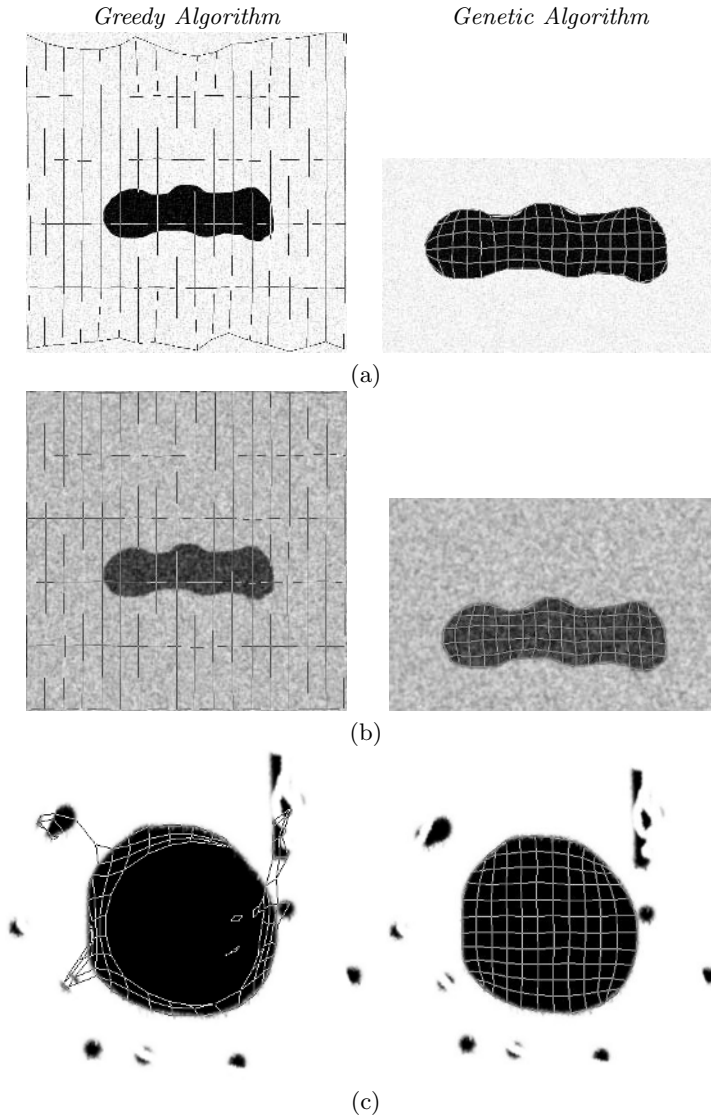


Fig. 7. Results obtained in noisy images. The genetic algorithm achieves the best adjustment independently of the amount and kind of noise. Opposite to this, the greedy algorithm is sensitive to noise and, so, it cannot segment the objects correctly.

a correct segmentation. Table 1 summarizes the TAN parameters used in the second evolutionary step of the segmentation processes. Regarding the first evolutionary step, the TAN parameters were $\beta = 0.00001$, $\rho = 1$, $\delta = 8$ in all the examples, whereas the remaining parameters were set to zero. Finally, in the GA segmentation, populations of 350 and 500 individuals were used for the segmentation of Fig. 5(a) and 5(b), respectively.

Table 3. TAN parameter sets in the segmentation processes of Fig. 7. δ was used only in the GA segmentation process.

| <i>Example</i> | <i>Size</i> | α | β | ω | ρ | ξ | δ |
|----------------|----------------|----------|---------|----------|--------|-------|----------|
| 7(a) | 18×6 | 0.5 | 0.1 | 1.5 | 1.5 | 4.5 | 6 |
| 7(b) | 18×6 | 1.2 | 0.1 | 2 | 1.5 | 4.5 | 6 |
| 7(c) | 12×12 | 1.5 | 0.00001 | 2 | 3 | 3 | 4 |

The mesh size and the parameter set of the TAN model are empirically selected since there is no general rule to choose the appropriate values in every image. In particular, the mesh size is based on the height and width of the object to segment, and remains constant in the two evolutionary steps.

This fact implies that the final segmentation depends on the selected parameter set and mesh size. As Fig. 6 shows, the greedy algorithm is very sensitive to the model variables. Opposite to this, the GA method gets good results independently of the mesh size and the parameter set. This is due to the fact that genetic algorithms usually find the best energy minimization since they take into account more alternatives (searching areas) than the greedy approach. Table 2 summarizes the TAN parameters in the examples. In all these genetic examples, we have used a population of 500 individuals.

The greedy method is also very sensitive to noise since the mesh can stop its contraction in image areas with a low S/N ratio. Also, with the genetic algorithm the mesh explores globally the search space and, then, it can avoid the local minima due to noise. Figure 7 shows the segmentation of some images with different kinds of noise. In these cases, the genetic algorithm produces a correct segmentation whereas the greedy algorithm stops the mesh minimization in local minima. Table 3 shows the parameters used in the segmentation of these noisy images. In the genetic segmentation, a population of 850 individuals has been used for the first example, 950 for the second one, and a population of 800 for the third example.

Regarding the computation times, the greedy method is faster than the GA. In an AMD Athlon at 1'2 GHz, the average execution time of the greedy algorithm is 5 seconds for a simple image and 30–60 seconds for a complex image whereas the GA spends 5–10 minutes in simple images and 15–25 in complex ones. These high execution times are mainly due to the large population sizes of the GA.

5 Conclusions

This paper presents a new approach to the energy minimization task in the Topological Active Nets model. Genetic algorithms are used in conjunction with the TAN model in order to find the lowest mesh energy, this is, the best fit to the scene objects. To this end, the TAN mesh has been coded as chromosomes, the crossover and the mutation operators have been implemented, and new operators have been proposed.

Table 4. Advantages and drawbacks of genetic and greedy algorithms

| Issue | Genetic Algorithm | Greedy Algorithm |
|---|-------------------|------------------|
| <i>Sensitive to changes in parameters</i> | No | Very sensitive |
| <i>Edge adjustment</i> | Very good | Very Good |
| <i>Sensitive to Noise</i> | No | Very sensitive |
| <i>Execution time</i> | High | Very low |

The genetic algorithm developed was tested with several images. In all the examples, the new method achieved a good adjustment to the objects. The results were even better than the results of the first approach to the minimization problem, this is, the greedy algorithm. Thus, the genetic algorithm is not sensitive to noise and it does not depend on the parameter set or the mesh size. The execution time is the main drawback of the genetic algorithm approach since it takes into account more alternatives than the greedy algorithm. Table 4 summarizes the features of genetic and greedy algorithms in the TAN minimization problem.

Future work includes the combination of genetic and greedy methods in order to take advantage of both methods and the parallelization of the genetic algorithm with the aim of reducing the execution times. Additionally, we will consider the energy components as independent objectives to optimize and, this way, we will use multiobjective evolutionary optimization techniques for the search of the optimal nets of the Pareto set.

References

1. M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(2):321–323, 1988.
2. K. S. Y. Tsumiyama and K. Yamamoto. Active net: Active net model for region extraction. *IPSJ SIG notes*, 89(96):1–8, 1989.
3. F. M. Ansia, M. G. Penedo, C. Mariño, and A. Mosquera. A new approach to active nets. *Pattern Recognition and Image Analysis*, 2:76–77, 1999.
4. F. M. Ansia, M. G. Penedo, C. Mariño, J. López, and A. Mosquera. Automatic 3D shape reconstruction of bones using active nets based segmentation. In *15th International Conference on Pattern Recognition*, volume 1, pages 486–489. IEEE Computer Society, 2000.
5. J. H. Holland. *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, Ann Arbor, MI, 1975.
6. David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.
7. L. Ballerini. Medical image segmentation using genetic snakes. In *Proceedings of SPIE: Application and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation II*, volume 3812, pages 13–23, 1999.
8. J. Tohka. Global optimization of deformable surface meshes based on genetic algorithms. In *ICIAP*, pages 459–464. IEEE Computer Society, 2001.
9. Y. Fan, T.Z. Jiang, and D.J. Evans. Volumetric segmentation of brain images using parallel genetic algorithm. *IEEE Transactions on Medical Imaging*, 21(8):904–909, 2002.
10. K. Sakaue. Stereo matching by the combination of genetic algorithm and active net. *Systems and Computers in Japan*, 27(1), 1996.

An Evaluation Measure of Image Segmentation Based on Object Centres

J.J. Charles¹, L.I. Kuncheva¹, B. Wells², and I.S. Lim¹

¹ School of Informatics, University of Wales, Bangor,
LL57 1UT, United Kingdom

² Conwy Valley Systems Ltd, United Kingdom

Abstract. Classification of organic materials obtained from rock and drill cuttings involves finding multiple objects in the image. This task is usually approached by segmentation. The quality of segmentation is evaluated by matching the whole detected objects to a reference segmentation. We are interested in representing each object by a single reference point called the “centre”. This paper proposes an evaluation measure of image segmentation for such representation. We argue that measures based only on distance between obtained centres and a set of predefined centres are insufficient. The proposed measure is based on a list of desirable properties of the segmentation. The three components of the measure evaluate the under/over segmentation of the objects, the proportion of centres placed in the background rather than in objects, and the distance between the guessed and the true centres. The ability of the measure to distinguish between segmentation results of different quality is illustrated on three sets of examples including an image containing microfossils and pieces of inert material.

Keywords: Image segmentation, evaluation measures, discrepancy methods, microfossils, palynomorphs.

1 Introduction

Image segmentation is a fundamental process within automatic image analysis. The large variety of practical applications has resulted in a spectrum of generic and specific segmentation algorithms being currently available. The choice of a segmentation algorithm suitable for the problem at hand is not simple. To aid this choice, here we propose a measure of quality of object segmentation using “centres” of the objects.

A survey was conducted by Zhang [1] where evaluation methods for image segmentation were categorised as analytical and empirical. Analytical methods examine the principles and properties of the segmentation algorithms themselves. Empirical methods evaluate the output of the segmentation algorithms on test images. Of the two groups, Zhang recommends the empirical methods.

The quality of segmentation of an image is judged by the so called “goodness” measure [1]. Examples of goodness measures are the entropy of the partitioned

image, intraregion uniformity, region shape, colour uniformity etc. Goodness measures are usually defined by human perception of the ideal segmentation. They are evaluated on the segmented image alone without requiring a reference segmentation, i.e. without a ground truth. Empirical methods that use goodness measures are known as *goodness methods*. Alternatively, the performances of segmentation algorithms can be evaluated relative to a ground truth; these types of methods are called *discrepancy methods*. They measure the inconsistency or some form of distance between the ground truth (ideal segmentation) and the actual segmented image.

There are four types of low level discrepancy approaches presented by Beauchemin and Thompson [2]:- pixel, area, point-pair and boundary. The pixel-based discrepancy approach is the most common one and consists of counting the number of misclassified pixels in the segmentation output relative to a reference partition. Using a similar approach Cardoso and Corte-Real [3] formulate a general measure, an important asset of which is that it is a metric. Area-based methods evaluate area of overlap between corresponding segments [4,5,6] while boundary-based schemes compare the perimeters of the segments. The point-pair discrepancy approach measures the agreement between two segmented images [7] without explicitly solving the correspondence problem between the regions. Segmentation produces a partition of the pixels and hence can be thought of as a clustering technique. Thus the discrepancy between the obtained segmentation and a ground truth segmentation can be evaluated by any measure of agreement or similarity between two partitions. Along with the Rand index, various other measures of similarity have been proposed in the literature, the most widely used being Jaccard index, adjusted Rand index, correlation, mutual information and entropy [7,8,9].

In this paper we propose an evaluation measure that belongs to the point-pair group within the discrepancy approach. The segmented image is represented as a set of coordinates of object centres. A reference image is used whereby the centres are marked by hand. The proposed measure consists of three indices evaluating different aspects of the positioning of centres. The rest of the paper is organised as follows. Section 2 describes the real-life problem which prompted this study. The new measure is proposed in Section 3. Toy examples and real images are used to demonstrate the insufficiency of simple distance metrics for evaluating the match between manually placed centres and centres obtained from automatic segmentation. Experiments with microscopy images of palynomorphs are reported in Section 4. Section 5 concludes the study.

2 Segmentation of Microscopy Images of Palynofacies

Microfossil analysis is an essential task in micropaleontology, aiding the interpretation of the history of regional and global climate and of the evolution of the biosphere [10]. Figure 1 shows a typical image of a slide containing microfossils and other organic debris. There are hundreds of types of microfossil and each image may contain many of them. The end task is to locate and subsequently classify the microfossils in the image.

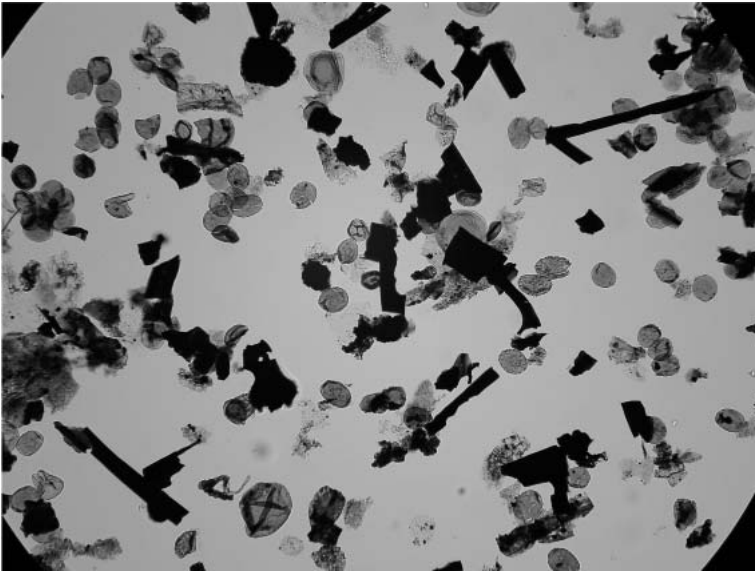


Fig. 1. Thin section of rock containing microfossils

Figure 1 demonstrates the difficulties in locating the objects. First, pieces of inert material (darker objects of irregular shapes and sizes) have to be eliminated so that the image contains only the microfossils. Second, the microfossils appear in different orientation, partly or completely overlapping, clustered together, overshadowed by inert material, distorted by pressure forces in the rock, etc. Third, the colour of some of the microfossils is barely distinguishable from the image background. Fourth, the background is not uniform across the image. Pixel intensities which correspond to background at some places may well be classed as object at a different place in the image.

Although the different types of fossils have different texture and structure, they can be roughly perceived as round or elliptical objects. Hence we are interested in finding “centres” so that we can crop sub-images of microfossils at a later stage applying a higher resolution.

Before object segmentation is attempted, the background must be removed. The difficulty in our case comes from the specific illumination of microscope slides. Usually the centre of the slide is brighter and the intensity fades towards the edges. Also, since the microscopic view is a circle, dark corners might appear. The approach adopted here was to look for and eliminate dark corners, model the background as a function of x and y and remove it from the image. A parabola was fitted to model the intensity of the background for each x and then for each y . The pixels whose true intensity was substantially lower than the scores on both parabolas were marked as non-background (intensity margin of 20 was empirically chosen here).

The measure proposed in the next section is intended for any segmentation method which produces a set of object centres. The standard watershed segmentation was tried as well as a recently developed algorithm called floodfill segmentation [11]. Here we look for a measure to compare object segmentation methods on the basis of their output.

3 Evaluation Measure for Segmentation Methods

Here we construct an empirical discrepancy measure to determine how close two segmented images are.

3.1 Definition of “Centre” of an Object

Suppose that a distance transform is applied to a black and white image so that each pixel, p , is associated with a function $D(p)$ [12]. The value of $D(p)$ gives the Euclidean distance from p to the nearest white pixel.

Definition 1. A *centre* of an object is the pixel p with the largest distance $D(p)$ within the object. If there is a tie, any of the tied pixels can be chosen to be the centre.

Note that this definition does not imply that the centres will be positioned at the centres of gravity of the objects. Also, one object may have infinite amount of candidate centres with the same highest $D(p)$. This will happen, for example, in a ring-shaped object. The geometric position of points at equal highest $D(p)$ will be a circle with radius equal to the average of the inner and outer radii of the ring. Any point on the circle will be a valid centre according to Definition 1.

The obtained centres are required for extracting the microfossils with a higher resolution for the purposes of subsequent classification. However, more generally, centres of objects may be required for other purposes such as setting the initial position of an active contour [13] or object tracking within moving images [14]. The centres provide a handle for our evaluation method.

3.2 Comparison of Sets of Centres Based on Distances

Let $C^* = \{c_1^*, \dots, c_n^*\}$ be the true centres and $C = \{c_1, \dots, c_m\}$ be the centres obtained through automatic segmentation. The most intuitive matching measure would be the sum of distances to the nearest centre. Let

$$R(C^* \rightarrow C) = \sum_{i=1}^n \min_{j=1}^m \{\text{dist}(c_i^*, c_j)\} \quad (1)$$

be the representation of C^* by C . In the ideal case where the two sets of centres are identical, $C^* = C$, we have

$$R(C^* \rightarrow C) = R(C \rightarrow C^*) = 0.$$

If $C \subset C^*$, we have under-segmentation. In this case $R(C^* \rightarrow C) > 0$ and $R(C \rightarrow C^*) = 0$. If $C^* \subset C$, the image is oversegmented, $R(C^* \rightarrow C) = 0$ and $R(C \rightarrow C^*) > 0$. To account for both under- and over-segmentation, and also for the discrepancies in the centre location, we can use the following measure

$$M_d(C^*, C) = R(C^* \rightarrow C) + R(C \rightarrow C^*) \tag{2}$$

$$= \sum_{i=1}^n \min_{j=1}^m \{ \text{dist}(c_i^*, c_j) \} + \sum_{j=1}^m \min_{i=1}^n \{ \text{dist}(c_j, c_i^*) \} \tag{3}$$

Here “dist” can be any distance. In the illustration below we use Euclidean and City-block distances. The smaller the value of M_d , the more similar the two segmentations are. We note that M_d is a metric on the space of sets of centres because $M_d(C^*, C) \geq 0$ with $M_d(C^*, C) = 0$ iff $C^* = C$ (nonnegativity); $M_d(C^*, C) = M_d(C, C^*)$ (symmetry) and it can be proved that $M_d(C_1, C_3) \leq M_d(C_1, C_2) + M_d(C_2, C_3)$ (triangle inequality), where C_1, C_2 and C_3 are sets of centres.

The problem with the distance-based measures is that they do not take into account the specific objectives of segmentation. An example is shown in Figure 2. The true centre of the grey object, according to Definition 1, is situated in the point with the largest $D(p)$ (marked with ‘x’). Two guessed centres are displayed in the figure. Clearly Guess 1 is closer to the true centre and any distance measure will favour it over Guess 2. However, Guess 2 sits on the next highest peak of $D(p)$ in the object and is a much better representation of the object than Guess 1. This deficiency of the distance-based measures is addressed by the measure proposed below.

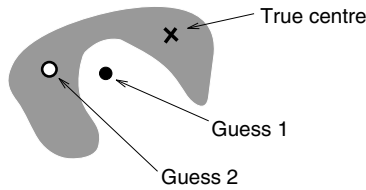


Fig. 2. Examples of a true and two guessed centres

3.3 Comparison of Sets of Centres Based on Segmentation Heuristics

The three most desirable properties of a segmented image can be specified as follows

- A perfectly segmented image exhibits no under- or over-segmentation.
- There are no centres of objects which lie outside objects boundaries.
- The centre of each segment should coincide with the relevant object centre.

We shall assume that a reference segmentation is available to represent the ground truth. We also assume that the segmentation process returns a set of m centres, C . We propose to evaluate the quality of segmentation by the following three measures.

Definition 2. Let n_i be the number of centres placed by the automatic segmentation within object i (the object is defined by the ground truth segmentation). The measure of *under- or over-segmentation* of i is

$$r_i = \begin{cases} 1 - 1/n_i, & \text{if } n_i > 0 \\ 1, & \text{if } n_i = 0 \end{cases} \quad (4)$$

If there is no centre in the object or if there are large number of centres there, r_i approaches 1. The most desirable value of r_i is 0 which is achieved if there is only one centre in the object.

Definition 3. The measure of *background segmentation* is

$$v = 1 - \frac{1}{m} \sum_{i=1}^n n_i. \quad (5)$$

Note that v is the proportion of automatic centres that are not contained within the boundaries of any objects. Thus $v = 0$ corresponds to the ideal situation where the background is free of centres placed by mistake by the segmentation algorithm.

The values r and v completely represent the under and over-segmentation of the image regardless of the location of the centres within the objects. Hence the third measure evaluates how close the approximations are to the ideal centres within the objects.

Definition 4. Let c_i^* be the true centre of object i and let $c' \in C$ be the nearest centre from the automatic segmentation which lies within object i , i.e.,

$$c' = \arg \min_{c \in \text{object } i} \{\text{dist}(c_i^*, c)\} \quad (6)$$

The *centre discrepancy* is defined as

$$q_i = 1 - D(c')/D(c_i^*), \quad (7)$$

where $D(c)$ is the distance transform value for point c . By Definition 1, $D(c_i^*)$ is the maximum distance within object i therefore $D(c') \leq D(c_i^*)$, and $q_i \in [0, 1]$. Approximated centres near the boundaries of objects should be assigned high discrepancy value, q_i , while those in the middle of objects should be assigned a low discrepancy value.

To illustrate the rationale for introducing the centre discrepancy, q_i , consider an elongated object as shown in Figure 3. There are infinitely many possible centres, according to Definition 1, situated along the ridge of the distance function for this object.

Thus any centre on the ridge should have a lower error value q_i than centres on the edge of the object. Figure 3 shows that Euclidean distance will be misleading in this case as a centre at the periphery will be preferred to one of the true centres with the largest $D(p)$.



(a) Small q_i , large Euclidean distance (preferred) (b) Large q_i , small Euclidean distance

Fig. 3. Illustration of the advantage of the centre discrepancy measure q_i over Euclidean distance in evaluating an approximated centre (circle) with respect to the true centre (cross)

The three measures r_i , v and q_i can be combined so that the quality of the segmentation is measured by a single value.

Definition 5. The measure of *quality of segmentation* represented by the set of centres C with respect to a ground truth segmentation with a set of centres C^* is

$$S(C, C^*) = \frac{1}{3} \left(v + \frac{1}{n} \sum_{i=1}^n (r_i + q_i) \right), \tag{8}$$

where r_i , v and q_i are calculated as in Definitions 2-4, respectively.

4 Experimental Results

4.1 A Single-Object Illustration

We start with a simple example showing why distance-based measures $M_d(C, C^*)$ are insufficient for the purposes of segmentation. Figure 4 displays seven copies of an image containing a single elliptical object with different nonempty sets of centres C . In all 7 cases the set C^* consists of one element which is the geometrical centre of the ellipse.

Three measures of quality of the segmentation represented by the centres C (dots) are shown in the table:- $M_d(C, C^*)$ for Euclidean and City-block distance and the quality of segmentation $S(C, C^*)$ as in Definition 5. The rows in the table are sorted with respect to $S(C, C^*)$ starting with the best case (minimum, $S = 0$) and ending with the worst case (maximum, $S = 1$). There are discrepancies in the ranking of the 7 cases according to the three measures. While the trivial case where the single guessed centre coincides with the true centre is the most preferred case for all the measures, $S(C, C^*)$ disagrees with both distance-based measures M_d about the least preferred case. Based on distances, case 4 is the worst because the guessed centre is far from the true centre. However, it is important for our segmentation purposes that the centres lie within the objects.








| No | Image (one object) | M_d Euclidean | M_d City-block | r | v | q | S |
|----|---|--------------------|---------------------|------|------|------|------|
| 1 |  | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 2 |  | 1.43 | 1.90 | 0.00 | 0.50 | 0.00 | 0.17 |
| 3 |  | 3.70 | 3.70 | 0.50 | 0.00 | 0.00 | 0.17 |
| 4 |  | 7.40 | 7.40 | 0.00 | 0.00 | 0.70 | 0.23 |
| 5 |  | 2.41 | 3.00 | 0.50 | 0.33 | 0.00 | 0.28 |
| 6 |  | 2.95 | 3.80 | 0.00 | 0.50 | 0.80 | 0.47 |
| 7 |  | 4.88 | 6.80 | 1.00 | 1.00 | 1.00 | 1.00 |

Fig. 4. Seven examples of segmentation of one object using centres and the values of the measures of quality of the segmentation. The true centre is marked with ‘ \times ’ and the guessed centres are marked with ‘ \bullet ’.

Thus case 7 should be the least preferable one because the object has not been found as the single centre lies in the background. Also, the oversegmented case 5 will be preferred to case 3 by both distance measures. In both cases there is a perfect centre within the set C . Note that there is an extra centre in the background in case 5, while there is no such centre in case 3. The disagreement between the two distance-based measures is minimal. They rank differently only cases 3 and 6. The major flaw of the distance measures is that they do not take into account any object boundaries. Hence the advantage of M_d over S would be speed of calculations. However, the better match with the desiderata for segmentation quality leads us to choose S .

4.2 A Multiple-Object Illustration

An example involving multiple objects is considered next. Figure 5 (a) presents the objects and the ideal segmentation. Three special cases are shown in plots (b), (c) and (d). The centres are placed manually in all three images.

For the task of classifying fossils cropped around the centres, missing an object should be penalised stronger than placing an extra centre in the background. In the former case, an important piece of information may be overlooked. In the latter case the analysis time will increase due to the extra centres of nonexistent objects but all the microfossils will be detected in the image. For this reason, the undersegmented image (c) should be ranked worse than (b) and (d) should

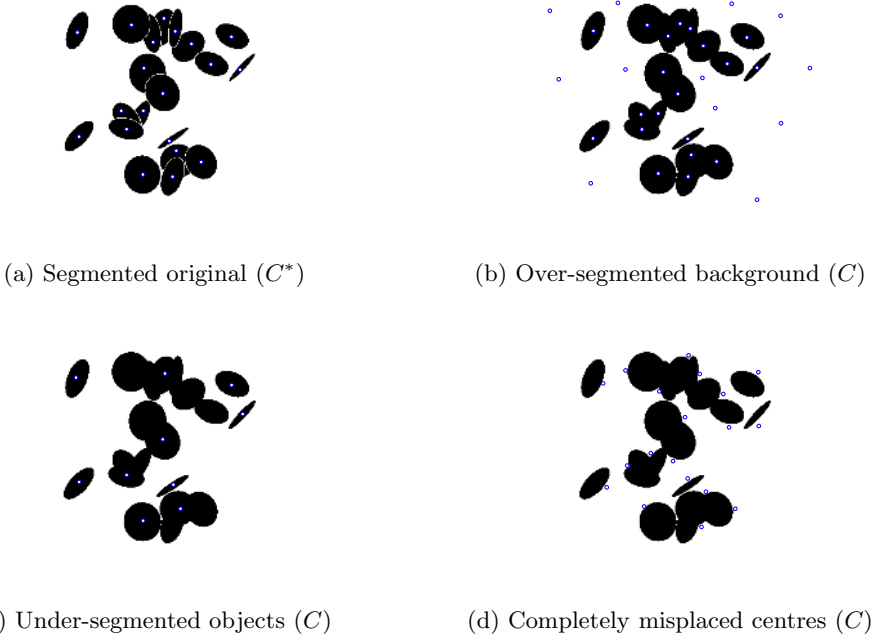


Fig. 5. The original segmentation of a multiple object image and three special cases of segmentation (centres have been placed manually in (b), (c) and (d))

Table 1. Measures of segmentation quality for the images in Figure 5 (b), (c) and (d) with respect to the ground truth (a)

| Subplot | M_d | M_d | r | v | q | S |
|----------------------------------|-----------|------------|------|------|------|------|
| | Euclidean | City-block | | | | |
| (b) Over-segmented background | 1180 | 1513 | 0.00 | 0.18 | 0.38 | 0.18 |
| (c) Under-segmented objects | 585 | 735 | 0.50 | 0.61 | 0.00 | 0.37 |
| (d) Completely misplaced centres | 1068 | 1327 | 1.00 | 1.00 | 1.00 | 1.00 |

be ranked worse than (c). Table 1 displays the three measures M_d -Euclidean, M_d -City block and S for the images in plots (b), (c), and (d).

The table shows that the distance-based measures fail to produce the required ranking while S clearly distinguishes between the three cases.

4.3 Results on Microfossil Images

The results from applying the watershed [15] and floodfill [11] segmentation methods on the image shown in Figure 1 are displayed in Figures 6 and 7, respectively. Table 2 compares the two segmentation approaches. The distance-based measures M_d -Euclidean and M_d -City block agree with S that the floodfill method performs better than the watershed algorithm. There is a considerable difference in the score of the distance-based measures between the two types of

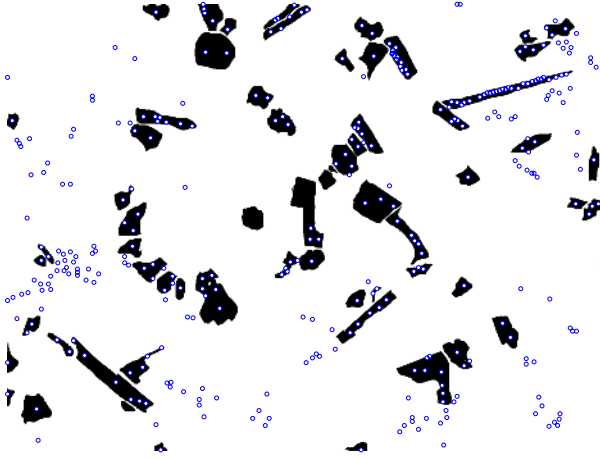


Fig. 6. Manually segmented inert material from Figure 1 overlaid with centres found through the *watershed* segmentation method

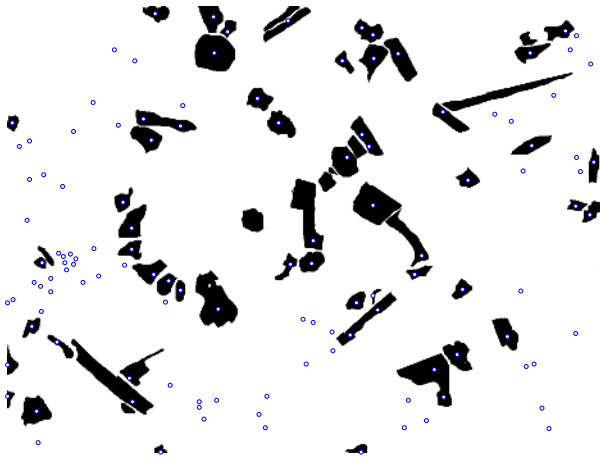


Fig. 7. Manually segmented inert material from Figure 1 overlaid with centres found through the *floodfill* segmentation method

segmentations indicating that these two approaches are very different, whereas S shows that floodfill method only slightly improves over the watershed method. This corresponds precisely to how we would interpret the data by visual inspection. The watershed algorithm (Figure 6) has failed to capture 3% of the objects, with most of the captured objects being oversegmented. The floodfill algorithm (Figure 7) has failed to capture 13% of the objects but the detection was

Table 2. Segmentation quality of watershed and floodfill segmentation of Figure 1

| Method | M_d | M_d | r | v | q | S |
|-----------|-----------|------------|------|------|------|------|
| | Euclidean | City-block | | | | |
| Watershed | 35407 | 44784 | 0.56 | 0.45 | 0.45 | 0.49 |
| Floodfill | 13254 | 16783 | 0.28 | 0.53 | 0.39 | 0.40 |

completed with nearly no oversegmentation. The watershed method makes up for its oversegmentation by achieving a high capture rate. The small difference between the two values of measure S in Table 2 account for the fact that both methods have assets and flaws and a choice between the two cannot be made with high certainty.

5 Conclusions

This paper proposes a new measure of the quality of segmentation of images containing objects. The measure operates on a set of ideal centres of the objects, C^* , assumed to be the ground truth and a set of guessed centres, C , obtained through a segmentation algorithm. Thus the proposed measure falls into the *discrepancy* category of evaluation methods, as detailed by Zhang [1].

Based on a list of desired properties and examples with generated and real images, we argue that the proposed measure of quality, $S(C, C^*)$ is better than measures based on the distances (M_d) between the centres in sets C and C^* . The three components of the proposed measure comply with the intuition for evaluating segmentation results represented by centres. Another advantage of S over M_d is that S has practically useful lower and upper limits while the distance-based measures have only a lower limit. A disadvantage of S is that it is slower to calculate than M_d because it requires knowledge of the objects in the image. We are more concerned with locating the inert material through segmentation rather than extracting the material, so even though two different segmentations can result in the same set of centres the value S will still provide an accurate comparison between segmentations.

The intended application of S is for choosing a segmentation method and tuning it for the specific practical application. In our case, a small number of images will be labelled manually by an expert palynologist, and used as the ground truth. After tuning, the segmentation method will be applied as a standard routine to find the microfossils in other images coming from the same domain.

Acknowledgements

The EPSRC CASE grant Number CASE/CNA/05/18 is acknowledged with gratitude.

References

1. Zhang, Y.J.: A survey on evaluation methods for image segmentation. *Pattern Recognition* **29**(8) (1996) 1335–1346
2. Beauchemin, M., Thomson, K.P.B.: The evaluation of segmentation results and the overlapping area matrix. *Remote Sensing* **18**(18) (1997) 3895–3899
3. Cardoso, J.S., Corte-Real, L.: Toward a generic evaluation of image segmentation. *IEEE Transactions on Image Processing* **14**(10) (2005) 1773–1782
4. Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P., Bunke, H., Goldgof, D., Bower, K., Eggert, D., Filtzbiggbon, A., Fisher, R.: An experimental comparison of range image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **18** (1996) 673–689
5. Borsotti, M., Campadelli, P., Schettini, R.: Quantitative evaluation of color image segmentation results. *Pattern Recognition Letters* **19** (1998) 741–747
6. Liu, J., Yang, Y.H.: Multiresolution color image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **16**(7) (1994) 689–700
7. Rand, W.: Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association* **66** (1971) 846–850
8. Elisseeff, A.B.H.A., Guyon, I.: A stability based method for discovering structure in clustered data. In: *Proc. Pacific Symposium on Biocomputing*. (2002) 6–17
9. Hubert, L., Arabie, P.: Comparing partitions. *Journal of Classification* **2** (1985) 193–218
10. Bollmann, J., Quinn, P.S., Vela, M., Brabec, B., Brechner, S., Cortes, M.Y., Hilbrecht, H., Schmidt, D.N., Schiebel, R., Thierstein, H.R.: Automated particle analysis: Calcareous microfossils. *Image Analysis, Sediments and Paleoenvironments* **7** (2004) 229–252
11. Charles, J.J., Kuncheva, L.I., Wells, B., Lim, I.S.: Object location within microscopic images of palynofacies. (2006) (submitted).
12. Borgfors, G.: Distance transformations in digital images. *Computer Vision Graphics and Image Processing* **34**(3) (1986) 344–371
13. Wang, Y., Chou, J.: Automatic segmentation of touching rice kernels with an active contour model. *Transactions of the ASAE* **47**(5) (2004) 1803–1811
14. Paul, G.V., Beach, G.J., Cohen, C.J.: A realtime object tracking system using a color camera. *Applied Imagery Pattern Recognition Workshop* (2001) 137–142
15. Vincent, L., Soille, P.: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **13**(6) (1991) 583–598

Active Shape Model Based Segmentation and Tracking of Facial Regions in Color Images

Bogdan Kwolek

Rzeszów University of Technology, W. Pola 2, 35-959 Rzeszów, Poland
bkwolek@prz.rzeszow.pl

Abstract. An approach for segmenting and tracking a face in a sequence of color images is presented. It enables reliable segmentation of facial region despite variation of skin-color perceived by a camera. A second order Markov model is utilized to forecast the skin distribution of facial regions in the next frame. The histograms that are constructed from the predicted distribution are backprojected to generate candidates of facial regions. Afterwards, a connected component labeling takes place. Spatial morphological operations, such as size and hole filtering are employed next. The Active Shape Model seeks to match a set of model points to the image. This statistical model of shape supports the segmentation of facial region undergoing tracking. Histograms are accommodated over time using feedback from shape, newly classified skin pixels and predictions of the skin-color evolution. This evolution is described by translation, rotation and scaling. In this context, the novelty of our approach lies in the introduction of Active Shape Model dealing with translation, rotation and scaling of the target to support face verification as well as to guide the evolution of skin distribution. The kernel histograms characterize the face during tracking in subsequent frames. The proposed algorithm achieves reliable detection and tracking results. The resulting system runs in real-time on standard PC computer.

1 Introduction

Skin-color has proven to be effective and robust cue for face detection, localization and tracking. A large part of image processing techniques uses skin detection as a first primitive for subsequent extraction of image features. Skin pixel candidates can be further processed to extract shape or motion cues. Well-known methods of color modeling, such as histograms and Gaussian mixture models have enabled the creation of appropriately exact and fast detectors of skin. Many available vision systems are now applying such techniques to extract skin-color patches for face detection and tracking in video sequences. In particular, skin color based methods are robust to changes in scale, resolution and partial occlusion. However, such techniques are not as good as can be for use in real environments because skin-color perceived by a camera usually changes when the lighting condition varies. Therefore, for reliable detection of skin pixels a dynamic color model that can cope with nonstationary skin-color distribution over time should be applied in vision systems.

The task of finding a human face in an image is referred to as face localization or face segmentation. The grouping of extracted facial features into face candidates, the heuristic rules and knowledge about a typical face and the correlation to statistical face template are examples of approaches commonly employed to detect the face [22]. Two types of information are typically used to perform segmentation during face tracking. The first is color information [4][7][10][14][19]. The second is the geometric configuration of the face shape and even a given set of facial features, e.g. both eyes, the nose, mouth etc. [5]. It is often not easy to separate skin colored objects from non-skin objects like wood, which can appear to be skin colored. Therefore, both skin-color modeling and contours are used to separate the facial region undergoing tracking [1]. The oval shape of the head is often approximated by an ellipse [1][20]. To cope with varying illumination conditions the color model is accommodated over time using the past color distribution and newly extracted distribution from the ellipse's interior. The kernel density based tracking has recently emerged as robust and accurate method due to its robustness to appearance variations and its low computational complexity [4][7][14].

Many color-based tracking approaches assume controlled lighting. In real scenarios an object undergoing tracking may be shadowed by other objects or even by the object itself. Updating the color model is thus one of the crucial issues in color-based tracking. A technique for color model adaptation was addressed in [15]. A Gaussian mixture model was used to represent the color distribution and the linear extrapolation was utilized to adapt the model parameters via a set of labeled training data from a subimage within the bounding box. A non-parametric method that in histogram adaptation employs only pixels which fall in the skin locus was proposed in work [16]. In work [18] the modeling of the color distribution over time is realized through predictive histogram adaptation. Histograms are dynamically updated using affine transformations, warping and resampling. The pixel-wise skin color segmentation is often not sufficient to provide the pixels for color model adaptation because pixels in the image background may also have colors similar with skin colors and this can then lead to over-segmentation. Another issue which should be taken into account is that nearby pixel from skin-colored background may blend with the true skin regions and this can have an adverse effect on subsequent processing of skin regions. The adaptive skin-color filter [6] performs initial skin candidate detection at the beginning and then more accurate tuning of skin model takes place. The adaptation takes into account the skin-like background colors. The method uses HSV color space in which the H coordinates are additionally shifted by 0.5. A comparative study of four state-of-the-art techniques of skin detection under changing illumination conditions can be found in [17].

The proposed algorithm of face tracking under time-varying illumination begins with separating skin and non-skin colors using a database of skin and non-skin pixels. The statistical shape model is utilized in selection of face candidates, yielding the best face candidate on the basis of shape and color criterions. The facial regions are detected during tracking by looking for pixels that have skin

colors. The presence of such pixels in the input image is detected using the skin and non-skin histograms. A kernel color histogram is used in frame-to-frame appearance matching. The detected skin-colored regions are then refined using homogeneity property which exhibit skin regions. In particular, the connected component analysis is applied to label separate regions. Spatial morphological operations for hole and object size filtering are used afterwards. The statistical shape model provides an effective method for fitting oval head shapes to detected face candidates. The algorithm reviews the image focusing the action around the position where the tracked face was detected in the previous frame. The outcome of this stage is a shape fitted to the face candidate. It provides additional information about expected target location. Using outline determined in such a way a local search is conducted to determine the pose of the face allowing for both color and shape information. The key idea of the proposed approach is improved selection of skin pixels to determine the parameters of models expressing the skin evolution over time. Even when a background region situated close to a face region has skin colored pixels, there always exists a boundary between the true skin region and the background. Our aim is to detect such a boundary using Active Shape Models. A second order Markov model is applied to predict the evolution of colors of such skin pixels, gathered in certain number of the last frames. The predicted skin distribution is quantized using a kernel to construct the histogram.

The Active Shape Models, which were originally proposed by Cootes [8], have been modified in work [13] to incorporate color cues. The cited approach does not apply color segmentation to the images. It is based on the minimization of energy functions in the color components. Therefore it admits of only a small change in illumination between two successive frames.

The following section briefly outlines some topics related to statistical shape models. The details of the shape alignment are given in Section 3. Section 4. describes how the Active Shape Model is used to conduct tracking and to support the skin segmentation in video under time-varying illumination conditions. The model of skin colors and their evolution is described in Section 5. Experimental results are shown in Section 6. We report some conclusions in the last section.

2 Facial Shape Constraints

The method of segmentation and tracking of facial regions proposed in this paper utilizes the statistical shape models. A shape model is utilized to constrain the configuration of a set of candidate skin pixels. An efficient algorithm allows the facial pixels detections to be tested and verified. The method allows for skin detector failures by predicting the locations of missing skin pixels using the shape model. The non-skin pixels outside the shape are not considered in the skin-color model as well.

During shape guided verification of the facial region a set of candidate skin pixels is inspected using shape constraints in two ways. Firstly, a shape model is fitted to the candidate facial region. Secondly, limits are prescribed on the

position, orientation and scale of a set of candidate skin pixels relative to the position, orientation and scale according to their values from the last frame. The aim is to extract pixels belonging only to the tracked face, using the candidate facial mask and the shape constraints. The facial mask is generated from a skin probability image. The skin probability image is extracted on the basis of skin histogram that is accommodated over time.

There are two broad approaches for representing a two-dimensional shape: region-based and contour-based. The region-based methods encode the place occupied by the object through a mask. The methods belonging to this group are sensitive to noise and they cannot cope with partly obscured objects. In contour-based approach the boundary of the object is modeled as an outline. Therefore, such methods can deal better with partially obscured objects and partial occlusions. A contour-based model can be built by placing landmark markers on distinctive features and at some pixels in between. The contour-based instances are usually normalized to canonical scale, translation and rotation in order to make possible comparison among distinct shapes. A distance between corresponding points from the two normalized shapes can be utilized to express the similarity.

Active Shape Models (ASMs or smart snakes) were originally designed as a method for locating given shapes or outlines within images [8]. An ASM-based procedure starts with the mean shape, approximately aligned to the object, iteratively distorts it and refines the pose to obtain a better fit. It seeks to minimize the distance between model points and the corresponding pixels found in the image. A shape consisting of n points can be considered as one data point in $2n$ -dimensional space. A classical statistical method for dealing with redundancy in multivariate data is the principal component analysis (PCA). PCA determines the principal axes of a cloud of n points at locations \mathbf{x}_i . The principal axes, explaining the principal variation of the shapes, compose an orthonormal basis $\Phi = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n\}$ of the covariance matrix $\Sigma = \frac{1}{n-1} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$. It can be shown that the variance across the axis corresponding to the i -th eigenvalue λ_i equals the eigenvalue itself. An instance of shape can be generated by deforming the mean shape $\bar{\mathbf{x}}$, using a linear combination of eigenvectors Φ , weighted by so-called modal deformation parameters \mathbf{b} . Thus, the new shape can be expressed in the following manner: $\mathbf{x} = \bar{\mathbf{x}} + \Phi\mathbf{b}$. By varying the elements of \mathbf{b} we can modify the shape. By applying constraints we ensure that the generated shape is similar to the mean shape from the original training data. The deformation of shapes is limited to a subspace spanned by a few eigenvectors corresponding to the largest eigenvalues. We can achieve a trade-off between the constraints on the shape and the model representation by varying the number of eigenvectors.

3 Shape Alignment

Given two 2D shapes, \mathbf{x}_2 and \mathbf{x}_1 our aim is to determine the parameters of a transformation T , which, when applied to \mathbf{x}_2 can best align it with \mathbf{x}_1 with

one-to-one point correspondence. During alignment we utilize an alignment metric that is defined as the weighted sum of the squares of the distances between corresponding points on the considered shapes. Thus we seek to choose the parameters t of the transformation T to minimize:

$$E = \sum_{i=1}^n (\mathbf{x}_{1i} - T_t(\mathbf{x}_{2i}))^T \mathbf{W}_i (\mathbf{x}_{1i} - T_t(\mathbf{x}_{2i})), \quad (1)$$

where \mathbf{W} is a diagonal matrix of weights $\{w_1, w_2, \dots, w_n\}$. Expressing T_t in the following form:

$$T_t \equiv \begin{bmatrix} s \cos(\theta) & -s \sin(\theta) & t_x \\ s \sin(\theta) & s \cos(\theta) & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{2i} \\ y_{2i} \\ 1 \end{bmatrix} \quad (2)$$

and denoting $a_x = s \cos(\theta)$, $a_y = s \sin(\theta)$ we can rewrite (1) in the following form: $E = \sum_{i=1}^n w_i ((a_x x_{2i} - a_y y_{2i} + t_x - x_{1i})^2 + (a_y x_{2i} + a_x y_{2i} - t_y - y_{2i})^2)$. The error E assumes a minimal value when all the partial derivatives are zero. For example, differentiating the last equation with regard to a_x we obtain: $\sum_{i=1}^n w_i (a_x (x_{2i}^2 + y_{2i}^2) + t_x x_{2i} + t_y y_{2i} (x_{1i} x_{2i} + y_{1i} y_{2i})) = 0$. Differentiating w.r.t. remaining parameters and equating to zero gives:

$$\begin{bmatrix} C_1 \\ C_2 \\ X_1 \\ Y_1 \end{bmatrix} = \begin{bmatrix} D & 0 & X_2 & Y_2 \\ 0 & D & -Y_2 & X_2 \\ X_2 & -Y_2 & W & 0 \\ Y_2 & X_2 & 0 & W \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ a_x \\ a_y \end{bmatrix}, \quad (3)$$

where $X_i = \sum_{k=1}^n w_k x_{ik}$, $Y_i = \sum_{k=1}^n w_k y_{ik}$, $C_1 = \sum_{k=1}^n w_k (x_{1k} x_{2k} + y_{1k} y_{2k})$, $C_2 = \sum_{k=1}^n w_k (y_{1k} x_{2k} + x_{1k} y_{2k})$, $D = \sum_{k=1}^n w_k (x_{2k}^2 + y_{2k}^2)$, $W = \sum_{k=1}^n w_k$. The parameters t_x , t_y , a_x and a_y constitute a solution which best aligns the shapes. An iterative approach to find the minimum of square distances between corresponding model and image points is as follows [8]:

1. Initialize \mathbf{b} to zero.
2. Generate the model points using $\mathbf{x} = \bar{\mathbf{x}} + \Phi \mathbf{b}$
3. Find the pose parameters using (3)
4. Project image pixels \mathbf{Y} into the model co-ordinates using $\mathbf{y} = T_t^{-1}(\mathbf{Y})$
5. Scale \mathbf{y} as follows $\mathbf{y}' = \mathbf{y}/(\mathbf{y} \cdot \bar{\mathbf{x}})$
6. Update \mathbf{b} in the following manner $\mathbf{b} = \Phi^T (\mathbf{y}' - \bar{\mathbf{x}})$
7. If not converged, go to step 2.

4 Active Shape Model Based Tracking

Tracking can be perceived as a problem of assigning consistent labels to objects being tracked. This is done through maintaining the observations of objects in order to label these so that all observations of a given object in a sequence of images are given the identical label. During shape aligning our algorithm reviews

the binary image focusing the action around the pose that has been determined in the previous frame. The algorithm requires that the new shape center remains within the face mask centered on the previous location of the target. Such an assumption is utilized in kernel based trackers [4][7]. Limits are prescribed on the position, orientation and scale according to their values in the last frame. The binary image is generated in advance on the basis of the skin histogram that is accommodated over time.

The standard ASM aligns the shape model to outlines in an image using only edges. To obtain a rough pose of the face we first utilize the edges of the mask indicating the face area. The final pose of the shape is determined on the basis of the intensity gradient near the edge of the outline, a matching score of colors from the candidate outline and from the outline determined in the previous iteration and the location of the mask. In work [12] a search for the edges in direction perpendicular to the border has been shown optimal. Therefore, a search for the points along profiles normal to the shape boundary has been implemented.

The shape model has been generated using 10 manually segmented images with frontal faces, each represented by 30 characteristic points. The faces have been normalized with regard to orientation and size in order to obtain a set of points with similar physical correspondence across the training collection. All training faces were manually aligned by eyes position.

The oval shape of the head can be reasonably well approximated by an ellipse. Therefore, in this work the model shapes are normalized by aligning the average shape to a fixed circle of landmark points. Such an approach has the advantage that the model can be scaled to size needed by the application through setting only the size of the circle.

Figure 1. demonstrates the performance of the ASM attempting to match the head model to a given binary mask that has been extracted during tracking. To demonstrate the usefulness of statistical shape models in tracking two artifacts at the left and the right side of face border have been manually added. Despite large deformation of shape outline we can observe how precisely the algorithm can align the model shape to the face mask. The shape on the left represents the initial pose that has been utilized in depicted shape alignment. This exemplifies also how the statistical shape models can support the selection of pixels for color model adaptation and thus the prediction of skin evolution over time. The next section is devoted to description of skin-color based image segmentation under time-varying illumination.

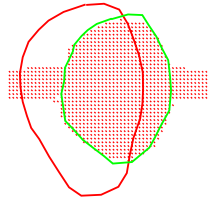


Fig. 1. Shape alignment in presence of artifacts

5 Skin Color Segmentation Under Time-Varying Illumination

The face detection scheme within tracking framework must operate flexibly and reliably regardless of lighting conditions, background clutter in the image, as well as variations in face position, scale, pose and expression. Some tracking applications, for example using a moving camera, do require good detection rates even in case of abrupt changes of illumination. Fast and reliable face segmentation techniques in image sequences are highly desirable capability for many vision systems. Skin color based detection methods are independent to scale, resolution and to some degree of face orientation in the image. A problem with robust detection of skin pixels arises under varying lighting conditions. The same skin patch can look like two different patches under two different conditions. An important issue for any skin-color based tracking system is to provide an accommodation mechanism which could cope with varying illumination conditions that may occur during tracking. In our approach, color distributions are estimated over time and then are predicted under the assumption that lighting conditions vary smoothly over time. The prediction is used to reflect the changing tendency in appearance of the object being tracked. A ground-truth is an evident need during adapting a color model over time to changing illumination conditions [15]. In this approach the evolution of distribution is constrained via statistical shape model and skin locus mechanism. In work [18] the current segmentation and predictions of Markov model were applied to provide a feedback for accommodation. In other work [15] the accommodation process is controlled via mechanism for detecting errors accompanying tracking.

One significant element that should be considered while constructing a statistical model of skin color is the choice of color space. One of the advantages of the HSV color space is that it yields minimum overlap between skin and non-skin distributions. Hue is invariant to certain types of highlights, shadows and shading. A shadow cast does not change significantly the hue color component. It decreases mainly the illumination component and changes the saturation. This color space was utilized in several face detection systems [15][18][19]. The only disadvantage of the HSI color space is the costly conversion from the RGB color space. We handled this problem by using lookup tables.

The histogram is the oldest and most broadly employed non-parametric density estimator. In the standard form it is computed by counting the number of pixels that have given color in region of interest. This operation allows alike colors to be clustered into the separate bin. The quantization into bins reduces the memory and computational requirements. Due to their statistical nature the color histograms can only reflect the content of images in a limited way [21]. Therefore, such representation of color densities is tolerant to noise. Histogram-based techniques are effective only when the number of bins can be kept relatively low and when sufficient data are in disposal [15]. One of the drawbacks of the histogram based density estimation is the lack of convergence to the true density

if the data set is small. In certain applications, the color histograms are invariant to object translations and rotations. They vary slowly under change of angle of view and with change in scale.

The target is represented by the set $S = \{\mathbf{u}_i\}_{i=1}^N$, where N is the number of pixels and \mathbf{u}_i denotes vector with HSV components of the i -th pixel. Given a set of samples S we can obtain estimate of $p(\mathbf{u})$ using multivariate kernel density estimation [7][9]:

$$p(\mathbf{u}) = p(u^{(1)} = H, u^{(2)} = S, u^{(3)} = V) = \frac{1}{N} \sum_{i=1}^N \prod_{l=1}^3 K_h(u^{(l)} - u_i^{(l)}), \quad (4)$$

where $K_h(\mathbf{u}) = \frac{1}{(\sqrt{2\pi}h)^d} \exp\left(-\frac{1}{2} \left(\frac{\|\mathbf{u}\|^2}{h^2}\right)\right)$ is a Gaussian kernel of bandwidth h , whereas d denotes the dimension. The quantization with $32 \times 32 \times 32$ bins has been used to represent both the target as well as the background.

An initial skin histogram, along with the model for non-skin background pixels, has been used in order to compute the probability of every pixel in the first input color image and thus to give the skin likelihood. A model for human skin color distribution was built using a repository of labeled skin pixels that has been prepared in advance. Given the histograms ϕ_{fg} and ϕ_{bg} , the log-likelihood ratio for a pixel with color \mathbf{u} is given by [11]:

$$L(\mathbf{u}) = \max\left(-1, \min\left(1, \log \frac{\max(\phi_{fg}(\mathbf{u}), \delta)}{\max(\phi_{bg}(\mathbf{u}), \delta)}\right)\right), \quad (5)$$

where δ is a very small number, whereas $\phi_{fg}(\mathbf{u})$, $\phi_{bg}(\mathbf{u})$ denote the frequency of pixels with color \mathbf{u} in the foreground and background, respectively.

Given the probability image the thresholding takes place. After that, the binary image is analyzed using a labeling procedure, which isolates connected components in order to detect the presence of face candidates in the image. Next, the candidate regions are subjected to morphological operations, such as size and hole filtering, to clean up the mask and to generate the mask indicating which pixels belong to the face. After alignment of the model shape with the current mask, the refined face mask is utilized to select from the newly classified pixels the representation of the skin distribution. Using such samples gathered over an initial sequence of frames the sequence-specific motion patterns are learned. A second-order Markov process has been chosen to model the evolution of the color distribution over time [3][18].

Many studies have indicated that the skin tones differ mainly in their intensity value while they form compact cluster in chrominance coordinates [23]. Hence, the evolution of skin cluster can be parameterized at each time instant t by translation, rotation and scaling. The translation parameters \mathbf{t}_p can be extracted on the basis of means from samples constituting a learning distribution, whereas the scaling parameters \mathbf{s}_p can be estimated from their standard deviations. The eigenvectors of the covariance matrices of samples from two consecutive frames define two coordinate frames, which can be then used to estimate the rotations \mathbf{r}_p [18].

The work [3] demonstrated that affine motion can be described via a second-order auto-regressive Markov process:

$$\mathbf{X}(t_{k+1}) - \bar{\mathbf{X}} = A_2(\mathbf{X}(t_{k-1}) - \bar{\mathbf{X}}) + A_1(\mathbf{X}(t_k) - \bar{\mathbf{X}}) + B_0\mathbf{w}_k, \quad (6)$$

where $\mathbf{X} = \{\mathbf{t}_p^T, \mathbf{s}_p^T, \mathbf{r}_p^T\}$ is the vector parameterizing the skin evolution. The parameters which should be learned are A_0 , A_1 , and $C = BB^T$ because B cannot be observed directly. It was shown in [2] that the matrices A_0 and A_1 can be estimated on the basis of the following equations:

$$S_{20} - \hat{A}_0 S_{00} - \hat{A}_1 S_{10} = 0 \quad (7a)$$

$$S_{21} - \hat{A}_0 S_{01} - \hat{A}_1 S_{11} = 0, \quad (7b)$$

where $S_{ij} = \sum_{k=1}^{m-2} (\mathbf{X}(t_{(k-1)+i})\mathbf{X}^T(t_{(k-1)+j}))$, $i, j = 0, 1, 2$, and m denotes number of learning frames. Having A_0 and A_1 we can estimate C from the following equation: $\hat{C} = \frac{1}{m-2}Z(A_0, A_1)$, where $Z(A_0, A_1) = S_{22} + A_1 S_{11} A_1^T + A_0 S_{00} A_0^T - S_{21} A_1 - S_{20} A_0^T + A_1 S_{10} A_0^T - A_1 S_{12} - A_0 S_{02} + A_0 S_{01} A_1^T$.

On the basis of predicted distribution the histogram $\phi_{fg(p)}$ of skin colors is extracted. After normalization of the histogram we perform an adaptation which combines the histogram that had been obtained from the predicted distribution and the histogram from the last frame. Adaptation is made according to the following equation:

$$\phi_{fg(u)}(t) = (1 - \alpha)\phi_{fg}(t-1) + \alpha\phi_{fg(p)}(t), \quad (8)$$

where the adaptation coefficient α has been determined empirically. The histogram $\phi_{fg(u)}(t)$ has been subjected to segmentation procedure to produce the face mask. The refined face mask by statistical shape model, as discussed in Section 4., has been then used to collect the newly classified skin pixels in a list.

The refined face mask by statistical shape model can contain non-skin pixels. Experiments demonstrated that the part of face below the hair was a source of such inadequate pixels. To deal with this undesirable effect, the pixels collected in the mentioned above list were additionally inspected if they fall within the prepared in advance skin locus. A prepared off-line two-dimensional table defining possible skin chromaticities has been used at this stage. It has shown to be useful especially in eliminating non-skin pixels from the representation of the skin distribution in a sudden change of illumination.

The list prepared in such a way has been utilized to generate the histogram $\phi_{fg(n)}$. Finally, this histogram has been updated in the following manner:

$$\phi_{fg}(t) = (1 - \alpha)\phi_{fg}(t-1) + \alpha\phi_{fg(n)}(t). \quad (9)$$

This histogram has been utilized to generate the skin image probability during tracking.

6 Experiments

To test the proposed method of face tracking we performed various experiments on real images. We utilized the Carphone sequence as our first test set. The

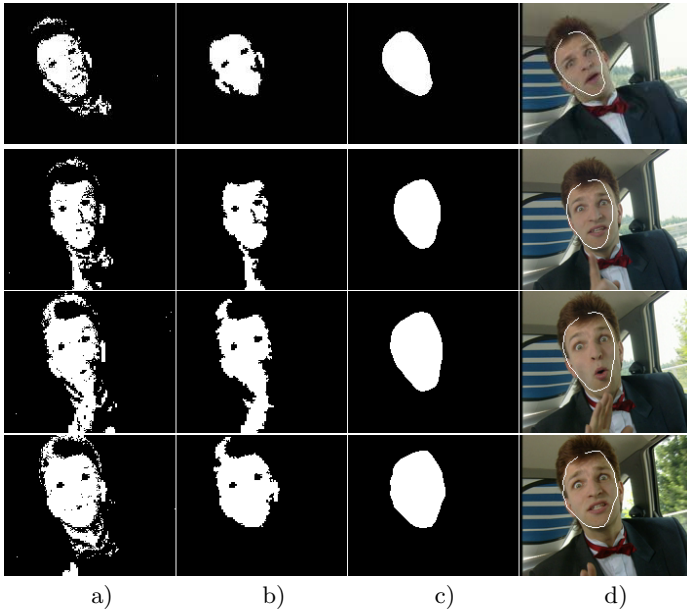


Fig. 2. Face tracking using the Carphone test sequence, frames #84, #125, #176, #200 (from top to bottom). Binary image (a). Hole and object size filtering (b). ASM-based refinement of the face mask (c). Input image with the extracted face outline (d).

model of the face shape was prepared on images not containing the face from the considered test sequence. Through this sequence we want to highlight the behavior of the tracking algorithm in case of errors in target segmentation. We can notice in Fig. 2. that even if the segmentation does not separate the object of interest from the background, the contour generated from the active shape model supports greatly the extraction of the object. For example, the images from the second column demonstrate that without the ASM based shape refinement the color model would be influenced by the hair colors as well as by the bow-tie colors, see also frame #176.

To study the adaptation performance in time-varying illumination conditions we conducted experiments with two configurations of the tracking algorithm. In the first configuration we utilized the predictions of the skin evolution, whereas in the second one only the newly classified pixels have been used to accommodate the histogram. The number of learning images has been set to 20. Typically in almost 90% images, the predictions lead to better fidelity in approximation of the face, see also Fig. 3. In particular, the first configuration detected smaller number of skin-pixels in the background in all images, compare Fig. 3a with Fig. 3c. Other tracking results using this image sequence can be found in [1][5]. The presented system runs at 176x144 image resolution at frame rates of 12-15 Hz on a 2.4 GHz PC. The algorithm has also been tested with Claire and Foreman test sequences as well as PETS-03 meeting recordings. The superior

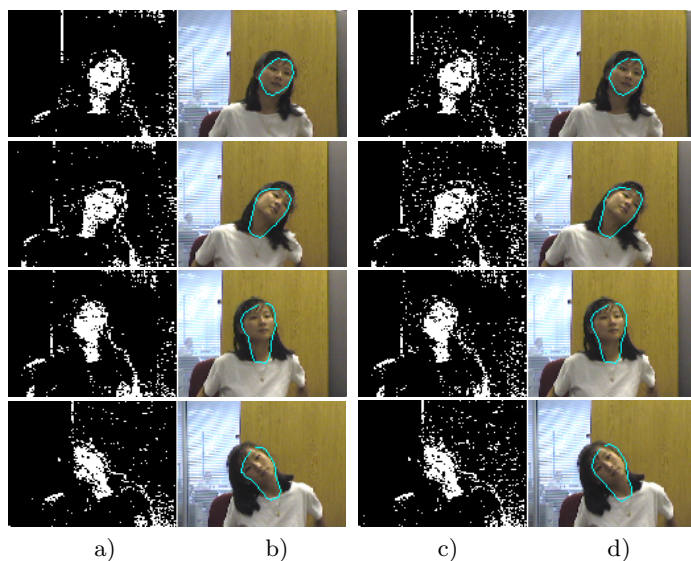


Fig. 3. Tracking in time-varying illumination, frames #301, #310, #319, #330 (from top to bottom). Histogram update using: predictions (a,b), newly classified pixels (c,d).

tracking performance over face tracking algorithm using intensity gradients and kernel histograms was observed in all above mentioned sequences. Particularly, smaller "jumps" of the shape indicating the face location from frame to frame have been perceived. In varying illumination can arise the superiority over Mean-Shift because of reduced adaptation capabilities of Mean-Shift methods.

7 Conclusion

A face detection and tracking strategy has been described. The accommodation of the skin histograms over time takes place using feedback from shape, newly classified skin pixels and predictions of the skin color evolution. Once a face is being tracked, the color model adapts according to changes in appearance and therefore improves tracking performance.

Acknowledgment

This work has been supported by MNSzW within the project 3 T11C 057 30.

References

1. Birchfield, S.: Elliptical Head Tracking Using Intensity Gradients and Color Histograms, In Proc. IEEE Conf. on Comp. Vis. and Patt. Rec., (1998) 232–237
2. Blake, A., Isard, M., Reynard, D.: Learning to Track the Visual Motion of Contours, Artificial Intelligence, **78** (1995) 101–133

3. Blake, A., Isard, M.: Active Contours, Springer (1998)
4. Bradski, G. R.: Computer Vision Face Tracking as a Component of a Perceptual User Interface, In Proc. IEEE Workshop on Appl. of Comp. Vision (1998) 214–219
5. Chen, Y., Rui, Y., Huang, T.: Mode-based Multi-Hypothesis Head Tracking Using Parametric Contours, In Proc. IEEE Int. Conf. on Aut. Face and Gesture Rec. (2002) 112–117
6. Cho, K. M., Jang, J. H., Hong, K. S.: Adaptive Skin Color Filter, Pattern Recognition, **34**(5) (2001) 1067–1073
7. Comaniciu, D., Ramesh, V., Meer, P.: Real-Time Tracking of Non-Rigid Objects Using Mean Shift, In Proc. IEEE Conf. on Comp. Vis. Patt. Rec. (2000) 142–149
8. Cootes, T.: An Introduction to Active Shape Models, Model-Based Methods in Analysis of Biomedical Images, [in:] Image Processing and Analysis, Eds., R. Baldoock and J. Graham, Oxford University Press (2000)
9. Elgammal, A., Duraiswami, R., Davis L. S.: Probabilistic Tracing in Joint Feature-Spatial Spaces, In Proc. IEEE Conf. on Comp. Vis. and Patt. Rec. (2003) 16–22
10. Fieguth, P., Terzopoulos, D.: Color-Based Tracking of Heads and Other Mobile Objects at Video Frame Rates, In Proc. IEEE Conf. on Comp. Vis. and Patt. Rec., Hilton Head Island (1997) 21–27
11. Han, B., Davis, L.: Robust Observations for Object Tracking, In Proc. Int. Conf. on Image Processing (2005) 442–445
12. Isard, M., Blake, A.: Contour Tracking by Stochastic Propagation of Conditional Density, European Conf. on Computer Vision, Cambridge (1996) 343–356
13. Koschan, A., Kang, A., Paik, J., Abidi, B., Abidi, M.: Color Active Shape Models for Tracking Non-Rigid Objects, Pattern Recognition Letters, **24** (2003) 1751–1765
14. Perez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-Based Probabilistic Tracking, European Conf. on Computer Vision (2002) 661–675
15. Raja, Y., McKenna, S. J., Gong, S.: Color Model Selection and Adaptation in Dynamic Scenes, Proc. European Conf. on Computer Vision (1998) 460–474
16. Soriano, M., Martinkauppi, B., Huovinen, S., Laaksonen, M.: Adaptive Skin Color Modelling Using the Skin Locus for Selecting Training Pixels, Pattern Recognition, **36** (2003) 681–690
17. Soriano, M., Martinkauppi, B., Pietikainen M., Detection of Skin under Changing Illumination: A Comparative Study, Int. Conf. on Image Analysis and Proc. (2003) 652–657
18. Sigal, L., Sclaroff, S., Athitsos, V.: Estimation and Prediction of Evolving Color Distributions for Skin Segmentation under Varying Illumination, In Proc. IEEE Conf. on Comp. Vis. and Patt. Rec. (2000) 2152–2159
19. Sobottka, K., Pitas, I.: Segmentation and Tracking of Faces in Color Images, In Proc. of the Sec. Int. Conf. on Aut. Face and Gesture Rec. (1996) 236–241
20. Srisuk, S., Kurutach, W., Limpitikeat, K.: A Novel Approach for Robust Fast and Accurate Face Detection, Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems, **9**(6) (2001) 769–779
21. Swain, M. J., Ballard, D. H.: Color Indexing, Int. J. of Comp. Vision **7**(1) (1991) 11–32
22. Yang, M.-H., Krigman, D., Ahuja, N.: Detecting Faces in Images: A survey, IEEE Trans. on Pattern Analysis and Machine Intelligence, **24**(1) (2002) 34–58
23. Yang, J., Weier, L., Waibel, A.: Skin-Color Modelling in Color Images, In Proc. Asian Conf. on Computer Vision (1998) II:687–694

On the Adaptive Impulsive Noise Attenuation in Color Images

Bogdan Smolka*

Silesian University of Technology, Department of Automatic Control,
Akademicka 16 Str, 44-100 Gliwice, Poland
smolka@ieee.org

Abstract. In this paper a novel method of impulsive noise suppression in color images is described. The new approach is based on a soft-switching scheme, whose output is the weighted average of the central pixel and the vector median of the local filtering window. The noise detection component of the switching filtering framework is based on the difference between accumulated distances assigned to the vector median of the local data and the central pixel in the filtering mask. The results of simulations performed on a set of test images show that the proposed method is capable of reducing even strong impulsive noise while retaining the image structures.

1 Introduction

During image *acquisition, formation, storage and transmission* many types of distortions limit the quality of digital images. Periodic or random motion of the camera system during exposure, electronic instabilities of the image signal, electromagnetic interferences from natural or man-made sources, sensor malfunctions, optic imperfections, electronics interference or aging of the storage material, all decrease the image quality. Very often, images are corrupted by the so called *impulsive noise* and the removal of this type of noise in color images is addressed in this paper.

Mathematically, a $N_1 \times N_2$ multichannel image is a mapping $\mathbb{Z}^l \rightarrow \mathbb{Z}^m$ representing a two-dimensional matrix of three-component samples (pixels), $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{im}) \in \mathbb{Z}^l$, where l is the dimension of the image domain and m denotes the number of channels, (in the case of standard color images, parameters l and m are equal to 2 and 3, respectively). Components x_{ik} , for $k = 1, 2, \dots, m$ and $i = 1, 2, \dots, N$, $N = N_1 \cdot N_2$, represent the color channel values quantified into the integer domain.

Let us assume that an image is represented in the RGB color space and let $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ be a set of n samples from the sliding filter window W , with \mathbf{x}_1 being the central pixel. The pixels from W can be ordered by assigning to each vector a measure of its dissimilarity to other vectors in the filtering window. The

* Supported by Grant 3 T11C 01629 of the Polish Ministry of Education and Science.

vector with the lowest cumulated dissimilarity measure is most centrally located and is the the output of the so called rank ordered filters, [1].

Mostly the ordering based on the cumulative distance function is utilized: $R(\mathbf{x}_i) = \sum_{j=1}^n \rho(\mathbf{x}_i, \mathbf{x}_j)$, where $\rho(\mathbf{x}_i, \mathbf{x}_j)$ is the distance among \mathbf{x}_i and \mathbf{x}_j . The increasing ordering of the scalar quantities $\{R_1, \dots, R_n\}$ generates the ordered set of vectors $\{\mathbf{x}_{(1)}, \mathbf{x}_{(2)}, \dots, \mathbf{x}_{(n)}\}$ and the vector $\mathbf{x}_{(1)}$ is called the *vector median*.

2 Proposed Filtering Design

Quite often filters based on local statistics are applied to suppress the noise disturbing the images. These filters make use of the local mean and the variance of the input set W and define the filter output as, [2]

$$\mathbf{y}_1 = \hat{\mathbf{x}}_1 + \alpha(\mathbf{x}_1 - \hat{\mathbf{x}}_1) = \alpha\mathbf{x}_1 + (1 - \alpha)\hat{\mathbf{x}}_1, \quad (1)$$

where $\hat{\mathbf{x}}_1$ is the arithmetic mean of the image pixels belonging to the filter window W , whose center is occupied by pixel \mathbf{x}_1 and α is a filter parameter, usually estimated using the local statistical features. Such a filter smooths with the local mean, when the noise is not very intensive and leaves the pixel value unchanged when a strong signal activity is detected. Equation (1) can be rewritten as

$$\mathbf{y}_1 = (1 - \alpha)(\psi_1\mathbf{x}_1 + \mathbf{x}_2 + \dots + \mathbf{x}_n) / n, \quad (2)$$

with $\psi_1 = (1 - \alpha + n\alpha)/(1 - \alpha)$ and in this way the local statistic filter (1) appears to take the form of the *central weighted average filter*, with a weighting coefficient ψ_1 .

The structure of the new filter called *Soft Switching VMF* (SSVMF) is similar to the presented above approach, [3,4]. However, as our aim is to construct a filter capable of removing impulsive noise, instead of the mean value, the VMF output is utilized and the output of the proposed filter is defined as

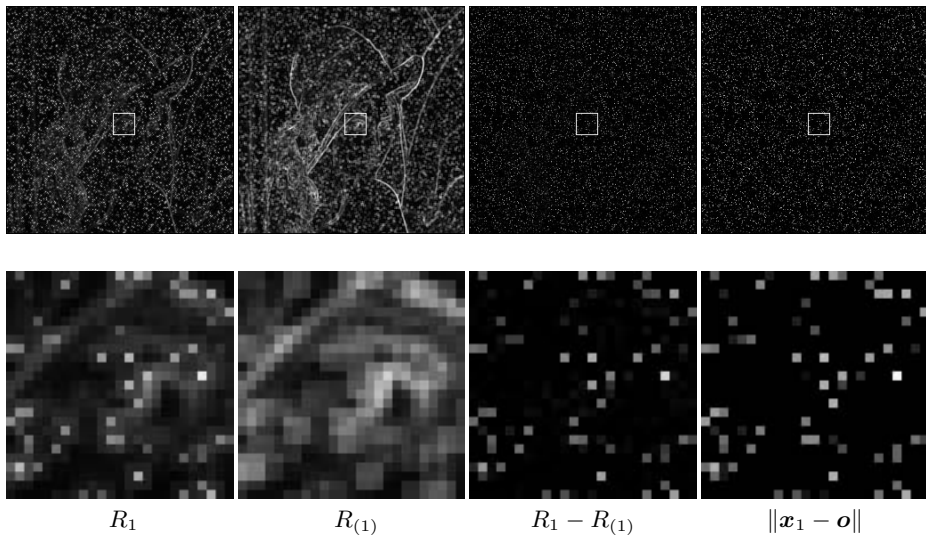
$$\mathbf{y}_1 = \mathbf{x}_{(1)} + \alpha \cdot (\mathbf{x}_1 - \mathbf{x}_{(1)}) = \alpha\mathbf{x}_1 + (1 - \alpha)\mathbf{x}_{(1)}. \quad (3)$$

In this way, the proposed technique is a compromise between the VMF and identity operation. When an impulse is present, then it will be replaced with the VMF, otherwise the central pixel should not be changed. Thus the efficiency of the proposed filter is dependent on the coefficient α which should be able to detect impulses in the center of the filtering window.

In this paper the difference between the cumulated distances R_1 and $R_{(1)}$ serve as an indicator of the presence of an impulsive distorsion. The local difference defined as $r = R_1 - R_{(1)}$ proves to detect quite well the impulses. This is shown in Fig. 1, which depicts the comparison between the difference r and the distance between the original, undisturbed pixels \mathbf{o}_i and pixel \mathbf{x}_i of the contaminated image.

As the α in (3) should be a decreasing function of the impulse magnitude, various decreasing functions known from the theory of non-parametric estimation have been used:

LENA



GOLDHILL

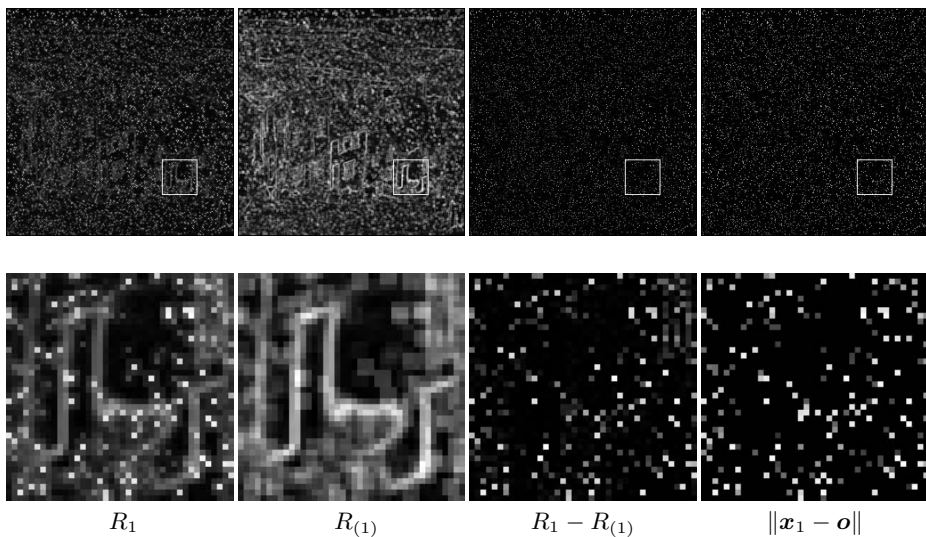


Fig. 1. Illustration of the efficiency of the proposed impulse detector for the LENA and GOLDHILL images. From the left: images depicting the R_1 , $R_{(1)}$ and $r = R_1 - R_{(1)}$ values compared with $\|\mathbf{x}_1 - \mathbf{o}\|$. The bottom images depict the zoomed regions. Note the similarity between the images of $r = R_1 - R_{(1)}$ and $\|\mathbf{x}_1 - \mathbf{o}\|$, which proves that the difference r between R_1 and $R_{(1)}$ detects well the impulses introduced by the noise process.

$$f_1 = e^{-\frac{x}{h}}, f_2 = e^{-\frac{x^2}{h^2}}, f_3 = \frac{1}{1 + \frac{x^2}{h^2}}, f_4 = \left[1 - \frac{x}{h}\right]^+, f_5 = \left[1 - \frac{x^2}{h^2}\right]^+, \quad (4)$$

where $[f(x)]^+ = f(x)$ if $0 < x < h$ and 0 otherwise.

Experiments revealed as expected, that the shape of the kernel function is not of great importance for the noise reduction efficiency of the proposed filter as quite different functions yield comparable results in terms of PSNR as shown in Fig. 6. For the comparison with existing filters the Gaussian function f_2 from (4) has been adopted and the α in (3) is defined as $\alpha = \exp\{-r^2/h^2\}$, where h is a smoothing parameter.

If the central pixel is not disturbed by the noise process then the α coefficient is close to 1 and the output is near to the original value \mathbf{x}_1 , otherwise it is close to 0 and its output is the vector median $\mathbf{x}_{(1)}$.

It is interesting to observe that the filter output \mathbf{y}_1 lies on the line joining the vectors \mathbf{x}_1 and $\mathbf{x}_{(1)}$ and depending on the value of the α coefficient it slides between the identity operation and the vector median, (Fig. 2).

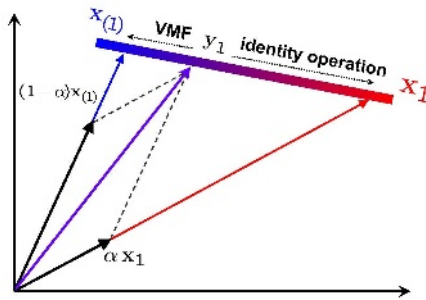


Fig. 2. The filter output slides between central pixel \mathbf{x}_1 and vector median $\mathbf{x}_{(1)}$

The conducted experiments have revealed that the optimal h value depends on the noise intensity p , which shows that an estimator of the contamination level is needed, so that the smoothing parameter can adapt to the noise process.

It has been observed that for the functions in (4) the product of the optimal value of the h parameter, denoted as h^* , which yields the best possible PSNR value, and the noise intensity p is a constant, which means that $1/h^*$ is proportional to p , as depicted in Fig. 3 a) for the Gaussian function f_2 .

Another important observation is that the mean values m of the histograms of the $r = R_1 - R_{(1)}$ values depicted in Fig. 4 are also proportional to p , which is shown in Fig. 3 b). This observation shows that the contamination intensity can be estimated from the information provided by the distribution of the r values.

Based on this observations, it can be concluded that there exists a simple relation between the optimal bandwidth parameter h^* and the mean value m of the histogram of r values, namely $m \cdot p = \beta$, where $\beta = 110$ is a constant, which was found experimentally using a set of natural images. The mean value of the

histogram $H(r)$ can be easily measured using the whole image or randomly chosen samples. This experimental observation is the most important contribution of the paper, as it enables fast and robust estimation of the noise corruption intensity.

3 Experimental Results

In this paper the noisy image is modelled as $\mathbf{x}_i = \{x_{i1}, x_{i2}, x_{i3}\}$, where

$$x_{ik} = \begin{cases} v_{ik} & \text{with probability } \pi, \\ o_{ik} & \text{with probability } 1 - \pi, \end{cases} \tag{5}$$

and the contamination component v_{ik} is a random variable. We will assume three models, [18,19] which will be called *random-valued* or *uniform* noise (NM1), when $v_{ik} = [0, 255]$, impulsive *fixed-valued* noise (NM2), when $v_{ik} \in \{0, 255\}$ and extended noise model (NM3) defined as, [18,20]

$$\mathbf{x}_i = \begin{cases} \mathbf{o}_i, & \text{with probability } 1 - p, \\ \{v_{i1}, o_{i2}, o_{i3}\}, & \text{with probability } p_1 p, \\ \{o_{i1}, v_{i2}, o_{i3}\}, & \text{with probability } p_2 p, \\ \{o_{i1}, o_{i2}, v_{i3}\}, & \text{with probability } p_3 p, \\ \{v_{i4}, v_{i4}, v_{i4}\}, & \text{with probability } p_4 p, \end{cases} \tag{6}$$

where p is the sample corruption probability and p_1, p_2, p_3 are corruption probabilities of each color channel, so that $\sum_1^4 p_k = 1$. In this work, NM3 will generally denote the case with $p_k = 0.25, k = 1, \dots, 4$, and $v_{ij} = 0$ or 255 for $j = 1, \dots, 4$.

It can be noticed that the second model is a special case of the *uniform* noise, as this noise can take on only two values 0 or 255 with the same probability, assuming 8-Bit per channel color image representation. In noise models NM1 and NM2 the contamination of the color image components is uncorrelated, and the overall contamination rate is $p = 1 - (1 - \pi)^3$.

For the measurement of the restoration quality the commonly used *Root Mean Squared Error* (RMSE) expressed through the *Peak Signal to Noise Ratio* (PSNR) was used as the RMSE is a good measure of the efficiency of impulsive noise suppression. The PSNR is defined as

$$PSNR = 20 \log_{10} \left(\frac{255}{\sqrt{MSE}} \right), \quad MSE = \frac{\sum_{i=1}^N \sum_{k=1}^m (x_{ik} - o_{ik})^2}{Nm}, \tag{7}$$

where N is the total number of image pixels, and x_{ik}, o_{ik} denote the k -th component of the noisy image pixel channel and its original, undisturbed value at a pixel position i , respectively.

For the evaluation of the detail preservation capabilities of the proposed filtering design the *Mean Absolute Error* (MAE) has been used

$$MAE = \frac{\sum_{i=1}^N \sum_{k=1}^m |x_{ik} - o_{ik}|}{Nm}. \tag{8}$$

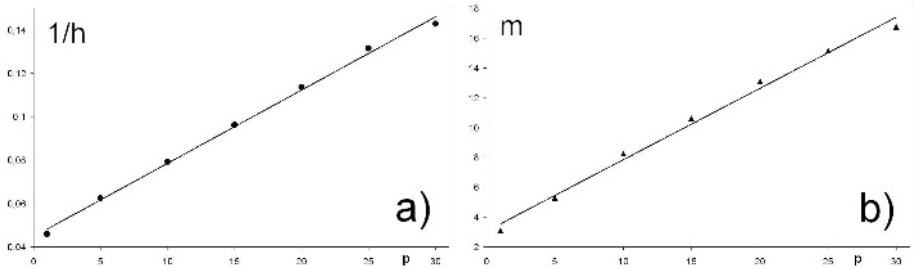


Fig. 3. Dependence of $1/h^*$ and m on the noise intensity p for the random valued impulsive noise and LENA image

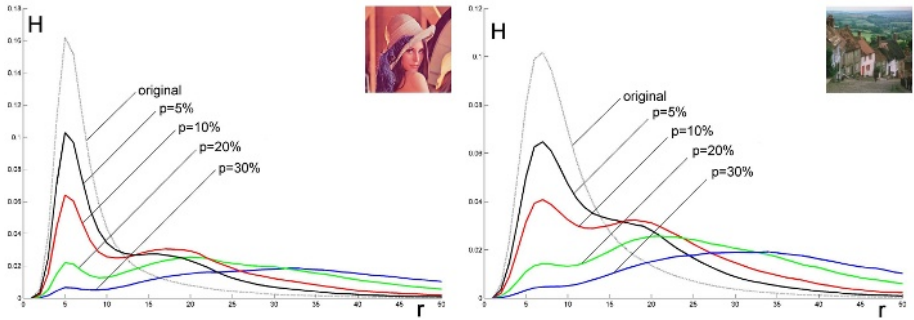


Fig. 4. Histograms of r for the LENA and GOLDHILL images for increasing random impulsive noise intensity

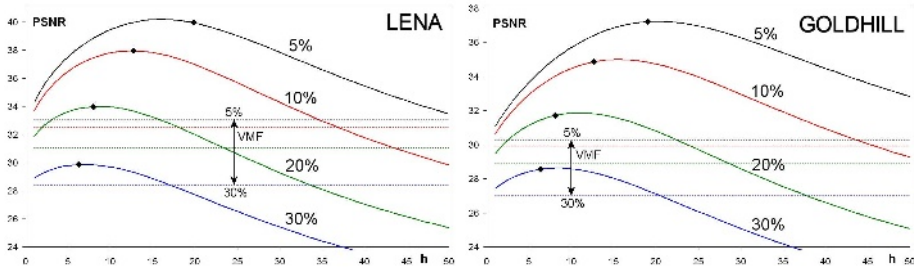


Fig. 5. Dependence of PSNR on parameter h for various noise intensity p . The results obtained using the proposed estimator of the optimal h value are depicted by \blacklozenge

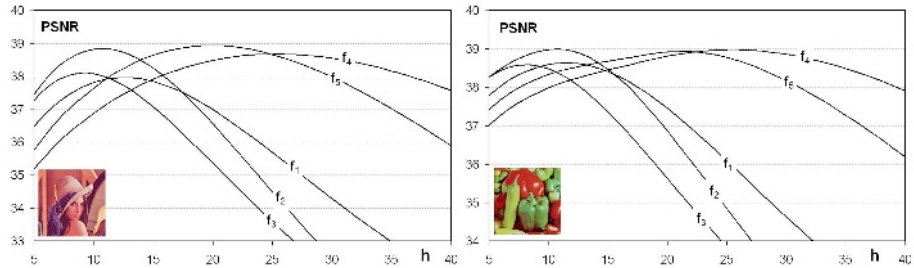


Fig. 6. Dependence of PSNR on the smoothing parameter h using various kernel functions, (see Eq. 4) for the LENA and PEPPERS images contaminated by 10% random valued impulsive noise NM1

Table 1. Filters used for the comparisons with the proposed noise removal method

| FILTERS USED FOR COMPARISONS | |
|------------------------------|--|
| VMF [1] | Vector Median Filter |
| VDOSF [5] | Vector Directional Order-Statistics Filter |
| HDF [6] | Hybrid Directional Filter |
| LIMF [7] | Local Information Measure Filter |
| VLUMS [8] | Vector LUM Smoother |
| AVLUMS [9] | Adaptive Vector LUM Smoother |
| MICM [10] | Modified ICM Method |
| AVFF [11] | Adaptive Video Filtering Framework |
| AVMF [12] | Adaptive Vector Median Filter |
| ACIF [13] | Adaptive Color Image Filter |
| CWVMF [14] | Central Weighted Vector Median Filter |
| MCWVMF [14] | Modified Central Weighted Vector Median Filter |
| FANRF [15] | Fast Adaptive Noise Reduction Filter |
| SANRF [16] | Self-Adaptive Noise Reduction Filter |
| GAVSF [17] | Generalized Adaptive Vector Sigma Filter |

Table 2. Comparison of the new filtering design with the denoising techniques listed in Tab. 1 using the *LENA* test image and three noise models for $p = 10\%$

| NOISE FILTER | NM1 | | | NM2 | | | NM3 | | |
|-----------------|-------------|--------------|-------------|-------------|-------------|-------------|-------------|--------------|-------------|
| | MAE | PSNR | NCD | MAE | PSNR | NCD | MAE | PSNR | NCD |
| NONE | 7.70 | 18.65 | 86.73 | 4.42 | 19.75 | 85.94 | 6.41 | 18.15 | 83.23 |
| SSVMF | 0.64 | 38.84 | 7.03 | 0.62 | 39.2 | 7.54 | 0.63 | 38.96 | 7.39 |
| FPGF | 0.58 | 38.38 | 6.36 | 0.58 | 37.58 | 7.53 | 0.59 | 37.56 | 7.25 |
| VMF | 3.53 | 32.49 | 40.82 | 3.41 | 32.85 | 39.93 | 3.45 | 32.71 | 40.10 |
| VDOSF | 0.90 | 32.46 | 7.78 | 0.72 | 34.72 | 8.42 | 2.92 | 22.53 | 27.59 |
| HDF | 3.66 | 32.52 | 41.91 | 3.56 | 32.82 | 41.26 | 3.59 | 32.70 | 41.30 |
| LIMF | 2.07 | 34.03 | 24.66 | 2.32 | 34.38 | 28.50 | 2.14 | 34.55 | 25.91 |
| VLUMS | 0.57 | 37.97 | 6.44 | 0.54 | 38.41 | 7.32 | 0.55 | 38.28 | 7.09 |
| AVLUMS | 0.95 | 37.29 | 10.50 | 0.90 | 37.85 | 10.34 | 0.92 | 37.53 | 10.36 |
| MICM | 0.65 | 37.15 | 7.72 | 0.45 | 38.72 | 6.81 | 0.51 | 38.41 | 6.78 |
| AVFF | 0.56 | 38.33 | 6.67 | 0.42 | 40.29 | 5.71 | 0.45 | 39.64 | 5.84 |
| AVMF | 0.65 | 37.08 | 7.19 | 0.63 | 36.56 | 8.92 | 0.63 | 36.65 | 8.44 |
| ACIF | 0.72 | 34.79 | 6.70 | 0.61 | 37.45 | 6.60 | 3.37 | 22.61 | 31.94 |
| CWVMF | 1.60 | 35.18 | 18.18 | 1.44 | 35.95 | 16.84 | 1.49 | 35.56 | 16.98 |
| MCWVMF | 0.93 | 36.48 | 10.37 | 0.85 | 37.14 | 10.16 | 0.91 | 36.87 | 10.62 |
| FANRF | 0.54 | 38.73 | 6.08 | 0.55 | 38.49 | 6.95 | 0.54 | 38.39 | 6.98 |
| SANRF | 0.52 | 39.08 | 5.89 | 0.53 | 38.63 | 6.96 | 0.54 | 38.61 | 6.72 |
| GAVSF | 1.37 | 31.84 | 14.72 | 1.14 | 34.62 | 14.04 | 3.13 | 22.68 | 31.37 |

Since RGB is not a perceptually uniform space in the sense that differences between colors in this space do not correspond to color differences perceived by humans, the restoration errors are often analyzed using the perceptually uniform color spaces. In this paper we will use the CIE LUV color space and the

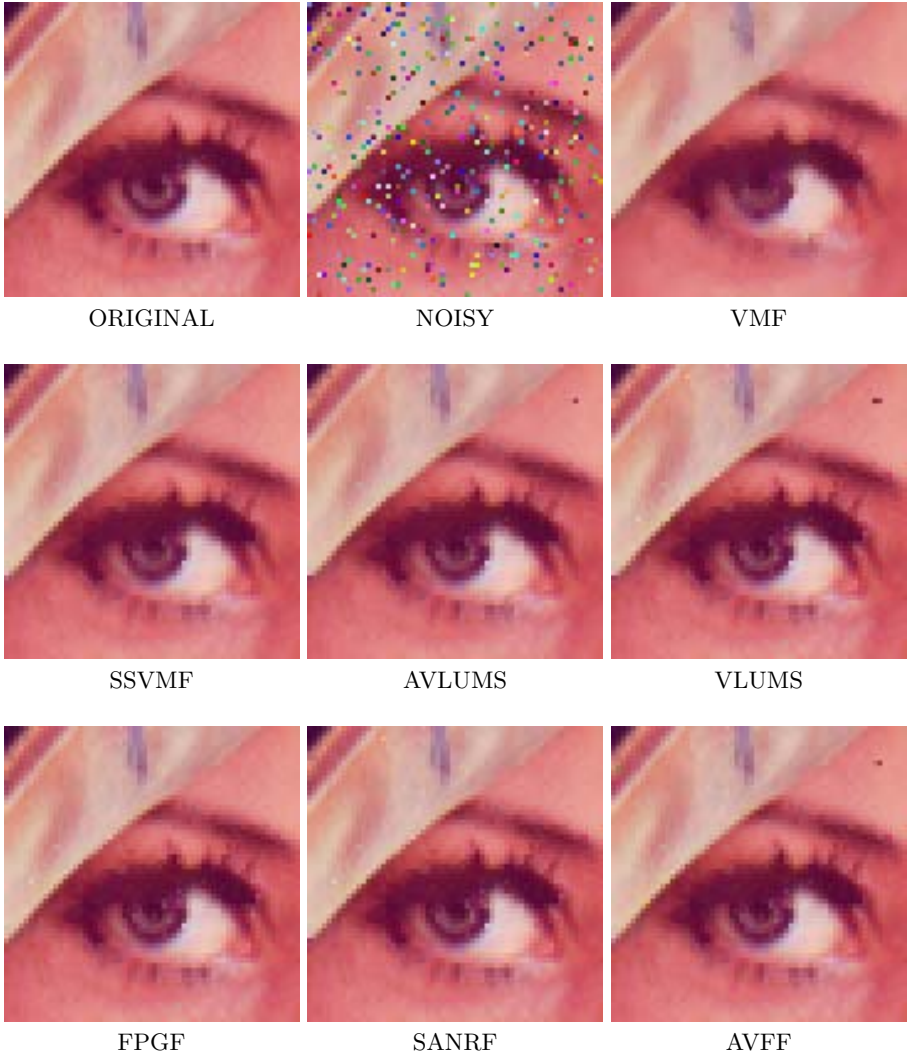


Fig. 7. Illustrative comparison of the filtering efficiency with some other noise removal algorithms from Tab. 1

Normalized Color Difference (NCD) defined as [18]

$$\Delta E = \frac{1}{N} \sum_{i=1}^N \sqrt{(L_{\mathbf{o}_i}^* - L_{\mathbf{x}_i}^*)^2 + (u_{\mathbf{o}_i}^* - u_{\mathbf{x}_i}^*)^2 + (v_{\mathbf{o}_i}^* - v_{\mathbf{x}_i}^*)^2}, \quad (9)$$

$$NCD = \frac{N \Delta E}{\sum_{i=1}^N \sqrt{(L_{\mathbf{o}_i}^*)^2 + (u_{\mathbf{o}_i}^*)^2 + (v_{\mathbf{o}_i}^*)^2}}, \quad (10)$$

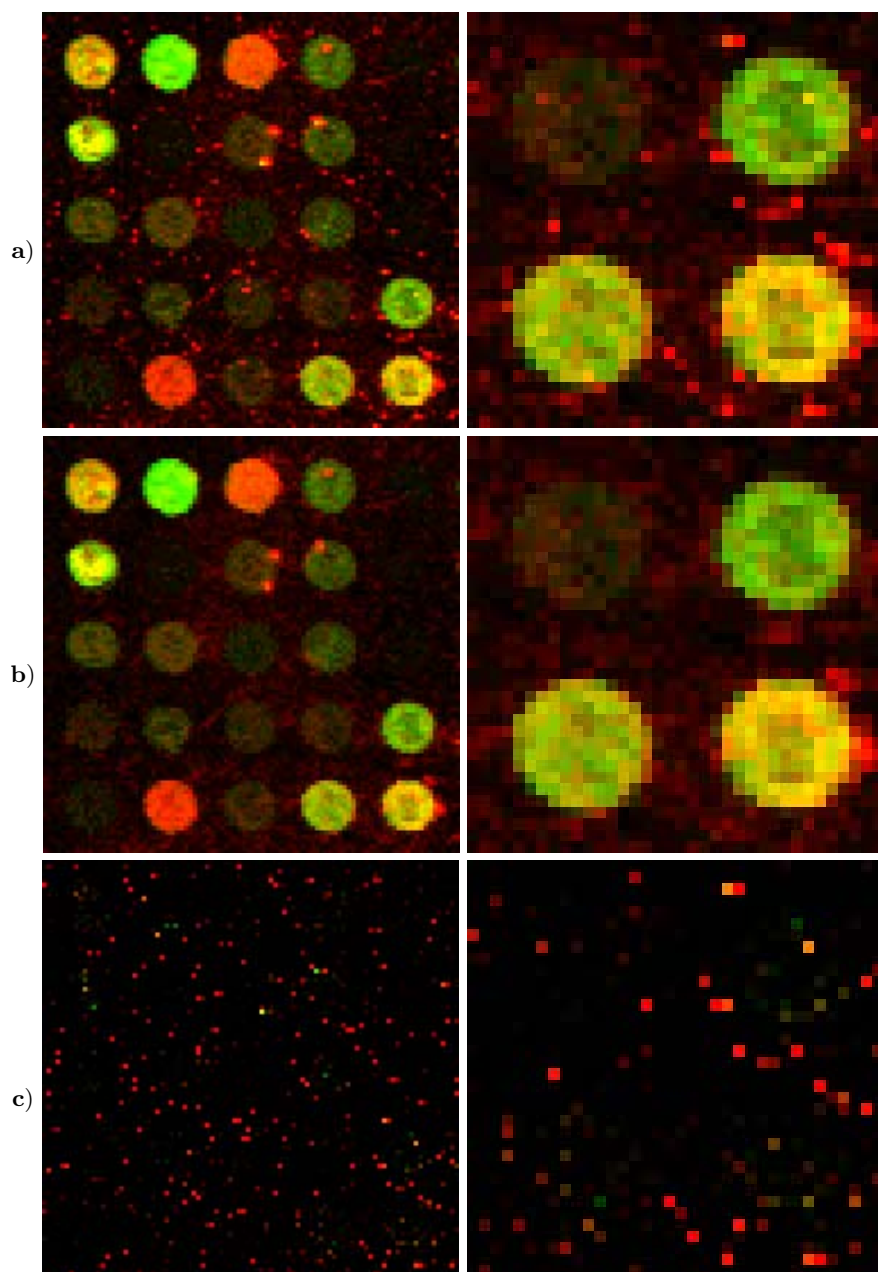


Fig. 8. Results of noise reduction in a cDNA image: a) noisy image and the zoom of its down-right corner, b) the output of the proposed adaptive filter and c) the difference between the original and restored images

where L^* represents lightness values and u^*, v^* chrominance values corresponding to original \mathbf{o}_i and noisy (filtered) \mathbf{x}_i samples expressed in the CIE LUV color space.

The efficiency of the proposed noise cancellation algorithm has been verified on a set of commonly used test images contaminated with the described noise models. The comparison of the noise suppression efficiency with the algorithms listed in Tab. 1 presented in Tab. 2 shows that the new technique is comparable with the best available techniques of impulsive noise suppression. The plots shown in Fig. 5 indicate that the new technique is significantly superior to the VMF. Also the illustrative comparison provided in Fig. 7 shows that the proposed filter efficiently removes the impulses and preserves edges and small image details.

The efficiency of the proposed filtering framework has also been confirmed by the filtering results obtained when denoising the cDNA microarray images with unknown amount of noise, Fig. 8. It can be noticed that the proposed filter adapts to the noise level and removes the spikes only, while preserving the textural information.

4 Conclusion

In the paper a fully adaptive soft-switching scheme, based on the vector median has been presented. The proposed filtering structure is comparable with the best available filtering schemes and can be applied for the removal of impulsive noise in natural images. It is relatively fast and the proposed adaptation scheme enables automatic filtering independently of the noise intensity. It is worth noticing that the proposed algorithm is also efficient in the case of gray scale images.

References

1. Astola, J., Haavisto, P., Neuvo, Y.: Vector Median Filters. Proc. of the IEEE, Vol. 78, No. 4 (1990) 678-689
2. Kuan, D.T., Sawchuk, A.A., Strand, T.C., Chavel, P.: Adaptive Noise Smoothing Filter for Images with Signal-Dependent Noise. IEEE Trans. on PAMI, Vol. 7, No. 2 (1985) 165-177
3. Smolka, B., Plataniotis, K.N.: Soft-Switching Adaptive Technique of Impulsive Noise Removal in Color Images. Lecture Notes in Computer Science, Vol. 3656 (2005) 686-694
4. Smolka, B., Chydzinski, A.: Fast Detection and Impulsive Noise Removal in Color Images. Real-Time Imaging, Vol. 11 (2005) 389-402
5. Lukac, R.: Color Image Filtering by Vector Directional Order-Statistics. Pattern Recognition and Image Analysis, Vol. 12, No. 3 (1990) 279-285
6. Gabbouj, M., Cheickh, FA: Vector median - Vector Directional Hybrid Filter for Colour Image Restoration. Proc. of EUSIPCO, Trieste, Italy, September 10-13, (1996) 879-881
7. Beghdadi, A., Khellaf, K.: A Noise-Filtering Method Using a Local Information Measure. IEEE Transactions on Image Processing, Vol. 6 (1997) 879-882

8. Lukac, R.: Vector LUM Smoothers as Impulse Detector for Color Images. Proc. of European Conference on Circuit Theory and Design (ECCTD), Vol. III, Espoo, Finland, (2001) 137-140
9. Lukac, R., Marchevsky, S.: Adaptive Vector LUM Smoother. Proc. of IEEE International Conference on Image Processing (ICIP), Vol. 1, Thessaloniki, Greece, (2001) 878-881
10. Park, J., Kurz, L.: Image Enhancement Using the Modified ICM Method. IEEE Transactions on Image Processing, Vol. 5, No. 5, (1996) 765-771
11. Lukac, R., Fischer, V., Motyl, G., Drutarovsky, M.: Adaptive Video Filtering Framework. International Journal of Imaging Systems and Technology, Vol. 14, No. 6, (2004) 223-237
12. Lukac, R.: Adaptive Vector Median Filtering. Pattern Recognition Letters, Vol. 24, No. 12, (2003) 1889-1899
13. Lukac, R.: Adaptive Color Image Filtering Based on Center-Weighted Vector Directional Filters. Multidimensional Systems and Signal Processing, Vol. 15, No. 2, (2004) 169-196
14. Smolka, B.: Efficient Modification of the Central Weighted Vector Median Filter. Lecture Notes in Computer Science, 2449 (2002) 166-173
15. Smolka, B., Lukac, R., Chydzinski, A., Plataniotis, K.N., Wojciechowski, K.: Fast Adaptive Similarity Based Impulsive Noise Reduction Filter. Real Time Imaging, Vol. 9, (2003) 261-276
16. Smolka, B., Plataniotis, K.N., Chydzinski, A., Szczepanski, M., Venetsanopoulos, A.N., Wojciechowski, K.: Self-Adaptive Algorithm of Impulsive Noise Reduction in Color Images. Pattern Recognition, Vol. 35, (2002) 1771-1784
17. Lukac, R., Smolka, B., Plataniotis, K.N., Venetsanopoulos, A.N.: Generalized Adaptive Vector Sigma Filters. In: Proc. of IEEE International Conference on Multimedia and Expo (ICME), Baltimore, USA, Vol. I, (2003) 537-540
18. Plataniotis, K.N., Venetsanopoulos, A.N.: Color Image Processing and Applications. Springer Verlag, (2000)
19. Viero, T., Öistämö, K., Neuvo, Y.: Three-dimensional Median-related Filters for Color Image Sequence Filtering. IEEE Trans. on Circuits and Systems for Video Technology, Vol. 4, No. 2, (1994) 129-142
20. Smolka, B., Plataniotis, K.N., Venetsanopoulos, A.N.: Nonlinear Techniques for Color Image Processing. In: Nonlinear Signal and Image Processing, Theory, Methods, and Applications, Editors K.E. Barner, G.R. Arce, CRC Press, Boca Raton, FL., USA, (2004) 445-505

A Simple Method for Designing 2D \mathbf{IM} -Channel Near-PR Filter Banks with Linear Phase Property[†]

Guangming Shi, Liang Song, Xuemei Xie, and Danhua Liu

School of Electronic Engineering Xidian University, Xi'an, P.R. China
gmshi@xidian.edu.cn, lsong@mail.xidian.edu.cn

Abstract. Two dimensional (2D) nonseparable filter banks with linear phase (LP) are desired for image sub-band coding and compression. In this paper, we propose a method for designing 2D nonseparable filter banks with the LP and near perfect reconstruction (near-PR) properties. By combining the unimodular transformation and a separable \mathbf{IM} -channel LP filter bank, the design problem is simplified to that of two one-dimensional (1D) LP filter banks. For 1D case, a novel method by employing partial cosine modulation is used to design near-PR filter banks with LP analysis and synthesis filters. For 2D case, we cascade two 1D near-PR LP filter banks in the form of tree structure to design a separable LP filter bank. With the unimodular transformation, a nonseparable LP filter bank is obtained. In addition, the filter bank achieves the near-PR property without sophisticated nonlinear optimization procedures. Design example shows the efficiency and simplicity of the proposed method.

1 Introduction

Multidimensional (MD) filter banks have been studied extensively due to their wide range of applications in image, video and other fields. Among the various MD filter banks, separable filter banks in [1], [2] are easy to implement and design, simply by factoring the 1D-components. In contrast to separable systems, nonseparable filter banks in [3-9], [12] offer more flexibility and usually provide better performance, and thus, are more appropriate for dealing with MD signals. One key property of a filter bank is the perfect reconstruction (PR) measure, which guarantees that the original input can be perfectly reconstructed from the output. However, the design of PR nonseparable filter banks by nonlinear constrained optimization methods in [3-5] is very difficult because of the large numbers of variables and optimization constraints. The methods of constructing two-dimensional (2D) cosine modulated filter banks (CMFB) with the PR property are proposed in [6] and [7]. The design of the system involves only a general nonlinear constrained optimization with the objective function representing the stopband energy of the prototype filter. But in these systems, the analysis and synthesis filters do not possess the linear phase (LP) property, which is highly desired in image coding application. Motivated by the development of 1D LP CMFB,

[†] Work supported by National Natural Science Foundation of China (Grant No. 60372047) and Natural Science Foundation of Shaanxi (Grant No. 2005F18).

the design of 2D PR CMFB with LP analysis and synthesis filters is studied in [8]. The design problem is reduced to the design of a LP prototype with a parallelogram support. Since the configuration of the resulting LP CMFB lacks vertex permissibility, the resulting analysis and synthesis filters can not have good frequency selectivity. In [9], the authors proposed a two subsystem (one subsystem is a cosine modulated and the other is a sine modulated version of the prototype) decomposition scheme to construct CMFB with the LP and PR properties. However, the filter bank has special form where two filters occupy the same band in this case.

In this paper, we propose a simple design method for 2D nonseparable near-PR filter banks with LP analysis and synthesis filters. We combine the unimodular transformation introduced in [10-13] and a separable LP filter bank, by which the design problem can be simplified to that of two 1D filter banks with LP analysis and synthesis filters. For 1D case, a novel method, namely partial cosine modulation, is used to design LP near-PR filter banks. Constraints to eliminate the significant aliasing and amplitude distortion are given. By employing the Parks-McClellan algorithm in constructing the prototype filter, the first and the last analysis filters, other analysis filters are cosine modulated versions of the prototype. Then we get the corresponding synthesis filters according to time-reverse property. In this way, all filters achieve ideal magnitude responses. For 2D case, we concatenate two 1D LP filter banks in the form of tree structure to design a separable LP one. We show that with the unimodular transformation, the 2D filters in the designed nonseparable system are partially cosine modulated, possess the LP property along with good frequency selectivity. The resulting 2D system possesses the near-PR property as its 1D counterpart. In addition, our method has no restriction to the number of channels and is of conceptual simplicity.

Notations: In this paper, \mathbf{M} denotes a 2×2 nonsingular integer matrix and the notation $\Xi(\mathbf{M})$ represents the set of all integer vectors of the form $\mathbf{n} = \mathbf{M}\mathbf{x}$ for $\mathbf{x} \in [0,1)^2$. The number of elements in $\Xi(\mathbf{M})$ is equal to the absolute value of determinant of \mathbf{M} , which is denoted by $|\mathbf{M}|$. The sampling lattice generated by \mathbf{M} is denoted by $LAT(\mathbf{M})$. $SPD(\mathbf{M})$ is the symmetric parallelepiped of \mathbf{M} , defined as the set of vectors of the form $\mathbf{M}\mathbf{x}$, $\mathbf{x} \in [-1,1)^2$. An integer matrix \mathbf{U} is a unimodular matrix if $|\mathbf{U}|=1$. The inverse \mathbf{U}^{-1} of a unimodular matrix \mathbf{U} is also a unimodular matrix.

The rest of this paper is organized as follows: in section 2, the principle of 2D maximally decimated filter banks will be introduced. Section 3 presents the method on designing 2D $|\mathbf{M}|$ -channel LP near-PR filter bank by combining the design method of 1D M -channel LP filter banks and the unimodular transformation, followed an example in section 4. Finally, some conclusions will be drawn in section 5.

2 Principle of 2D Maximally Decimated Filter Banks

A general structure of 2D $|\mathbf{M}|$ -channel maximally decimated filter banks with decimation matrix \mathbf{M} is depicted in Fig. 1.

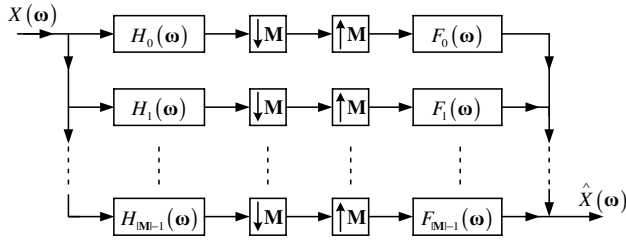


Fig. 1. $|M|$ -channel maximally decimated filter bank

The input-output relationship of this system can be expressed as

$$\hat{X}(\omega) = X(\omega)T(\omega) + \sum_{\substack{\mathbf{m}_i \in \Xi(\mathbf{M}^T) \\ \mathbf{m}_i \neq \mathbf{0}}} X(\omega - 2\pi\mathbf{M}^{-T}\mathbf{m}_i)A_{\mathbf{m}_i}(\omega), \tag{1}$$

where

$$T(\omega) = \frac{1}{|M|} \sum_{i=0}^{|M|-1} H_i(\omega)F_i(\omega), \tag{2a}$$

$$A_{\mathbf{m}_i}(\omega) = \frac{1}{|M|} \sum_{j=0}^{|M|-1} H_j(\omega - 2\pi\mathbf{M}^{-T}\mathbf{m}_i)F_j(\omega), \tag{2b}$$

$$\mathbf{m}_i \in \Xi(\mathbf{M}^T), \mathbf{m}_i \neq \mathbf{0}$$

It can be seen from Eq.(1) that the reconstructed signal $\hat{X}(\omega)$ is a linear combination of the input signal $X(\omega)$ and its shifted versions $X(\omega - 2\pi\mathbf{M}^{-T}\mathbf{m}_i)$. We refer to the terms, $T(\omega)$, $X(\omega - 2\pi\mathbf{M}^{-T}\mathbf{m}_i)$ and $A_{\mathbf{m}_i}(\omega)$, as the overall transfer function, alias components and alias transfer functions, respectively.

Similar to the 2D case, in a 1D filter bank with sampling factor M , the z -domain input-output relationship can be expressed as

$$\hat{X}(z) = X(z)T(z) + \sum_{l=1}^{M-1} X(zW_M^l)A_l(z), \tag{3}$$

where

$$T(z) = \frac{1}{M} \sum_{k=0}^{M-1} H_k(z)F_k(z), \tag{4a}$$

$$A_l(z) = \frac{1}{M} \sum_{k=0}^{M-1} H_k(zW_M^l)F_k(z), \tag{4b}$$

$$W_M = e^{-j2\pi/M}, 1 \leq l \leq M - 1, 0 \leq k \leq M - 1.$$

Correspondingly, we respectively refer to the terms, $T(z)$, $X(zW_M^l)$ and $A_l(z)$, as the overall transfer function, alias components and the alias transfer functions.

Two quantities are defined to measure the reconstruction degree of the filter bank. One is denoted by E_{pp-2D} in 2D case or E_{pp} in 1D case, which measures the worst possible amplitude distortion (AMD), describing the maximum peak to peak ripple of $|M \|T(\omega)|$ or $M |T(e^{j\omega})|$. The other E_{a-2D} in 2D case or E_a in 1D case that is used

to evaluate the worst possible peak ALD is the maximum value of $\sqrt{\sum_{\substack{m_i \in \Xi(M^T) \\ m_i \neq 0}} |A_{m_i}(\omega)|^2}$

or $\sqrt{\sum_{l=1}^{M-1} |A_l(e^{j\omega})|^2}$. As a general rule, under the condition that the filter bank is free from phase distortion (PHD), if the values of E_{pp-2D} and E_{a-2D} are in the order of 10^{-3} or below, we classify the filter bank as a near-PR one; on the other hand, if both values are in the order of 10^{-12} or below, we consider that the filter bank achieves the PR property.

3 2D IMI-Channel Linear Phase Near-PR Filter Banks

We introduce a novel method for designing 1D M -channel near-PR filter banks with LP analysis and synthesis filters in section 3.1. Section 3.2 gives a brief introduction of unimodular transformation. In section 3.3, we combine the methods shown in section 3.1 and section 3.2 to obtain a 2D IMI-channel LP near-PR filter bank.

3.1 1D LP Near-PR Filter Banks with Partial Cosine Modulation

In a 1D M -channel CMFB, the analysis filters $h_k(n)$ and synthesis filters $f_k(n)$ are obtained by cosine modulation of a prototype $p(n)$ with order N

$$h_k(n) = 2p(n) \cos\left(\frac{\pi}{2M}(2k+1)\left(n - \frac{N}{2}\right) + \theta_k\right), \tag{5a}$$

$$f_k(n) = 2p(n) \cos\left(\frac{\pi}{2M}(2k+1)\left(n - \frac{N}{2}\right) - \theta_k\right), \tag{5b}$$

$$0 \leq n \leq N, 0 \leq k \leq M - 1,$$

where $\theta_k = (-1)^k \pi/4$; $h_k(n)$ and $f_k(n)$ are related by

$$f_k(n) = h_k(N - n) \text{ or } F_k(z) = z^{-N} H_k(z^{-1}). \tag{6}$$

In this case, the overall transfer function $T(z)$ in Eq.(4a) can be expressed in frequency-domain as

$$T(e^{j\omega}) = \frac{e^{-jN\omega}}{M} \sum_{k=0}^{M-1} |H_k(e^{j\omega})|^2, \tag{7}$$

which guarantees the elimination of PHD in the filter bank.

It is obvious that the analysis and synthesis filters of the CMFB system in Eq.(5) do not have the LP property. To obtain LP filter banks using the cosine modulation theorem and make the aliasing error negligible, we design the first analysis filter $h_0(n)$ to be symmetric, and the last one $h_{M-1}(n)$ to be symmetric for odd M or anti-symmetric for even M ; $h_k(n)$, $1 \leq k \leq M - 2$, is generated by Eq.(5a) where θ_k becomes

$$\theta_k = \pi/4 - (-1)^k \pi/4, 1 \leq k \leq M - 2. \tag{8}$$

Each synthesis filter $f_k(n)$, $0 \leq k \leq M - 1$, is generated from Eq.(6). In this way, the analysis and synthesis filters $H_k(z)$ and $F_k(z)$, $0 \leq k \leq M - 1$, are all linear phase and the system is approximately free of ALD. We call this method partial cosine modulation. It can be proved that the analysis filters $h_k(n)$, $0 \leq k \leq M - 1$, satisfy the alternative symmetric relation and the synthesis filters are given by

$$f_k(n) = (-1)^k h_k(n), 0 \leq k \leq M - 1. \tag{9}$$

Now we introduce a simple method for designing the prototype filter $P(z)$, $H_0(z)$ and $H_{M-1}(z)$ to eliminate the significant AMD in the system. With Eq.(7), we use the similar tricks in [15] by employing Parks-McClellan algorithm to impose approximately power complementary constraint on $H_0(z)$, $P(z)$ and $H_{M-1}(z)$. For simplicity, we only describe the following constraint between $P(z)$ and $H_0(z)$:

$$|H_0(e^{j\omega})|^2 + |P(e^{j(\omega-3\pi/2M)})|^2 = 1, \omega \in \left(\frac{\pi}{2M}, \frac{3\pi}{2M}\right). \tag{10}$$

Thus, if we force the transition bands of $H_0(e^{j\omega})$ and the frequency shifted prototype filter $P(e^{j(\omega-3\pi/2M)})$ to follow a cosine roll-off function such that $\cos^2 \theta(\omega) + \sin^2 \theta(\omega) = 1$, Eq. (10) is satisfied. If the similar constraint between $P(z)$ and $H_{M-1}(z)$ is met in the same way, $|T(e^{j\omega})|$ is approximately flat in $-\pi < \omega < \pi$.

Employing the Parks-McClellan algorithm with cosine roll-off characteristic in designing the prototype filter $P(z)$, $H_0(z)$ and $H_{M-1}(z)$, a LP near-PR filter bank can be obtained. This is performed by using *remez* (or *firpm*) function in MATLAB.

3.2 Unimodular Transformation

The unimodular transformation proposed in [10-13] is an efficient method to convert a 2D separable filter bank to a nonseparable one. To begin with, let us consider a separable filter bank denoted by S_Λ with diagonal decimation matrix Λ in Fig. 1. Fig. 2(a) shows the i th branch of this system. Since a unimodular decimator \mathbf{U} or a unimodular expander \mathbf{U} merely permutes the input signal, we carry out \mathbf{U} -fold decimation before the system S_Λ and \mathbf{U} -fold interpolation after S_Λ . Fig. 2(b) and 2(c) show the i th sub-band of the resulting system S_M with decimation matrix $\mathbf{M} = \mathbf{U}\Lambda$. This transformation of S_Λ to S_M is called the unimodular transformation.

Using noble identities in [14], each separable analysis filter $G_i(\omega)$ and synthesis filter $Q_i(\omega)$ in S_Λ are converted to nonseparable filters $H_i(\omega)$ and $F_i(\omega)$ by

$$H_i(\omega) = G_i(\mathbf{U}^T \omega), F_i(\omega) = Q_i(\mathbf{U}^T \omega). \tag{11}$$

Note that the support of each filter is the union of four shifts of $SPD(\frac{\pi}{2}\mathbf{M}^{-T})$. For \mathbf{M} is not diagonal and $|\mathbf{M}| = |\Lambda|$, the system S_M is not separable and has $|\Lambda|$ channels.

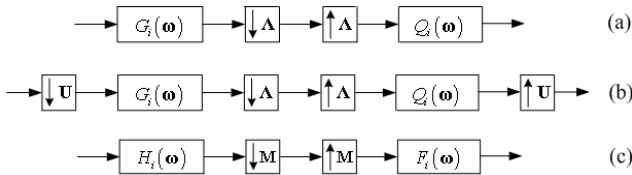


Fig. 2. Unimodular transformation. (a) The i th branch of S_Λ . (b) The i th branch of S_M . (c) The i th equivalent branch of S_M .

3.3 2D LP Near-PR Filter Banks with Partial Cosine Modulation

We start from a 2D separable $|\lambda_0\lambda_1|$ -channel filter bank S_Λ with diagonal decimation matrix $\Lambda = \begin{bmatrix} \lambda_0 & 0 \\ 0 & \lambda_1 \end{bmatrix}$. This can be done by concatenating two 1D filter banks denoted by S_{λ_0} and S_{λ_1} with $|\lambda_0|$ and $|\lambda_1|$ channels respectively, that is

$$g_{ij}(n_1, n_2) = h_i^1(n_1)h_j^2(n_2), q_{ij}(n_1, n_2) = f_i^1(n_1)f_j^2(n_2), \tag{12}$$

$$0 \leq i \leq |\lambda_0| - 1, 0 \leq j \leq |\lambda_1| - 1; 0 \leq n_1 \leq N_1, 0 \leq n_2 \leq N_2.$$

Here $g_{ij}(n_1, n_2)$ and $q_{ij}(n_1, n_2)$ are the impulse responses of the analysis filter $G_{ij}(\omega)$ and synthesis filter $Q_{ij}(\omega)$ in S_Λ ; $h_i^1(n_1)$, $f_i^1(n_1)$, $h_j^2(n_2)$ and $f_j^2(n_2)$ are the impulse

responses of analysis and synthesis filters in S_{λ_0} and S_{λ_1} with order N_1 and N_2 respectively. Using the method we described in 3.1, all filters in these two 1D systems can be designed to possess the LP property, which leads to

$$g_{ij}(n_1, n_2) = g_{ij}(N_1 - n_1, N_2 - n_2); \quad q_{ij}(n_1, n_2) = q_{ij}(N_1 - n_1, N_2 - n_2). \quad (13)$$

This means that the filters in S_Λ are also LP. With Eq.(9) and (12), the overall transfer function $T_\Lambda(e^{j\omega_0}, e^{j\omega_1})$ of S_Λ , $T_{\lambda_0}(e^{j\omega_0})$ of S_{λ_0} and $T_{\lambda_1}(e^{j\omega_1})$ of S_{λ_1} are related by

$$T_\Lambda(e^{j\omega_0}, e^{j\omega_1}) = T_{\lambda_0}(e^{j\omega_0})T_{\lambda_1}(e^{j\omega_1}), \quad (14)$$

which suggests that the maximum peak to peak ripple of $|\lambda_0\lambda_1|T_\Lambda(\omega)$ equals nearly to the sum of that of two 1D systems. Eq.(14) also ensures the elimination of PHD in the filter bank S_Λ . Since two 1D systems designed using partial cosine modulation are approximately free of ALD, we can conclude that the system S_Λ is also approximately alias-free. Accordingly, the near-PR property is preserved in the designed 2D system. Furthermore, filters of S_Λ are partially cosine modulated by a 2D separable prototype filter $p(n_1, n_2) = p^1(n_1)p^2(n_2)$

$$g_{ij}(n_1, n_2) = 4p(n_1, n_2) \cdot \cos\left(\frac{\pi}{2|\lambda_0|}\left(2i+1\right)\left(n_1 - \frac{N_1}{2}\right) + \theta_i\right) \cdot \cos\left(\frac{\pi}{2|\lambda_1|}\left(2j+1\right)\left(n_2 - \frac{N_2}{2}\right) + \theta_j\right), \quad (15a)$$

$$q_{ij}(n_1, n_2) = 4p(n_1, n_2) \cdot \cos\left(\frac{\pi}{2|\lambda_0|}\left(2i+1\right)\left(n_1 - \frac{N_1}{2}\right) - \theta_i\right) \cdot \cos\left(\frac{\pi}{2|\lambda_1|}\left(2j+1\right)\left(n_2 - \frac{N_2}{2}\right) - \theta_j\right), \quad (15b)$$

$$0 \leq n_1 \leq N_1, 0 \leq n_2 \leq N_2,$$

where $\theta_i = \pi/4 - (-1)^i \pi/4$, $1 \leq i \leq |\lambda_0| - 2$ and $\theta_j = \pi/4 - (-1)^j \pi/4$, $1 \leq j \leq |\lambda_1| - 2$; $p^1(n_1)$ and $p^2(n_2)$ are the prototype filters of S_{λ_0} and S_{λ_1} respectively. With the relationship given by Eq.(11), the nonseparable analysis filters $h_{ij}(n_1, n_2)$ and synthesis filters $f_{ij}(n_1, n_2)$ can be obtained by

$$h_{ij}(\mathbf{n}) = \begin{cases} g_{ij}(\mathbf{U}^{-1}\mathbf{n}) & \text{if } \mathbf{n} \in \text{LAT}(\mathbf{U}) \\ \mathbf{0} & \text{otherwise} \end{cases}, \quad f_{ij}(\mathbf{n}) = \begin{cases} q_{ij}(\mathbf{U}^{-1}\mathbf{n}) & \text{if } \mathbf{n} \in \text{LAT}(\mathbf{U}) \\ \mathbf{0} & \text{otherwise} \end{cases}. \quad (16)$$

Accordingly, $h_{ij}(n_1, n_2)$ and $f_{ij}(n_1, n_2)$ hold similarly the LP property after performing the unimodular transformation. The overall transfer function $T_\Lambda(\omega)$ of S_Λ and $T_M(\omega)$ of S_M are related by

$$T_M(\omega) = T_\Lambda(U^T \omega), \tag{17}$$

which means $|T_M(\omega)|$ will be as flat as $|T_\Lambda(\omega)|$. It can be proved that if $T_\Lambda(\omega)$ has the LP property, then S_M is free of PHD. We can also conclude that the resulting 2D nonseparable filter bank S_M will be approximately free from ALD and partially cosine modulated. Thus, the system S_M possesses the same properties as S_Λ does.

4 Design Example

In this section, we present an example to illustrate the proposed method on designing LP near-PR filter banks.

To begin with, we should design two 1D M -channel LP filter banks. We set the numbers of the channels to be 4 and 3 respectively.

In the 1D 4-channel LP filter bank, the length of each analysis filter is $N = 100$. The stopband cutoff frequency of the prototype filter is $\omega_s = 0.240\pi(\text{rad})$. The magnitude responses of the analysis filters are shown in Fig. 3(a). The stopband attenuation is about 97 dB. The amplitude distortion $E_{pp} = 2.438 \times 10^{-3}$ and the aliasing error $E_a = 3.239 \times 10^{-4}$.

In the 1D 3-channel LP filter bank, the length of each analysis filter is $N = 99$ and the stopband cutoff frequency of the prototype is $\omega_s = 0.197\pi(\text{rad})$. Fig. 3(b) shows the magnitude responses of the analysis filters with the stopband attenuation being 97 dB, $E_{pp} = 2.036 \times 10^{-3}$ and $E_a = 6.766 \times 10^{-4}$.

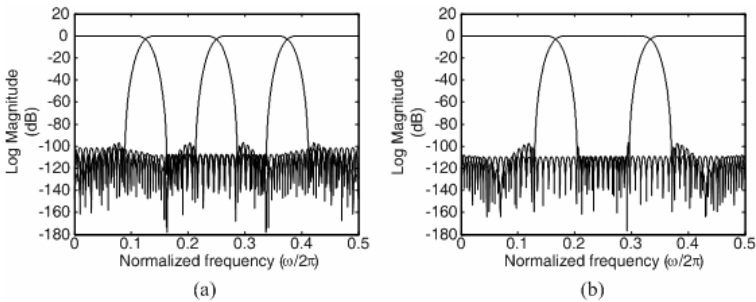


Fig. 3. (a) Magnitude responses of the analysis filters of the 4-channel case. (b) Magnitude responses of the analysis filters of the 3-channel case.

According to the proposed method, we can get a 2D filter bank based on the above two 1D filter banks.

This example is to evaluate the performance of the designed 2D $|M|$ -channel near-PR filter bank with LP filters. Let $\Lambda = \begin{bmatrix} 3 & 0 \\ 0 & 4 \end{bmatrix}$ and $U = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$, then a 12-channel nonseparable filter bank S_M with decimation matrix $M = U\Lambda = \begin{bmatrix} 6 & 4 \\ 3 & 4 \end{bmatrix}$ is obtained.

Spectral supports of the first four analysis filters, $H_0 \sim H_3$, in the desired filter bank are shown in Fig. 4(a) and the magnitude response of the designed lowpass analysis filter H_0 is shown in Fig. 4(b). Fig. 4(c) and (d) show respectively amplitude distortion E_{pp-2D} and the aliasing error E_{a-2D} with $E_{pp-2D} = 4.435 \times 10^{-3}$ and $E_{a-2D} = 2.004 \times 10^{-4}$. The stopband attenuation is about 97.0 dB. It can be seen that the near-PR requirement is satisfied.

We apply the filter bank designed above on the 512×512 picture ‘‘Lena’’. The original image is shown in Fig. 5(a). The first four sub-band images, corresponding to

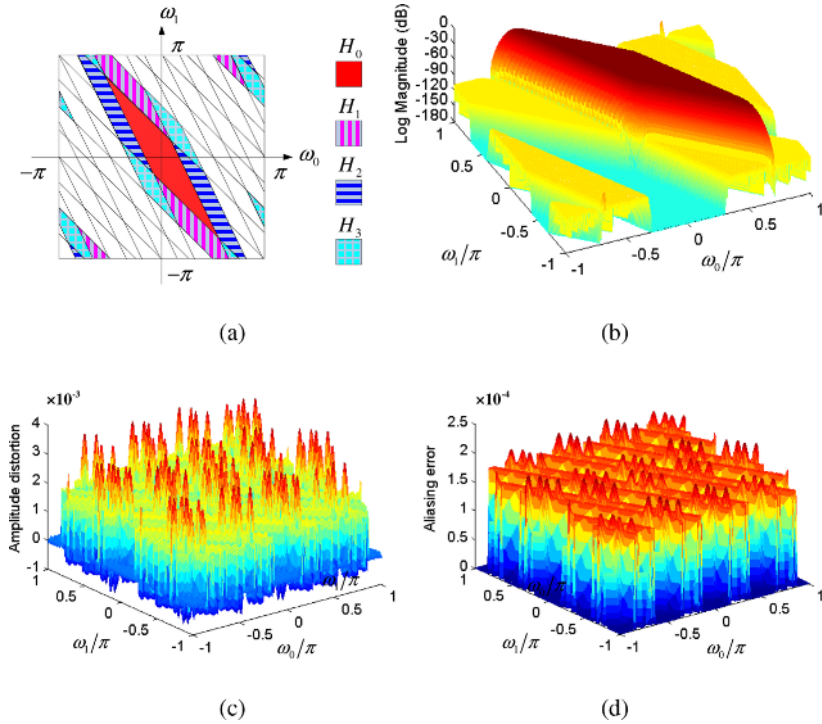


Fig. 4. 2D 12-channel LP near-PR filter bank. (a) Spectral supports of the first four analysis filters in the desired filter bank. (b) Magnitude response of H_0 . (c) Amplitude distortion. (d) Aliasing error.

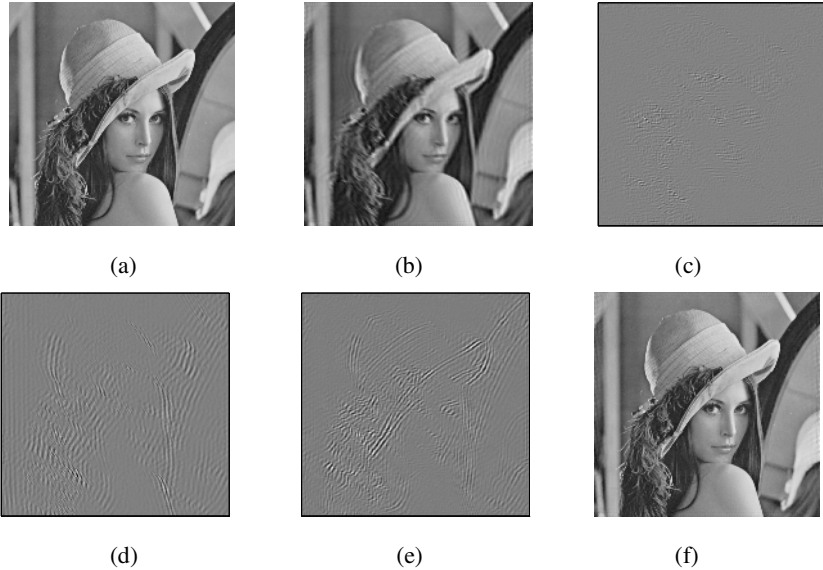


Fig. 5. (a) Original image. (b)-(e) The first four reconstructed sub-band images. (f) Reconstructed image (PSNR=40.25 dB).

$H_0 \sim H_3$, are shown respectively in Fig. 5(b)-(e). Fig. 5(f) shows the reconstructed image with peak signal-to-noise ratio (PSNR) being 40.25 dB. We notice that the reconstruction quality is very good.

5 Conclusions

In this paper, we have proposed a simple method for designing 2D nonseparable IMI-channel near-PR filter banks with the LP property. The filter banks are obtained by combining the unimodular transformation and a separable LP filter bank (hence two 1D filter banks). With the method for designing 1D LP near-PR filter banks, namely partial cosine modulation, the filters in 2D system are partially cosine modulated by a LP prototype filter, which can reduce the design and implementation costs. Moreover, all analysis and synthesis filters possess the linear phase property along with good frequency selectivity. We showed that the resulting system achieves the near-PR property as the 1D system. Simplicity and effectiveness of the proposed method have been testified through the design example.

References

1. Vetterli, M.: Multidimensional sub-band coding: Some theory and algorithms. *Signal Processing*, Vol. 6. February (1984) 97-112
2. Woods, J.W., and O'Neil, S.D.: Subband coding of images. *IEEE Trans. ASSP.*, Vol. 34. No.5. October (1986) 1278-1288

3. Karlsson, G., and Vetterli, M.: Theory of two-dimensional multirate filter banks. *IEEE Trans. ASSP.*, Vol. 38. No.6. June (1990) 925-937
4. Viscito, E., and Allebach, J.P.: The analysis and design of multidimensional FIR perfect reconstruction filter banks for arbitrary sampling lattices. *IEEE Trans. CAS.*, Vol. 38. No.1. January (1991) 29-42
5. Nguyen, T.T., and Orantara, S.: Multidimensional filter banks design by direct optimization. *IEEE ISCAS*, Vol. 2. May (2005) 1090-1093
6. Lin, Y.P., and Vaidyanathan, P.P.: Two-dimensional paraunitary cosine modulated perfect reconstruction filter banks. *IEEE ISCAS*, Vol. 1. April (1995) 752-755
7. Ikehara, M.: Cosine-modulated 2 dimensional FIR filter banks satisfying perfect reconstruction. *IEEE ICASSP*, Vol. 3. April (1994) 137-140
8. Lin, Y.P., and Vaidyanathan, P.P.: Two-dimensional linear phase cosine modulated filter banks. *IEEE ISCAS*, Vol. 2. May (1996) 73-76
9. Nagai, T., and Ikehara, M.: Cosine modulated 2-dimensional perfect reconstruction FIR filter banks with linear phase. *IEEE ICASSP*, Vol. 3. May (1996) 1483-1486
10. Vaidyanathan, P.P.: The role of Smith-form decomposition of integer-matrices, in multidimensional multirate systems. *IEEE ICASSP*, Vol. 3. April (1991) 1777-1780
11. Vaidyanathan, P.P.: New results in multidimensional multirate systems. *IEEE ISCAS*, Vol. 1. June (1991) 468-471
12. Lin, Y.P., and Vaidyanathan, P.P.: On the study of four-parallelogram filter banks. *IEEE Trans. SP.*, Vol. 44. No.11. November (1996) 2707-2717
13. Lin, Y.P., and Vaidyanathan, P.P.: Theory and design of two-dimensional filter banks: A review. *Multidimensional Systems and Signal Processing*, Vol. 7. (1996) 236-330
14. Vaidyanathan, P.P.: *Multirate Systems and Filter Banks*. Prentice Hall, Englewood Cliffs, New Jersey (1992)
15. Xie, X.M., Shi, G.M., and Chen, X.Y.: Optimal prototype filters for near-perfect-reconstruction cosine-modulated filter banks. *CIS 2005, Part II. LNAI 3802*. December (2005) 851-856

Fast Hermite Projection Method

Andrey Krylov¹ and Danil Korchagin²

¹ Moscow State University, Moscow, Russia
kryl@cs.msu.su

² European Organization for Nuclear Research, Geneva, Switzerland
Danil.Korchagin@cern.ch

Abstract. Fast Hermite projection scheme for image processing and analysis is introduced. It is based on an expansion of image intensity function into a Fourier series using full orthonormal system of Hermite functions instead of trigonometric basis. Hermite functions are the eigenfunctions of the Fourier transform and they are computationally localized both in frequency and spatial domains in the contrast to the trigonometric functions. The acceleration of this expansion procedure is based on Gauss-Hermite quadrature scheme simplified by the replacement of Hermite associated weights and Hermite polynomials by an array of associated constants. This array of associated constants depends on the values of Hermite functions. Image database retrieval and image foveation applications based on 2D fast Hermite projection method have been considered. The proposed acceleration algorithm can be also efficiently used in Hermite transform method.

1 Introduction

Trigonometric Fourier series expansion plays a very important role in image processing and image analysis. Nevertheless the trigonometric basis functions are not jointly localized in space and frequency domains of Fourier transform. At the same time, image parameterization with Fourier series enables to perform many of image processing procedures in a most effective way especially due to the existence of Fast Fourier Transform method. We propose to substitute trigonometric functions in Fourier series by Hermite functions basis (“Hermite projection method”) to obtain necessary computational localization. The aim of the work is to design fast Hermite projection method and to apply it to image database retrieval and image foveation problems.

Digital image consists of 2D array of intensities. An expansion of signal information into a series of Hermite functions enables us to perform information analysis of the signal and its Fourier transform at the same time, because the Hermite functions are the eigenfunctions of Fourier transform. These functions are widely used in pure mathematics, where the expansion into Hermite functions is also called as Gram-Charlie series [1], image processing [2-5] and streaming waveform data processing [6]. Joint localization of Hermite functions in the both frequency and spatial domains make using these functions very stable to information errors. Each coefficient of

expansion of the proposed Hermite projection method contains information on the whole treated image and these coefficients can be analyzed progressively. This is the main difference of Hermite projection method with Hermite transform method [2] and basing on this feature we can effectively solve pattern recognition problems like image database retrieval and image foveation.

2 2D Hermite Functions

2D Hermite functions form a full orthonormal in $L_2(-\infty, \infty) \times (-\infty, \infty)$ system.

The 2D Hermite functions are defined as:

$$\psi_{nm}(x, y) = \frac{(-1)^{n+m} e^{-\frac{x^2+y^2}{2}}}{\sqrt{2^{n+m} n! m! \pi}} \cdot \frac{d^n(e^{-x^2})}{dx^n} \cdot \frac{d^m(e^{-y^2})}{dy^m}$$

They can be also determined as superposition of the 1D recurrent formulas [3].

Moreover the 2D Hermite functions are the eigenfunctions of the Fourier transform: $F(\psi_{nm}) = i^{n+m} \psi_{nm}$, where F denotes Fourier transform operator.

The graphs of the 2D Hermite functions are shown in figure 1.

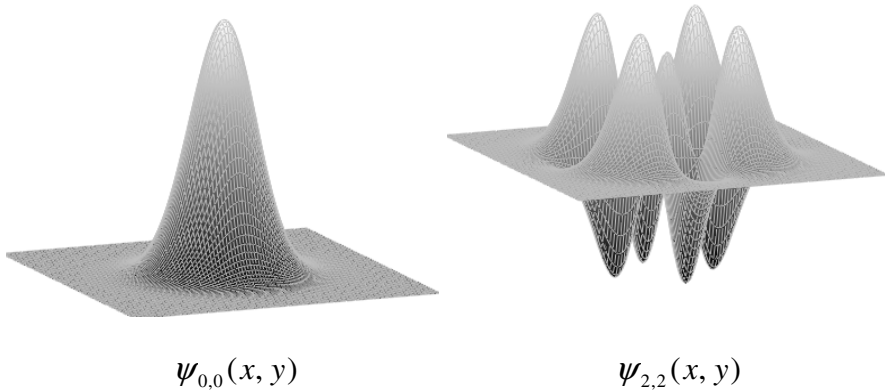


Fig. 1. 2D Hermite functions

These functions are computationally localized both in space and frequency domains. This localization of Hermite functions was first noted by Gabor [7]. The comparison of the localization of different sets of functions like Gabor functions, wavelets, Gaussian derivatives and Hermite functions is an open problem [8] and needs a formulation of an appropriate localization metrics. Nevertheless all sets of localized functions are widely used in image processing.

The set of Hermite functions has the following advantages:

It is an orthonormal set;

Even (odd) Hermite functions tends to cosine (sine) functions when $n \rightarrow \infty$ and the coefficients of Hermite series expansion give the “frequencies” that are analogues to Fourier frequencies.

The main disadvantage of Hermite projection method (expansion into series of Hermite functions) is its high computational complexity.

The idea of fast expansion into series of Hermite functions was first proposed by Eberlein [9]. Nevertheless a successful algorithmic implementation was not developed. Our fast implementation of Hermite projection method is presented in the following section.

3 Fast Hermite Projection Method

In common case 2D Hermite projection method of $f(x, y)$ is defined as:

$$F(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} c_{ij} \psi_{ij}(x, y),$$

where $\psi_{ij}(x, y)$ – 2D Hermite functions, c_{ij} – Hermite coefficients:

$$c_{ij} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \psi_{ij}(x, y) dx dy.$$

It can be represented as superposition of 1D Hermite projection methods:

$$F(x, y) = \sum_{j=0}^{\infty} \overline{c}_j \overline{\psi}_j(y), \quad \overline{c}_j(x) = \int_{-\infty}^{\infty} \overline{f}(x, \xi) \psi_j(\xi) d\xi,$$

$$\overline{f}(x, \xi) = \sum_{i=0}^{\infty} \overline{c}_i(\xi) \psi_i(x), \quad \overline{c}_i(\xi) = \int_{-\infty}^{\infty} f(\chi, \xi) \psi_i(\chi) d\chi.$$

Each coefficient can be redefined through Hermite polynomials as following:

$$c_n = \frac{1}{\alpha_n} \int_{-\infty}^{\infty} e^{-x^2} \left(f(x) e^{x^2/2} \right) H_n(x) dx,$$

where α_n – Hermite normalization constant: $\alpha_n = \sqrt{2^n n! \sqrt{\pi}}$ and $H_n(x)$ is Hermite polynomial:

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n (e^{-x^2})}{dx^n},$$

This integral can be approximated by Gauss-Hermite quadrature [10]:

$$\begin{aligned} c_n &= \frac{1}{\alpha_n} \int_{-\infty}^{\infty} e^{-x^2} \left(f(x) e^{x^2/2} \right) H_n(x) dx \approx \\ &\approx \frac{1}{\alpha_n} \sum_{m=1}^N A_m \left(f(x_m) e^{x_m^2/2} \right) H_n(x_m), \end{aligned}$$

where x_m are zeros of Hermite polynomials $H_N(x)$, A_m are associated weights:

$$A_m = \frac{2^{N-1} N! \sqrt{\pi}}{N^2 H_{N-1}^2(x_m)}.$$

These associated weights are subjected to increase complication of calculation with the increase of N [10]:

$$N = 1, \quad A_1 = 1, 772\ 453\ 850;$$

$$N = 3, \quad \begin{matrix} A_2 = 1, 181\ 635\ 900 \\ A_1 = A_3 = 0, 295\ 408\ 975; \end{matrix}$$

$$A_3 = 0, 945\ 308\ 720$$

$$N = 5, \quad A_2 = A_4 = 0, 393\ 619\ 323.$$

$$A_1 = A_5 = 0, 019\ 953\ 242$$

This problem can be solved by replacement of Hermite polynomials by Hermite functions in the formula for Hermite coefficients:

$$c_n \approx \frac{2^{N-1} N! \sqrt{\pi}}{\alpha_n N^2} \sum_{m=1}^N \frac{\alpha_n \psi_n(x_m) e^{x_m^2/2} f(x_m) e^{x_m^2/2}}{\left(\alpha_{N-1} \psi_{N-1}(x_m) e^{x_m^2/2} \right)^2}.$$

After its simplification we can receive the following formula:

$$c_n \approx \frac{1}{N} \sum_{m=1}^N \mu_{N-1}^n(x_m) f(x_m),$$

where $\mu_{N-1}^n(x_m)$ is the array of associated constants:

$$\mu_{N-1}^n(x_m) = \psi_n(x_m) / \psi_{N-1}^2(x_m).$$

Error of Gauss-Hermite quadrature can be estimated as [10]:

$$R(f) = \frac{N! \sqrt{\pi}}{2^N (2N)!} \cdot \left. \frac{d^{2N}}{dx^{2N}} \left(f(x) e^{x^2/2} \right) \right|_{x=\eta}.$$

For arrays' coding ($f(x)$ – discrete function) by fast Hermite projection method we used averaged values in mesh points calculated by averaging values in \mathcal{E}_m - vicinity of point x_m instead of $f(x_m)$. A domain of array is also corresponding to the domain of Hermite function's localization $[-\gamma_N, \gamma_N]$, where $\gamma_N = \arg(\max_x(\psi_N(x)))$, $\forall N \geq 2$.

Results of common/fast coding of 512 arrays 512 elements each on processor Pentium-M 1500MHz by described algorithms using C++ Win32 program under Windows XP are shown in the figure 3 (horizontal axis – number of functions, standing axis – time, relative units):

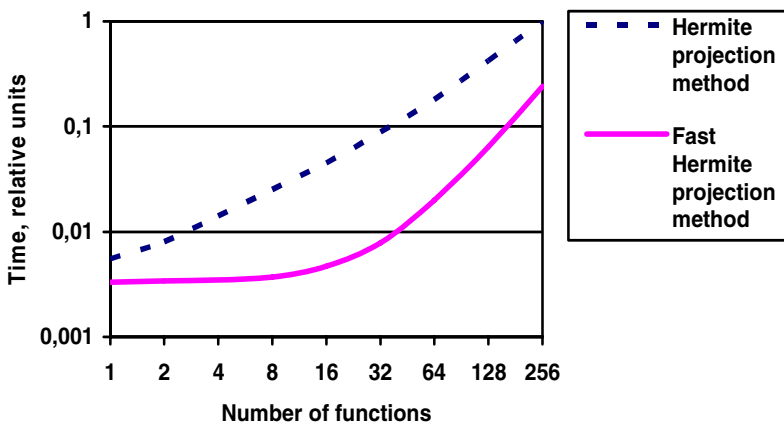


Fig. 3. Chart of coding time from number of functions

4 Applications

The proposed fast Hermite projection method is applicable in signal processing and can be successfully used in the areas where Hermite projection method shows promising results like image filtration and deblocking [4], image foveation [5], waveform data processing and indexing [6], etc.

In the following section we will consider its application for image database retrieval for the task of automatic paintings' identification using digital camera with following limitations: photo type - 1600x1200x24b or higher quality; there must be part of a painting in the center of a photo; there must be only one painting on a photo; a painting must be completely inside on a photo; maximal camera rotation from vertical axis - 26°; a painting must be on monotone absorbent background with color for easy detecting border of a painting; a painting must be rectangular; a painting must have uniform illumination; camera flash must be off.

5 Image Database Retrieval

For image database retrieval two main steps were used: preprocessing algorithm (fig. 4) and database query by fast Hermite coefficients to retrieve record with minimal quadratic discrepancy from the database.

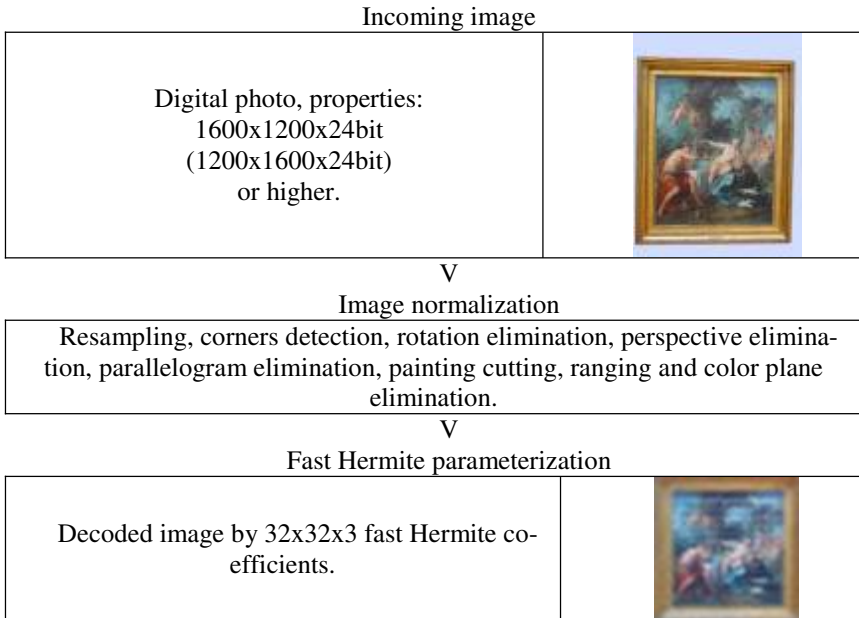


Fig. 4. Preprocessing algorithm

For acquiring statistics were used 4100 different photos for database training and 12 photos for testing purpose. Fast Hermite projection method has been compared with Hermite projection method and discrete Fourier transform (DFT). Discrepancies, each averaged for 12 photos, are given in Table 1.

Table 1. Discrepancy

| Number of functions in the series | Fast Hermite proj. method | Hermite proj. method | DFT |
|-----------------------------------|---------------------------|----------------------|---------|
| 1 | 0,00842 | 0.00859 | 0.00376 |
| 2 | 0,02179 | 0.02193 | 0.00955 |
| 5 | 0,12151 | 0.12583 | 0.05358 |
| 10 | 0,13737 | 0.14170 | 0.06051 |
| 100 | 0,17022 | 0.17400 | 0.07480 |
| 1000 | 0,20081 | 0.20421 | 0.08804 |
| 4100 | 0,38318 | 0.39052 | 0.16732 |

A discrepancy $d_{l,m}$ between coefficients' sets c_l and c_m with average intensities $\eta_{l,\gamma}$ and $\eta_{m,\gamma}$ was calculated by formula (γ numerates color planes in RGB space):

$$d_{l,m} = \sqrt{\frac{1}{3n^2} \sum_{\gamma=0}^2 \sum_{y=0}^n \sum_{x=0}^n \left(\frac{c_m(x, y, \gamma)}{\eta_{m,\gamma}} - \frac{c_l(x, y, \gamma)}{\eta_{l,\gamma}} \right)^2}.$$

All calculations were performed in automatic mode on Pentium-M 1500 MHz under OS MS Windows XP Professional. Searching time was on average 3-7 seconds (server), parameterization with preprocessing time was on average 4-6 seconds (client, 2-3 Mp), but only 0.05 seconds were spent for fast Hermite parameterization.

6 Foveation

Foveation refers to the creation and display of the signal where the resolution varies across the signal. Foveation operator T of a function $f(x)$ can be treated as an integral operator [11]:

$$(Tf)(x) = \int_{-\infty}^{\infty} k(x,t) f(t) dt,$$

where $k(x,t)$ is the kernel of the operator.

The kernel for Hermite foveation has been considered in detail in [5]:

$$k(x, t) = \sum_{i=0}^{\frac{n}{K}-1} \psi_i \left(A_{\frac{n}{K}-1} \frac{2x-w+1}{w} \right) \psi_i \left(A_{\frac{n}{K}-1} \frac{2t-w+1}{w} \right) + \sum_{j=1}^{K-1} \left(\max \left(\min \left(\frac{r}{r-1} \left(1 - \frac{2r^j |\gamma-x|}{w} \right), 1 \right), 0 \right) \cdot \sum_{i=0}^{\frac{n}{K}-1} \psi_i \left(A_{\frac{n}{K}-1} \frac{2x-w+1}{wr^j} \right) \psi_i \left(A_{\frac{n}{K}-1} \frac{2t-w+1}{wr^j} \right) \right),$$

where $K \geq 2$ is a number of foveation steps (lays), $r > 1.0$ is decreasing coefficient for foveation area between lays, n is a number of Hermite functions ($K \leq n \leq 750K$), $\gamma \in [0, w-1]$ is a fovea.

By replacing Hermite projection method by fast Hermite projection method in Hermite foveation we can obtain fast Hermite foveation. The scheme of fast foveation-coding is depicted on fig. 5.

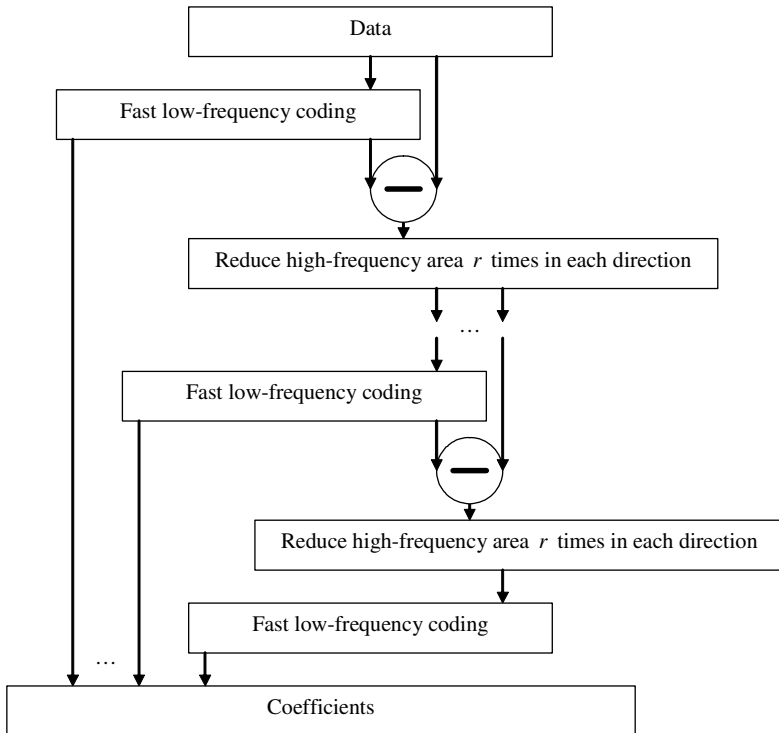


Fig. 5. Scheme of the fast foveation-coding

The acceleration of Hermite projection method by the proposed fast algorithm for 2D image foveation is about 2 times. Practically all the computational time was spent at the foveation decoding steps. Fig. 6 shows the results of fast Hermite foveation of Lena image with the fovea at right Lena eye.



Fig. 6. Fast Hermite foveation with image compression 6 (left) and 12 (right) times

7 Conclusion

In this paper we have designed the algorithm of fast Hermite projection method, which shows 11x acceleration comparing with Hermite projection method for the task of image database retrieval and 2x acceleration for image foveation. Here we used an expansion into series of eigenfunctions of the Fourier transform, which enabled us to use advantages of time-frequency analysis, and accelerated our method by Gauss-Hermite quadrature. The proposed acceleration algorithm can be also efficiently used in Hermite transform method [2].

References

1. Gabor Szego, "Orthogonal Polynomials", American Mathematical Society Colloquium Publications, vol. 23, NY (1959)
2. Jean-Bernard Martens, "The Hermite Transform – Theory", IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 38 (1990) 1595-1606
3. Andrey Krylov and Danil Kortchagine, "Projection filtering in image processing", Proceedings Int. Conference Graphicon 2000, Moscow (2000) 42-45
4. Najafi M., Krylov A., Kortchagine D., "Image deblocking with 2-D Hermite transform", Proceedings Int. Conference Graphicon 2003, Moscow (2003) 180-183
5. Krylov A., Kortchagine D., "Hermite Foveation", Proceedings Int. Conference Graphicon 2004, Moscow (2004) 166-169

6. Krylov A.S., Kortchagine D.N., Lukin A.S., "Streaming Waveform Data Processing by Hermite Expansion for Text-Independent Speaker Indexing from Continuous Speech", Proceedings Int. Conference Graphicon 2002, Nizhny Novgorod (2002) 91-98
7. D. Gabor, "Theory of Communication", J. IEE (London), vol. 93, part III, no. 26 (1946) 429-457
8. J. A. Bloom, T. R. Reed, "An Uncertainty Analysis of Some Real Functions for Image Processing Applications", Proceedings of ICIP 1997 (1997) III-670-67
9. W.F. Eberlein, "A New Method for Numerical Evaluation of the Fourier Transform", Journal of Mathematical Analysis and Application 65 (1978) 80-84
10. V.I. Krylov, "Numerical Integral Evaluation", Moscow: Science (1967) 116-147
11. Ee-Chien Chang, Stephane Mallat and Chee Yap, "Wavelet Foveation", J. Applied and Computational Harmonic Analysis, volume 9, number 3 (2000) 312-335

Posterior Sampling of Scientific Images

Azadeh Mohebi and Paul Fieguth

Department of Systems Design Engineering, University of Waterloo
Waterloo, Ontario, Canada, N2L 3G1

Abstract. Scientific image processing involves a variety of problems including image modelling, reconstruction, and synthesis. We are collaborating on an imaging problem in porous media, studied in-situ in an imaging MRI in which it is imperative to infer aspects of the porous sample at scales unresolved by the MRI. In this paper we develop an MCMC approach to resolution enhancement, where a low-resolution measurement is fused with a statistical model derived from a high-resolution image. Our approach is different from registration/super-resolution methods, in that the high and low resolution images are treated only as being governed by the same spatial statistics, rather than actually representing the same identical sample.

1 Introduction

Scientific imaging plays a significant role in research, especially with the availability of sophisticated imaging tools, including magnetic resonance imaging, scanning electron microscopy, confocal microscopy, computer aided X-ray tomography, and ultrasound, to name only a few. Because of the significant research funding and public interest in medical imaging and remote sensing, these aspects of scientific imaging have seen considerable attention and success.

However there is an enormous variety of imaging problems outside of medicine and remote sensing, where we would argue the current image processing practice to be relatively rudimentary, and where substantial contributions remain to be made. One such area is that of porous media [1] — the science of water-porous materials such as cement, concrete, cartilage, bone, wood, and soil, with corresponding significance in the construction, medical, and environmental industries.

The research in this paper is predicated on the quandary of imaging such porous media. High-resolution two-dimensional images of a porous surface can be produced, however viewing the interior of a sample requires cutting, polishing, and exposure to air, all of which may alter the sample. In-situ three-dimensional images of a medium can be measured using an imaging MRI, however the spatial resolution is very limited, such that only the largest pores are resolved.

Our long-term objective is the fusion of data from multiple imaging modalities to produce high-resolution images of porous media; for example, the fusion

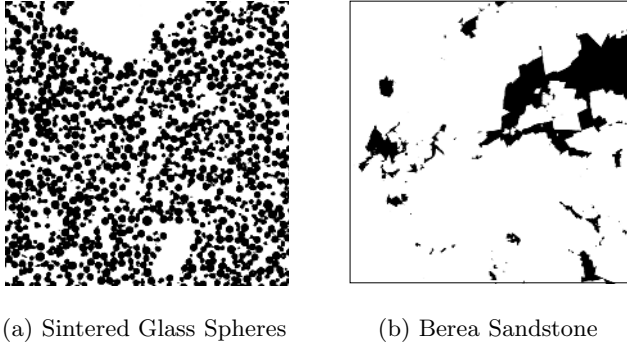


Fig. 1. Two examples of high-resolution 2D slices of irregular porous media

of high-resolution 2D and low-resolution 3D measurements. As it will never be feasible to acquire hundreds or thousands of 2D slices from a single sample, the given 2D measurements will image only an infinitesimal fraction of the 3D sample, so we are *not* interested in a 2D-3D co-registration problem, as is common in medical imaging and remote sensing. Instead, we view the 2D image as *characterizing* the sample, such that we infer a statistical model from the image, and fuse this model with the 3D data set to infer details at a higher resolution.

As an initial exploration of the above long-term research objective, this paper explores the statistical fusion of low and high-resolution data sets, limiting our attention here to two-dimensional random fields for simplicity.

2 Bayesian Image Analysis

Most porous media [1], such as those shown in Fig. 1, possess clear spatial patterns and relationships which characterize the medium. While simple image prior models can be obtained from image statistics such as spatial variances and correlation functions, such models are particularly poor for discrete-state (pore/solid) problems, in which case Gibbs random-field models are widely used [8],[5].

As the generation of a high resolution realization from a low resolution observation is highly ill-posed, some sort of regularization constraint is needed. In a non-Bayesian case (Fig. 2 (I,II)), multiple input images can be combined using methods of super resolution and data-fusion [4]. However, in our scientific imaging application in porous media we do not have the luxury of multiple images, although we may have available high-resolution measurements from statistically-equivalent samples (Fig. 2 (III)). Therefore, we consider a Bayesian approach in which the prior model can be characterized according to a high resolution 2D image and a low resolution 2D image or 3D volume which is the measurements characterizing a sample from which we wish to infer high resolution details.

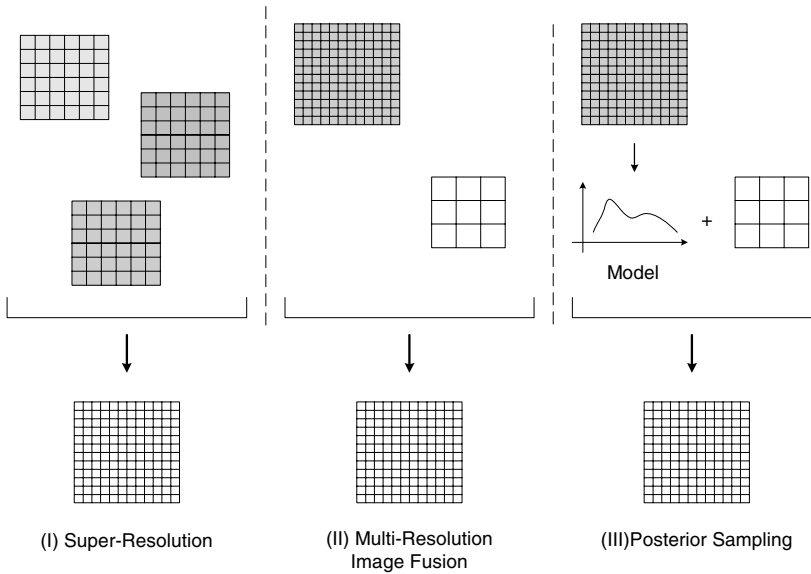


Fig. 2. Different perspectives in data fusion: We may be registering and fusing multiple images (I), fusing data across resolution (II), or statistical fusion through a model (III), as we propose in this paper. Although (II) and (III) appear superficially similar, in (II) the two data sets correspond to the identical underlying image, whereas in (III) the data are only assumed to obey the same statistics.

3 Posterior Sampling

There are two common objectives associated with Bayesian random-field models [9], [10]:

1. To generate random realizations consistent with the prior statistics of the model, referred to as random synthesis or prior sampling.
2. To solve for the optimal image based on measurements and a prior model, known as as estimation.

Prior sampling is not a function of measurements, and is therefore of limited use in settings where we wish to enhance a low-resolution data set, whereas estimation produce only those image features or structures which are inferable from the measurements, and therefore of limited utility in porous media which exhibit behavior over a wide range of scales. Instead, we propose to do posterior sampling (Fig. 3) — the synthesis of random images simultaneously obeying both the measurements and the prior models, producing results similar to estimates in densely-measured area, and producing a random synthesis in those area not constrained by measured values, thereby creating a high resolution result containing structure on a variety of scales.

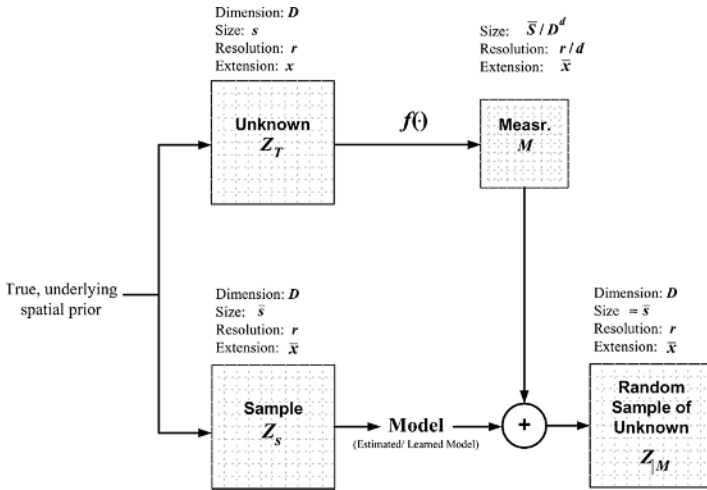


Fig. 3. Posterior Sampling. The sample image Z_s is used for learning/estimating the prior model. An irregular and incomplete measurement from the unknown image Z_T is available as M which is considered in the prior model. The posterior sample, unknown image and the sample image obey the same statistics.

3.1 Problem Formulation

To formulate the problem, we have used Gibbs Random Fields (GRFs) theory. GRFs are lattice models, used to quantify the spatial interactions of observed values at the nodes of a grid and to compute a probability for any configuration of that grid [12] [3]. GRFs were originally used in statistical physics to study the thermodynamic characteristics of interacting neighboring particles in a system [12]. Any Gibbs probability distribution takes the form

$$p(Z) = \frac{e^{-\beta H(Z)}}{\mathcal{Z}} \tag{1}$$

where $H(\cdot)$ is an energy function:

$$H(Z) = \sum_{c \in \mathcal{C}} V_c(Z) \tag{2}$$

written as a function of interactions over a local clique \mathcal{C} , where \mathcal{Z} is a normalization factor, the *partition function*, and $\beta = 1/T$ is related to the temperature T . As the joint prior probability $p(Z)$ is strictly a function of H , the energy H implicitly encodes all of the characteristics of the random field. In order to solve the posterior sampling problem we will need to sample from the posterior probability [10]:

$$p(Z|M) = \frac{e^{-\beta H(Z|M)}}{\mathcal{Z}} \tag{3}$$

where

$$H(Z|M) = H(Z) + \alpha \|M - f(Z)\| \tag{4}$$

for some norm $\|\cdot\|$, where $f(\cdot)$ is the forward model, describing how measurements are derived from unknowns,

$$M = f(Z). \tag{5}$$

Two questions remain: the selection of H , and the balancing of α and β in producing a random sample. The selection of H entails identifying the prior model and the measurement model, which are $H(Z)$ and $\|M - f(Z)\|$ respectively, shown in Eq. (4).

3.2 Prior Models

We consider two types of prior model:

1. *Ising model* [12]: Each site can take two possible values and a first-order neighborhood structure is considered for each site, i.e. for every site $z_{i,j}$ in the field, sites $z_{i,j+1}$, $z_{i,j-1}$, $z_{i+1,j}$, $z_{i-1,j}$ are defined as its neighbors. The potential function for a this neighborhood is defined as:

$$H(Z) = \sum_{i,j} -z_{i,j} (z_{i,j+1} + z_{i,j-1} + z_{i+1,j} + z_{i-1,j}) \tag{6}$$

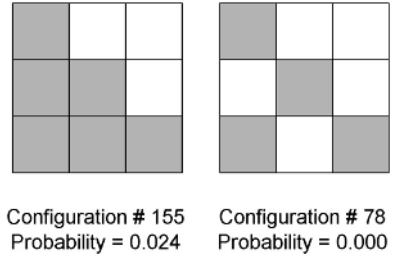
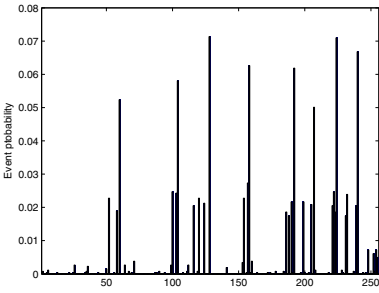
We do not for a moment consider the Ising model a meaningful representation of porous media. We include it only because it is simple, widely understood, and therefore a convenient point of comparison.

2. *Histogram model* [2]: This model is non-parametric, keeping the entire joint probability distribution of local set of pixels within a neighborhood. We have chosen the neighborhood of a pixel to be the eight adjacent pixels, leading to a non-parametric model containing of a histogram of $2^9 = 512$ probabilities. For reasons of convenience we compute two histograms, one for configurations with a black pixel at the center and another for white. Fig. 4 shows such histogram, learned from the image in Fig. 1(a).

3.3 Gibbs Sampling Subject to Constraint

It is hardly possible to sample directly from the posterior distribution since the configuration space for an image with N sites has 2^{N^2} elements. When the random field is considered to be continuous, hierarchical and multi-scale approaches [7] can be applied. However, for the discrete case Monte Carlo Markov Chain (MCMC) methods [12] are most fitting.

The well known Gibbs sampler [10], based on an MCMC approach, generates random samples from the Gibbs distribution by producing a Markov chain whose elements are a sequence Z_1, Z_2, \dots , such that Z_{i-1} and Z_i can differ in at most one pixel [11].



(a) Probability of 256 possible configurations which have black pixel at the center

(b) Examples of two configurations and their probability

Fig. 4. An example of probability distribution of all possible configurations with black pixels at the center (a), with examples of two configurations and their probability (b). The random field is modeled non-parametrically via the probability of every configuration of the eight binary pixels surrounding a central pixel.

In principle, the posterior sampling appears straightforward: specify the energy function and run the Gibbs sampler. In practice the problem is not at all straightforward.

The process of annealing involves generating a sequence of samples, applying the Gibbs sampler while gradually increasing β (decreasing T) in the energy function. This annealing process is started at small β (high temperature), where $p(Z)$ is only a weak function of Z , thus Z is relatively unconstrained, and as β increases the system is driven to lower energy until the minimum energy, the most probable Z maximizing $p(Z)$, is obtained.

But here lies the problem:

1. We are not seeking the most probable realization, rather we want a *random* realization faithful to the prior model and measurements, that is, we wish to draw a random sample from $p(Z)$ at $\beta = \infty$ ($T \neq 0$).
2. However the energy function $H(Z)$, empirically derived from porous media, is really only valid at $\beta = \infty$ ($T = 0$). That is, Z is unlikely to look like a porous medium unless the empirical constraints in $H(Z)$ are rigidly asserted.

Astonishingly, almost all porous media MCMC papers ignore the above distinction and do simulated annealing to generate “random samples” from the prior model, a procedure which succeed because the annealing process fails to find the optimum Z maximizing $p(Z)$, and finds a random, near-optimum Z instead.

Therefore, we are left with three possible approaches:

1. Posterior sampling from the Gibbs distribution, problematic as described above.
2. Maximizing the Gibbs posterior distribution [11], by simulated annealing.

3. Constrained maximization of the posterior distribution [11], also by simulated annealing.

The latter two cases are distinguished by the relative choices of weighting factors α , β in (3), (4). If α is fixed and $\beta \rightarrow \infty$ (the second approach) then we have regular annealing on a fixed model. However if β is fixed, or slowly increasing such that $\beta/\alpha \rightarrow 0$ as $\alpha \rightarrow \infty$ (the third approach), then we are annealing subject to generating samples within the following constrained space [10], [11]

$$\{Z \mid \|M - f(Z)\| = 0\}. \tag{7}$$

That is, this set contains Z which matches the low-resolution measurement perfectly.

4 Results and Evaluation

To evaluate and test the proposed approach, we have applied Gibbs sampling with hard constraint in the form of low-resolution measurement to a portion of two images shown in Fig. 1. The results for those images are shown in Fig. 5, Fig. 6 and Fig. 7 three different samples are generated for the two prior models (Ising and Histogram).

If the original high-resolution image is $n \times n$ and the measurement is $m \times m$, the defined down-sampling parameter is $d = \frac{n}{m}$. For the result shown in Fig. 5, Fig. 6 and Fig. 7, $d = 8$.

The results can be evaluated in terms of the goodness of fit to the prior and measurement models. For the reconstructed image $Z_{|M}$ from true, underlying sample Z_s , we define $J(Z_{|M}|M)$ and $J(Z_{|M}|Z_T)$ to be the goodness of fit to the prior and measurement, respectively

$$J(Z_{|M}|M) = \|M - f(Z_{|M})\|^2 \tag{8}$$

$$J(Z_{|M}|Z_T) = \|Z_T - Z_{|M}\|^2. \tag{9}$$

Satisfying the prior model perfectly implies finding Z such that $H(Z) = 0$. However, a random sample found by suboptimal annealing, will be expected to have $H(Z) > 0$. The actual numerical value of $H(Z)$ is difficult to interpret, therefore we will assess all generated samples on the basis of goodness of fit to the measurement $J(Z|M)$ and the Mean-Squared Error (*MSE*) $J(Z|Z_T)$ to the underlying, true high-resolution ground-truth Z_T .

For the Ising model, parameter β in Eq. (3) is estimated using the method in [6]. As a one-parameter model the Ising model is able to represent only a very limited range of structures, so the reconstruction are relatively poor.

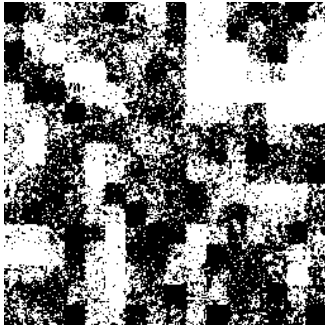
The histogram model, although with a structure nearly as local as Ising, has many more degrees of freedom (512), and is able to give a much clearer, more convincing reconstruction.



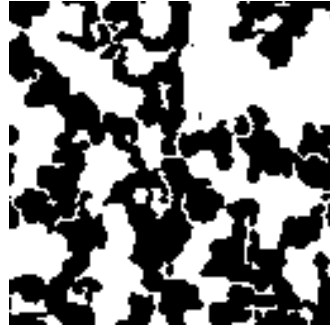
(a) Original image



(b) Measurement



(c) Reconstructed using Ising model



(d) Reconstructed using Histogram model

Fig. 5. reconstructed original image (a), given measurement as (b) and using two different models. The result using Ising and Histogram model is illustrated respectively in (c) and (d). The resolution of original image is 8 times greater than the measurement, i.e., $d=8$.

We can also study the *MSE* variations for different values of the down-sampling parameter d , i.e. as a function of measurement resolution. Fig. 8 shows this variation. Clearly, as expected, $J(Z|Z_T)$ is monotonically increasing with d , however the increase is relatively modest, implying that even for large d the method is producing meaningful estimates. The *MSE* for the proposed method is also compared to the following reconstruction methods:

1. Random reconstruction, which assigns values $\{0, 1\}$ to the sites in the image with probability 0.5,
2. Constrained random reconstruction, which assigns values $\{0, 1\}$ to the sites in the image with probability 0.5 constrained by the measurement.

The correlation between the reconstructed and the original images is more subtle to measure. A pixel in the middle of a large pore or solid is likely to be the same in the original and reconstructed images, as opposed to a pixel on a

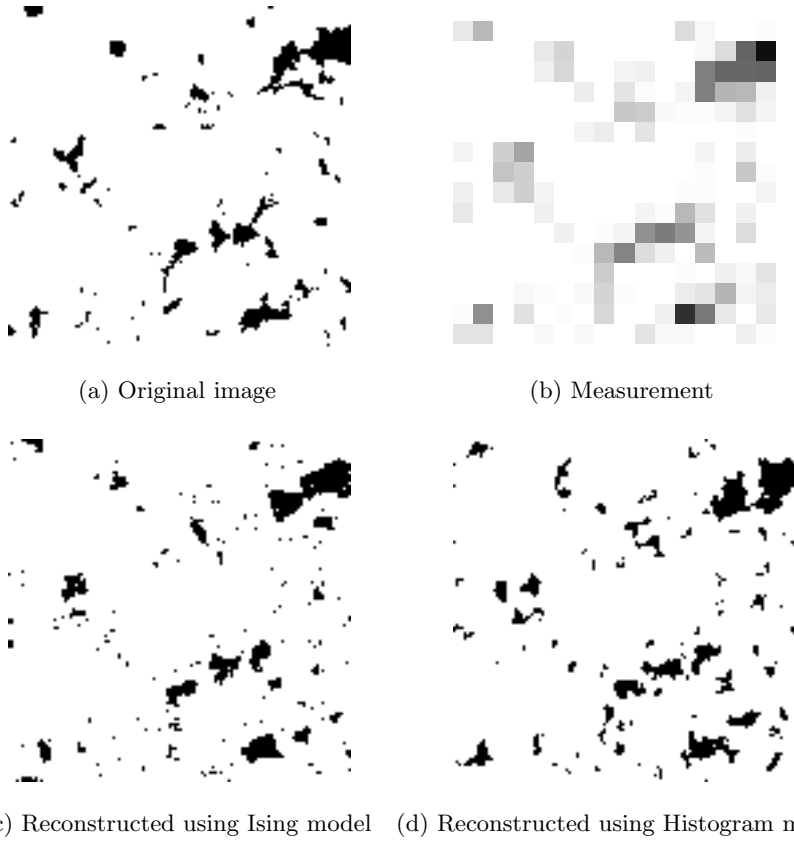


Fig. 6. reconstructed original image in (a) which is a small portion of Berea image, given measurement in (b) and using two different models. The result using Ising and Histogram model is illustrated respectively in (c) and (d). The resolution of original image is 8 times greater than the measurement, i.e. $d=8$.

pore/solid boundary. That is, the original-reconstructed correlation is likely a function of structure size. If we measure structure size at a pixel as the number of times a pixel value is unchanged by sampling, then we can compute correlation as a function of size. The results of Fig. 9 confirm our expectations, and quantify how many scales below the measurement resolution the posterior samples accurately reflect the measured sample.

A final comparison examines pore statistics as a function of pore size. Given an $(n \times n)$ image Z , we define $Z^{(k)}$, $k = 1, \dots, n - 1$, as

$$Z^{(k)}(i, j) = \frac{1}{k^2} \sum_{h=i}^{h+k-1} \sum_{g=j}^{g+k-1} Z(g, h). \tag{10}$$

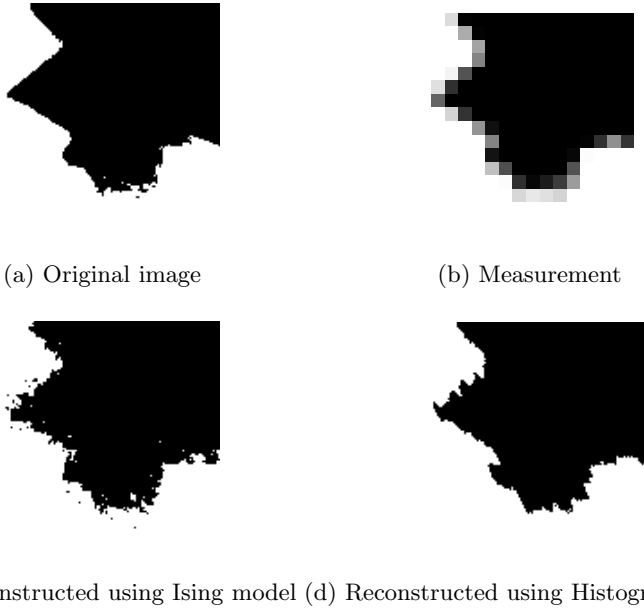


Fig. 7. reconstruction of a portion of Berea image in (a), given measurement in (b) and using two different models. The result using Ising and Histogram model is illustrated respectively in (c) and (d). The resolution of original image is 8 times greater than the measurement, i.e. $d=8$. Although the reconstructed image shown in (d) is more consistent with the original image shown in (c), the prior model still needs to have more contribution in the process to avoid artifacts caused by the measurement.

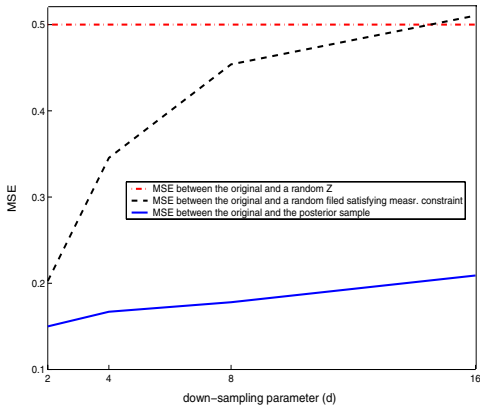


Fig. 8. How the MSE $J(Z|Z_T)$ of the reconstruction process for the image in Fig. 1(a) changes as a function of measurement resolution. $d = n/m$ is the down-sampling parameter for $m \times m$ measurement and $n \times n$ original image. The MSE $J(Z|Z_T)$ of the proposed method is also compared to two other methods: random reconstruction with probability 0.5 and constrained random reconstruction also with probability 0.5 but constrained by the measurement.

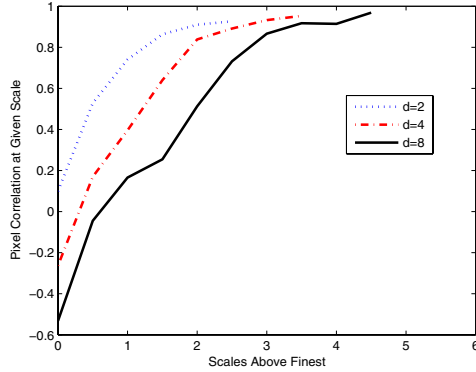


Fig. 9. Correlation between the original and reconstructed image, with the correlation computed as a function of structure scale. As would be expected, the details at the finest scales are only weakly correlated with truth, but much more strongly for larger structures.

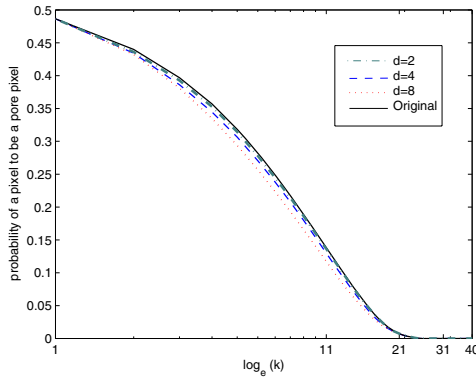


Fig. 10. Pore probability as a function of size k for the image in Fig. 1(a): By down-sampling each image into overlapping blocks of size $k = 2, 3, \dots, n - 1$ and counting the number of pore blocks, we have a sense of pore distribution vs. scale.

In other words, each element of $Z^{(k)}$ is the sum of a $(k \times k)$ block in Z . Then we consider the probability of a pixel to be a pore pixel as the fraction of pore pixels in $Z^{(k)}$. Therefore, for each k the probability of a pixel to be a pore pixel is obtained. Fig. 10 plots the pore probability as a function of k , shown for different values of the down-sampling parameter, d . The comparison shows the effect of measurement resolution on the reconstruction process and the remarkable consistency of the reconstructed image with the pore statistics of the original.

5 Conclusion

In this paper, a model-based approach for image sampling and reconstruction of porous media is introduced. Separate from current approaches such as super-resolution and multi-resolution image fusion, we have considered a statistical model-based approach in which a given low-resolution measurement constrains the model. The approach shows considerable promise on the basis of the preservation of pore statistics and the production of meaningful structures at resolution fewer than given measurements. Comparing the Ising and histogram model in the reconstruction process shows that the later generates samples which are more consistent with the measurement and the prior model. However, a proper prior model needs to have more contribution in the process than the histogram model to avoid artifacts generated by the measurement.

References

1. P.M. Adler. *Porous Media, Geometry and Transports*. Butterworth-Heinemann series in chemical engineering. Butterworth-Heinemann, 1992.
2. S. Alexander and P. Fieguth. Hierarchical annealing for random image synthesis. *Lecture Notes in Computer Science (EMMCVPR 2003)*, (2683), 2003.
3. B. Chalmond. *Modelling and Inverse Problems in Image Analysis*, volume 155 of *Applied mathematical sciences*. Springer-Verlag, 2003.
4. Subhasis Chaudhuri. *Super Resolution Imaging*. Kluwer, 2001.
5. R. Chellappa and A. Jain. *MRF Theory and Applications*, chapter Image Modeling During 1980s: a brief overview. Academic Press, 1993.
6. H. Derin, H. Elliott, and J. Kuang. A new approach to parameter estimation for gibbs random fields. *IEEE*, 1985.
7. P. Fieguth. Hierarchical posterior sampling for images and random fields. *IEEE, ICIP*, 2003.
8. P. Fieguth and J. Zhang. *Handbook of Video and image processing*, chapter Random Field Models. Academic Press, 2000.
9. S. B. Gelfand and S. K. Mitter. *Markov Random Fields, Theory and Applications*, chapter On Sampling Methods and Annealing Algorithms, pages 499–515. Academic Press, 1993.
10. D. Geman. *Random fields and inverse problems in imaging*, volume 1427 of *Lecture Notes in Mathematics*. Springer-Verlag, 1990.
11. S. Geman and D. Geman. Stochastic relaxation, gibbs distribution, and the bayesian restoration of images. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 6(6), 1984.
12. G. Winkler. *Image analysis, Random Fields, and Markov Chain Monte Carlo Methods*. Springer-Verlag, second edition, 2003.

Image Denoising Using the Lyapunov Equation from Non-uniform Samples

João M. Sanches and Jorge S. Marques

Instituto Superior Técnico / Instituto de Sistemas e Robótica
jmrs@isr.ist.utl.pt

Abstract. This paper addresses two problems: an image denoising problem assuming dense observations and an image reconstruction problem from sparse data. It shows that both problems can be solved by the Sylvester/Lyapunov algebraic equation. The Sylvester/Lyapunov equation has been extensively studied in Control Theory and it can be efficiently solved by well known numeric algorithms. This paper proposes the use of these equations in image processing and describes simple and fast algorithms for image denoising and reconstruction.

1 Introduction

Image reconstruction aims to recover images from a partial set of observations, corrupted by noise. Several methods have been proposed to deal with this problem e.g., Bayesian methods [1], wavelets [2,3,4], anisotropic diffusion [5,6], level sets [7,8]. Bayesian methods based on Markov random fields are among the most popular. They require the specification of a prior model and a sensor model which are chosen by the user or learned from the data. The image estimate is then obtained by minimizing an energy function with a very high number of variables or by solving a huge set of linear equations. These operations are very time consuming and they are carried out by suboptimal approaches (e.g., block processing) or by iterative algorithms [1].

This paper proposes an alternative approach based on the Sylvester/Lyapunov (SL) algebraic equation. We consider a denoising problem (all the pixels are observed and corrupted by Gaussian noise) and an image reconstruction problem (only sparse observations are available) and show that the original image can be estimated in both cases using the SL equation, in a very simple way. In the first case (denoising) the estimation requires the solution of a single SL equation. In the second case (sparse data), this equation appears embedded in an iterative scheme which updates its parameters. Convergence is obtained after a small number of iterations. Both procedures are easy to implement since there are efficient algorithms to solve the SL equation (e.g., in Matlab). Experimental tests are presented in this paper showing that the proposed algorithms are efficient and fast and lead to good reconstruction results.

The paper is organized as follows. In section 2 we address the image denoising problem. Section 3 addresses the reconstruction problem from sparse and

non uniform samples. This problem arises for instance in the case of freehand 3D ultrasound or confocal microscopy imaging. Section 4 shows examples using synthetic data and section 5 concludes the paper.

2 Image Reconstruction and Denoising

This section addresses an image denoising problem with the SL equation. Let \mathbf{X} be a $m \times n$ matrix, representing an unknown image and let

$$\mathbf{Y} = \mathbf{X} + \mathbf{\Gamma} \tag{1}$$

be a noisy observation of \mathbf{X} where $\mathbf{\Gamma} \sim N(0, \sigma^2 I)$ is a random error with Gaussian distribution. We wish to estimate \mathbf{X} from the noisy image \mathbf{Y} . The matrix \mathbf{X} is assumed to be a Markov Random Field (MRF) which means that it has a Gibbs distribution, $p(\mathbf{X}) = \frac{1}{Z} e^{-\alpha U(\mathbf{X})}$ where Z is a normalization constant, α is a regularization parameter and $U(\mathbf{X})$ is the *internal energy*. This energy has the form $U(\mathbf{X}) = \sum_{(p,q) \in V} v(x_p - x_q)$ where V is the set of all pairs of neighbors and $v(\tau)$ is a *potential function*.

In this paper we adopt a quadratic potential function, $v(\tau) = \tau^2$ and a 4-neighborhood system. Therefore, the internal energy is

$$U(\mathbf{X}) = \text{tr} [(\theta_v \mathbf{X})^T (\theta_v \mathbf{X}) + (\theta_h \mathbf{X}^T)^T (\theta_h \mathbf{X}^T)] = \text{tr} [\mathbf{X}^T (\theta_v^T \theta_v) \mathbf{X} + \mathbf{X} (\theta_h^T \theta_h) \mathbf{X}^T]$$

where θ_v and θ_h are $n \times n$ and $m \times m$ difference operators. $\theta_v \mathbf{X}$ is a $n \times m$ matrix with all vertical differences of neighboring pixels and $\theta_h \mathbf{X}^T$ is a $m \times n$ -dimensional matrix with all the horizontal differences between neighboring pixels. Both matrices, θ_v and θ_h , have the following structure

$$\theta = \begin{pmatrix} 1 & -1 & 0 & \dots & 0 & 0 & 0 \\ -1 & 1 & 0 & \dots & \dots & \dots & 0 \\ 0 & -1 & 1 & \dots & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & \dots & \dots & -1 & 1 \end{pmatrix}. \tag{2}$$

If we adopt the MAP criterion, the estimate of \mathbf{X} is

$$\hat{\mathbf{X}} = \arg \max_{\mathbf{X}} E(\mathbf{X}) \tag{3}$$

where the energy function $E(\mathbf{X}) = \log [p(\mathbf{Y}|\mathbf{X})p(\mathbf{X})]$ is

$$E(\mathbf{X}) = \text{tr} \left[\frac{1}{2} (\mathbf{X} - \mathbf{Y})^T (\mathbf{X} - \mathbf{Y}) + \alpha \sigma^2 (\mathbf{X}^T (\theta_v^T \theta_v) \mathbf{X} + \mathbf{X} (\theta_h^T \theta_h) \mathbf{X}^T) \right] \tag{4}$$

The solution of (3) is obtained by finding a stationary point of $E(\mathbf{X})$, which obeys the equation

$$\frac{\partial E(\mathbf{X})}{\partial \mathbf{x}_{ij}} = 0 \tag{5}$$

for $i = 1, \dots, n$ and $j = 1, \dots, m$. After straightforward manipulation (5) can be written as

$$\mathbf{X} - \mathbf{Y} + 2\alpha\sigma^2(\mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{B}) = 0 \tag{6}$$

where $A = \theta_v^T \theta_v$ and $B = \theta_h^T \theta_h$ are $n \times n$ and $m \times m$ matrices respectively. This equation can be easily written in the form of a Sylvester equation

$$\mathcal{A}\mathbf{X} + \mathbf{X}\mathcal{B} + \mathbf{Q} = 0 \tag{7}$$

where $\mathcal{A} = I + 2\alpha\sigma^2 A$, $\mathcal{B} = I + 2\alpha\sigma^2 B$ and $\mathbf{Q} = -\mathbf{Y}$ and I is the identity matrix. The reconstructed image can be obtained by solving the Sylvester equation. In the case of square images, $n = m$, $\mathcal{A} = \mathcal{B}$ are square and symmetric matrices and (7) becomes the well known Lyapunov equation.

The Lyapunov equation plays an important role in many branches of control theory namely, in stability analysis, optimal control and stochastic control [9]. There are efficient algorithm to solve the SL equation, some are include in mathematical packages (e.g., in Matlab) [10,11,12]. Therefore, we can easily use one of these algorithms to solve the image denoising problem.

It is important to stress that all the matrices used in (7) have low dimensions ($n \times m$, $n \times n$ or $m \times m$) while a direct implementation of the MAP denoising method involves the computation and inversion of a huge $nm \times nm$ matrix. This is not possible in practice and has to be solved by iterative algorithms [1].

3 Image Reconstruction from Sparse Data

This section considers image reconstruction from sparse observations. It is assumed that the observations are made at non-uniform positions in the image domain (see Fig. 1). We will address this problem assuming a continuous model for the image. Let $f : \Omega \rightarrow \mathbb{R}$, ($\Omega \subset \mathbb{R}^2$) be a continuous image to be estimated. We will assume that f of obtained by interpolating a discrete image $X = \{x_i\}$ using a set of interpolation functions. Therefore, $f(z)$ belongs to finite dimension linear space spanned by the set of interpolation functions: $\{\phi_1(z), \phi_2(z), \dots, \phi_{nm}(z)\}$

$$f(z) = \sum_{i=1}^{nm} x_i \phi_i(z) \tag{8}$$

where x_i are the coefficients to be estimated, associated with each basis function. Herein, we assume that the basis functions, $\phi_i(z)$, are shifted versions of a known function $\phi(z)$ with finite supported, centered at the nodes of $n \times m$ 2D regular grid, i.e., $\phi_i(z) = \phi(z - \mu_i)$, where μ_i is the location of the i -th node (see Fig. 1).

Consider now a L -dimensional column vector of noisy observations, $\mathbf{y} = \{y_i\}$ taken at non-uniform locations $\mathbf{z} = \{z_i\}$ given by $\mathbf{y} = F(\mathbf{z}) + \mathbf{n}$ where $F(\mathbf{z}) = \{f(z_1), f(z_2), \dots, f(z_L)\}^T$ and $\mathbf{n} = \{n_1, n_2, \dots, n_L\}^T$ are L -dimensional column vectors with $n_i \sim \mathcal{N}(0, \sigma^2)$ being a Gaussian random variable. As before, the

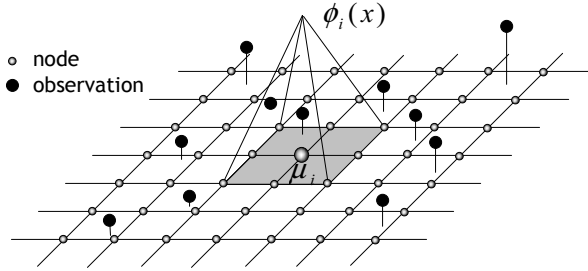


Fig. 1. 2D regular grid with a non-uniform sampling process

MAP estimate of \mathbf{X} is given by the minimization of the energy function (2). Assuming that the observations are conditionally independent, the data fidelity term is

$$E_Y = -\log(p(\mathbf{y}|\mathbf{X})) = \log \left[\prod_{i=1}^L (p(y_i|f(z_i))) \right] = \frac{1}{2} \sum_{i=1}^L (f(z_i) - y_i)^2. \quad (9)$$

and the partial derivative of E_Y with respect to each unknown, x_p , is

$$\begin{aligned} \frac{\partial E_Y}{\partial x_p} &= \sum_{i=1}^L (f(z_i) - y_i) \phi_p(z_i) = \sum_{i=1}^L \left[\sum_{k=1}^{nm} x_k \phi_k(z_i) - y_i \right] \phi_p(z_i) \\ &= \sum_{k=1}^{nm} (x_k h_{pk}) - d_p, \end{aligned} \quad (10)$$

where $f(z)$ was replaced by (8), $h_{pk} = \sum_{i \in V(p,k)} \phi_k(z_i) \phi_p(z_i)$, $d_p = \sum_{i \in V(p)} y_i \phi_p(z_i)$, $V(p)$ is the neighborhood of x_p and $V(p, k)$ is the intersection of $V(p)$ with $V(k)$. This derivative can be efficiently computed since the sum has only 8 terms different from zero, corresponding to the 8 neighbors of pixel p . Furthermore, it can be expressed as a 2D convolution, of x_p with a space varying impulse response, $h_{p,k}$ minus a difference term d_p , $\partial E_Y / \partial x_p = h_{p,k} * x_p - d_p$. This can be written in a compact way as,

$$\frac{\partial E_Y}{\partial \mathbf{X}} = H * \mathbf{X} - D \quad (11)$$

where $*$ denotes the convolution of the discrete image \mathbf{X} with the space varying impulse response H . In practice this amounts to convolving \mathbf{X} with the following space varying mask

$$H(i, j) = \begin{pmatrix} h_{i,j,i-1,j-1} & h_{i,j,i-1,j} & h_{i,j,i-1,j+1} \\ h_{i,j,i,j-1} & h_{i,j,i,j} & h_{i,j,i,j+1} \\ h_{i,j,i+1,j-1} & h_{i,j,i+1,j} & h_{i,j,i+1,j+1} \end{pmatrix} \quad (12)$$

which can be computed at the beginning once we know the sampling points. There are several equal coefficients among these masks due to symmetry.

Approximately only half of the coefficients have to be computed. The derivative of the energy is therefore,

$$\frac{dE}{d\mathbf{X}} = 0 \Rightarrow H * \mathbf{X} - D + 2\alpha\sigma^2 [\Phi_V \mathbf{X} + \mathbf{X} \Phi_H] = 0 \tag{13}$$

This equation can be written as a SL equation,

$$A\mathbf{X} + \mathbf{X}B + Q(\mathbf{X}) = 0 \tag{14}$$

where $A = 0.5I + 2\alpha\sigma^2\Phi_V$, $B = 0.5I + 2\alpha\sigma^2\Phi_H$, $Q = H * \mathbf{X} - \mathbf{X} - D$, $\Phi_V = \theta_V^T \theta_V$ and $\Phi_H = \theta_H^T \theta_H$.

In this case, $Q(\mathbf{X})$ depends on \mathbf{X} which means that the solution can not be obtained in one iteration. Therefore, an iterative algorithm is proposed. In each iteration a new estimate of \mathbf{X} , \mathbf{X}_t is obtained from the previous estimate of \mathbf{X} , \mathbf{X}_{t-1} by solving

$$A\mathbf{X} + \mathbf{X}B + Q(\mathbf{X}_{t-1}) = 0 \tag{15}$$

where $Q = [H(p) - \delta(p)] * \mathbf{X}_{t-1} - D$. Equation (15) is iteratively solved until convergence is achieved. When there is no overlap of basis functions, which is

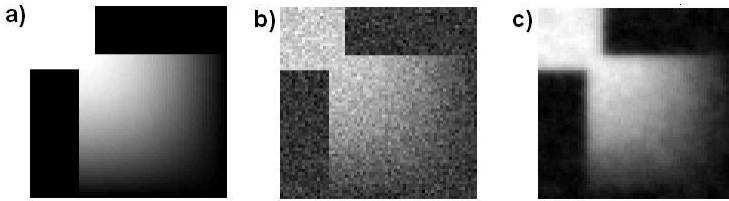


Fig. 2. Image denoising using a synthetic 64×64 pixel image. a)Original, b)noisy with zero mean additive Gaussian noise ($\sigma = 0.25$), c)denoised image. Processing times 530.12 and 5.31 seconds using the VECT and the LYAP methods respectively.

usually the case in image processing (square pixels), all coefficients $\varphi_{kp} = 0$ for $k \neq p$. In this case, the computation of the term $H(p) * \mathbf{X}$ is replaced by a simpler computation of $H \cdot \mathbf{X}$ where “.” denotes the Hadamard product and $H_p = \sum_{i \in V(p)} \phi_p^2(z_i)$. Therefore, in this case,

$$[Q]_p = \sum_{i \in V(p)} [\phi_p^2(z_i)] - 1 - d_p \tag{16}$$

where p is a bi-dimensional index. In image denoising problems with regular grids and without interpolation, there is only one observation per unknown. In this case, $H(p) = \delta_p$, that is, $Q = -Y$. In this particular case the denoising operation can be performed by solving only once the equation $A\mathbf{X} + \mathbf{X}B - Y = 0$. Finally, in the general case, it is possible to ignore the cross terms by making $\varphi_p(x_i)h_k(x_i) \approx 0$. In this case, the computation is strongly simplified with a minor degradation of the results, where Q is computed using (16). The degradation resulting from this simplification depends on the amount and spatial density of the data.

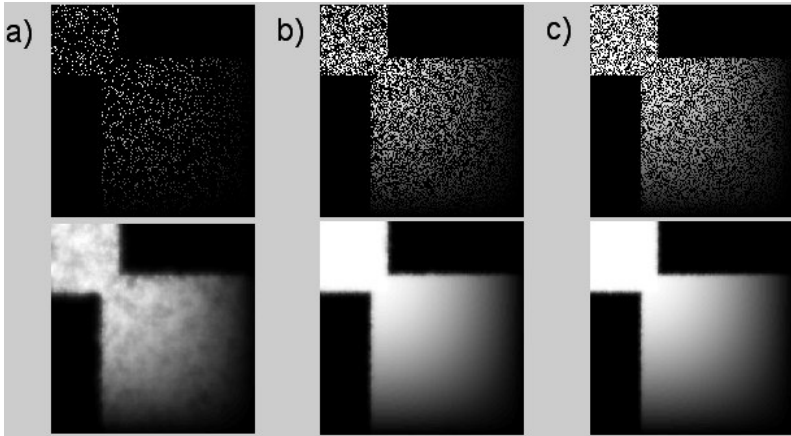


Fig. 3. Image reconstruction from sparse observations. a) $0.1L$ samples (28.34dB), b) $0.5L$ samples (46.36dB) and c) $0.75L$ samples where $L = 256^2$ is the number of pixels of the original 256×256 underlying image. (48.73dB).

4 Experimental Results

In this section we present experimental results obtained with synthetic data in a Pentium 4 PC running at $2.8GHz$. Two methods are used and compared in both methods. The first one, denoted VECT, vectorizes the images involved and reconstructs the images accurately using direct or pseudo matrix inversion. In the second method, denoted LYAP, we use the SL solver, without requiring the matrix inversion.

Fig.2.a) shows the 64×64 noiseless image used in this experiment. It consists of three constant regions and a smooth region where the intensity varies according to a cosine function. Fig.2.b) shows the noisy image, with additive Gaussian white noise with $\sigma = 0.25$ and Fig.2.c) shows the denoised image. The results using both methods are equal but the processing time was 530.12 seconds for the VECT algorithm and 5.31 seconds for the LYAP method.

The noiseless image was used to test the performance of both algorithms with several image sizes and noise energy. Table 1 shows the simulation results for several image dimensions, $N \times N$, and for three values of noise: small ($\sigma = 1e^{-4}$), medium ($\sigma = 0.15$) and severe ($\sigma = 0.5$). Several figures of merit are shown to assess the algorithm performance: the signal to noise ratio (SNR), the minimized energy values, E , and the processing time for each experiment and for both methods. Both algorithms produce the same result, same SNR and E (energy), but the LYAP method clearly outperforms the VECT in terms of processing time when $N > 16$. We note an unexpected jump in the processing time when the image dimensions increase from $N = 48$ to $N = 56$. This results from the fact that, for large enough images, the PC starts to use the HD virtual memory. In addition the algorithm was also tested with non-uniform sampling with the

Table 1. Simulation results for several image dimensions ($N \times N$) and three different noise energy values, $\sigma = \{10^{-4}, 1.5, 0.5\}$

| N | $\sigma = 1e^{-4}$ | | | | $\sigma = 0.15$ | | | | $\sigma = 0.5$ | | | |
|----|--------------------|------|---------------|-------------|-----------------|------|---------------|-------------|----------------|-------|---------------|-------------|
| | SNR (dB) | E | time(s) | | SNR (dB) | E | time(s) | | SNR | E | time(s) | |
| | | | VECT | LYAP | | | VECT | LYAP | | | VECT | LYAP |
| 8 | 12.5 | 2.1 | 0.00 | 0.04 | 12.4 | 2.2 | 0.04 | 0.49 | 5.9 | 4.0 | 0.03 | 0.78 |
| 16 | 22.5 | 5.5 | 0.08 | 0.06 | 22.3 | 5.8 | 0.09 | 0.32 | 17.9 | 10.4 | 0.09 | 0.17 |
| 24 | 28.1 | 8.5 | 0.68 | 0.06 | 27.1 | 9.6 | 0.72 | 0.76 | 23.6 | 20.2 | 0.72 | 0.54 |
| 32 | 30.4 | 11.9 | 3.91 | 0.07 | 29.1 | 13.8 | 3.94 | 0.06 | 23.2 | 33.2 | 3.93 | 0.08 |
| 40 | 33.4 | 14.7 | 14.46 | 0.07 | 31.5 | 17.7 | 14.36 | 0.10 | 23.5 | 48.7 | 14.35 | 0.07 |
| 48 | 35.7 | 17.4 | 43.85 | 0.08 | 33.9 | 21.6 | 43.63 | 0.08 | 24.8 | 65.9 | 43.65 | 0.43 |
| 56 | 36.8 | 20.8 | 133.27 | 4.50 | 34.9 | 26.5 | 120.26 | 2.65 | 25.5 | 82.8 | 120.31 | 2.13 |
| 64 | 38.4 | 23.5 | 516.88 | 7.85 | 36.3 | 31.1 | 381.87 | 4.89 | 25.1 | 110.9 | 432.32 | 5.80 |

LYAP method. Fig.3 shows the reconstruction results obtained with the LYAP method for three different numbers of samples, a) $0.1L$, b) $0.5L$ and c) $0.75L$, where $L = 256^2$ is the number of pixels of the original image. The SNR obtained in these experiments was (28.34dB), (46.36dB) and (48.73dB) for $0.1L$, $0.5L$ and $0.75L$ samples respectively. The iterative LYAP method (see equation (15)) spent 24.3 seconds to generate the 256×256 reconstructed images. The method was also applied to real images with good results as well.

5 Conclusions

This paper shows that image reconstruction and denoising with Gaussian noise can be performed using the SL equation. This equation can be efficiently solved by numeric methods leading to fast reconstruction and denoising algorithm. This method avoids the use of huge $nm \times nm$ matrices and their vectorization and inversion.

Experimental results comparing the proposed algorithm with a direct implementation of image denoising and reconstruction show that important computational time savings are achieved. A comparizon with other state of the art reconstruction algorithms in terms of computational efficiency will be presented in a future work.

References

1. M. R. Banham, A. K. Katsaggelos, Digital Image Restoration, IEEE Signal Processing Magazine, Vol. 14, n 2, March 1997.
2. D. Wei, A.C. Bovik, Wavelet denoising for image enhancement, in The Handbook of Image and Video Processing, A.C. Bovik, (Ed.), Academic Press, 2000.
3. M. Figueiredo, R. Novak, An EM Algorithm for Wavelet-Based Image Restoration, IEEE Trans. Image Processing, Vol. 12, 906-916, 2003.

4. Chang, G., Yu, B., and Vetterli, M. Adaptive wavelet thresholding for image denoising and compression, *IEEE Trans. Image Processing*, 9 (2000), 1532-1546.
5. M. J. Black, G. Sapiro, D. H. Marimont, D. Heeger, Robust anisotropic diffusion, *IEEE Trans. Image Processing*, Vol. 7, 421-432, 1998.
6. C. Riddell, H. Benali, I. Buvat, Diffusion Regularization for Iterative Reconstruction in Emission Tomography, *IEEE Trans. Nuclear Science*, VOL. 52, 669-675, 2005
7. J. C. Ye, Y. Bresler, P. Moulin, A Self-Referencing Level-Set Method for Image Reconstruction from Sparse Fourier Samples, *Int. Journal of Computer Vision*, Vol. 50, 253-270, 2002.
8. T. Deschamps, R. Malladi, I. Ravve, Fast Evolution of Image Manifolds and Application to Filtering and Segmentation in 3D Medical Images, *IEEE Trans. Visualization and Computer Graphics*, Vol. 10, 525-535, 2004
9. Brogan, William L. *Modern Control Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice Hall, 1991.
10. Bartels, R.H. and Stewart, G.W., Solution of the matrix equation $A X + X B = C$, *Comm. A.C.M.*, 15, pp. 820-826, 1972.
11. Barraud, A.Y., A numerical algorithm to solve $A X A - X = Q$., *IEEE Trans. Auto. Contr.*, AC-22, pp. 883-885, 1977.
12. D. Calvetti and L. Reichel, Application of ADI Iterative Methods to the Restoration of Noisy Images, *SIAM J. Matrix Anal. Appl.*, vol.17,no. 1, 1996.

Appendix - Basic algorithm

An exact and non-iterative optimization algorithm may be derived by vectorizing the matrices involved, that is, making $\mathbf{x} = \text{vect}(\mathbf{X})$. The corresponding energy function is $E(\mathbf{x}) = \frac{1}{2}(F(\mathbf{z}) - \mathbf{y})^T(F(\mathbf{z}) - \mathbf{y}) + \alpha\sigma^2((\Delta_V\mathbf{x})^T(\Delta_V\mathbf{x}) + (\Delta_H\mathbf{x})^T(\Delta_H\mathbf{x}))$ where Δ_V and Δ_H are $NM \times NM$ difference matrices, $F(\mathbf{z})$ and \mathbf{y} are L -dimensional column vectors, \mathbf{x} is a NM -dimensional column vector and $F(\mathbf{z}) = \Psi(\mathbf{z})\mathbf{x}$ where Ψ is the following $L \times NM$ matrix

$$\Psi = \begin{pmatrix} \phi_1(z_1) & \dots & \phi_{nm}(z_1) \\ \dots & \dots & \dots \\ \phi_1(z_L) & \dots & \phi_{nm}(z_L) \end{pmatrix} \quad (17)$$

The minimizer of $E(\mathbf{x})$ is $\hat{\mathbf{x}} = (\Psi^T\Psi + 2\alpha\sigma^2(\Delta_V^T\Delta_V + \Delta_H^T\Delta_H))^{-1}\Psi^T\mathbf{y}$. This computation of $\hat{\mathbf{x}}$ is difficult in practice because of the huge dimensions of the matrices involved.

New Method for Fast Detection and Removal of Impulsive Noise Using Fuzzy Metrics

Joan-Gerard Camarena¹, Valentín Gregori^{1,*},
Samuel Morillas^{2,**}, and Guillermo Peris-Fajarnés²

¹ Universidad Politécnica de Valencia, E.P.S. de Gandía, Departamento de Matemática Aplicada, Carretera Nazaret-Oliva s/n, 46730 Grao de Gandia (Valencia), Spain

² Universidad Politécnica de Valencia, E.P.S. de Gandía, Departamento de Expresión Gráfica en la Ingeniería, Carretera Nazaret-Oliva s/n, 46730 Grao de Gandia (Valencia), Spain

Abstract. A novel approach to impulsive noise detection in color images is introduced. The neighborhood of a central pixel using a fuzzy metric is considered for the fast detection of noisy pixels using a peer group concept. Then, a filter based on a switching scheme between the Arithmetic Mean Filter (AMF) and the identity operation is proposed. The proposed filter reaches a very good balance between noise suppression and detail-preserving outperforming significantly the classical vector filters. The presented approach is faster than recently introduced switching filters based on similar concepts showing a competitive performance.

1 Introduction

Several vector filters for color images taking advantage of both the existing correlation amongst the color image channels and the theory of robust statistics have been proposed to date. Well-known vector filters of this family are the *vector median filter* (VMF), [2], the *basic vector directional filter* (BVDF), [25], or the *distance directional filter* (DDF), [9]¹.

The classical vector filters mentioned above present a robust performance but they often tend to blur image edges and details. In order to approach this drawback the switching schemes have been used to their appropriate performance and computational simplicity. The switching approaches aims at selecting a set of pixels of the image to be filtered leaving the rest of the pixels unchanged. A series of methods for selecting the noise-likely pixels based on local statistical information or neighborhood tests have been proposed to date [1,3,11,13,14,15,16,18,21,22,23,24].

A *peer group* concept using classical metrics is employed in [23,24] to detect impulsive noisy pixels. In this paper, we consider the mentioned notion of *peer group* using a certain fuzzy metric since it has provided better results than classical metrics in

* The author acknowledge the support of Spanish Ministry of Science and Technology "Plan Nacional I+D+I", and FEDER, under Grant BFM2003-02302.

** The author acknowledges the support of Spanish Ministry of Education and Science under program "Becas de Formación de Profesorado Universitario FPU".

¹ Extensive information of these vector filters and their properties can be found in the recent overview made in [12].

impulsive noise filtering [17,18]. The proposed *fuzzy peer group* concept is employed to define a switching filter on the basis of the above mentioned works. This filter is faster than those in [23,24] presenting a competitive performance respect to recently introduced switching filters, outperforming them in some cases. Experimental results for performance comparison are provided to show that the proposed approach outperforms classical vector filters, as well.

The paper is organized as follows. In Section 2 the fuzzy metric used in this paper is introduced. Section 3 details the proposed filtering algorithm. Experimental study and some comparisons in front of recent and well-known vector filters are shown in Section 4. Finally, Section 5 presents the conclusions.

2 An Appropriate Fuzzy Metric

According to [5,8] a stationary fuzzy metric on a non-empty set X is a fuzzy set M on $X \times X$ satisfying the following conditions for all $x, y, z \in X$:

- (FM1) $M(x, y) > 0$
- (FM2) $M(x, y) = 1$ if and only if $x = y$
- (FM3) $M(x, y) = M(y, x)$
- (FM4) $M(x, z) \geq M(x, y) * M(y, z)$

where $*$ is a continuous t -norm.

$M(x, y)$ represents the degree of nearness of x and y and according to (FM2) $M(x, y)$ is close to 0 when x is far from y . It was proved in [6,7], that the above definition is an appropriate concept of fuzzy metric.

The next proposition will be established to be applied in next sections when working with color pixels \mathbf{x}_i that are characterized by terns of values in the set $\{0, 1, \dots, 255\}$. This proposition is a particular case of the one used in [18], inspired in [20], and its proof is omitted.

Proposition 1. *Let X be the set $\{0, 1, \dots, 255\}$. Denote by $(x_i(1), x_i(2), x_i(3))$ an element $\mathbf{x}_i \in X^3$ and let $K > 0$. The function $M : X^3 \times X^3 \rightarrow]0, 1]$ given by*

$$M(\mathbf{x}_i, \mathbf{x}_j) = \prod_{l=1}^3 \frac{\min\{x_i(l), x_j(l)\} + K}{\max\{x_i(l), x_j(l)\} + K} \tag{1}$$

is a fuzzy metric on X^3 , where the t -norm $$ is the usual product in $[0, 1]$.*

In this way, from now on $M(\mathbf{x}_i, \mathbf{x}_j)$ will be the fuzzy distance between the color image vectors \mathbf{x}_i and \mathbf{x}_j . According to [17,18], an appropriate value for K when comparing RGB color vectors is $K = 1024$.

3 Proposed Detection and Removal of Corrupted Pixels

A color RGB image is commonly represented as a multidimensional array $N_1 \times N_2 \times 3$, where every pixel \mathbf{x}_i ($i \in N_1 \times N_2$) is a three component vector of integer values in

Table 1. Filters taken for performance comparison and notation

| Notation | Filter |
|----------|--------------------------------------|
| VMF | Vector Median Filter [2] |
| DDF | Directional Distance Filter [9] |
| BVDF | Basic Vector Directional Filter [25] |
| FIVF | Fast Impulsive Vector Filter [18] |
| PGF | Peer Group Filter [23,24] |
| FMNF | Fuzzy Metric Neighborhood Filter |

the interval $[0,255]$. Now, attending to the concept of nearness (see axiom FM2 in the definition of fuzzy metric given in Section 2), for a central pixel \mathbf{x}_i in a $n \times n$ filtering window W and fixed $d \in]0, 1]$, we denote by $\mathcal{P}(\mathbf{x}_i, d)$ the neighborhood of \mathbf{x}_i contained in W

$$\{\mathbf{x}_j \in W : M(\mathbf{x}_i, \mathbf{x}_j) \geq d\}$$

that is, $\mathcal{P}(\mathbf{x}_i, d)$ is the set of pixels of the filtering window W whose fuzzy distance to \mathbf{x}_i is not less than d . Obviously, $\mathcal{P}(\mathbf{x}_i, d)$ is not empty for each \mathbf{x}_i , since $\mathbf{x}_i \in \mathcal{P}(\mathbf{x}_i, d)$. Now, we denote by $\mathcal{P}(\mathbf{x}_i, m, d)$ a subset of $\mathcal{P}(\mathbf{x}_i, d)$ with $m + 1$ elements, and following [23] it is a *peer group* of \mathbf{x}_i in W , for the fuzzy metric M and for a fixed $d \in]0, 1[$. In particular, if $\mathcal{P}(\mathbf{x}_i, d)$ has $c + 1$ elements then it can be denoted by $\mathcal{P}(\mathbf{x}_i, c, d)$.

The proposed impulsive noise detection and removal algorithm, called from now on *Fuzzy Metric Neighborhood Filter* (FMNF), is a modification of the given in [23] and is based on the *peer group* $\mathcal{P}(\mathbf{x}_i, m, d)$ defined above, which involves the fuzzy metric defined in Section 2, and it performs in two steps as follows.

In the first step the pixels of the image are *diagnosed*. The algorithm takes a $n \times n$ filtering window for a central pixel \mathbf{x}_i and, given $d \in]0, 1]$, it finds the set $\mathcal{P}(\mathbf{x}_i, d)$. Now, for a given $m \in \{0, \dots, n^2 - 1\}$, if we can find a *peer group* $\mathcal{P}(\mathbf{x}_i, m, d)$, then all the pixels in $\mathcal{P}(\mathbf{x}_i, d)$ are declared as *non-corrupted* pixels. Notice that, in this case, using the axiom FM4 in the definition of fuzzy metric in Section 2 we can guarantee that the fuzzy distance between any two pixels of the set $\mathcal{P}(\mathbf{x}_i, d)$ will be greater or equal to d^2 , thus, with an appropriate election of the d value, those pixels in the mentioned set can be declared as *non-corrupted* with quite reliability. This fact reduces drastically the number of calculations almost without decreasing the performance of the detection process as it will be shown in Section 4. The rest of the pixels in the window which do not belong to the set $\mathcal{P}(\mathbf{x}_i, d)$ are declared as *non-diagnosed*. If the found set $\mathcal{P}(\mathbf{x}_i, d)$ for the pixel \mathbf{x}_i contains less than $m + 1$ elements, then \mathbf{x}_i is declared *provisionally corrupted* and every pixel in the window is declared as *non-diagnosed*.

This process is repeated for all the pixels in the image but skipping the pixels already declared as *non-corrupted*. Notice that a *provisionally corrupted* pixel could be diagnosed as *non-corrupted* later when taking another window. Finally, the rest of the *provisionally corrupted* pixels are declared *corrupted*.

In the second step the *corrupted* pixels are substituted performing the filtering operation. Each *corrupted* pixel is replaced by the output of the arithmetic mean filter

(AMF) performing (mimicking [23]) over the *non-corrupted* neighbors of the pixel in substitution in a $n \times n$ window. Notice that a *corrupted* pixel may not have any *non-corrupted* neighbor in its $n \times n$ neighborhood. If it is the case, the window must be enlarged in size until at least one *non-corrupted* neighbor is included. The AMF is applied because of its computational simplicity, however other suitable filters as the VMF could be applied in the above conditions, as well.

4 Experimental Study and Performance Comparison

In this section, several images have been considered to evaluate the performance of the proposed filter FMNF explained in Section 3. The impulsive noise model for the transmission noise, as defined in [19], has been used to add noise to the images of Aerial1, Baboon and Lenna (see Fig. 1).

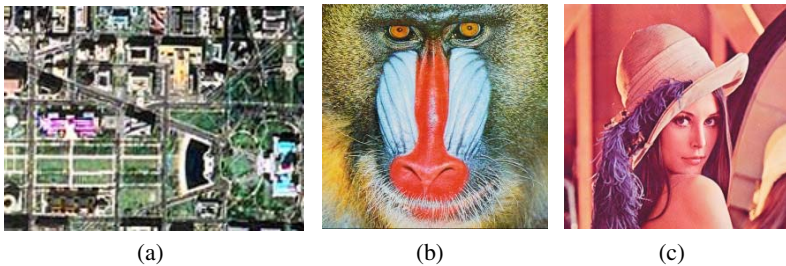


Fig. 1. Test Images: (a) Aerial1 image, (b) Baboon image, (c) Lenna image

The *Mean Absolute Error* (MAE), *Peak Signal to Noise Ratio* (PSNR) and *Normalized Color Difference* (NCD) have been used to assess the performance of the proposed filter. According to [19], these quality measures are defined as

$$MAE = \frac{\sum_{i=1}^N \sum_{j=1}^M \sum_{q=1}^Q |F^q(i, j) - \hat{F}^q(i, j)|}{N \cdot M \cdot Q} \quad (2)$$

and

$$PSNR = 20 \log \left(\frac{255}{\sqrt{\frac{1}{NMQ} \sum_{i=1}^N \sum_{j=1}^M \sum_{q=1}^Q (F^q(i, j) - \hat{F}^q(i, j))^2}} \right) \quad (3)$$

Table 2. Performance comparison for 150×200 *Aerial1* image

| p | | MAE | PSNR | NCD (10^{-2}) | Computation Time (sec) | Time reduction FMNF vs. best PGF |
|------|---------------------------------|--------|--------|----------------------|---------------------------|-------------------------------------|
| 0.05 | VMF | 17.271 | 19.530 | 9.204 | 70.97 | 40.4% |
| | DDF | 17.059 | 19.303 | 8.644 | 134.16 | |
| | BVDF | 21.842 | 16.884 | 10.401 | 78.03 | |
| | FIVF | 3.033 | 23.698 | 2.565 | 37.41 | |
| | PGF _{$m=2$} | 7.507 | 21.237 | 3.334 | 4.75 | |
| | PGF _{$m=3$} | 15.404 | 17.696 | 6.840 | 5.99 | |
| | FMNF | 3.343 | 23.852 | 2.475 | 2.83 | |
| 0.1 | VMF | 17.508 | 19.422 | 9.357 | 70.38 | 37.2% |
| | DDF | 17.441 | 19.209 | 8.845 | 134.09 | |
| | BVDF | 22.739 | 16.612 | 10.770 | 77.53 | |
| | FIVF | 5.243 | 21.898 | 3.936 | 37.20 | |
| | PGF _{$m=2$} | 9.198 | 20.392 | 4.249 | 4.95 | |
| | PGF _{$m=3$} | 16.802 | 17.253 | 7.761 | 6.17 | |
| | FMNF | 5.210 | 22.260 | 3.841 | 3.11 | |
| 0.15 | VMF | 17.719 | 19.298 | 9.514 | 70.41 | 33.9% |
| | DDF | 17.892 | 19.040 | 9.052 | 133.56 | |
| | BVDF | 23.324 | 16.400 | 11.010 | 77.61 | |
| | FIVF | 7.605 | 20.597 | 5.373 | 37.61 | |
| | PGF _{$m=2$} | 11.019 | 19.611 | 5.318 | 5.33 | |
| | PGF _{$m=3$} | 18.321 | 16.791 | 8.909 | 6.33 | |
| | FMNF | 7.753 | 20.713 | 5.366 | 3.52 | |
| 0.2 | VMF | 18.031 | 19.158 | 9.742 | 70.39 | 29.7% |
| | DDF | 18.598 | 18.827 | 9.364 | 133.39 | |
| | BVDF | 24.343 | 16.197 | 11.435 | 77.61 | |
| | FIVF | 10.371 | 19.549 | 6.589 | 37.25 | |
| | PGF _{$m=2$} | 13.087 | 18.774 | 6.617 | 5.42 | |
| | PGF _{$m=3$} | 19.608 | 16.448 | 10.252 | 6.61 | |
| | FMNF | 10.374 | 19.393 | 6.957 | 3.81 | |

where M, N are the image dimensions, Q is the number of channels of the image ($Q = 3$ for RGB images), and $F^q(i, j)$ and $\hat{F}^q(i, j)$ denote the q^{th} component of the original image vector and the filtered image, at pixel position (i, j) , respectively, and

$$NCD_{Lab} = \frac{\sum_{i=1}^N \sum_{j=1}^M \Delta E_{Lab}}{\sum_{i=1}^N \sum_{j=1}^M E_{Lab}^*} \tag{4}$$

where $\Delta E_{Lab} = [(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2]^{\frac{1}{2}}$ denotes the perceptual color error and $E_{Lab}^* = [(L^*)^2 + (a^*)^2 + (b^*)^2]^{\frac{1}{2}}$ is the *norm* or *magnitude* of the original image color vector in the $L^*a^*b^*$ color space.

The proposed filter is assessed in front of some classical and recent filters with good detail preserving ability enumerated in Table 1.

Experimental results² are shown in Tables 2-4. In the following, we will denote by p the noise intensity of the image which will be given as a probability of noise appearance. We have chosen in our experiences values $p \in \{0.05, 0.1, 0.15, 0.2\}$. Output detail

² Results obtained in a Pentium IV, 3.4 GHz, 512 MB RAM.

Table 3. Performance comparison for 256×256 Baboon image

| p | | MAE | PSNR | NCD (10^{-2}) | Computation Time (sec) | Time reduction FMNF vs. best PGF |
|------|--------------|--------|--------|----------------------|---------------------------|-------------------------------------|
| 0.05 | VMF | 10.878 | 23.116 | 4.496 | 156.28 | 24.9% |
| | DDF | 10.708 | 23.029 | 4.201 | 315.72 | |
| | BVDF | 11.891 | 22.065 | 4.548 | 171.38 | |
| | FIVF | 1.698 | 27.055 | 1.555 | 81.31 | |
| | PGF $_{m=2}$ | 2.678 | 26.931 | 1.062 | 7.72 | |
| | PGF $_{m=3}$ | 4.874 | 24.483 | 1.857 | 10.06 | |
| | FMNF | 2.140 | 27.328 | 1.180 | 5.80 | |
| 0.1 | VMF | 10.962 | 23.073 | 4.541 | 158.45 | 20.0% |
| | DDF | 10.920 | 22.952 | 4.311 | 316.86 | |
| | BVDF | 12.147 | 21.930 | 4.661 | 173.02 | |
| | FIVF | 2.891 | 25.519 | 2.258 | 81.47 | |
| | PGF $_{m=2}$ | 3.543 | 26.007 | 1.439 | 8.24 | |
| | PGF $_{m=3}$ | 5.714 | 23.962 | 2.220 | 10.94 | |
| | FMNF | 3.562 | 25.656 | 1.763 | 6.59 | |
| 0.15 | VMF | 11.083 | 22.999 | 4.607 | 155.88 | 17.3% |
| | DDF | 11.164 | 22.838 | 4.442 | 318.19 | |
| | BVDF | 12.585 | 21.723 | 4.853 | 173.92 | |
| | FIVF | 4.012 | 24.440 | 2.963 | 81.33 | |
| | PGF $_{m=2}$ | 4.386 | 25.275 | 1.867 | 8.97 | |
| | PGF $_{m=3}$ | 6.570 | 23.486 | 2.661 | 11.53 | |
| | FMNF | 4.882 | 24.589 | 2.331 | 7.42 | |
| 0.2 | VMF | 11.205 | 22.919 | 4.682 | 155.88 | 15.5% |
| | DDF | 11.401 | 22.740 | 4.569 | 328.14 | |
| | BVDF | 12.903 | 21.511 | 5.001 | 171.23 | |
| | FIVF | 5.329 | 23.977 | 3.024 | 81.09 | |
| | PGF $_{m=2}$ | 5.512 | 24.239 | 2.525 | 9.63 | |
| | PGF $_{m=3}$ | 7.530 | 22.952 | 3.229 | 12.20 | |
| | FMNF | 6.306 | 23.553 | 3.032 | 8.14 | |

images of the filters in comparison for Aerial1 with $p = 0.1$ and Baboon and Lenna with $p = 0.2$ are shown in Figures 3-5

We can conclude that the best results for our filter for 3×3 windows are achieved for $m = 3, 4$ and $d \in [0.88, 0.94]$ and, in particular, the value $d = 0.92$ can be taken as a robust setting. It can be seen that the proposed filter outperforms significantly the classical vector filters and its performance is competitive respect to the other considered switching filters. Moreover, as it is shown in Tables 2-4, the proposed filter gives better results for images presenting a lot of fine details and texture, as the Aerial1 or Baboon images, in front of other types of images. Figure 2 shows the performance of the FMNF in function of the d parameter in front of the VMF and the PGF (using the suggested parameter setting $d = 45$, [23]).

The computational improvement of the PGF with respect to the VMF was shown through the study of the computational complexity of the PGF. Now, recall that in [23] the concept of *peer group* (involving classical metrics) is used to diagnose only the central pixel in a sliding window, and this process is made on all the pixels in the image. In our algorithm it is possible to diagnose several pixels as *non-corrupted* when the central pixel is *non-corrupted* and so there are some pixels where the sliding window

Table 4. Performance comparison for 256×256 *Lenna* image

| p | | MAE | PSNR | NCD (10^{-2}) | Computation Time (sec) | Time reduction FMNF vs. best PGF |
|------|--------------|--------|--------|----------------------|---------------------------|-------------------------------------|
| 0.05 | VMF | 2.755 | 32.051 | 1.781 | 155.97 | 9.7% |
| | DDF | 2.764 | 31.808 | 1.663 | 317.50 | |
| | BVDF | 2.985 | 31.382 | 1.720 | 173.34 | |
| | FIVF | 0.414 | 35.685 | 0.482 | 81.38 | |
| | PGF $_{m=2}$ | 0.4037 | 37.886 | 0.297 | 6.11 | |
| | PGF $_{m=3}$ | 0.7386 | 34.384 | 0.445 | 8.02 | |
| | FMNF | 0.699 | 34.483 | 0.471 | 5.52 | |
| 0.1 | VMF | 2.874 | 31.767 | 1.847 | 155.89 | 11.2% |
| | DDF | 2.982 | 31.325 | 1.788 | 321.36 | |
| | BVDF | 3.248 | 30.773 | 1.869 | 174.73 | |
| | FIVF | 0.743 | 34.158 | 0.669 | 81.17 | |
| | PGF $_{m=2}$ | 0.696 | 35.257 | 0.553 | 6.78 | |
| | PGF $_{m=3}$ | 1.072 | 32.857 | 0.705 | 8.74 | |
| | FMNF | 1.067 | 32.580 | 0.786 | 6.02 | |
| 0.15 | VMF | 2.946 | 31.611 | 1.903 | 157.80 | 10.5% |
| | DDF | 3.159 | 31.006 | 1.915 | 316.08 | |
| | BVDF | 3.513 | 30.112 | 2.035 | 172.22 | |
| | FIVF | 1.031 | 32.972 | 0.936 | 81.09 | |
| | PGF $_{m=2}$ | 1.081 | 32.397 | 0.915 | 7.36 | |
| | PGF $_{m=3}$ | 1.442 | 31.639 | 0.982 | 9.45 | |
| | FMNF | 1.458 | 31.343 | 1.084 | 6.59 | |
| 0.2 | VMF | 3.073 | 31.265 | 1.990 | 156.52 | 11.4% |
| | DDF | 3.386 | 30.524 | 2.057 | 314.83 | |
| | BVDF | 3.788 | 29.540 | 2.196 | 172.67 | |
| | FIVF | 1.350 | 31.936 | 1.191 | 81.03 | |
| | PGF $_{m=2}$ | 1.527 | 30.515 | 1.334 | 8.06 | |
| | PGF $_{m=3}$ | 1.896 | 30.188 | 1.345 | 10.28 | |
| | FMNF | 1.941 | 29.965 | 1.456 | 7.14 | |

processing is no longer needed. Therefore, the main advantage of the proposed filter technique is its great computational speed (see Tables 2-4 and Figure 2). In addition

Table 5 shows a comparative of the computational load between PGF $_{m=2}$ (which gives the best results) and FMNF deduced from our experiences with the previous images. In this table, the values e_p and e_p^* denote the mean number of distance computations needed to diagnose a pixel in the PGF $_{m=2}$ and our FMNF, respectively.

Table 5. Comparative of the computational load between PGF $_{m=2}$ and FMNF

| p | e_p | e_p^* |
|------|-------|---------|
| 0.05 | 3.52 | 2.51 |
| 0.10 | 3.82 | 2.72 |
| 0.15 | 4.15 | 2.94 |
| 0.20 | 4.46 | 3.21 |

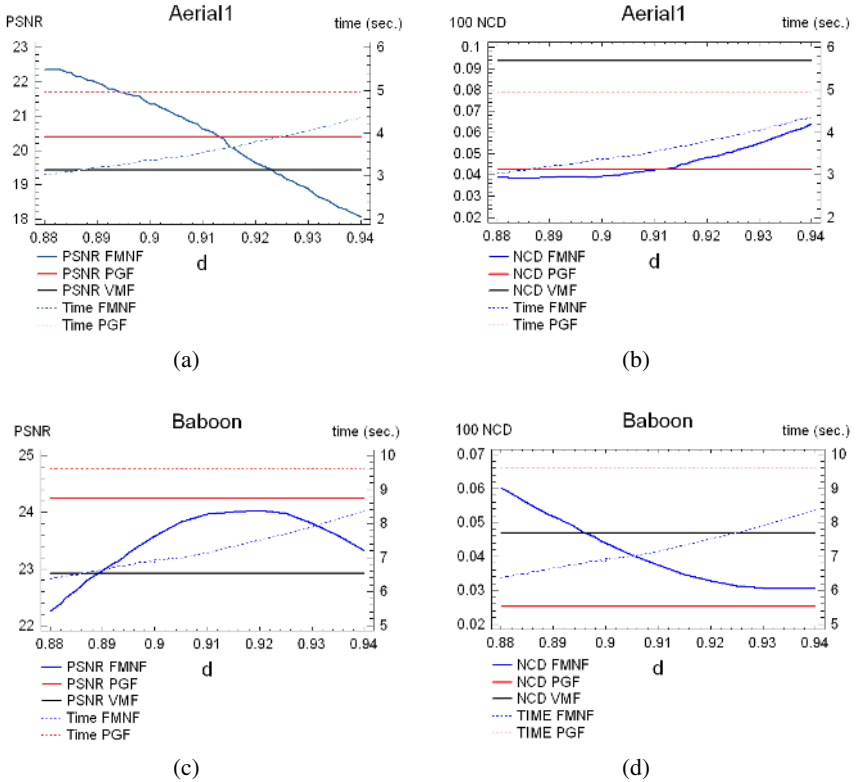


Fig. 2. Comparison of PSNR, NCD and computation time as a function of d in the following cases: (a)-(b) Aerial 1 image contaminated with 10% of impulsive noise and (c)-(d) Baboon image contaminated with 20% of impulsive noise

5 Conclusions

In this paper, we use a peer group concept based on a fuzzy metric to define a novel impulsive noise filter which is a convenient variant of the peer group filter proposed in [23].

The presented approach significantly outperforms classical vector filters and presents a similar performance than recently introduced vector filters reaching an appropriate balance between noise reduction and detail preserving, specially when the image presents a lot of fine details and texture. The computational complexity of the proposed filter is lower than the one of the peer group filter [23], which is based on a similar concept.

Fuzzy metrics suitability for image processing tasks is demonstrated again. Moreover, the use of the proposed fuzzy metric in substitution of classical metrics has supposed a gain in computational efficiency.

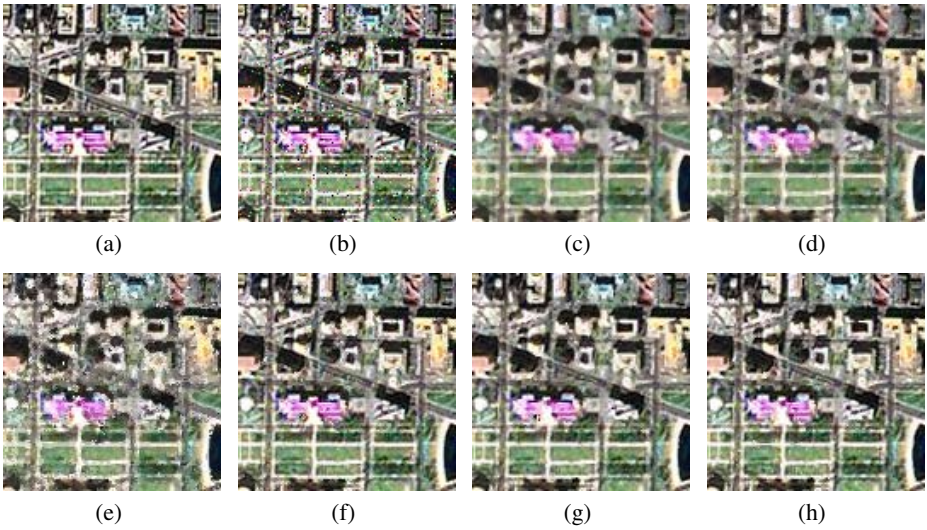


Fig. 3. Output Images: (a) Aerial1 original detail image, (b) Aerial1 with 10% of impulsive noise detail image, (c) VMF filtered detail image, (d) DDF filtered detail image, (e) BVFD filtered detail image, (f) FIVF filtered detail image, (g) $PGF_{m=2}$ filtered detail image, (h) FMNF filtered detail image

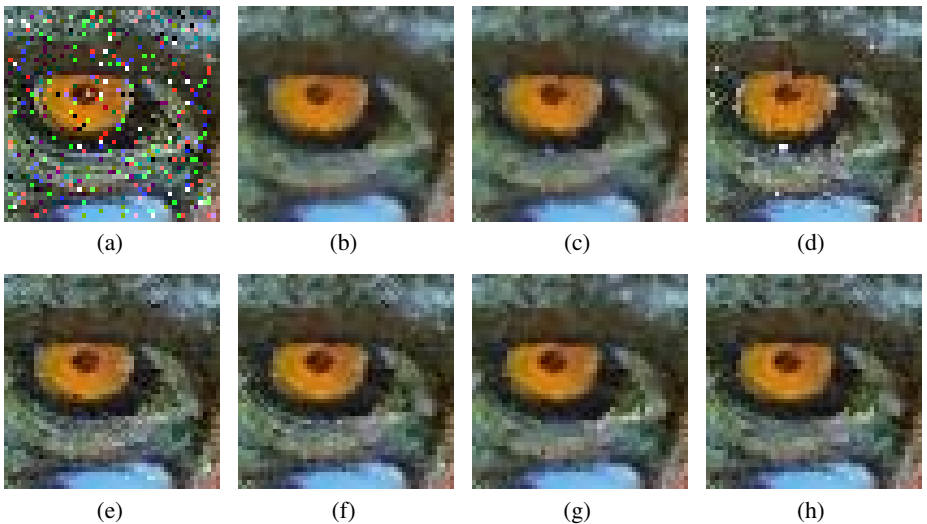


Fig. 4. Output Images: (a) Baboon with 20% of impulsive noise detail image, (b) VMF filtered detail image, (c) DDF filtered detail image, (d) BVFD filtered detail image, (e) FIVF filtered detail image, (f) $PGF_{m=2}$ filtered detail image, (g) $PGF_{m=3}$ filtered detail image, (h) FMNF filtered detail image

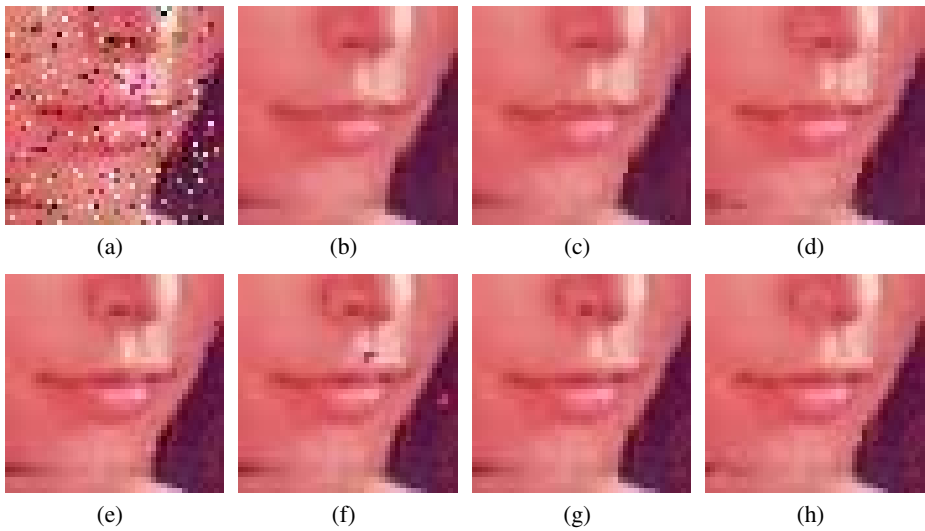


Fig. 5. Output Images: (a) Lenna with 20% of impulsive noise detail image, (b) VMF filtered detail image, (c) DDF filtered detail image, (d) BVFD filtered detail image, (e) FIVF filtered detail image, (f) $PGF_{m=2}$ filtered detail image, (g) $PGF_{m=3}$ filtered detail image, (h) FMNF filtered detail image

References

1. H. Allende, J. Galbiati, "A non-parametric filter for image restoration using cluster analysis", *Pattern Recognition Letters* **25** 8 (2004) 841-847.
2. J. Astola, P. Haavisto, Y. Neuvo, "Vector Median Filters", *Proc. IEEE*. **78** 4 (1990) 678-689.
3. J. Camacho, S. Morillas, P. Latorre, "Efficient impulsive noise suppression based on statistical confidence limits", *Journal of Imaging Science and Technology, to appear*
4. Y. Deng, C. Kenney, MS Moore, BS Manjunath, "Peer group filtering and perceptual color image quantization", *Proceedings of IEEE international symposium on circuits and systems* **4** (1999) 21-4.
5. A. George, P. Veeramani, "On Some results in fuzzy metric spaces", *Fuzzy Sets and Systems* **64** 3 (1994) 395-399.
6. A. George, P. Veeramani, "Some theorems in fuzzy metric spaces", *J. Fuzzy Math.* **3** (1995) 933-940.
7. V. Gregori, S. Romaguera, "Some properties of fuzzy metric spaces", *Fuzzy Sets and Systems* **115** 3 (2000) 477-483.
8. V. Gregori, S. Romaguera, "Characterizing completable fuzzy metric spaces", *Fuzzy Sets and Systems* **144** 3 (2004) 411-420.
9. D.G. Karakos, P.E. Trahanias, "Generalized multichannel image-filtering structure", *IEEE Transactions on Image Processing* **6** 7 (1997) 1038-1045.
10. C. Kenney, Y. Deng, BS Manjunath, G. Hewer, "Peer group image enhancement", *IEEE Transactions on Image Processing* **10** (2) (2001) 326-34.
11. R. Lukac, "Adaptive vector median filtering", *Pattern Recognition Letters* **24** 12 (2003) 1889-1899.

12. R. Lukac, B. Smolka, K. Martin, K.N. Plataniotis, A.N. Venetsanopoulos, "Vector Filtering for Color Imaging", *IEEE Signal Processing Magazine, Special Issue on Color Image Processing* **22 1** (2005) 74-86.
13. R. Lukac, "Adaptive Color Image Filtering Based on Center Weighted Vector Directional Filters", *Multidimensional Systems and Signal Processing* **15** (2004) 169-196.
14. R. Lukac, K.N. Plataniotis, A.N. Venetsanopoulos, B. Smolka, "A Statistically-Switched Adaptive Vector Median Filter", *Journal of Intelligent and Robotic Systems* **42** (2005) 361-391.
15. R. Lukac, B. Smolka, K.N. Plataniotis, A.N. Venetsanopoulos, "Vector sigma filters for noise detection and removal in color images", *Journal of Visual Communication and Image Representation* vol 17. num. 1, pp. 1-26, 2006.
16. Z. Ma, D. Feng, H.R. Wu, "A neighborhood evaluated adaptive vector filter for suppression of impulsive noise in color images", *Real-Time Imaging* **11** (2005) 403-416.
17. S. Morillas, V. Gregori, G. Peris-Fajarnés, P. Latorre, "A new vector median filter based on fuzzy metrics", *ICIAR 2005, Lecture Notes in Computer Science* **3656** (2005) 81-90.
18. S. Morillas, V. Gregori, G. Peris-Fajarnés, P. Latorre, "A fast impulsive noise color image filter using fuzzy metrics", *Real-Time Imaging* **11 5-6** (2005) 417-428.
19. K.N. Plataniotis, A.N. Venetsanopoulos, *Color Image processing and applications* (Springer-Verlag, Berlin, 2000).
20. A. Sapena, "A contribution to the study of fuzzy metric spaces", *Appl. Gen. Topology* **2 1** (2001) 63-76.
21. B. Smolka, R. Lukac, A. Chydzinski, K.N. Plataniotis, W. Wojciechowski, "Fast adaptive similarity based impulsive noise reduction filter", *Real-Time Imaging* **9 4** (2003) 261-276.
22. B. Smolka, K.N. Plataniotis, A. Chydzinski, M. Szczepanski, A.N. Venetsanopoulos, K. Wojciechowski, "Self-adaptive algorithm of impulsive noise reduction in color images", *Pattern Recognition* **35 8** (2002) 1771-1784.
23. B. Smolka, A. Chydzinski, "Fast detection and impulsive noise removal in color images", *Real-Time Imaging* **11** (2005) 389-402.
24. B. Smolka, K.N. Plataniotis, "Ultrafast technique of impulsive noise removal with application to microarray image denoising", *Lecture Notes in Computer Science* **3656** (2005) 990-997.
25. P.E. Trahanias, D. Karakos, A.N. Venetsanopoulos, "Directional processing of color images: theory and experimental results", *IEEE Trans. Image Process.* **5 6** (1996) 868-880.

Background Subtraction Framework Based on Local Spatial Distributions

Pierre-Marc Jodoin¹, Max Mignotte¹, and Janusz Konrad²

¹ D.I.R.O. Université de Montréal

P.O. Box 6128, Stn. Centre-Ville Montréal, Qc, Canada

² Department of Electrical and Computer Engineering Boston University
8 St-Mary's st. Boston, MA 02215

Abstract. Most statistical background subtraction techniques are based on the analysis of temporal color/intensity distributions. However, learning statistics on a series of time frames can be problematic, especially when no frames absent of moving objects are available or when the available memory isn't sufficient to store the series of frames needed for learning. In this paper, we propose a framework that allows common statistical motion detection methods to use spatial statistics gathered on one frame instead of a series of frames as is usually the case. This simple and flexible framework is suitable for various applications including the ones with a mobile background such as when a tree is shaken by wind or when the camera jitters. Three statistical background subtraction methods have been adapted to the proposed framework and tested on different synthetic and real image sequences.

Keywords: Background subtraction, spatial distributions.

1 Introduction

Motion detection methods are generally used to evaluate and locate the presence (or absence) of motion in a given animated scene. To this end, one class of solutions that enjoys tremendous popularity is the family of background subtraction methods [1]. These methods are based on the assumption that the scene is made of a static background in front of which animated objects with different visual characteristics are observed. Typical applications for background subtraction methods include camera surveillance [2], traffic monitoring [3,4] and various commercial applications [5].

As the name suggests, the most intuitive background subtraction method involves one background image and an animated sequence containing moving objects. These moving objects are segmented by simply thresholding the difference between the background and each frame. The threshold can be *a priori* known or estimated on the fly [6]. Unfortunately, such a simple solution is sensitive to background variations and can only meet the requirements of simple applications.

Background variations can be caused by all kinds of phenomena. For instance, noise induced by a cheap low-quality camera or by motion jitter caused by an

unstable camera are typical situations that can't be handled properly by simplistic background subtraction methods. Also, there are many applications for which some background objects aren't perfectly static and induce local false positives. It's the case, for instance, when a tree is shaken by wind or when the background includes animated texture such as wavy water. Another common source of variation is when the global illumination isn't constant in time and alters the appearance of the background. Such variation can be gradual such as when a cloud occludes the sun, or sudden such as when a light is turned on or off.

For all these situations, a more elaborate background subtraction strategy is required. In this perspective, many methods proposed in the literature, model each pixel of the background with a statistical model learned over a series of training frames. For these methods, the detection becomes a simple probability density function (PDF) thresholding procedure. For instance, a single-Gaussian distribution per pixel [7,8] can be used to compensate for uncorrelated noise. However, this single-Gaussian approach is limited by the assumption that the color distribution of each pixel is unimodal in time. Although this assumption is true for some indoor environments, it isn't true for outdoor scenes made up of moving background objects or for sequences shot with an unstable camera. Therefore, because the color distribution of moving background pixels can be multimodal, many authors use a mixture of Gaussians (MoG) [9,3,10] to model the color distribution of each pixel. The number of Gaussians can be automatically adjusted [9] or predefined based on the nature of the application [3]. Non-parametric modeling [2,11] based on a kernel density estimation has also been studied. The main advantage of this approach is the absence of parameters to learn and its ability to adapt to distributions with arbitrary shape. Let us also mention that block-based methods [12], Markovian methods [13] and predictive methods [14,15], to name a few, have been also proposed.

The main limitation with statistical solutions is their need for a series of training frames absent of moving objects. Without these training frames, a non-trivial outlier-detection method has to be implemented [9]. Another limitation with these methods is the amount of memory some require. For example, in [3], every training frame needs to be stored in memory to estimate the MoG parameters. Also, for kernel-based methods [2,11], a number of N frames need to be kept in memory during the entire detection process which, indeed, is costly memory-wise when N is large. In this paper, we propose a novel framework that allows training on only one frame and requires a small amount of memory during runtime. Our framework considers two kinds of illumination variations : a unimodal variation (caused by noise) and a multimodal variation (caused by local movement). The methods we have adapted to our framework are thus robust to noise and background motion.

The rest of the paper is organized as follows. In Section 2, we present our framework before Section 3 explains how statistical background subtraction methods can be adapted to it. Several results are then presented in Section 4 to illustrate how robust our framework is. Section 5 draws conclusions.

2 Proposed Framework

As mentioned earlier, the choice for a pixel-based background model is closely related to the nature of background variations. In this perspective, let us consider two kinds of background variations. The first one concerns variations due to noise. In this case, the background B is considered as being stationary and the color of each pixel s at time t is defined as $B_t(s) = B(s) + n$ where n is an uncorrelated noise factor and $B(s)$ is the *ideal* noise-free background color (or intensity) of site s . In this case, the distribution of $B_t(s)$ in time is considered to be unimodal and centered on $B(s)$. The second kind of variations we consider is the one due to background movement caused by, say, an animated texture or by camera jitter. Considering that variations are due to local movements, it can be assumed that the distribution of $B_t(s)$ in time is similar to the spatial distribution of $B(s)$, *i.e.*, the spatial distribution $\{B(r), \forall r \in \eta_s\}$ where η_s is a $M \times M$ neighborhood centered on s . As a matter of fact, as shown in Fig.1 (a)-(b), when a site s is locally animated, the color observed in time over s often corresponds to the color observed locally around s . Therefore, since the spatial distribution is often multimodal, the distribution of $B_t(s)$ often turns out to be multimodal too.

In this paper, we propose a novel framework which uses only one frame for training. Unfortunately, with only one training frame, it isn't possible to

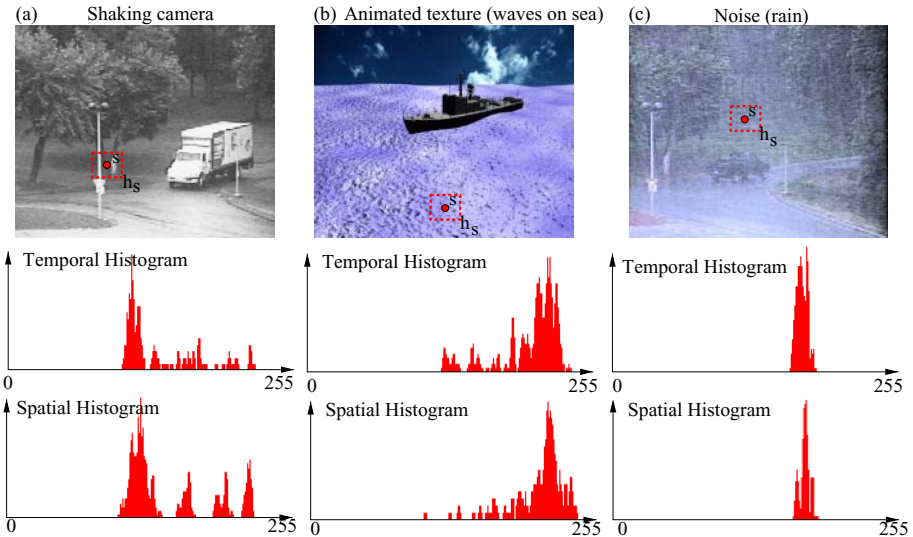


Fig. 1. Three image sequences exhibiting local illumination variations. From left to right: a sequence shot with an unstable camera, a sequence including animated texture and a noisy sequence. The histograms show that the spatial intensity distributions often resemble the temporal distribution.

determine whether the distribution of a site s in time is unimodal or multimodal. One can even think of applications where unimodal regions become multimodal after a while. A good example is when the wind starts to blow and suddenly animates a tree. In this way, since the modality of each pixel distribution isn't known *a priori* and can't be estimated with the analysis of only one background frame, we decided to use a *decision fusion* strategy. To this end, each pixel is modeled with two PDFs: one unimodal PDF (that we call P^u) and one multimodal PDF (called P^m). Both these PDFs are trained on one single background frame " B " (see Section 3 for more details on training). The goal of these two PDFs is to estimate a motion label field L_t which contains the motion status of each site s at time t (typically, $L_T(s) = 0$ when s is motionless and $L_T(s) = 1$ otherwise). The detection criterion can be formulated as follows

$$L_t(s) = \begin{cases} 0 & \text{if } P_s^u(I_t) > \tau \text{ OR } P_s^m(I_t) > \tau \\ 1 & \text{otherwise.} \end{cases} \quad (1)$$

where I_t is a frame observed at time t . Estimating the motion label field L_t with this equation turns out to be the same as blending two label fields L_t^u and L_t^m that would have been obtained after thresholding $P_s^u(I_t)$ and $P_s^m(I_t)$ separately. Other configurations for blending P^u and P^m as well as other thresholding procedures have been investigated during this research. It turned out that a decision criterion such as the one of Eq.1 is a good compromise between simplicity and efficiency.

3 Spatial Training

In this section, we present how P^u and P^m can be trained on data gathered on one background training frame B .

Single Gaussian

As mentioned earlier, P^u models variations due to uncorrelated noise and assumes that the background is perfectly motionless. To this end, P^u is modeled with a single Gaussian distribution

$$P_s^u(I_t) = \frac{1}{(2\pi)^{d/2} |\Sigma_s|^{-1/2}} \exp\left(-\frac{1}{2} (I_t(s) - \mu_s)^T \Sigma_s^{-1} (I_t(s) - \mu_s)\right) \quad (2)$$

where $d = 1$ for grayscale sequences and $d = 3$ for color sequences. Notice that for color sequences, Σ_s is a 3×3 variance-covariance matrix that, as suggested by Stauffer and Grimson [9], is assumed to be diagonal for efficiency purposes. Since only one training frame is available in this framework, μ_s and Σ_s have to be estimated with data gathered around s . Of course, by the very nature of P^u , the spatial data should ideally have an unimodal distribution that resembles the one observed temporally over site s . Although some spatial neighborhoods

of a scene offer that kind of unimodal distribution (see neighborhood A in Fig. 2), others are obviously multimodal (see neighborhood B in Fig. 2) and can't be used as is for training. In fact, using every pixels within a neighborhood η_s would often lead to corrupted (and useless) parameters. Thus, to prevent μ_s and Σ_s from being corrupted by outliers (the gray pixels of the street near B in Fig. 2 for example), a robust function ρ is used to weight the importance of each sample $\mathbf{B}(r)$ [16]. More specifically, the parameter estimation can be expressed as

$$\begin{aligned} \mu_s &= \frac{1}{\sum_{r \in \eta_s} \rho_{s,r}} \sum_{r \in \eta_s} \rho_{s,r} \mathbf{B}(r) \\ \Sigma_s(j) &= \frac{1}{\sum_{r \in \eta_s} \rho_{s,r}} \sum_{r \in \eta_s} \rho_{s,r} (\mathbf{B}(r, j) - \mu_s(j))^2 \quad \forall j \in [1, d] \end{aligned} \quad (3)$$

where $\Sigma_s(j)$ is the variance of the j^{th} color space dimension and η_s is a $M \times M$ neighborhood centered on s . As suggested by Huber [16], we defined $\rho(s, r)$ as

$$\rho(s, r) = \begin{cases} 1 & \text{if } \|\mathbf{B}(s) - \mathbf{B}(r)\|_2 \leq c \\ \frac{c}{\|\mathbf{B}(s) - \mathbf{B}(r)\|_2} & \text{otherwise} \end{cases} \quad (4)$$

where c is a constant that we set to 5. This robust estimator leads to interesting results as shown by the black dotted Gaussian of Fig. 2.

Notice that global lighting variations can be compensated by updating μ_s and Σ_s at every frame [7,8] as follows

$$\mu_s \leftarrow \alpha \mathbf{I}_t(s) + (1 - \alpha) \mu_s, \quad \forall L_t(s) = 0 \quad (5)$$

$$\Sigma_s(j) \leftarrow \alpha (\mathbf{I}_t(s, j) - \mu_s(j))^2 + (1 - \alpha) \Sigma_s(j) \quad \forall L_t(s) = 0, \forall j \in [1, d]. \quad (6)$$

where $\alpha \in [0, 1[$ is the so-called *learning rate* [3].

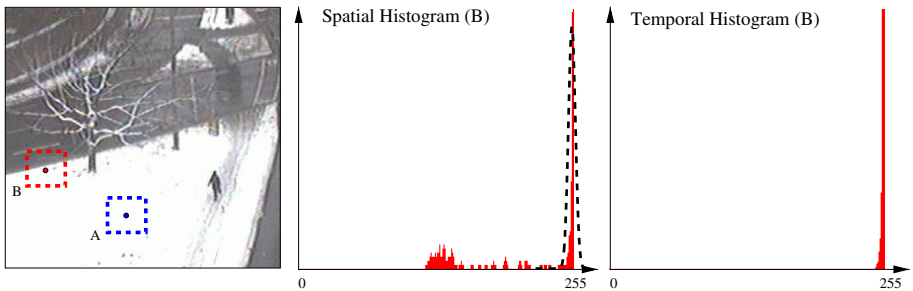


Fig. 2. A sequence shot with a perfectly static camera. While the temporal intensity distribution of pixel B is unimodal and centered on intensity 254, the spatial intensity distribution around B is bimodal. However, estimating the Gaussian parameters with Eq. (3) leads to a distribution (in black) centered on the main mode, uncorrupted by the graylevels of the street.

Mixture of Gaussians

Multimodal histograms such as the ones in Fig. 1 (a)-(b) can't be appropriately modeled with one single Gaussian. However, a mixture of K Gaussians can be a good choice to model such distributions:

$$P_s^m(I_t) = \sum_{i=1}^K w_{s,i} \mathcal{N}(I_t(s), \boldsymbol{\mu}_{s,i}, \Sigma_{s,i}) \quad (7)$$

where $\mathcal{N}(\cdot)$ is a Gaussian similar to the one of Eq.(2), $w_{s,i}$ is the weight of the i^{th} Gaussian and K is the total number of Gaussians (between 2 and 5 typically). In this context, a number of $3 \times K$ parameters per pixel need to be estimated during the training phase. To do so, the well known K -means algorithm has been implemented. The objective of K -means is to iteratively estimate the mean of K clusters by minimizing the total intra-cluster variance. In our framework, K -means takes as input for each pixel s , the $M \times M$ background pixels $\{\mathbf{B}(r), r \in \eta_s\}$ contained within the square neighborhood η_s . When the algorithm converges and the mean $\boldsymbol{\mu}_{s,i}$ of every cluster has been estimated, the variance $\Sigma_{s,i}$ and the weight $w_{s,i}$ are then estimated. In this paper, the number of K -means iterations was set to 6. For further explanations on K -means, please refer to [17].

As we mentioned for P_s^u , the MoG parameters can be updated at every frame to account for illumination variations. As suggested by Stauffer and Grimson [9], at every frame I_t , the parameters of the Gaussian that matches the observation $I_t(s)$ can be updated as follows

$$\boldsymbol{\mu}_{s,i} \leftarrow \alpha I_t(s) + (1 - \alpha) \boldsymbol{\mu}_{s,i}, \quad \forall L_t(s) = 0 \quad (8)$$

$$\Sigma_{s,i}(j) \leftarrow \alpha (I_t(s, j) - \mu_{s,i})^2 + (1 - \alpha) \Sigma_{s,i}(j) \quad \forall L_t(s) = 0 \quad (9)$$

$$w_{s,i} \leftarrow (1 - \alpha) w_{s,i} + \alpha M_{s,i} \quad (10)$$

where $M_{s,i}$ is 1 for the Gaussian that matches and 0 for the other models. The weights $w_{s,i}$ are then normalized.

Nonparametric Density Estimation

Since the color distribution of each pixel can't always be assumed to follow a parametric form, a multimodal density can be estimated with an unstructured approach. One such nonparametric approach is the kernel-based density estimation (also called *Parzen density estimate* [18]) which locally estimates density from a small number of neighboring samples. The kernel method we have implemented has been inspired by the work of Elgammal *et al.* [2].

Considering η_s as being a $M \times M$ neighborhood centered on site s , the density at s is estimated with

$$P_s^m(I_t) = \frac{1}{M \times M} \sum_{r \in \eta_s} K_\sigma(I_t(s) - B(r)) \quad (11)$$

for grayscale sequences and, for color sequences,

$$P_s^m(I_t) = \frac{1}{M \times M} \sum_{r \in \eta_s} \prod_{j=1}^3 K_{\sigma_j}(I_t(s, j) - B(r, j)). \quad (12)$$

where j is the color-space index (red, green or blue) and B is a background frame. Here, K is a *kernel function* –i.e. some PDF– which, in practice, turns out to be normal or uniform [18]. As suggested by Elgammal *et al.* [2], we implemented K_σ as being a zero-mean Gaussian of the form

$$K_\sigma(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-x^2}{2\sigma^2}. \quad (13)$$

Although σ can be estimated on the fly for each pixel [2], we used a constant value. In this way, a single global 256-float-long lookup table can be precalculated to allow significant speedup during runtime. The table values are accessed with the intensity value difference $I_t(s, j) - B(r, j)$ as index.

Let us mention that the background frame B used to model P^m can be updated at every frame to account for lighting variations. This can be easily done with the following operation:

$$\mathbf{B}(s) \leftarrow \alpha \mathbf{B}(s) + (1 - \alpha) \mathbf{I}_t(s). \quad (14)$$

4 Experimental Results

To validate our framework, we have segmented sequences representing different challenges. These tests aim at validating how stable and robust our framework is with respect to traditional methods using a temporal-training approach. For each example presented in this section, a neighborhood η_s of size between 11×11 and 15×15 has been used.

The first sequence we segmented is the one presented in Fig.3 (a) which contains strong noise caused by rain. The segmentation has been performed by thresholding independently a single-Gaussian PDF (see Eq. (2)) trained over each pixel. At first, the Gaussian parameters have been computed with a 20-frame temporal training. Then, the parameters have been trained spatially on one frame using Eq. (3). As shown in Fig. 3, the label fields obtained with both approaches are, to all practical ends, very much similar. They both exhibit some few isolated false positives (due to rain) and some false negatives over the truck window.

The second sequence we have segmented (see Fig. 3 (b)) is one with no training frame absent of moving objects. The background frame B needed to learn the Gaussian parameters has been computed with a simple five-frame median filter : $\mathbf{B}(s) = \text{Med}[I_0(s), I_{40}(s), I_{80}(s), I_{120}(s), I_{160}(s)]$. This median filter led to the middle frame of Fig. 3 (b) which, after a spatial training, gave the result of Fig. 3 (c).

The third sequence we have segmented is the one shown in Fig. 4. This sequence has been shot with a perfectly static camera and contains a highly animated background made of trees shaken by wind. The sequence has been segmented with the MoG and the kernel-based background subtraction method

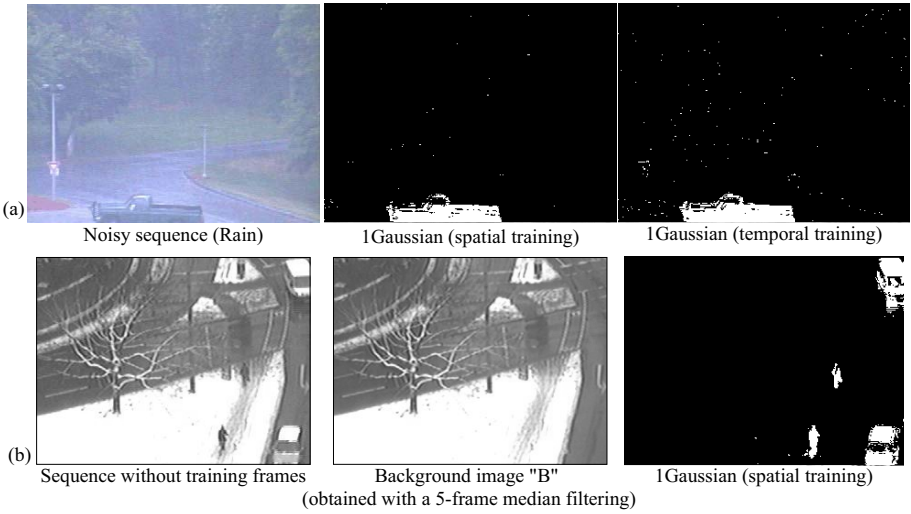


Fig. 3. (a) A noisy sequence segmented with one Gaussian per pixel whose parameters have been spatially and temporally trained. (b) From a sequence without training frames absent of moving objects, a five-frame median filter has been used to produce the background frame *B*.

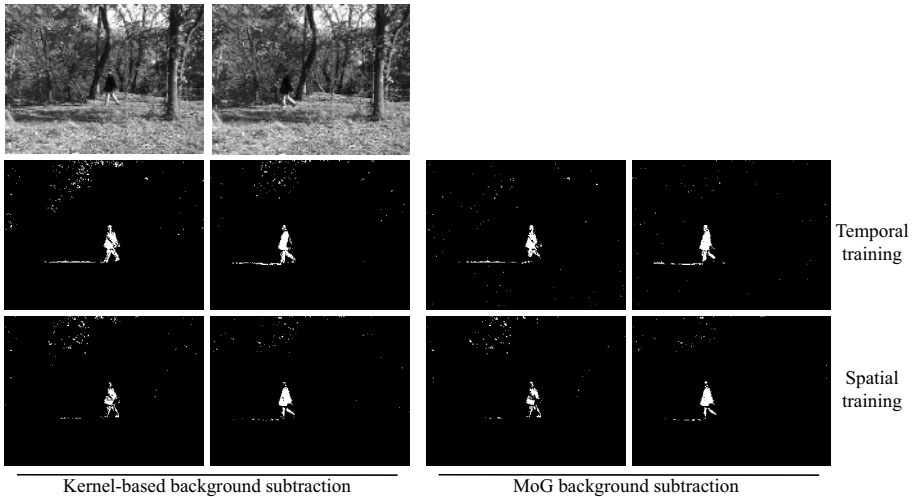


Fig. 4. Two frames of a sequence exhibiting a multimodal background made of trees shaken by wind. The MoG and the kernel-based method have been trained either temporally on a series of frames or spatially, on single frame.

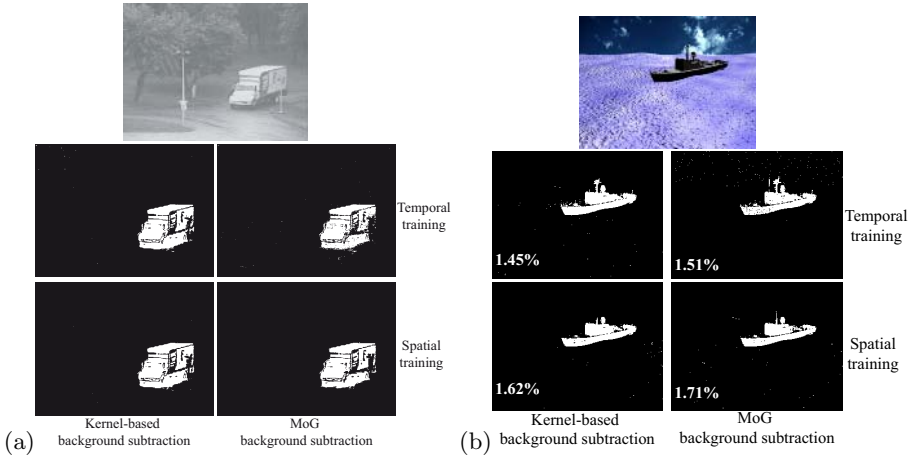


Fig. 5. (a) Sequence shot with an unstable camera and (b) a synthetic sequence made of a boat sailing on a wavy sea. The numbers in the lower left corner of the boat label fields, indicate the average percent of mis-matched pixels. In each case, the MoG and the kernel-based method have been used. Both were trained either temporally on a series of frames or spatially on one frame.

presented in Section 3. Both have been trained temporally on a series of 150 frames and spatially on one single frame. Since the sequence shows a great deal of movement in the background, the parameters of each method have been updated at every frame following Section 3's explanations. In Fig.4, two frames are presented. As can be seen, the spatial results are very much similar to the temporal ones although the latter shows a bit more precision. In both cases, a few isolated false positives due to wind are present at the top, and false negatives due to camouflage can be seen on the walker. Notice that these isolated pixels can be efficiently eliminated with a basic 3×3 or 5×5 median filter.

The third sequence we have segmented is the one presented in Fig. 5 (a) which exhibits a moving truck. The sequence was shot with an unstable camera making the background highly multimodal (notice that for the purposes of this paper, additional camera jitter has been added to the original scene). Again, the MoG and the Kernel-based approach have been used to segment the sequence. Here, 20 frames were used for temporal training. Again, the results obtained after a temporal training are similar to the ones obtained after a spatial training. However, the latter seems a bit more precise, mostly because only 20 frames were available for temporal training. A larger number of training frames would have had certainly a positive influence on the results' sharpness (at the expense of processing time of course).

The fourth sequence shows a boat sailing on a wavy sea. The sequence consists of 200 frames from which the first 80 have been used for temporal training. The sequence has been computer generated and has a ground-truth label field for

each frame. With these ground-truths, the average percentage of mis-matched pixels has been computed to empirically compare the four methods. The average percentage presented in Fig.5 (b) shows, again, how small is the difference between the spatially-trained and the temporally-trained methods.

5 Discussion

In this paper, a novel spatial framework for the background subtraction problem has been presented.

Our framework is based on the idea that, for some applications, the temporal distribution observed over a pixel corresponds to the statistical distribution observed spatially around that same pixel. We adapted three well known statistical methods to our framework and showed how these methods can be trained spatially instead of temporally.

Our framework offers three main advantages. First, the statistical parameters can be learned over one single frame instead of a series of frames as is usually the case for single-Gaussian or the MoG model. This has the advantage of requiring much less memory and being more flexible in presence of sequences having no training frames absent of moving objects. Second, as opposed to the kernel-based method, only one frame (instead of N) is kept in memory during runtime. This, again, is a major advantage memory-wise. Also, it makes our kernel-based methods much easier to be implemented on programmable graphics hardware [19] having a limited amount of memory. Last, but not least, our framework maintains the conceptual simplicity and strength of the background subtraction methods adapted to it. The segmentation function, the adaptation to global illumination variations and the statistical learning phase are implemented in a way that is very similar to the one originally proposed under a temporal framework.

Finally, we have shown that the results obtained with our framework on sequences with high noise, camera jitter and animated background are, to all practical ends, identical to the ones obtained with methods trained temporally.

Acknowledgment

The authors are very grateful to the group of Prof. Dr. H.-H. Nagel for the winter sequence of Fig. 3 (b) and to Dr. Ahmed Elgammal for the truck sequences of Fig. 3 (a) and Fig. 5 (a) and the walker sequence of Fig. 4.

References

1. McIvor A. Background subtraction techniques. In *Proc. of Image and Video Computing*, 2000.
2. Elgammal A., Duraiswami R., Harwood D., and Davis L.S. Background and foreground modeling using nonparametric kernel density for visual surveillance. In *Proc of the IEEE*, volume 90, pages 1151–1163, 2002.

3. Friedman N. and Russell S.J. Image segmentation in video sequences: A probabilistic approach. In *UAI*, pages 175–181, 1997.
4. Cheung S.-c. and Kamath C. Robust techniques for background subtraction in urban traffic video. In *proc of the SPIE, Volume 5308*, pages 881–892, 2004.
5. Zhou D. and Zhang H. 2d shape measurement of multiple moving objects by gmm background modeling and optical flow. In *proc of ICIAR*, pages 789–795, 2005.
6. Rosin P. and Ioannidis E. Evaluation of global image thresholding for change detection. *Pattern Recognition Letters*, 24:2345–2356, 2003.
7. Wren C.R., Azarbayejani A., Darrell T., and Pentland A.P. Pfunder: Real-time tracking of the human body. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):780–785, 1997.
8. Jabri S., Duric Z., Wechsler H., and Rosenfeld A. Detection and location of people in video images using adaptive fusion of color and edge information. In *ICPR*, pages 4627–4631, 2000.
9. Stauffer C. and Grimson E.L. Learning patterns of activity using real-time tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):747–757, 2000.
10. Y. Dedeoglu. Moving object detection, tracking and classification for smart video surveillance. Master’s thesis, Bilkent University, 2004.
11. Mittal A. and Paragios N. Motion-based background subtraction using adaptive kernel density estimation. In *Proc. of CVPR*, pages 302–309, 2004.
12. Ohya T. Matsuyama T. and Habe H. Background subtraction for non-stationary scenes. In *Proc. of 4th Asian Conference on Computer Vision*, pages 662–667, 2000.
13. Sheikh Y. and Shah. Bayesian object detection in dynamic scenes. In *Proc of CVPR*, pages 74–79, 2005.
14. Toyama K., Krumm J., Brumitt B., and Meyers B. Wallflower: Principles and practice of background maintenance. In *ICCV*, pages 255–261, 1999.
15. Zhong J. and Sclaroff S. Segmenting foreground objects from a dynamic textured background via a robust kalman filter. In *Proc of ICCV*, pages 44–50, 2003.
16. Huber P.J. *Robust Statistical Procedures, 2nd Ed.* SIAM, Philadelphia, 1996.
17. Bishop C. *Neural Networks for Pattern Recognition.* Oxford University Press, 1996.
18. Fukunaga K. *Introduction to statistical pattern recognition (2nd ed.)*. Academic Press Professional, Inc., San Diego, CA, USA, 1990.
19. <http://www.gpgpu.org/>.

Morphological Image Interpolation to Magnify Images with Sharp Edges

Valérie De Witte¹, Stefan Schulte¹, Etienne E. Kerre¹
Alessandro Ledda², and Wilfried Philips²

¹ Ghent University

Department of Applied Mathematics and Computer Science
Fuzziness and Uncertainty Modelling Research Unit
Krijgslaan 281 (Building S9), B-9000 Gent, Belgium

² Telin Department - IPI Group

Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium

Valerie.DeWitte@UGent.be

<http://www.fuzzy.ugent.be>

Abstract. In this paper we present an image interpolation method, based on mathematical morphology, to magnify images with sharp edges. Whereas a simple blow up of the image will introduce jagged edges, called ‘jaggies’, our method avoids these jaggies, by first detecting jagged edges in the trivial nearest neighbour interpolated image, making use of the hit-or-miss transformation, so that the edges become smoother. Experiments have shown that our method performs very well for the interpolation of ‘sharp’ images, like logos, cartoons and maps, for binary images and colour images with a restricted number of colours.

1 Introduction

Image interpolation has many applications such as simple spatial magnification of images (e.g. printing low-resolution documents on high-resolution printer devices, digital zoom in digital cameras), geometrical transformation (e.g. rotation), etc. Different image interpolation methods have already been proposed in the literature, a.o. [[1] - [11]]. In this paper we will describe a new morphological image interpolation technique, for binary images as well as for colour images with a limited number of colours, with sharp edges. First we review some basic definitions of mathematical morphology, including the hit-or-miss transformation. In section 3 we introduce our new interpolation approach. We explain the pixel replication or nearest neighbour interpolation, used as the first ‘trivial’ interpolation step in our method. Thereafter we discuss our corner detection method, using different kinds of structuring elements, and describe our corner validation, first for magnification by a factor 2 and then for magnification by an integer factor $n > 2$. Finally, in section 4 we have compared our interpolation method experimentally to other well-known approaches. The results show that our method provides a visual improvement in quality on existing techniques: all jagged effects are removed so that the edges become smooth.

2 Basic Notions

2.1 Modelling of Images

A digital image I is often represented by a two-dimensional array, where (i, j) denotes the position of a pixel $I(i, j)$ of the image I .

Binary images assume two possible pixel values, e.g. 0 and 1, corresponding to black and white respectively. White represents the objects in an image, whereas black represents the background. Mathematically, a 2-dimensional binary image can be represented as a mapping f from a universe X of pixels (usually X is a finite subset of the real plane \mathbb{R}^2 , in practice it will even be a subset of \mathbb{Z}^2) into $\{0, 1\}$, which is completely determined by $f^{-1}(\{1\})$, i.e. the set of white pixels, so that f can be identified with the set $f^{-1}(\{1\})$, a subset of X , the so-called domain of the image. A 2-dimensional grey-scale image can be represented as a mapping from X to the universe of grey-values $[0, 1]$, where 0 corresponds to black, 1 to white and in between we have all shades of grey. Colour images are then represented as mappings from X to a ‘colour interval’ that can be for example the product interval $[0, 255] \times [0, 255] \times [0, 255]$ (the colour space RGB). Colour images can be represented using different colour spaces; more information about colour spaces can be found in [13], [14].

2.2 Binary and Colour Morphology

Consider a binary image X and a binary structuring element A , which is also a binary image but very small in comparison with X . The translation of X by a vector $y \in \mathbb{R}^n$ is defined as $T_y(X) = \{x \in \mathbb{R}^n \mid x - y \in X\}$; the reflection of X is defined as $-X = \{-a \mid a \in X\}$.

Definition 1. *The binary dilation $D(X, A)$ of X by A is the binary image*

$$D(X, A) = \{y \in \mathbb{R}^n \mid T_y(A) \cap X \neq \emptyset\}.$$

The binary erosion $E(X, A)$ of X by A is defined as

$$E(X, A) = \{y \in \mathbb{R}^n \mid T_y(A) \subseteq X\}.$$

The dilation enlarges the objects in an image, while the erosion reduces them.

Property 1. *If A contains the origin (i.e. $\mathbf{0} \in A$), then*

$$E(X, A) \subseteq X \subseteq D(X, A).$$

With this property the binary image $D(X, A) \setminus E(X, A)$ can serve as an edge-image of the image X , which we call the morphological gradient $G^A(X)$ of X . Analogously we define the external morphological gradient $G^{A,e}$ and the internal morphological gradient $G^{A,i}$ as

$$G^{A,e}(X) = D(X, A) \setminus X \quad \text{and} \quad G^{A,i}(X) = X \setminus E(X, A),$$

which will give us the external and inner edge-image of X respectively. We will use the internal morphological gradient to detect the positions of jaggies in an image. More information about binary morphology can be found in [[15] - [17]]. In [18] we have extended the basic morphological operators dilation and erosion to vector morphological operators for colour images by defining a new vector ordering for colours in the RGB colour space. Therefore we will also use the notations $D(X, A)$ and $E(X, A)$ for colour images.

2.3 The Hit-or-Miss Transformation

Consider again a binary image X and two binary structuring elements A and B . The hit-or-miss operator is defined by

$$X \otimes (A, B) = E(X, A) \cap E(X^c, B),$$

where X^c is the complement of X w.r.t. \mathbb{R}^n . The result is empty if $A \cap B \neq \emptyset$. The name hit-or-miss operator can be explained as follows: a pixel h belongs to the hit-or-miss transformation $X \otimes (A, B)$ iff $T_h(A)$ does not hit (intersect with) X^c and $T_h(B)$ does not hit X . The hit-or-miss operator is very useful for the detection of points inside an image with certain (local) geometric properties, e.g. isolated points, edge points, corner points.

As an example we show the detection of the upper-left corner points of objects in an image:

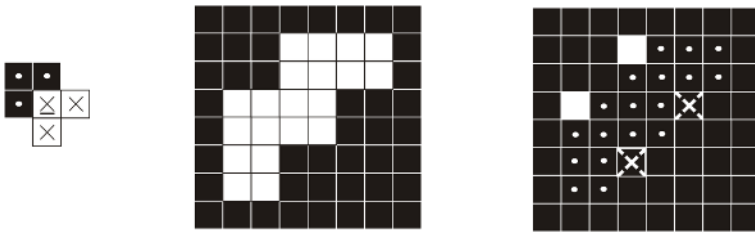


Fig. 1. From left to right: The structuring elements (A, B) , where A contains the white pixels (\times) and B the black pixels (\cdot), the original binary image X and the hit-or-miss transformation $X \otimes (A, B)$ (only the white pixels)

3 New Morphological Image Interpolation Method

3.1 Pixel Replication or Nearest Neighbour Interpolation

The resolution of an image is the number of pixels per unit area, and is usually measured in pixels per inch. When we magnify an image V times, the number of pixels will increase (V^2 times). The easiest way to enlarge an image is to copy the existing pixel values to the new neighbouring pixels. If we magnify

an image V times, one pixel in the original image will be replaced by a square of $V \times V$ pixels in the new image. This is called pixel replication or nearest neighbour interpolation. The result is quite poor, but we can use it as a first ‘trivial’ interpolation step.

3.2 Corner Detection

To detect the unwanted jaggies in the nearest neighbour interpolated image, we first determine the inner edge-image of the blown up image and then apply the hit-or-miss operator to obtain the positions of the object corner edge pixels. The advantage of the internal morphological gradient is that this gradient will give the correct pixel positions of corners in an image. When our original image is a binary image, the blown up image will also be a binary image, and so will the inner edge-image. On the other hand, if our original image is a colour image, the inner edge-image of the blown up image will be a colour image, but we transform it into a binary image by giving all non-black pixels a white value.

Let’s call O the nearest neighbour interpolated image of the original image X , O^c the complement of O , O_{edge} the inner edge-image of O , and O_{edge}^c the inner edge-image of the complement image O^c . With a given pair of structuring elements (A, B) we first determine the hit-or-miss transformation $O_{edge} \otimes (A, B)$ and secondly the hit-or-miss transformation $O_{edge}^c \otimes (A, B)$. This way we will not only detect corners of objects in O , but also corners of objects in O^c .

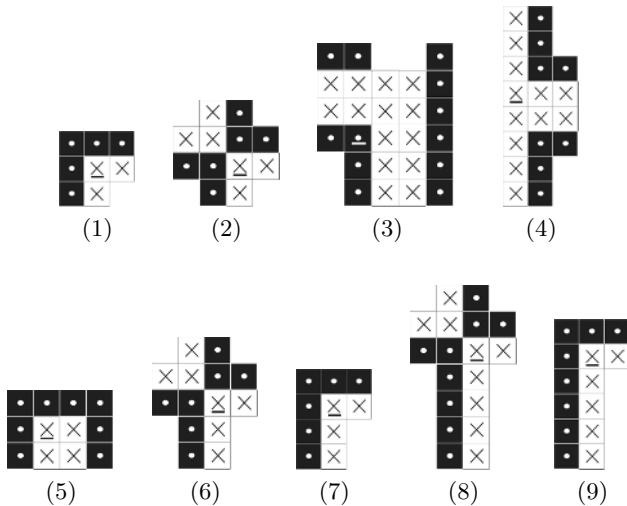


Fig. 2. The used structuring elements (1) (A_1, B_1) ... (9) (A_9, B_9) (the underlined element corresponds to the origin of coordinates) for magnification by a factor 2

Not all the corner pixels in the image should be changed, because some corners are ‘real’ corners, which have to be preserved in the magnified image, whereas others are part of jaggies and have to be removed. Therefore we will use different kinds of structuring elements in the hit-or-miss transformation. The structuring elements used for magnification by a factor 2 are shown in fig. 2, where A_i contains the white pixels (\times) and B_i the black pixels (\cdot), $i = 1 \dots 9$. The other structuring elements are rotated or reflected versions of these. For example, the structuring elements (A_1, B_1) will allow us to detect all upper-left corner pixels, while using structuring elements that are properly rotated versions of (A_1, B_1) we will detect all upper-right, lower-left and lower-right corners. We not only look for corners of the ‘foreground’ objects, but also for corners of the ‘background’ objects. Consequently we will have 8 different kinds of corner pixels (4 ‘object’ corner pixels and 4 ‘background’ corner pixels). In the example in section 2.3 (fig. 1), not only the white pixels ($X \otimes (A, B)$) will be detected, but also the black pixels with white \times -symbol ($X^c \otimes (A, B)$).

3.3 Corner Validation

In this section we explain our method for magnification by a factor 2.

Step 1. We look for corners determined by the structuring elements (A_1, B_1) and their rotations. For example, if we detect an upper-left object corner pixel a or an upper-left background corner pixel a^* at position (i, j) in the edge-image O_{edge} (see figure 3). An upper-left object corner pixel will be determined by the hit-or-miss transformation $O_{edge} \otimes (A_1, B_1)$, while an upper-left background corner pixel will be determined by the hit-or-miss transformation of the complement of O_{edge} , i.e. $O_{edge}^c \otimes (A_1, B_1)$. Then we change the colour of the pixel at position (i, j) in the blown up image O by a mixture of the ‘foreground’ and ‘background’ colour of the surrounding pixels. We obtain this by adding the pixel values of the erosions $E(O, A_1)$ and $E(O, B_1)$ at position (i, j) , that is, $O(i, j) = E(O, A_1)(i, j) + E(O, B_1)(i, j)$. Let $[r_c, g_c, b_c]$ and $[r_{c'}, g_{c'}, b_{c'}]$ be the RGB colour vector of the pixel $E(O, A_1)(i, j)$ and $E(O, B_1)(i, j)$ respectively, we define the new colour value of $O(i, j)$ with RGB components (r, g, b) as

$$r \stackrel{def}{=} (r_c + r_{c'})/2, \quad g \stackrel{def}{=} (g_c + g_{c'})/2, \quad b \stackrel{def}{=} (b_c + b_{c'})/2.$$



Fig. 3. Step 1 worked out for the structuring elements (A_1, B_1)

Step 2. We detect corners with the pair (A_2, B_2) . If we detect such an object corner a or a background corner a^* at position (i, j) in O_{edge} (see figure 4), we fill the corner pixels at position $(i - 1, j)$ and $(i, j - 1)$ or at positions $(i - 1, j - 1)$ and (i, j) , in the blown up image O with the foreground colour. Note that the foreground colour is defined by the minority colour in the image. Here, for the foreground colour of the pixel $O(i - 1, j)$ we determine which of the two colour values $O(i - 1, j)$ and $O(i, j)$ is less present in the image. The foreground colour of the pixel $O(i, j - 1)$ is determined by the minority colour of the pixels $O(i, j - 1)$ and $O(i, j)$. Analogously for $O(i - 1, j - 1)$ and $O(i, j)$.

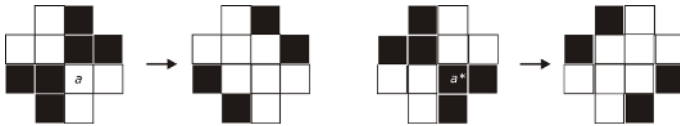


Fig. 4. Step 2 illustrated with structuring elements (A_2, B_2)

Step 3. We have added the structuring elements (A_3, B_3) , (A_4, B_4) , (A_5, B_5) and rotated or reflected structuring elements because we experienced that they are representative for the corner structures that should not be changed in an image. When we find a corner determined by one of these structuring elements, we leave the observed pixels unchanged in the interpolated image to avoid that real corners will be removed in the magnified image.

Step 4. We look at structuring elements of the form (A_6, B_6) , (A_7, B_7) and rotated or reflected versions. See figure 5. In the first case (1), when an object corner a or background corner a^* is determined by (A_6, B_6) at position (i, j) in O_{edge} , we replace the pixel value at position $(i + 1, j - 1)$ or position $(i + 1, j)$ in the image O by the colour with RGB components $r \stackrel{def}{=} (1/4 \times r_c + 3/4 \times r_{c'})$, g and b defined analogously, where $[r_c, g_c, b_c]$ and $[r_{c'}, g_{c'}, b_{c'}]$ are the RGB colour vectors of the pixels $E(O, A_1)(i, j)$ and $E(O, B_1)(i, j)$. In the second case (2), if a corner a or a^* is determined by the structuring elements (A_7, B_7) or by the structuring elements (A_6, B_6) in the special composition as illustrated in fig. 5(a) for a (for a^* we get an analogous figure), at position (i, j) in O_{edge} , we replace the original colour at position $(i + 1, j)$ or position $(i + 1, j - 1)$ in O by the colour with RGB components $r \stackrel{def}{=} (3/4 \times r_c + 1/4 \times r_{c'})$, g and b analogous, where $[r_c, g_c, b_c]$ and $[r_{c'}, g_{c'}, b_{c'}]$ are the RGB colour vectors of the pixels $E(O, A_1)(i, j)$ and $E(O, B_1)(i, j)$. The colour value of $O(i, j)$ or $O(i, j - 1)$ is changed to the RGB colour (r', g', b') with $r' \stackrel{def}{=} (1/4 \times r_c + 3/4 \times r_{c'})$, g' and b' analogous.

Step 5. At last we consider the pairs of structuring elements (A_8, B_8) , (A_9, B_9) and their rotated or reflected structuring elements. When we find such an object

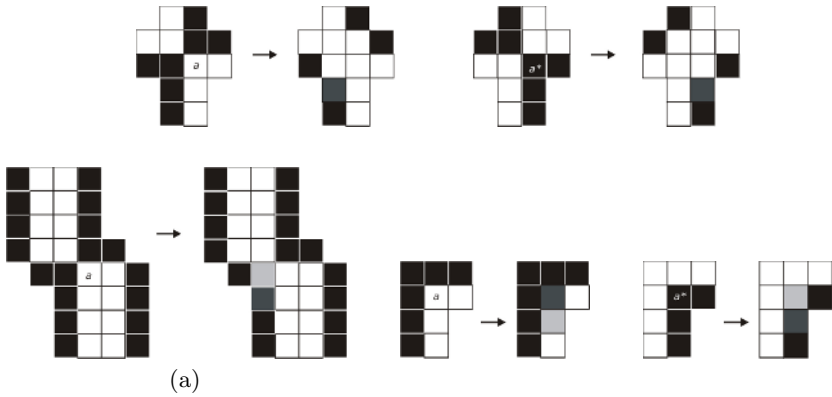


Fig. 5. Step 4, at the top: case (1) and at the bottom: case (2)

corner a (or background corner a^*) at position (i, j) , we move the colour of pixel $O(i + 1, j - 1)$ ($O(i + 1, j)$) to pixel $O(i + 2, j - 1)$ ($O(i + 2, j)$) and give pixel $O(i + 1, j - 1)$ ($O(i + 1, j)$) an intermediate colour value between the foreground and background colour of pixel $O(i, j)$.

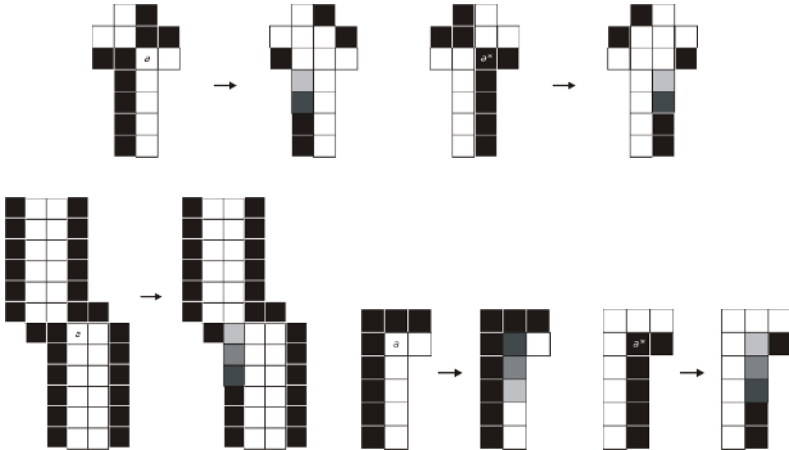


Fig. 6. Step 5, at the top: case (1) and at the bottom: case (2)

3.4 Magnification by an Integer Factor $n > 2$

Now, for magnification by an integer factor $n > 2$, we have to extend the structuring elements to a larger size but a similar shape, and the way of filling up

the edge pixels will change a bit, but is very analogous. In figure 7 we have illustrated this process for magnification by a factor 3:



Fig. 7. The corner validation for magnification by a factor 3

4 Experimental Results

Fig. 8 and 9 show some results of our interpolation method.

Fig. 9 and 10 illustrate the result of several interpolation methods. We have compared our technique with the following state-of-the-art methods: the high-quality magnification filter Hq [10], which analyses the 3×3 area of the source pixel and makes use of lookup tables to get interpolated pixels of the filtered image, the anti-alias filter [12], which detects aliasing and changes only aliased edges

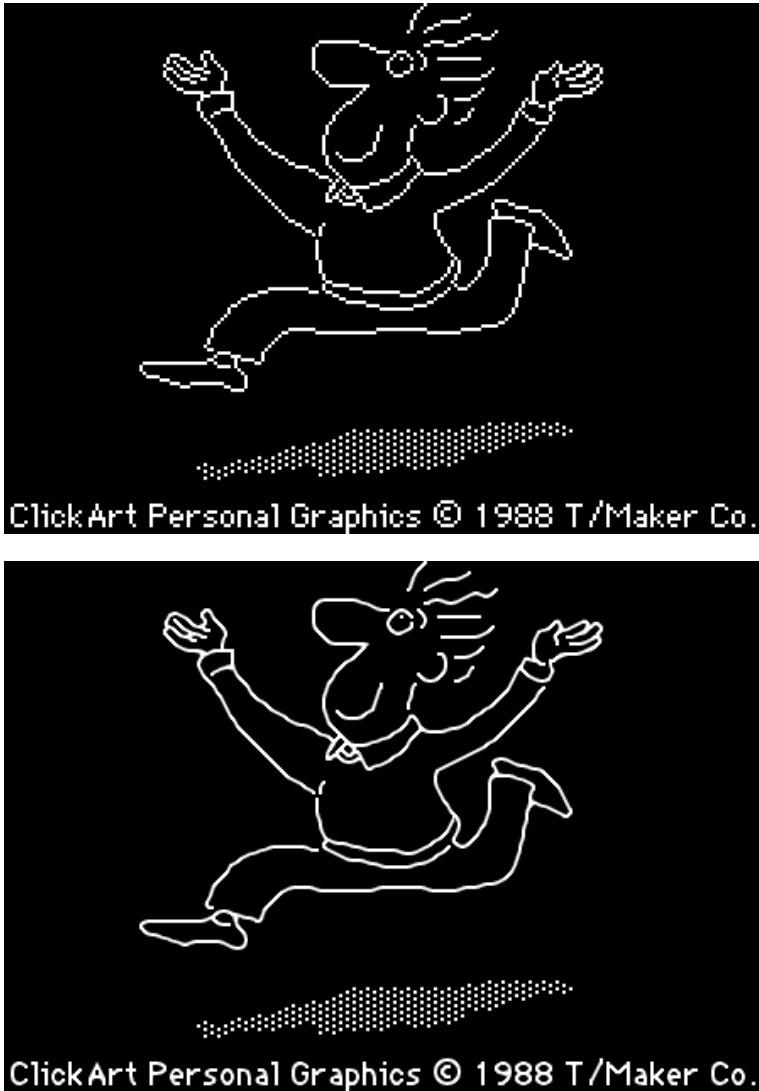


Fig. 8. At the top: the pixel replication ‘cartoon’ image for magnification by a factor 2, at the bottom: the result of our morphological interpolation method

and pixels, and some classical linear interpolation methods [1], in particular, bi-linear and bicubic interpolation, which use the (weighted) mean of respectively 4 and 16 closest neighbours to calculate the new pixel value, and sinc interpolation, which makes use of windowed sinc functions. The main advantages and drawbacks of these linear interpolation filters are pointed out in [1].

We may conclude that our new method provides very beautiful results. Improvements in visual quality can be noticed: unwanted jaggies have been removed

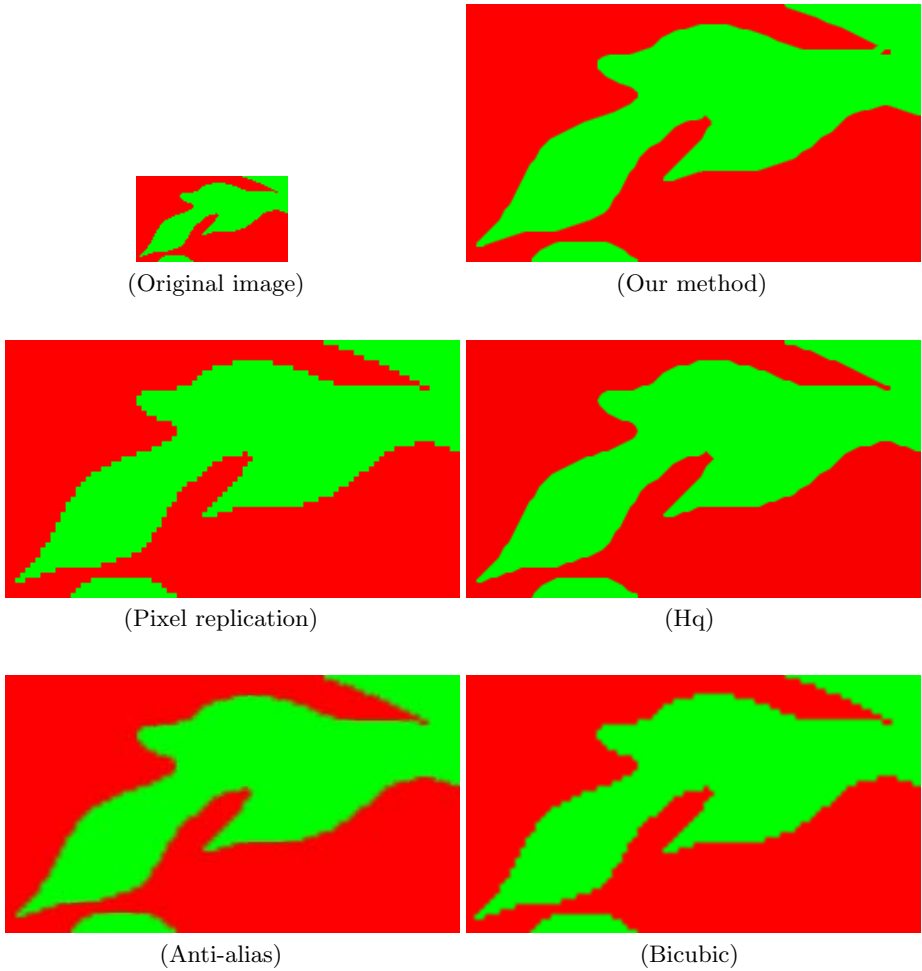


Fig. 9. Interpolation results for magnification by a factor 3

so that edges have become smoother. Good results are also obtained with the hq interpolation method, but our method outperforms all the others.

Our method was implemented in Matlab, which makes it hard to compare the computational complexity of this method with the others. As future work we will reimplement all methods in the same program language, Java or C++, to make them comparable with each other.

Remark: Sometimes it is desired that binary logos, cartoons and maps remain binary, or that no new colours are introduced in a colour image after magnification. Our method can also produce such a result: we only have to insert a threshold in our method, that is, in section 3.3 all new coloured pixels with RGB colour values greater than or equal to $(1/2 \times [r, g, b])_{\text{foreground colour}}$

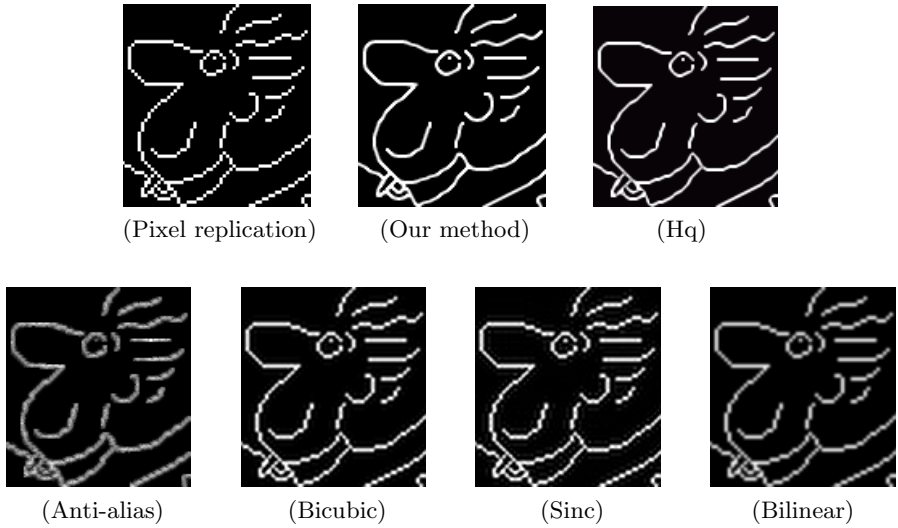


Fig. 10. Result of several interpolation methods for magnification by a factor 2



Fig. 11. Our new morphological interpolation method, giving as result a binary image of the original binary image 'cartoon' (magnification by a factor 2)

$+ 1/2 \times [r, g, b]_{\text{background colour}}$) are assigned to the foreground colour value, while all other colour pixels are transformed to the background colour value. The main visual improvements can be seen in fig. 11: the contours are smooth and text is also interpolated very well.

For a binary image interpolation method making use of mathematical morphology and giving a black-and-white result, we also refer to [19].

5 Conclusion

This paper presents a morphological method that improves the visual quality of magnified binary images with sharp boundaries. The problem of interpolating an image is the introduction of unwanted jagged edges in the blown up image. We developed a new approach to avoid these jaggies, making use of mathematical morphology. We also demonstrated that our method gives beautiful results for the magnification of colour images with sharp edges with a limited number of colours. As future work we will extend our approach towards all colour images with sharp edges, therefore we will define a new vector ordering for colours in the RGB colour space, and towards images with ‘vague’ edges, again using mathematical morphology.

Acknowledgement. This research was financially supported by the GOA-project 12.0515.03 of Ghent University.

References

1. Lehmann, T., Gönner, C., Spitzer, K.: Survey: Interpolations Methods In Medical Image Processing, *IEEE Transactions on Medical Imaging*, Vol. 18, No. 11, (1999) 1049–1075
2. Digital Image Interpolation: <http://www.cambridgeincolour.com/tutorials/image-interpolation.htm>
3. Allebach, J., Wong, P.W.: Edge-Directed Interpolation, *Proceedings of the IEEE International Conference on Image Processing ICIP 96*, Vol. 3, Switzerland, (1996) 707–710
4. Li, X., Orchard, M.T.: New Edge-Directed Interpolation, *IEEE Transactions on Image Processing*, Vol. 10, No. 10, (2001) 1521–1527
5. Muresan, D.D., Parks, T.W.: New Edge-Directed Interpolation, *IEEE Transactions on Image Processing*, Vol. 13, No. 5, (2004) 690–698
6. Tschumperlé, D.: PDE’s Based Regularization of Multivalued Images and Applications, PhD thesis, Université de Nice, France, 2002
7. Morse, B.S., Schwartzwald, D.: Isophote-Based Interpolation, *Proceedings of the IEEE International Conference on Image Processing ICIP 98*, USA, (1998) 227–231
8. Luong, H.Q., De Smet, P., Philips, W.: Image Interpolation Using Constrained Adaptive Contrast Enhancement Techniques, *Proceedings of the IEEE International Conference on Image Processing ICIP 05*, Italy, (2005) 998–1001
9. Honda, H., Haseyama, M., Kitajima, H.: Fractal Interpolation for Natural Images, *Proceedings of the IEEE International Conference on Image Processing ICIP 99*, Japan, (1999) 657–661
10. Stepin M.: hq3x Magnification Filter, <http://www.hiend3d.com/hq3x.html> (2003)
11. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-Based Super-Resolution, *IEEE Computer Graphics and Applications*, Vol. 22, No. 2, (2002) 56–65

12. Power Retouche: Anti-aliasing Filter,
http://www.powerretouche.com/Antialias_plugin_introduction.htm (2001-2006)
13. Sangwine, S. J., Horne, R.E.N.: *The Colour Image Processing Handbook*, Chapman & Hall, London (1998)
14. Sharma, G.: *Digital Color Imaging Handbook*, CRC Press, Boca Raton (2003)
15. Heijmans, H. J.A.M., Ronse, C.: *The Algebraic Basis of Mathematical Morphology, Part1: Dilations and Erosions*, *Computer Vision, Graphics and Image Processing*, Vol. 50, (1990) 245–295
16. Ronse, C., Heijmans, H. J.A.M.: *The Algebraic Basis of Mathematical Morphology, Part2: Openings and Closings*, *Computer Vision, Graphics and Image Processing*, Vol. 54, (1991) 74–97
17. Heijmans, H. J.A.M.: *Morphological Image Operators*, *Advances in Electronics and Electron Physics*, Academic Press, Inc., London (1994)
18. De Witte, V., Schulte, S., Nachtegael, M., Van der Weken, D., Kerre, E.E.: *Vector Morphological Operators for Colour Images*, *Proceedings of the Second International Conference on Image Analysis and Recognition ICIAR 2005, Canada*, (2005) 667–675
19. Ledda, A., Luong, H.Q., Philips W., De Witte, V., Kerre, E.E.: *Image Interpolation Using Mathematical Morphology*, Accepted for the Second International Conference on Document Image Analysis for Libraries DIAL 06, France, (2006)

Adaptive Kernel Based Tracking Using Mean-Shift

Jie-Xin Pu and Ning-Song Peng

Electronic Information Engineering College, Henan University of Science and Technology,
Luoyang, 471039, China
{pjax, pengningsong}@mail.haust.edu.cn

Abstract. The mean shift algorithm is an kernel based way for efficient object tracking. However, there is presently no clean mechanism for selecting kernel bandwidth when the object size is changing. We present an adaptive kernel bandwidth selection method for rigid object tracking. The kernel bandwidth is updated by using the object affine model that is estimated by using object corner correspondences between two consecutive frames. The centroid of object is registered by a special backward tracking method. M-estimate method is used to reject mismatched pairs (outliers) so as to get better regression results. We have applied the proposed method to track vehicles changing in size with encouraging results.

1 Introduction

The mean shift algorithm is an efficient way to do mode seeking by using kernel density estimation [4,5]. Recently, this nonparametric statistical method has been used into the visual tracking fields [1,2,3] where object is represented by kernel histogram and from the same location in the next frame, mean shift iterations based on maximizing the Bayes error associated with the model and candidate distributions continues until its convergence. However, the unchanged kernel bandwidth limits its performance in tracking object changing in size [7]. The problem we address in the paper is selecting the bandwidth of the mean shift kernel, which directly determines the size of the tracking window in which object is enclosed. Our method is proposed under the assumption that is the object we tracked is rigid and its motion satisfies affine model in two consecutive frames. The bandwidth of mean shift kernel is selected by discovering the scale parameters of object affine model described by corner correspondences. By applying a special backward tracking method to register object centroid in different frames, we can not only simplify the affine model but also provide an opportunity to use our simple local-matching method to find corner correspondences. Finally, in order to get better regression result, robust M-estimate is employed to reject outliers caused by mismatching pairs, noise and some other factors.

The paper is organized as follows. In section 2 we review the mean shift theory and illuminate the limitation of using unchanged bandwidth in visual tracking. In section 3

the bandwidth selection method is developed. Experimental results are given in section 4, and the conclusion is in section 5.

2 Mean Shift Object Tracking

2.1 Mean Shift Theory

The mean shift algorithm is a nonparametric method based on kernel density estimation (KDE) for mode seeking [4,5]. Let data be a finite set A embedded in the n -dimensional Euclidean space X . The sample mean at $x \in X$ is

$$mean(x) = \frac{\sum_a K(a-x)w(a)a}{\sum_a |K(a-x)w(a)|}, \quad a \in A \quad (1)$$

where K is kernel function, and w is weight function that can be negative value [7]. The difference $mean(x) - x$ is called mean shift vector. The repeated movement of data points to the sample means is called the mean shift algorithm. An important property of the mean shift algorithm is that the local mean shift vector computed at position x using kernel K points opposite to the gradient direction of the convolution surface

$$J(x) = \sum_a G(a-x)w(a) \quad (2)$$

Here, kernel K and G must satisfy the relationship

$$g'(r) = -ck(r), \quad r = \|a-x\|^2, \quad c > 0 \quad (3)$$

where g and k are profile of kernel G and K respectively. A commonly used kernel is Gauss kernel function

$$k(x) = \frac{1}{2\pi} \exp\left(-\frac{1}{2}\|x\|^2\right) \quad (4)$$

whose corresponding function g is also a Gauss kernel function.

In literature [1,2], $J(x)$ is designed by using object kernel histogram, and tracking problem finally converts to mode seeking problem by using Bhattacharyya coefficient to measure similarity of two kernel histograms belonging to object image and image in the searching area respectively.

2.2 Kernel Histogram and Kernel Bandwidth

In the mean shift tracking algorithm [1,2], kernel histogram plays an important role in object representation. Let $\{X_i\}_{i=1 \dots n}$ be the pixel locations of object image in a frame, centered at position Y . In the case of gray level image, the kernel histogram is defined as

$$\begin{aligned}
 h_{\mu}(Y) &= C \sum_{i=1}^n k\left(\left\|\frac{Y-X_i}{r}\right\|\right) \delta[b(X_i)-\mu] \\
 C &= \frac{1}{\sum_{i=1}^n k(\|X_i\|^2)}, \quad \mu = 1 \dots m
 \end{aligned}
 \tag{5}$$

where μ is the bin index of kernel histogram and m is the number of bins. $h_{\mu}(Y)$ is the entry of kernel histogram with the index μ . $\{h_{\mu}(Y)\}_{\mu=1 \dots m}$ is the object kernel histogram. $b(X_i)$ is the quantization value of the pixel intensity at X_i . $r = \{r_x, r_y\}$ is called kernel bandwidth which normalizes coordinates of image so as to make radius of the kernel profile to be one. Imposing the condition $\sum_{\mu=1}^m h_{\mu}(Y) = 1$ derives the constant C . The bandwidth $r = \{r_x, r_y\}$ is also the radius of the tracking window covering the object we want to track. If the object scale expands gradually, the unchanged bandwidth leads to poor localization because the kernel is so small as to find similar color distribution inside the expanded blob. On the contrary, if the object shrinks, tracking window will contain many background pixels, which makes kernel histogram diluted and multiple modes will be included.

There is presently no clean mechanism for selecting bandwidth while object is changing in size. In literature [1,2], by using the Bhattacharyya coefficient, author expands or reduces the tracking window size, i.e., bandwidth, by 10 percent of the current size to evaluate the best scale. This method has no ability to keep the window inflated because any location of a window that is too small will yield the similar value of the Bhattacharyya coefficient. Scale kernel is used to get best tracking window size [7] and the scale space is defined as $\{\delta 1.1^b\}_{b=-2, -1, 0, 1, 2}$ where δ is the initial scale. By using Epanechnikov kernel, its mean shift iterations for finding best scale is equal to average all the scale in scale space under the condition that the spatial location has been matched. Therefore, this method has the same limitation analyzed above.

3 Method Proposed

In this section we consider a method for automatically selecting bandwidth for rigid object tracking. We consider only the two kinds of motions encountered most frequently in image sequences: translation and scaling. Since translation motion doesn't need bandwidth update, we should emphasize on scaling motion. Given two consecutive frames i and $i + 1$, we define the affine model as [8]

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} m_x \\ m_y \end{bmatrix}
 \tag{6}$$

where (x, y) is the location of object feature point in frame i and (x', y') is the counterpart in frame $i+1$. $m = \{m_x, m_y\}$ specifies the translation parameter while $s = \{s_x, s_y\}$ denotes the scaling magnitude. If we know the scaling magnitude, the kernel bandwidth can be updated as follows.

$$\begin{cases} r_x = r_x \cdot s_x \\ r_y = r_y \cdot s_y \end{cases} \quad (7)$$

We use object corners as the feature points to estimate scaling magnitude for bandwidth selection. Our method first uses corner detector to extract the feature points from the tracking windows in both frame i and $i+1$. Corner correspondences are then obtained by a local-matching method with the help of the object centroid registration that is implemented by a special backward tracking. Finally, robust regression, i.e., M-estimate is applied to estimate scaling magnitude by rejecting outliers. Fig. 1. illustrates the outline of our proposed method when object expands its size.

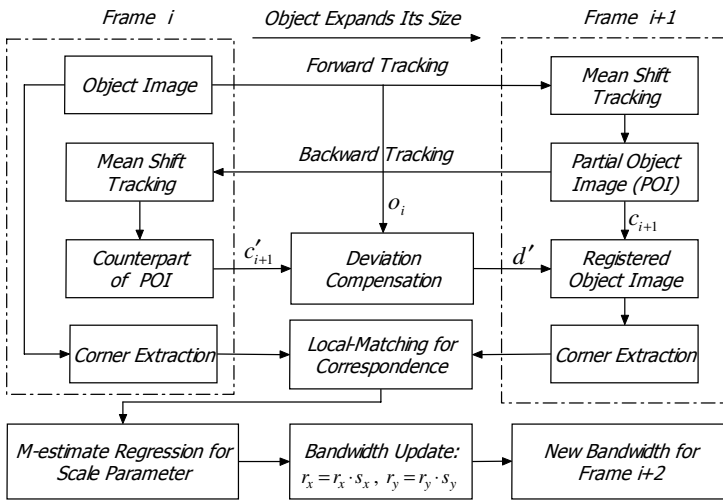


Fig. 1. Outline of proposed method

3.1 Detection of Corner Correspondences

We extract corners from object tracking window according to the operator [6]

$$R(x, y) = \det[B] - l \times \text{trace}^2[B] \quad (8)$$

where l is a constant and the matrix B is defined as

$$B = \begin{bmatrix} \sum E_x^2 & \sum E_x E_y \\ \sum E_x E_y & \sum E_y^2 \end{bmatrix} \tag{9}$$

E_x and E_y are the spatial image gradients on the horizontal direction and vertical direction respectively.

Given two corner sets from two tracking windows in consecutive frames, we use a simple local-matching method to find corner correspondences on condition that the object centroid is registered and used as original point, see also section 3.2. Let us suppose that the number of corners extracted from t_i , the tracking window in frame i are N while corners in t_{i+1} , the tracking window in frame $i+1$ are M . In the case of gray level image, given a corner point $p_c|t_i$ in t_i , its correspondence point $p_c|t_{i+1}$ in the t_{i+1} should satisfies

$$I(p_c|t_{i+1}) = \min \{ |I(p|t_i) - I(p_j|t_{i+1})| \}_{j=1,2,\dots,n} \tag{10}$$

where I is pixel intensity and $n (n < M)$ is the number of candidate corners within a small given window centered at the location of $p_c|t_i$ in the t_{i+1} . $n=0$ means there is no correspondence for $p_c|t_i$. For each $p_c|t_i$ in the t_i , we use (10) to find its correspondence in t_{i+1} . Obviously, our local-matching method is simple and fast to implement comparing to the maximum-likelihood template matching method [8,9] whose computational complexity is $O(NM)$.

3.2 Object Centroid Registration by Backward Tracking

Using an unchanged bandwidth to track an object by mean shift algorithm often get poor localization when the object expands its scale [7]. On the contrary, when the object shrinks its scale, though tracking window will contain many background pixels as well as the foreground object pixels, the center of tracking window always indicates the object centroid [7], which gives us a cue to register the object centroid when object expands its scale. Let us suppose at frame i , the object with its centroid at o_i is well enclosed by an initial tracking window t_i centered at $c_i = o_i$. When the object expands its scale at frame $i+1$, there should be a bit offset $d = c_{i+1} - o_{i+1}$ that is the tracking deviation caused by mean shift tracking algorithm, where o_{i+1} is the real centroid of the object in the current frame $i+1$ and c_{i+1} is the center of the current tracking window t_{i+1} . Obviously, some parts of the object are in t_{i+1} because the current object scale is bigger than the size of t_{i+1} . In order to register the two centroid of object which is changing in scale, first, we generate a new kernel

histogram representing the partial object enclosed by t_{i+1} . Actually, c_{i+1} indicates the centroid of this partial object. Secondly, we seek c'_{i+1} , i.e., the correspondence of c_{i+1} , backwards in frame i also by mean shift tracking algorithm. Because from frame $i+1$ to i , this partial object shrinks its scale, it is possible for us to find its accurate centroid c'_{i+1} in frame i . Therefore, there should be another offset $d' = c_i - c'_{i+1}$ between c_i and c'_{i+1} . Under the assumption that the object scale doesn't change so much during the consecutive frames, we can compensate the deviation d by d' . Finally, the object centroid in current frame $i+1$ is evaluated by

$$o_{i+1} \approx c_{i+1} + d' \quad (11)$$

Given the object image from initial tracking window t_i , we can use mean shift tracking algorithm and equation (11) to register object centroid at frame $i+1$, which reduces the potential mismatching pairs remarkably. In addition, we can get translation parameter of object affine model by directly using

$$m = o_{i+1} - o_i, \quad (12)$$

which simplifies the affine model (6) so as to reduce the computation of regression. When the object shrinks its scale from frame i to $i+1$, the method described above may suffer from incorrect registration result. However, since object at frame $i+1$ is almost within the tracking window t_{i+1} , the registration error influences little on local-matching results.

3.3 Bandwidth Selection by Robust Regression

Although centroid registration help us to reduce the mismatching correspondences, it is inevitable to reject all the outliers caused by noise, occlusions and fake similarities as well as our local-matching method (10). Therefore, we employ M-estimate to perform outlier rejection while modeling matching pairs with affine model. With corners set $\{(x_1, y_1), \dots, (x_N, y_N)\}$ in the t_i , the modeling process can be defined as a search for $s = \{s_x, s_y\}$ of the model $\hat{f}(s, x_i, y_i)$ that best fits the correspondence point set $\{(x'_1, y'_1), \dots, (x'_N, y'_N)\}$ in t_{i+1} . We define the residual errors as

$$e(x_i, y_i) = (x'_i - s_x \cdot x_i - m_x)^2 + (y'_i - s_y \cdot y_i - m_y)^2 \quad (13)$$

where m_x and m_y are solved by (12). A robust M-estimate for $s = \{s_x, s_y\}$ minimizes function $\varepsilon(s)$ of the deviations of the $\{(x'_1, y'_1), \dots, (x'_N, y'_N)\}$ from the estimate $\hat{f}(s, x_i, y_i)$:

$$\varepsilon(s) = \sum_{i=1}^N \rho\left(\frac{e(x_i, y_i)}{\alpha}\right) \tag{14}$$

Parameter α is previously computed and ρ is a robust loss function. Let $\psi(s, x, y) = \frac{\partial(\rho(s, x, y))}{\partial(s)}$, the necessary condition for a minimum is

$$\begin{cases} \psi\left(\frac{e(x_i, y_i)}{\alpha}\right) \cdot x_i = 0 \\ \psi\left(\frac{e(x_i, y_i)}{\alpha}\right) \cdot y_i = 0 \end{cases} \tag{15}$$

Introducing a set of weighting parameters

$$\omega(x_i, y_i) = \begin{cases} \frac{\psi\left(\frac{e(x_i, y_i)}{\alpha}\right)}{\alpha} & \text{if } e(x_i, y_i) \neq 0 \\ 1 & \text{if } e(x_i, y_i) = 0 \end{cases} \tag{16}$$

Equation (15) can be rewritten as follows.

$$\begin{cases} \omega(x_i, y_i) \cdot x_i \cdot e(x_i, y_i) = 0 \\ \omega(x_i, y_i) \cdot y_i \cdot e(x_i, y_i) = 0 \end{cases} , \tag{17}$$

which can be solved iteratively via iteratively re-weighted least squares (IRLS) [10].

4 Experimental Results

In our experiments, object kernel histogram computed by Gauss kernel has been derived in the *RGB* space with $32 \times 32 \times 32$ bins. In Fig. 2., (a) and (b) are two consecutive frames named by i and $i + 1$. The vehicle enclosed by green block with the size 52×37 in frame i is the initial object that is tracked in frame $i + 1$ where vehicle not only moves but also expands its size. By using mean shift tracking algorithm with an unchanged kernel bandwidth, see the tracking windows (52×37 green block) in (b), localization is poor because the vehicle expands its size from frame i to $i + 1$. The red block is the registered tracking window by backward tracking method in which new kernel histogram is created from the image in green block at frame $i + 1$. The corner operator [6] is initialized with the deviation of smoothing Gauss set by 1 and threshold by 2000. Fig. 2. (c) illustrates the corners extracted from initial tracking window, and (d) shows the corners extracted from the extended unregistered tracking windows. Corners from the extended registered tracking window are shown in (e). The two extended windows size are both 1.1 times of their previous size. To extend the windows size is for getting more candidate

correspondence. In Fig. 3., object corners (*plus*) in frame i and their correspondences (*dot*) in frame $i + 1$ are shown in a same coordinates with origin point at tracking window center and lines represent the correspondent relationship. Fig. 3. (b) illustrates the results with the support of centroid registration in contrast to the results in (a). It is clear that the lines in (b) are almost with the same orderly and outstretched tide comparing to the lines in (a), which indicates outliers can be partly restrained by centroid registration method in terms of our special backward tracking.

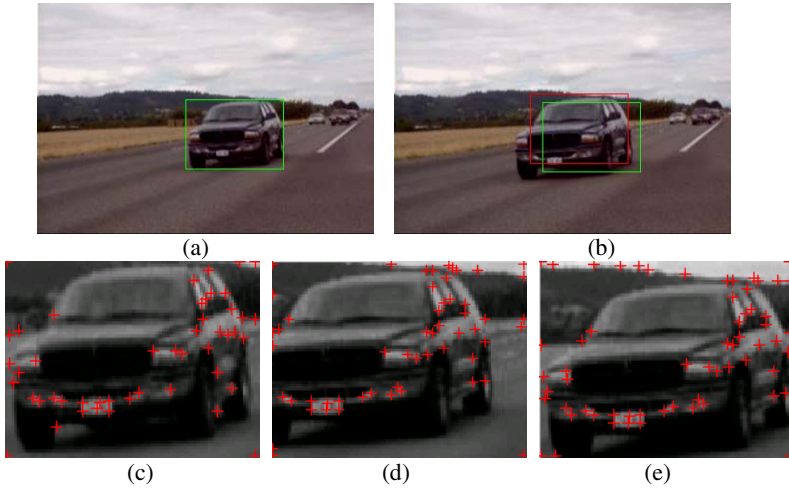


Fig. 2. Tracking windows and corners: (a) initial tracking window (green block) in frame i ; (b) mean shift tracking result (green block) and registered tracking window (red block) in frame $i + 1$; (c) corners in initial tracking window; (d) corners in expanded unregistered tracking window; (e) corners in expanded registered tracking window

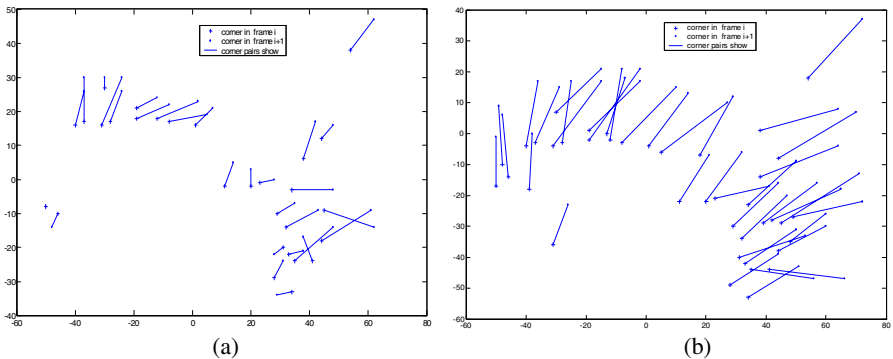


Fig. 3. Corners and correspondences: (a) result without centroid registration; (b) result with centroid registration

The weight function used in E-estimate regression is the *Huber Minimax* with the expression:

$$\omega(x_i, y_i) = \begin{cases} \frac{\alpha}{|e(x_i, y_i)|} & \text{if } |e(x_i, y_i)| > \alpha \\ 1 & \text{if } |e(x_i, y_i)| \leq \alpha \end{cases} \quad (18)$$

The regression comparison between M-estimate (black line) and Least-squares estimator (blue line) is shown in Fig. 4. Referenced by real scale magnitude $s_x = 1.2335$ measured manually in horizontal direction, the M-estimate regression yield $s_x = 1.2231$ comparing to 1.167 by Least-squares estimator.

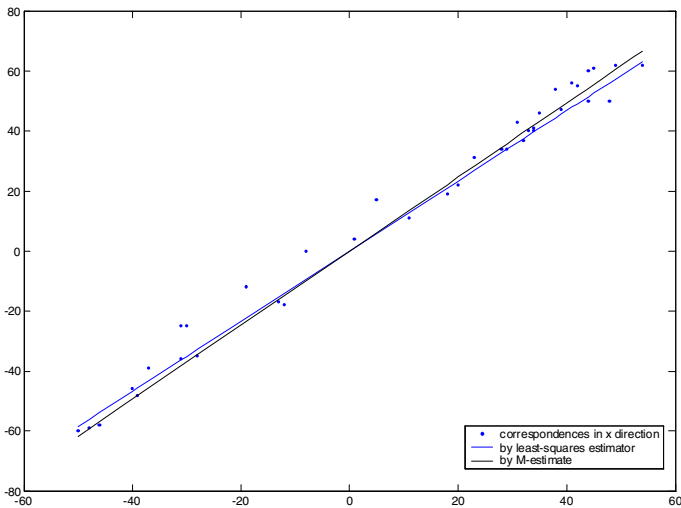


Fig. 4. M-estimate regression vs. Least-squares regression

5 Conclusion

In this paper, we proposed a novel kernel bandwidth selection method for mean shift based rigid object tracking. The method is mainly based on estimating scale magnitude which subjects to object affine transform. Corner correspondences are used for discovering affine model by M-estimate regression. To reduce computational complexity for finding corner correspondences, we apply a local-matching method in which object centroid registration plays an indispensable role. When object expands its size, by tracking backwards, the deviation caused by mean shift tracking algorithm is compensated so as to register object centroid in consecutive frames. However, mismatching pairs are inevitable due to noise, changeable scenes and some other factors. Robust regression technique, i.e. M-estimate is employed in our system to reject outliers so as to get better scaling magnitude for bandwidth selection. In

contrast to the empirical bandwidth selection method in literature [1,2], where 3 times mean shift tracking should be performed, our method is not only efficient but also robust to outliers. Experimental results show that our method provides a valid way to update kernel bandwidth and can be embedded into the mean shift based tracking system.

Reference

1. D. Comaniciu, V. Ramesh, P. Meer, 2000. Real-time tracking of non-rigid objects using mean shift. *IEEE Int. Proc. Computer Vision and Pattern Recognition*. 2, 142-149.
2. D. Comaniciu, V. Ramesh, P. Meer, 2003. Kernel-based object tracking. *IEEE Trans. Pattern Analysis Machine Intelligence*. 25(5), 564-575.
3. Alper Yilmaz, Khurram Shafique, Mubarak Shah, 2003. Target tracking in airborne forward looking infrared imagery. *Image and Vision Computing*. 21, 623-635.
4. Cheng, Y., 1995. Mean Shift, mode seeking and clustering. *IEEE Trans. Pattern Analysis and Machine Intelligence*. 17(8), 790-799.
5. Fukunaga, K. and Hostetler, L.D., 1975. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Trans. Information Theory*, 21, 32-40.
6. C. Harris, M. Stephens, 1988. A combined corner and edge detector. *Int. Conf. Alvey Vision Conference*. 189-192.
7. Ning Song Peng, 2005. Mean-Shift Object Tracking with Automatic Selection of Kernel-Bandwidth. *Journal of Software*. 16(9), 1542-1550
8. Robert T. Collins, 2003. Mean-shift blob tracking through scale space. *IEEE Int. Proc. Computer Vision and Pattern Recognition*. 2, 234-240.
9. Weiyu Hu, Song Wang, Rwei-Sung Lin, Stephen Levinson, 2001. Tracking of object with SVM regression. *IEEE Int. Conf. Computer Vision and Pattern Recognition*. 2, 240-245.
10. C.F. Olson, 2000. Maximum-Likelihood template matching. *IEEE Int. Conf. Computer Vision and Pattern Recognition*. 2, 52-57.
11. P. J. Besl, J. B. Birch, L. T. Watson, 1989. Robust window operators. *Machine Vision and Applications*. 2, 179-192.

Real Time Sobel Square Edge Detector for Night Vision Analysis

Ching Wei Wang

Vision and Artificial Intelligence Group,
Department of Computing and Informatics,
University of Lincoln, Brayford Pool,
Lincoln LN6 7TS, United Kingdom
cweiwang@lincoln.ac.uk

Abstract. Vision analysis with low or no illumination is gaining more and more attention recently, especially in the fields of security surveillance and medical diagnosis. In this paper, a real time sobel square edge detector is developed as a vision enhancer in order to render clear shapes of object in targeting scenes, allowing further analysis such as object or human detection, object or human tracking, human behavior recognition, and identification on abnormal scenes or activities. The method is optimized for real time applications and compared with existing edge detectors. Program codes are illustrated in the content and the results show that the proposed algorithm is promising to generate clear vision data with low noise.

1 Introduction

Night vision analysis is increasingly seen as important both for academic research and industrial development. Applications involving surveillance system and medical diagnosis are obliged to perform video monitoring under circumstances of low illumination or a complete lack of visible light, in order to comply with security or medical requirements. Thermal imaging and infrared imaging techniques are generally utilized for assisting capture of video information for such applications. Due to the high price of thermal cameras and the limitation on the nature of targets, which are with or without heat, infrared imaging is more popularly adopted.

Although night vision devices improve vision capability, low illumination from limited infrared lighting is still a critical constraint for video monitoring and analysis. In addition, infrared content is constrained in grayscale format without color information provided. Thus, analytical methods, such as skin color models, are not applicable for video analysis using night vision devices. As a result, edge information that may be used to recover shapes of objects is suggested in this work for further analysis, i.e. object detection, object tracking, human activity recognition, and scene identification.

Fig. 1 presents three main functions of image processing, namely data cleaning, data transformation and data improvement, within the system flow in computer vision systems. As edge detector significantly reduces the amount of data and filters out useless information while preserving the important structural properties in an

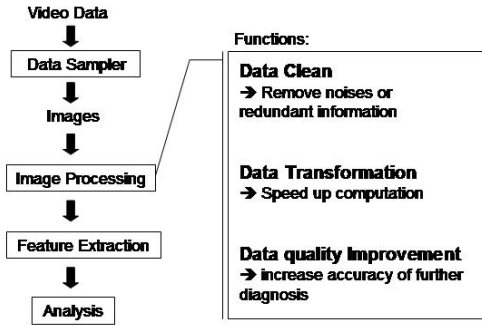


Fig. 1. Main Functions of Image Processing

image, it is thus broadly utilized for fast comparison, compression and speeding up analysis like pattern recognition. In this work, an improved edge detector is specifically developed for night vision analysis in order to extract important information in targeting scenes. Four popular adopted edge detection algorithms [1], i.e. Sobel Kernels, Prewitt Kernels, Kirsch Compass Kernels, Laplacian, and one quick edge detector, which is constructed in this work for experiments, are compared with the proposed approach, and the evaluation results show that the proposed method greatly outperforms others from processing speed, data enhancement and data cleaning viewpoints. The infrared video frames were acquired with resolution of $320 * 240$ and grayscale pixel format. The algorithm is optimized to process video in real time, i.e. 0.03 second per frame. The system is tested over a PC with CPU P4-2.4G and 256M Ram. The prototype is implemented in C#, a high level language for fast development.

Furthermore, the improved edge detector is implemented and applied to an artificial intelligent vision analysis system [2], which monitors patients' behavior during sleep. The diagnosis is generally performed without visible light, providing a comfortable environment for patients. It shows very positive results for human activity recognition.

The paper is organized as follows. Related work and the technical challenges are presented in Section 2. After explaining the details of the existing edge detection algorithms in Section 3, the proposed edge detector algorithm, resulting images and program code are described in Section 4. Finally, future development follows in Section 5.

2 Related Work

Infrared imaging and thermal imaging play important roles for night vision analysis. Under relatively low illumination levels, night vision devices provide enormous benefits for many operational environments. Security Surveillance detects intruders and

abnormal events in a 24 hours basis; car night vision assists human driving safety; military targeting acquires specific objects' locations; medical diagnosis monitors patients' behavior without visible lights.

Plenty of research works have been devoted to human activity recognition, human detection, scene identification, object tracking and abnormal moving direction detection [3, 4, 5]. However, the performance of most methods is greatly influenced by the illumination level of the environment, contributing to unreliable and instable system performance. There is therefore a need for more robotic methods generating high-quality night vision data for further analysis.

2.1 Technical Challenges and Analysis

The limitation of night vision using infrared cameras is that no color information is provided. Images are rendered in grayscale format. In addition, the length of the illumination of the infrared lights is also constrained, contributing to low visibility in interested regions, which diminishes the data quality for further analysis. In other words, for general grayscale images in ordinary lighting status, the range of each pixel value is within 0 to 255. However, under low illumination, vision data is constrained into a small range, i.e. 0 to 30. Information in the targeting scenes is reduced and shrinks into a smaller scope. As a result, a sophisticated image processing technique is necessary here as a vision enhancer and further extract useful information from the reduced-quality data. Fig.2 illustrates that each pixel value of the infrared content shrinks into a smaller range, contributing to data in lower level contrast. Thus, to overcome poor visibility of infrared content, the main idea of this work is to exaggerate differences of each pixel.

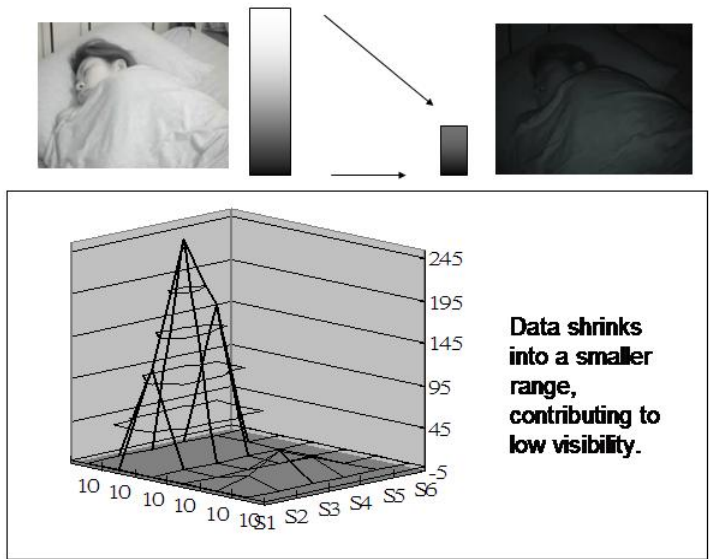


Fig. 3. General Lighting versus Low Illumination

3 Existing Edge Detectors

In this work, several popular edge detectors [1] are built and compared, including first-derivative methods, such as Kirsch compass gradient kernels, Sobel kernels, Prewitt kernels, and a second derivative method, i.e. Laplacian. In this section, existing edge detectors are firstly introduced and then a short discussion is presented on the shortages of these approaches for night vision data.

Edge Detector. Let $f(x,y)$ be the image intensity function. It has derivatives in all directions. The gradient is a vector pointing in the direction in which the first derivative is highest, and whose length is the magnitude of the first derivative in that direction. If f is continuous and differentiable, then its gradient can be determined from the directional derivatives in any two orthogonal directions - standard to use x and y .

$$gradient = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \tag{1}$$

$$magnitude = (\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2})^{\frac{1}{2}} \tag{2}$$

$$direction = \arctan \begin{bmatrix} \frac{\partial f}{\partial y} \\ \frac{\partial f}{\partial x} \end{bmatrix} \tag{3}$$

3.1 Kirsch Compass Kernels

Given k operators, g_k is the image obtained by convolving $f(x,y)$ with the k -th operator. The k defines the edge direction. The gradient is defined as:

$$g(x, y) = \max_k g_k(x, y) \tag{4}$$

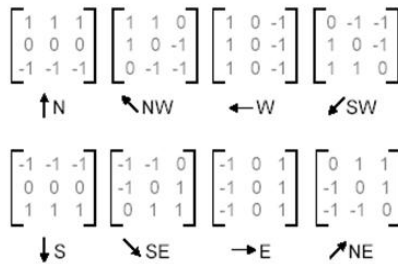


Fig. 4. Kirsch Compass Operators

3.2 Prewitt Kernels

The Prewitt edge detection masks are one of the oldest and best understood methods of detecting edges in images. Basically, there are two masks, one for detecting image derivatives in X and one for detecting image derivatives in Y. To find edges, a user convolves an image with both masks, producing two derivative images (dx & dy). The strength of the edge at any given image location is then the square root of the sum of the squares of these two derivatives. The orientation of the edge is the arc tangent of dy/dx.

3.3 Sobel Kernels

Sobel detector is similar to the Prewitt detector. They both rely on central differences, but greater weights are given to the central pixels in Sobel Kernels. The Sobel kernels can also be thought of as 3 * 3 approximations to first-derivative-of-Gaussian kernels. The equations of sobel kernels are presented below.

$$Gx(x, y) = [f(x + 1, y - 1) + 2 * f(x + 1, y) + f(x + 1, y + 1) - f(x - 1, y - 1) - 2 * f(x - 1, y) - f(x - 1, y + 1)] / 4 \tag{5}$$

$$Gy(x, y) = [f(x - 1, y + 1) + 2 * f(x, y + 1) + f(x + 1, y + 1) - f(x - 1, y - 1) - 2 * f(x, y - 1) - f(x + 1, y - 1)] / 4 \tag{6}$$

$$f(x, y) = \sqrt{Gx(x, y)^2 + Gy(x, y)^2} \tag{7}$$

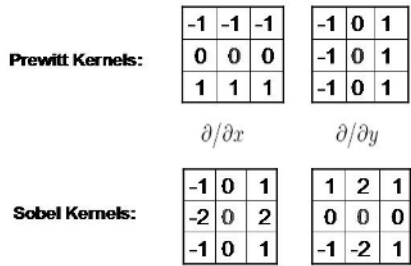


Fig. 5. Templates to calculate the gradient value

3.4 Laplacian

The Laplacian $L(x,y)$ of an image with pixel intensity values $I(x,y)$ is given by:

$$I(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \tag{8}$$

The equation is derived from the following equations.

$$\begin{aligned}\frac{\partial f^2(x, y)}{\partial^2 x} &= f''(x, y) = f'(x+1, y) - f'(x, y) \\ &= [f(x+1, y) - f(x, y)] - [f(x, y) - f(x-1, y)] \\ &= f(x+1, y) - 2f(x, y) + f(x-1, y)\end{aligned}\quad (9)$$

$$\begin{aligned}\frac{\partial f^2(x, y)}{\partial^2 y} &= f''(x, y) = f'(x, y+1) - f'(x, y) \\ &= [f(x, y+1) - f(x, y)] - [f(x, y) - f(x, y-1)] \\ &= f(x, y+1) - 2f(x, y) + f(x, y-1)\end{aligned}\quad (10)$$

$$I(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}\quad (11)$$

Particularly, the higher the derivative, the more sensitivity the operator has.

| | | | | | | | | |
|---|----|---|----|----|----|----|----|----|
| 0 | 1 | 0 | 1 | -2 | 1 | -1 | -1 | -1 |
| 1 | -4 | 1 | -2 | 4 | -2 | -1 | 8 | -1 |
| 0 | -1 | 0 | 1 | -2 | 1 | -1 | -1 | -1 |

Fig. 6. Three commonly used discrete approximations in Laplacian

3.5 Simple Edge Detect

In order to develop a suitable edge detector, a simple edge detector is also produced for experiments. It uses a convolution filter with value below.

| | | |
|----|----|----|
| -1 | -1 | -1 |
| 0 | 0 | 0 |
| 1 | 1 | 1 |

Fig. 7. Convolution filter for simple edge detect

3.6 Shortage on Existing Edge Detectors for Night Vision Analysis

The main issue for night vision data is that vision information is constrained into a much smaller range as demonstrated in Fig 3. Therefore, existing approaches, which are suitable for general lighting condition, are not feasible to extract enough edge information due to low differential degree among pixel datum in each frame. Although threshold mechanism is utilized as high and low pass filters to avoid noises or redundant data, the high / low pass filters cannot exaggerate the differential level among each pixel, and thus they cannot further extract important edges of objects in the targeting scenes.

4 Proposed Sobel Square Edge Detectors

As data under low illumination is constrained into a smaller range, high pass filter may remove important information. In order to defeat the low data quality issue for night vision analysis, exaggeration on differential degree among data is proposed to obtain edge information from the original scenes. A simple illustration is given in Fig 8, which shows that low pass filter is not useful whereas high pass filter removes precious data under low illumination. The proposed approach is to increasing differential degrees among data for recovering information under general lighting condition. The proposed edge detector is adapted from Sobel kernels. The original G_x and G_y are enlarged to A_x and B_y . In addition, $f(x,y)$ is also magnified into $F(x,y)$. Resulting images with existing approaches and proposed method are presented in Fig. 9 whereas the original image is shown in Fig. 9. It is found that the image processed from the proposed method obtains more highlighting on important edges with lowest production on redundant information. The algorithm is equipped with equation 12, 13 and 14.

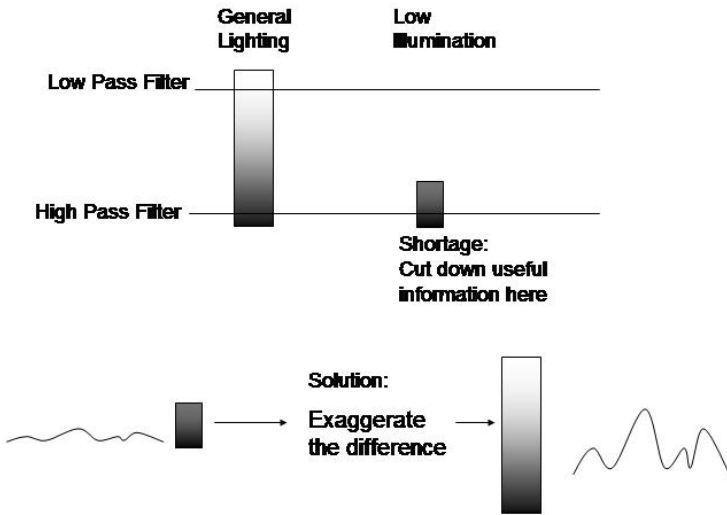


Fig. 8. Thresholds versus Exaggeration on Differences

4.1 Proposed Method

$$Ax(x, y) = f(x + 1, y - 1) + 2 * f(x + 1, y) + f(x + 1, y + 1) - f(x - 1, y - 1) - 2 * f(x - 1, y) - f(x - 1, y + 1) \tag{12}$$

$$By(x, y) = f(x - 1, y + 1) + 2 * f(x, y + 1) + f(x + 1, y + 1) - f(x - 1, y - 1) - 2 * f(x, y - 1) - f(x + 1, y - 1) \tag{13}$$

$$F(x, y) = \frac{Ax(x, y)^2 + By(x, y)^2}{8} \tag{14}$$


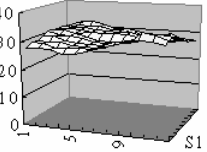
| | | | |
|----------------|---|---|---|
| Original Image |  | Sample Pixels: Pixel(X,Y), where $98 < X < 109$ $119 < Y < 132$ |  |
|----------------|---|---|---|

Fig. 9. Original Infrared Content


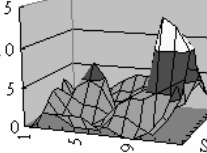
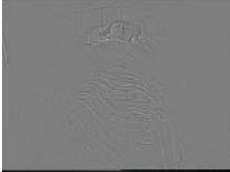
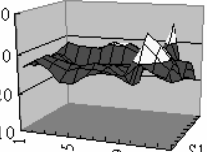

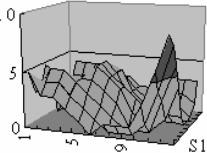

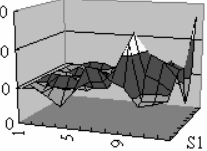

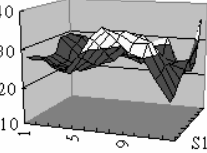

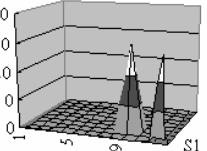
| | | | |
|------------------------------|---|---|---|
| Sobel Kernels |  | Sample Pixels: Pixel(X,Y), where $98 < X < 109$ $119 < Y < 132$ |  |
| Laplacian |  | Sample Pixels: Pixel(X,Y), where $98 < X < 109$ $119 < Y < 132$ |  |
| Prewitt Kernels |  | Sample Pixels: Pixel(X,Y), where $98 < X < 109$ $119 < Y < 132$ |  |
| Kirsch Compass Kernels |  | Sample Pixels: Pixel(X,Y), where $98 < X < 109$ $119 < Y < 132$ |  |
| Simple Edge |  | Sample Pixels: Pixel(X,Y), where $98 < X < 109$ $119 < Y < 132$ |  |
| New Edge Detector |  | Sample Pixels: Pixel(X,Y), where $98 < X < 109$ $119 < Y < 132$ |  |

Fig. 10. Resulting Images from Different Edge Detectors

4.2 Resulting Images

In this section, an original infrared image is displayed in Fig. 9 and a portion of the image is extracted and presented in a 3D chart, containing coordinates of X and Y and the pixel value of points. The image is in grayscale format, and hence the value of individual pixel is valid from 0 to 255. Moreover, the resulting images processed by different edge detectors are shown in Fig. 10. The experimental results demonstrate that the proposed new edge detector is able to effectively extract edges and reduce redundant information.

4.3 System Performance

Image processing is the preliminary step for computer vision analysis, which may include more heavy computation tasks such as pattern recognition. Therefore, the preliminary process should not become the burden of overall performance both for the concerns of computation speed and CPU consumption. The system is tested over a pc with CPU P4-2.4G and 256M Ram. All algorithms are implemented in C#, a high level language for fast development. The proposed real time sobel square is able to process 30 frames per second for real time application. For a detailed examination, it costs 0.03 second to process a 320*240 frame. In comparison, existing approaches such as Sobel or Kirsch after optimization spend 0.3 second to process each frame. As a result, the proposed edge detector is suitable for applications, which take system performance as first priority. The reason why the proposed method is ten times faster than the conventional sobel detector is because the square root is removed and the number of division is reduced.

5 Conclusion and Future Development

The aim of this work is to provide a more robotic image processing technique in computer vision, removing constraints on night vision analysis. The proposed new edge detector appears to improve video analysis result by assisting night vision capability for diagnosis in obstructive sleep apnoea [2]. In future development, we hope that the proposed method will be applied and examined in more wide usages like surveillance system. Also, more generic algorithms are expected to be built for obtaining stable performance invariant to changes on environmental variables.

Acknowledgement

Ching Wei Wang obtains her PhD scholarship jointly funded by United Lincolnshire Hospital NHS Trust and University of Lincoln.

References

1. Lukac, R., Plataniotis, K.N., Venetsanopoulos, A.N., Bieda, R., Smolka, B.: "Color Edge Detection Techniques", *Signaltheorie und Signalverarbeitung, Akustik und Sprachakustik, Informationstechnik*, W.E.B. Universitat Verlag, Dresden, vol. 29, pp. 21-47, 2003

2. Wang, C.W., Ahmed, A., Hunter, A: "Robotic Video Monitor with intelligence for diagnosis on Obstructive Sleep Apnoea". (Unpublished)
3. Lipton, A. J., Heartwell , C. H., Haering, N., and Madden, D.: "Critical Asset Protection, Perimeter Monitoring, and Threat Detection Using Automated Video Surveillance", Object Video white paper.
4. Gavrilu D.M., "The visual analysis of human movement – a survey, computer vision and image understanding", vol. 73, no.1, pp.82-98, 1999
5. Zhou, J., Hoang, J.: "Real Time Robust Human Detection and Tracking System", IEEE-Computer vision and pattern recognition, Volume: 3, On page(s): 149- 149, 2005
6. Desmond C. Adler, Tony H. Ko, Paul R. Herz, and James G. Fujimoto: "Optical coherence tomography contrast enhancement using spectroscopic analysis with spectral autocorrelation", Vol. 12, No. 22 / OPTICS EXPRESS 5487, 2004

A New Video Images Text Localization Approach Based on a Fast Hough Transform

Bassem Bouaziz¹, Walid Mahdi², and Abdelmajid Ben Hamadou²

MIRACL, Multimedia InfoRmation system and Advanced Computing Laboratory
Higher Institute of Computer Science and Multimedia,
Sfax, BP 1030 - Tunisia 69042
Bassem.bouaziz@fsegs.rnu.tn,
{walid.mahdi, benhamadou.abdelmajid}@isims.rnu.tn

Abstract. The amount of digital video data is increasing over the world. It highlights the need for efficient algorithms that can retrieve this data by content. The full use of this media is currently limited by the opaque nature of the video which prevents content-based access. To facilitate video indexing and browsing, it is essential to allow non-linear access, especially for long programs. This can be achieved by identifying semantic description captured automatically from video story structure. Among these descriptions, text within video frames is considered as rich features that enable a good way for browsing video. In this paper we propose a fast Hough transformation based approach for automatic video frames text localization. Experimental results, that we drove on a large sample of video images issued from portions of news broadcasts, sports program, advertisements and movies, shows that our method is very efficient, capable to locate text regions with different character sizes, directions and styles even in case of complex image background.

1 Introduction

Automatic text regions detection in digital images is an indispensable preprocessing step within the framework of multimedia indexing and video structuring. Content-based video indexing aims at providing an environment both convenient and efficient for video storing and retrieving [2], especially for content-based searching as this exists in traditional text-based database systems. Most of research works have been focused on the extraction of the structure information or simple spatial-temporal based features within a video, such as the basic shot boundaries detection [3], color histogram [5], texture or movement, for indexing purpose. However, besides these basic features which are important to characterize the video content, one has also to segment or provide underlying semantic descriptors in order to fully enable intelligent video content access.

In our last work, we have made use of the video syntax (special effect, rhythm, specific syntax for news) to drive some semantic hierarchical video structures embedded within a video [6] [7]. Unfortunately, such a semantic hierarchical video structure, described as shots, scenes and sequences does not provide enough the semantic

description of each segmented units. One automatic way, to reach more semantic description, is to segment texts which often occur in video images, especially for Film fictions, TV news, Sports or Advertisement. For instance, most of movies begin with credit titles, and when a jump occurs in the film narration, it is often indicated by textual terms, called ellipses, such as “5 months before” or “3 years latter”. TV news makes an extensive use of textual information to indicate the subject of each topic, the name of interviewee, the place of report, etc. Thus textual information when embedded within a video are very high level clues for the content-based video indexing and it is generally associated to an important event. In this paper, we propose an efficient automatic text regions locating technique for video images. These text regions resulted from our technique can be then passed to a standards OCR in order to obtain ASCII texts. The remainder of the paper is organized as follows. In section 2, we briefly review related works in the literature, emphasizing strengths and drawbacks of each technique. In section 3, we summarize text regions features. Section 4 details our proposed method to extract these text regions features (ie. directionality, regularity, alignment). In section 5, we describe the use of this method to locate text within video frames. In the last section we give exhaustive experimentations.

2 Related Work

Text detection in real-life videos and images is still an open problem. The first approach proposes a manual annotation [9]. Clearly manual-based solution is not scalable as compared to the amount of video programs to be indexed. Current automatic text detection methods can be classified into three categories. The first category is connected components-based methods [4] [10] [11], which can locate text quickly but have difficulties when text embedded in complex background or touches other graphical objects. For example, Zhong et al [10] extracted text as those connected components of monotonous color that follow certain size constraint and horizontal alignment constraint. The second category is texture based [12] [13] [14]. This category is hard finding accurate boundaries of text areas and usually yields many false alarms in “text like” background texture areas. Indeed, texture analysis-based methods could be further divided into top-down approaches and bottom-up approaches. Classic top-down approaches are based on splitting images regions alternately in horizontal and vertical direction based on texture, color or edge. On the contrary, the bottom-up approach intends to find homogenous regions from some seed regions. The region growing technique is applied to merge pixels belonging to the same cluster [15] [16]. Some existing methods do solve the problem to a certain extent, but, not perfectly. The difficulty comes from the variation of font-size, font-color, spacing, contrast and text language, and mostly the background complexity. The third category is edge-based methods [8][17][18][19]. Generally, text regions can be decomposed by analyzing the projection profiles of edge intensity maps. However, this kind of approaches can hardly handle large-size text. As conclusion, the three categories methods presented above are limited to many special characters embedded in text of video frames, such as text size and the contrast between text and background in video images. To detect the text efficiently, all these methods usually define a lot of rules that are largely dependent of the content of video. Unlike other approaches, our approach

has nor a restriction on the type of characters neither on the place where text regions should appear within an image. Our text region detection technique is based on the basic features of texts appearing in digital videos such alignment and orientation.

3 Text Region Characteristics

Textual information in audiovisual program can be classified into two kinds: natural text which appears as a part of the scene (e.g. street names or shop names in the scene), and artificial text which is produced separately from the video shooting and inserted into the scene during the post-processing step by video title machines. Both of them, when they occur within a video program, are of capital importance and they are good clues for content-based indexing and retrieval. However, by the opposition to the natural text which accidentally appears in the scene, the inclusion of artificial text is carefully selected, and thus is subject to many constraints so that the video program is easily read by viewers. Below we summarize the main characteristics of these constraints by the following features:

- Text characters are in the foreground.
- Text characters contrast with background since artificial text is designed to be read easily.
- Text characters can be monochrome or not.
- Text characters are generally upright but it can appear in different direction.
- Text character has size restrictions; character size should not be smaller than a certain number of pixels otherwise they are illegible to viewers.

Our method makes use of these basic features to localize text regions in video images. It also takes into account the characteristics of video programs, such as the low resolution, presence of noises, and absence of control parameters in the cameras. In our approach we do not use a filtering method to reduce image noises because it may cause problems for the detection of the characters having a very small size. Actually, the majority of text region detection methods in the literature support the hallucination of some text regions which is more acceptable than the false detection. Making use of these basic features, the experimentation that we have conducted on various types of video programs shows that our technique is robust for contrast, font-size, font-color, background complexity and direction. Besides we also note in our experimentation that our method decreases the number of false detections (false alarm).

4 Proposed Approach

To locate text within a video frame we propose a new method offering both accuracy and rapidity. This method is based on local application of Hough transform. Local application of Hough transform minimizes influence of noise provoked by isolated pixels. This influence is insignificant in segment detection context.

Textual information embedded within an image present in general a regular signature. This signature can be described, on the one hand, by vertical line segment

separated by similar distance (directionality) and on the other hand, by the alignment of these segments (orientation)[1].

In the following we will detail our approach which is adapted to textual information features extraction (ie. regularity and directionality). This approach presents four steps: Sweeping the image, detecting segments, storing information about these segments and detecting regularity.

4.1 Image Sweeping

Initially we studied access manners to line segment pixels appearing in an edge image. Let S be a set of collinear pixels forming a line segment within an image. A simple and rapid way to get extremities of a given segment is to sweep sequentially the image from top to bottom and from left to right. In this case, the angle θ formed by a given segment with the horizontal axis verify the inequalities, $0 < \theta < \Pi$. When a line segment is detected, stored and removed from the image, the sequential search continues until sweeping the whole image.

4.2 Segment's Detection

When a line segment extremity is reached, we sweep all directions θ between 0 and Π to find direction where most connected pixels exists. In all cases, when the number of connected pixels is important or exceed a given threshold information concerning the segment (r, θ) are stored and the segment is removed from the image (figure 1). In other cases it's considered as noise (isolated pixels, short segments) and it's simply removed.

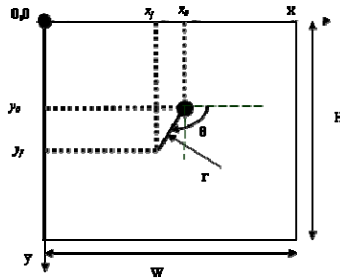


Fig. 1. Features segment representation

The detection of segment direction is the most costly step in line detection algorithm, so requires a particular attention.

In the discrete image space, line segment pixels having a direction θ , length r and beginning coordinates (x_0, y_0) can be represented by:

$$P(X_{0..r,0..0}, Y_{0..r,0..0}) = P(x_0 + E(m \cdot \cos(\theta)), (y_0 + E(m \cdot \sin(\theta))) . \tag{1}$$

Where $1 \leq m \leq r$, $0 < \theta < \Pi$ and E is the entire part of a number.

In order to improve performances and avoid call of trigonometric functions inside our algorithm, we compute two transformation matrixes in the initialization step.

$TX_{0...r,0...0}$ (table 1) and $TY_{0...r,0...0}$ (table 2).

Table 1. Transformation matrix $TX_{0...r,0...0}$

| | | |
|----------------|---------|------------------|
| 0 | ←-----→ | 180 |
| $E(r.\cos(0))$ | ... | $E(r.\cos(180))$ |
| ... | ... | ... |
| $E(2.\cos(0))$ | ... | $E(2.\cos(180))$ |
| $E(1.\cos(0))$ | ... | $E(1.\cos(180))$ |

Table 2. Transformation matrix $TY_{0...r,0...0}$

| | | |
|----------------|---------|------------------|
| 0 | ←-----→ | 180 |
| $E(r.\sin(0))$ | ... | $E(r.\sin(180))$ |
| ... | ... | ... |
| $E(2.\sin(0))$ | ... | $E(2.\sin(180))$ |
| $E(1.\sin(0))$ | ... | $E(1.\sin(180))$ |

We denote by r the maximal length of line segment that we should detect in a direction between 0 and 180°.

x_0, y_0 are the coordinates of line segment extremity identified when sweeping the image. So each element $X=x_0+TX_{0...r,0...0}$ and $Y = y_0+TY_{0...r,0...0}$ represents pixel coordinates in the image space.

4.3 Segment’s Information Storing

Let $V_{0...r,0...0}$ be an accumulator, which values are initialized to zero in a new line segment extremity detection process, Its content is updated as follows :

$$V_{0...r,0...0} = \begin{cases} 1 & \text{if}(I(X(m, \theta), Y(m, \theta))=1 \\ 0 & \text{if}(I(X(m, \theta), Y(m, \theta))=0 \end{cases}$$

Where $1 \leq m \leq r$, $0 < \theta < \Pi$

The obtained matrix $V_{0...r,0...0}$ represents neighborhood’s information of a detected extremity concerning connected pixels.

Starting from the assumption that some images are imperfect we propose an improvement of our approach. So, in the one hand we introduce a tolerance pace p_s that when exceeded we stop the computation in a given direction. In the other hand we introduce a neighborhood θ_v to a given direction. So an element of accumulator $V_{0...r,0...0}$ is set to 1 only if equation 2 is verified :

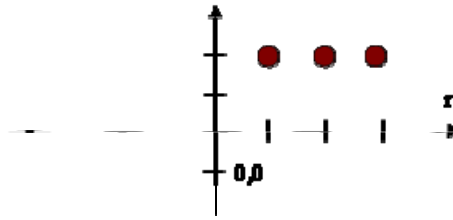


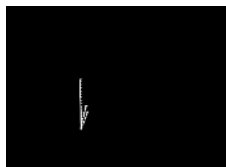
Fig. 2. Space parameter (r, θ) of a detected line segment

$$(V_{r-\rho_s, \theta==1}) \text{ or } (V_{r, \theta-\theta_v==1}) \text{ or } (V_{r, \theta+\theta_v==1}) . \tag{2}$$

This consideration of neighborhood makes our method able to detect imperfect segment as the case of an edge image.

4.4 Segment’s Removing

The last step of our method consists on removing segment’s pixels having length exceeding a threshold S_r . This threshold represents the minimal length of segment that we should detect. This step is fast because information about these segments were already stored in the accumulator $V_{0..r,0..\theta}$. Indeed an accumulator’s cell set to 1 represents the transform of an edge pixel in parameter space.



(a) initial image



(b) Detected segment

Fig. 3. Segment detection having direction $=86^\circ$ and length $r > S_r$

4.5 Detecting Segment’s Regularity

Let D' be the line passing through the origin (x_0, y_0) . This line is perpendicular to D which is the support line of the segment to be detected.

r' is the perpendicular distance from the origin to the line which supports the segment to be detected.

θ' is the angle between the norm of the line and the x axis.

So, for a given angle θ' we can determinate, for a given direction θ , all segments perpendicular to D' but having different values of r' (figure 4).

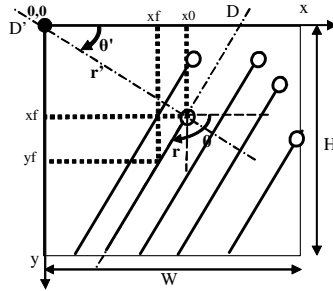


Fig. 4. Segments having same direction and different distances from origin.

The regularity can be detected if distance (Dps) between parallel segments is similar for a specified value of θ' . This representation is very important and makes possible responding for a query that must return regular texture with a given direction or with any direction.

To quantify regularity, first we should express r' and θ' in step with r and θ that we had stored in the accumulator.

For $\theta \in [0, \Pi]$ two cases are presented:

$$\begin{aligned} \text{If } \theta \in [0, \Pi/2] \text{ then } \theta' &= \theta + \Pi/2 \\ \text{If } \theta \in [\Pi/2, \Pi] \text{ then } \theta' &= \theta - \Pi/2 \end{aligned}$$

As for r' it can be written in all $[0, \Pi]$ as :

$$r' = |x_0 \sin(\theta) - y_0 \cos(\theta)| \tag{3}$$

5 Video Text Localization

Starting from hypothesis that a text is characterized by a regular texture and that characters forming a text line are oriented and aligned [1][9][10], we propose a new technique to detect text within video image.

The regularity can be described by a similar distance between edge segments having the same orientation, as for alignment, it needs another feature to be determinate. (Fig. 5 shows different categories of segments that can be occurred in an edge video frame).

Let $dist$ be the distance between the extremity point of a segment and the intersection point between the support line of the same segment and the perpendicular line passing from the origin. Aligned segment are those which have similar values of $dist$ where $dist$ can be presented as follow:

$$dist = |y_0 \sin(\theta) + x_0 \cos(\theta)| \tag{4}$$

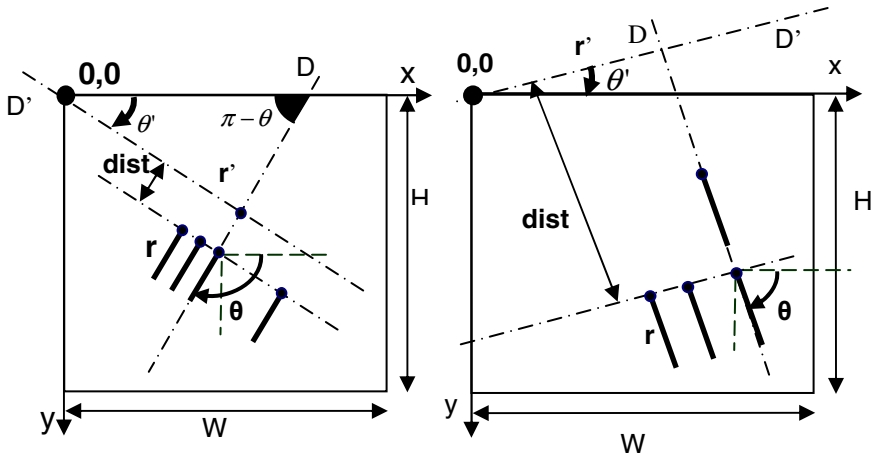


Fig. 5. Three-dimensional representation of segments (r' , θ' and $dist$)

Having a three-dimensional representation of segments within a video frame (r' , θ' and $dist$), we proceed by grouping aligned segments with similar θ' and $dist$ to obtain potential text regions.

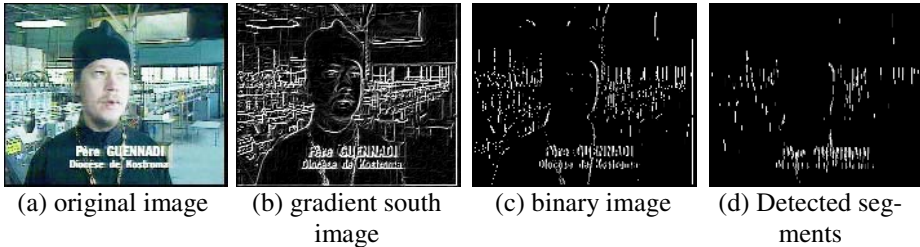


Fig. 6. Detection of segments within a video frame

In order to localize effective text regions, these regions will be filtered according two criteria:

- distance (Dps) between aligned segments and,
- minimal number of segments (ns) forming a text line

A result of such filtering process is shown by fig. 7.



Fig. 7. Localization text regions after filtering process with $2 < Dps < 6$ and $ns=6$

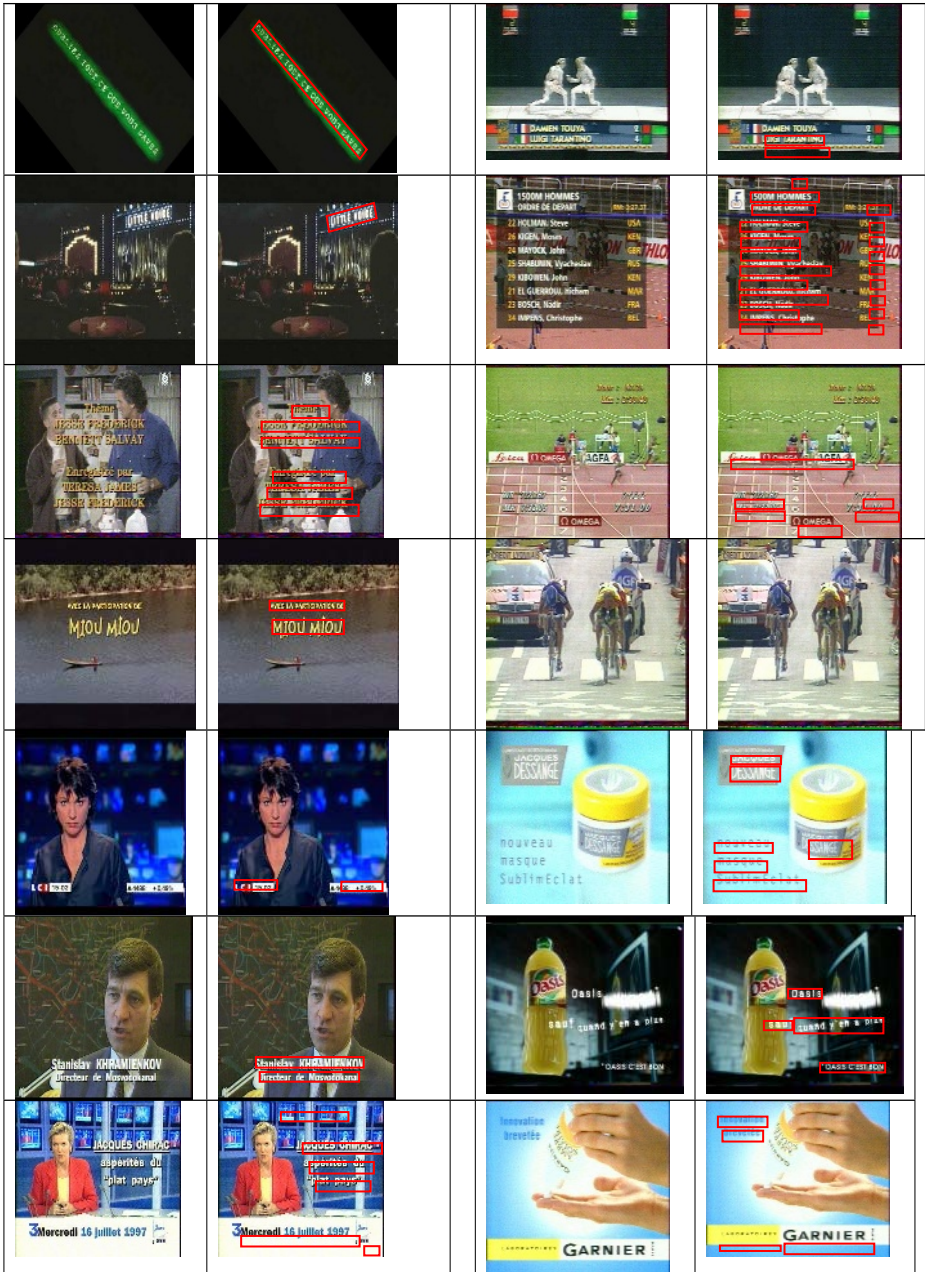


Fig. 8. Illustration of the video frames experimentation database (movies, news, sports and advertisement)

6 Experimental Result

Our test consists on ten video sequences totaling 13259 frames. The sequences were captured at 25 frames per second with a 352x240 frame size. The sequences are portions of news broadcasts, sports program, advertisements and movies. The test database is challenging due to the poor quality and low contrast of these broadcasts. Text appears in a variety of colors, sizes, fonts, orientation and background complexity.

All regions distinguishable as text by humans were included in the experimentation process. Closely spaced words lying along the same horizontal were considered to belong to the same text instance. Test data contains a total of 280 temporally unique text instances of which 205 are artificial text instances and 75 are scene (or natural) text instances.

During evaluation, each text region in the test database is placed into one of three categories:

- Detection: regions identified as text by the localization algorithm.
- False Alarm: regions identified by the detection algorithm but not belonging to text regions.
- Missed Detection: Pixels belonging to the text regions and not identified by the algorithm.

The performance of an algorithm is quantified by its recall and precision, given by equations 5 and 6.

$$\text{Recall} = \frac{\text{detections}}{\text{detections} + \text{missed detection}} \quad (5)$$

$$\text{Precision} = \frac{\text{detections}}{\text{detections} + \text{False alarms}} \quad (6)$$

To evaluate the performance of our algorithm, F1 measure given by Equation 7, was computed.

$$\text{F1} = 2 * \text{precision} * \text{recall} / (\text{precision} + \text{recall}) \quad (7)$$

Table 3. Text Localization Performance

| | Recall | Precision | F1 |
|-----------------|--------|-----------|--------|
| Scene text | 84,2% | 85,71% | 84,95% |
| Artificial text | 97,05% | 93,83% | 95,42% |

Table 3 shows that the overall results of our method were 97,05% Recall and 93,83% Precision for artificial text and 84,2% Recall and 85,71% Precision for scene

text. According to these results it is clear that more enhanced images processing techniques can be applied based on the characteristics of scene text detection characterized by low contrast and poor quality of video.

7 Conclusion

For automatic video browsing purpose we have introduced a new text regions localization technique in digital video frames. This method, based on fast Hough transform, uses the basic features of text appearance. It provides both speed and effectiveness.

The experimentation, that we have driven on a large set of video images selected from various kinds and genres, shows that our technique is very efficient, capable to locate text regions with different character sizes, styles and orientation, even in case of texts occurring within complex image background.

Moreover, many applications can be derived from this automatic text locating technique. For instance, automatic extraction of credit titles, automatic generation of video content table and exploration of other spatial clues to define other techniques enabling further video browsing and retrieval. In this context we work on complete tools as a new prototype product for video structuring and indexing.

References

- [1] B. bouaziz, W. Mahdi, M. Ardebilian, A. Ben Hamadou, " A new approach for texture features extraction : application for text localization in video images ", Proceedings of the IEEE International Conference on Multimedia and Exposition, ICME06, July 2006, Ontario, Canada.
- [2] Ph. Aigrain, Ph. Joly et V. Longueville, Representation based user interfaces for the Audiovisual Library of Year 2000", Proceedings of IS&T/SPIE Conference on Multimedia Computing and Networking, pages 35-45, 1995.
- [3] M. Ardebilian, X. W. Tu, L. Chen, Improvement of Shot Detection Methods Based-on Dynamic Threshold Selection, Proc. SPIE : Multimedia Storage and Archiving Systems II, Dallas, USA, 1997.
- [4] K. Jung, H. Han, Hybrid approach to efficient text extraction in complex color images Pattern Recognition Letters, Edition Elsevier, Volume 25, Issue 6, April 2004, ISSN: 0167-8655. pp. 679-699.
- [5] M. Swain and D. Ballard, Color Indexing, International Journal of Computer Vision, Vol. 7 (N° 1, 1991), pages 11-32.
- [6] W. Mahdi, M. Ardebilian, L. Chen, Automatic Video Content Parsing Based on Exterior and Interior Shots Classification. In the seventh International conference on ADVANCED COMPUTER SYSTEMS, ACS'2000 October 23-25, 2000, ISBN 83-87352-24-7, Szczecin, POLAND, pp. 571-578.
- [7] W. Mahdi, M. Ardebilian, L. Chen, Automatic Video Scene Segmentation based on Spatial-temporal Clues and Rhythm, in International Journal of Networking and Information Systems . EDITION HERMES, Vol N°3, n° 1/2000, pp 27-51, ISSN 1290-2926, ISBN 2-7462-0288-3, <http://www.hermes-journals.com>.

- [8] Lyu, J. Song, M. Cai. A comprehensive method for multilingual video text detection, localization, and extraction, *Circuits and Systems for Video Technology*, IEEE Transactions on Volume 15, Issue 2, Feb. 2005 pp.243 - 255
- [9] D. Marc, Media stream : Representing Video for Retrieval and Repurposing, ACM Multimedia proceeding, page 478-479, sans Fransisco, CA, ,USA, octobre 15-20, 1994.
- [10] A.K. Jain and B. Yu, Automatic Text Location In Images and Videos Frames, *Pattern Recognition*, Vol. 31, N° 12, pp. 2055-2076, 1998.
- [11] Y. Zhong, K. Karu and A.K. Jain, Locating text in complex color images, *Pattern Recognition*, 28(10):1523- 1535, 1995.
- [12] H. Li, D. Doermann, and O. Kia, Automatic Text Detection and Tracking in Digital Video, *IEEE Trans, Image Processing*, Vol. 9, N°1, Jan 2000.
- [13] Y. Zhong, H. Zhang, and A. K. Jain, Automatic Caption Extraction of Digital Video, *Proc IICIP'99, Kobe*, 1999.
- [14] T. Sato and T. Kanade, Video OCR: Indexing digital news livrairies by recognition of superimposed caption“, *ICCV Workshop on Image and Video retrieval*, 1998.
- [15] V. Wu, R. Manmatha, and E.M. Riseman, Finding text in images, in *Proc of the 2nd Intl., Conf on Digital Librairies*. Philadalpha, PA, pp 1-10, Juuly 1997.
- [16] K. Sobotka, H. Bunke, and H. Kronenberg, Identification of Text on Colored Book and Journal Covers, in *Proc of the 5th Intl., Conf on Document Analysis and Recognition*, pp. 57-62, 1999.
- [17] T. Sato, T. Kanade, E. Hughes, and M. Smith, Video OCR for digital News Archives, *IEEE International Workshop on Content-Based Access of Images and Video Databases*, pp. 52-60, January, 1998.
- [18] W. Qi, et al., Integrating Visual, Audio and Text Analysis for news Video, *7th IEEE International Conference on Image Processing (ICIP2000)*, Vancouver, British Columbia, Canada, 10-13 September 2000.
- [19] Y. Hao, Y. Zhang, H. Zeng-guang and T. min, Automatic Text Detection In Video Frames Based on Bootstrap Artificial Neural Network And CED, in *Journal of WSCG*, Vol.11, N°1., ISSN 1214-6972 WSG'2003, February 3-7, 2003, Plzen, Czech Republic. Copyright UNION Agency – Science Press.
- [20] C. Wolf , J.M Jolion and F. Chassaing. Text Localization, Enhancement and Binarization in Multimedia Documents. In *Proceedings of the International Conference on Pattern Recognition (ICPR) 2002*, volume 4, pages 1037- 1040, IEEE Computer Society. August 11th-15th, 2002, Quebec City, Canada

Video Sequence Matching Using Singular Value Decomposition

Kwang-Min Jeong¹, Joon-Jae Lee^{2*}, and Yeong-Ho Ha³

¹ Dept. of Information Communication, Kyungnam College of Information & Technology,
Busan, 617-701, South Korea
kmjeong@kit.ac.kr

² Dept. of Computer and Information Engineering, Dongseo University,
Busan, 617-716, South Korea
jjlee@dongseo.ac.kr

³ School of Electrical Engineering and Computer Science, Kyungpook National University,
Daegu, 702-701, South Korea
yha@ee.knu.ac.kr

Abstract. This paper proposes a novel signature based on singular value decomposition (SVD) for video sequence matching. By considering the input image as a matrix, a partition procedure is first performed to separate the matrix into non-overlapping sub-images of a fixed size. The SVD process then individually decomposes each partitioned sub-image into a singular value and the corresponding singular vector factorization. As a result, several dominant singular values are obtained for each sub-image, allowing the dissimilarity to be determined between the reference video clip and the query one. Experimental results based on receiver operating characteristics (ROC) curves confirm the effective performance of the proposed video signature.

1 Introduction

Recently, the most widely used technique for content-based copy detection is a sequence matching approach, where multiple sequence frames are used as the basis for matching, as opposed to matching single video frames. Features like intensity rankings and color histograms are extracted from the original video frames to create the reference signatures in a database. The same feature, extracted from the query video sequences, is then matched to the reference signatures to determine if the query video sequence is a copy of the original. Related work includes the following.

Jain et al. [1] proposed a sequence matching method based on a set of key frames. Although motion information is included with the key frames, it is not yet clear. Meanwhile, Naphade et al. [2] proposed an algorithm for matching video clips that uses the histogram intersection of YUV histograms of the DC sequence of the MPEG video. However, this technique does not evaluate variations between copies, such as signal modifications or display format conversions. Mohan used

* This work was supported by grant No. R01-2004-000-10851-0 from Ministry of Science & Technology.

the ordinal measure originally proposed by Bhat et al. for video sequence matching [3,4], then Hampapur et al. compared the ordinal measure technique with techniques using a motion signature and color signature [5], and showed that matching based on an ordinal signature produced the best performance. The ordinal measure is not sensitive to intensity value changes and also has a low memory requirement for storing/indexing the signature.

Singular value decomposition (SVD) is already known for its ability to derive a low-dimensional refined feature space from a high-dimensional raw feature space, and capturing the essential structure of a data set within a feature set, and several studies have already focused on the use of SVD for texture analysis in image processing. For example, Luo and Chen utilized SVD for texture discrimination [6], where the proportion of the two dominant singular values of an image matrix was used as the textural feature to discriminate texture images

Accordingly, this paper proposes a novel signature based on SVD for video sequence matching. By considering the input image as matrix A , the SVD process decomposes the image into an singular value and the corresponding singular vector factorization, where the singular values represent the energy of matrix A projected on each subspace. The singular values and their distribution carry useful information about the contents of A , and can vary drastically from image to image, which makes the singular values suitable for discriminating image patterns or contents. For most images, only a very few number is enough as features because a few larger singular values dominate, and yet all others close to small values. The similarity between two video clips can be treated as the similarity between the shapes represented by singular values of two corresponding hyper-ellipses, if the orientation is not considered. Therefore, the distance or dissimilarity metric between image frames can be represented by the Euclidian distance of the singular values. Experimental results demonstrate the matching performance in comparison with the ordinal measure.

The rest of the paper is organized as follows. Section 2 describes the properties of the singular value as an image signature, then Section 3 provides a detailed explanation of the video sequence matching algorithm using the proposed feature. Experimental results are presented in Section 4, and some final conclusions given in Section 5.

2 Signature Extraction

2.1 Singular Value Decomposition (SVD)

Let an image of size $M \times N$ be image matrix A of dimensions $M \times N$, where $M \geq N$. It is possible to represent this image in the r -dimensional subspace, where r is the rank of A , and $r \leq N$. Singular value decomposition (SVD) is then a factorization of matrix X into orthogonal matrices.

$$A = USV^T \quad (1)$$

where U is an $M \times r$ matrix and consists of the orthonormalized singularvectors of $A^T A$, and S is an $r \times r$ diagonal matrix consisting of the ‘singular values’ of A , which are the non-negative square roots of the singular values of $A^T A$. These singular values, denoted by $\lambda_i, i=1, 2, \dots, r$, are sorted in a non-increasing order, i.e.

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_r \geq 0 \tag{2}$$

An important property of U and V is that they are mutually orthogonal. The singular values (λ) represent the importance of individual singular vectors in the composition of the matrix. In other words, the singular vectors corresponding to large singular values include more information about the matrix than the other singular vectors.

2.2 Singular Value as Image Signature

The singular values represent the energy of matrix A projected on each subspace, where the singular values and their distribution carry useful information about the contents of V , and can vary drastically from image to image. For most images, only a very few larger singular values dominate, while all the other singular values are quite small. Figures 1 and 2 show some sample images and their corresponding 10-dominant singular values, respectively. The distributions of the singular values for the images are quite different, even though they have similar content patterns.



Fig. 1. The example images: image taken from (a) movie, (b) TV drama, (c) golf

One of the important characteristics for an image signature is robustness to media transform such as image size variation and compression format change. This is simply achieved by normalizing the singular value,

$$\sigma_i = \frac{\lambda_i}{\sum_{i=1}^r \lambda_i}, \quad i = 1, 2, \dots, r \tag{3}$$

where σ_i is the i th normalized singular value and λ_i is the i th singular value.

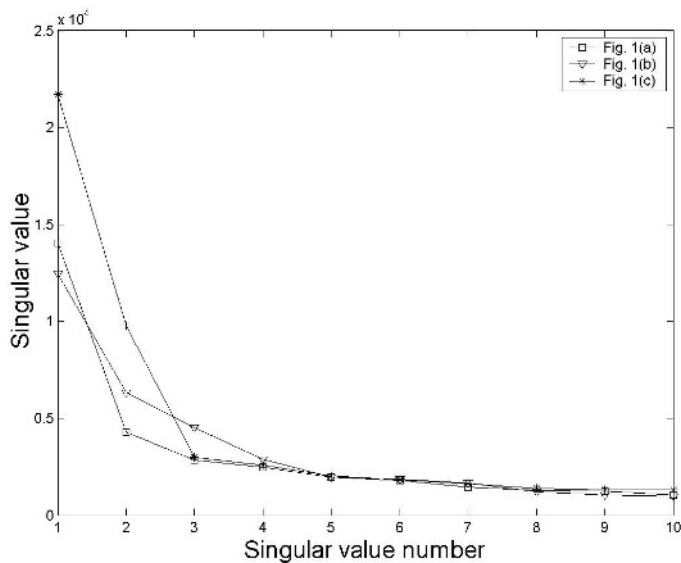


Fig. 2. The 10 largest singular values for sample images of Fig. 1

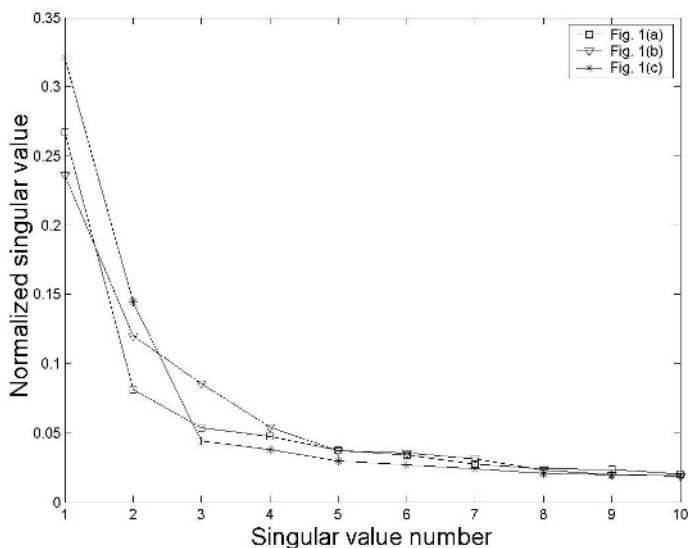


Fig. 3. The 10 largest normalized singular values for sample images of Fig. 1

As shown in Figure 3, the magnitude of the first singular value is 0.32, which means it includes 32% of the image information. These singular values are sorted in a non-increasing order using Eq.(2) so they can be used as a signature in order of importance. In the proposed system, spatial information on an image frame is incorporated by dividing each frame into 3×3 blocks, and calculating the normalized singular values for each block.

2.3 Distance Measure

Assuming that v_i is one of the singular vectors V and λ_i is the corresponding singular value, then v_i determines the orientation of one semi-axis of the hyper-ellipse and λ_i measures the length of the corresponding axis. As such, the shape and orientation of the hyper-ellipse is a description of the characteristics of the image. Figure 4 illustrates two different hyper-ellipses in a 2-dimensional plane as an example. As a result, the similarity between two video clips can be treated as the similarity between the shapes representing by singular values of two corresponding hyper-ellipses, if the orientation is not considered.

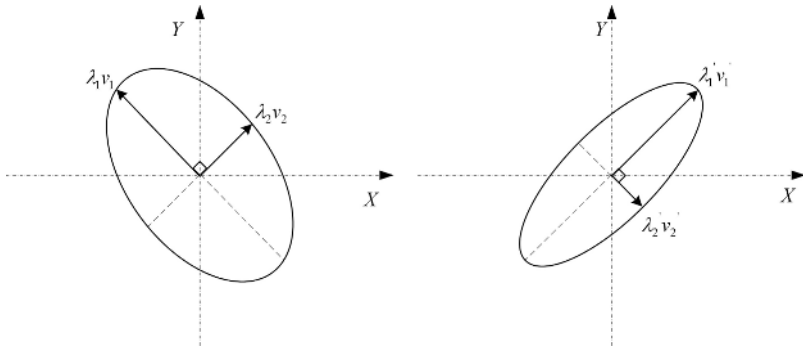


Fig. 4. Illustrations of two 2-D hyper-ellipses with two dominant feature vectors

Therefore, the above description leads to the following distance or dissimilarity metric between image frame A and frame A' .

$$Dist_{nor}(A, A') = \sum_{i=1}^r |\sigma_i - \sigma'_i| \tag{4}$$

3 Video Sequence Matching

3.1 Construction of the Proposed Signature and Distance Measure

A video is composed of continuous image frames. Therefore, a video sequence clip with n frames is denoted by $C = \{C[1], C[2], \dots, C[n]\}$, and the i th frame with

m partitions can be expressed as follows $C[i] = \{C^1[i], C^2[i], \dots, C^m[i]\}$, where C^j denotes the sequence of the j th partition. A sub-video of C is also defined as $C[p : p + N - 1]$, where the number of frames is N and the first frame is $C[p]$, $1 \leq p \leq n - N - 1$.

Let the normalized singular value matrix for the i th frame of the query video clip $C_q[i]$ be

$$\sigma_{q,1}(i), \sigma_{q,2}(i), \dots, \sigma_{q,k}(i), k = 1, \dots, r \tag{5}$$

and if the frame has m partitions, the normalized singular value matrix for the j th partition of the i th frame $C_q^j[i]$ can be defined as

$$\sigma_{q,1}^j(i), \sigma_{q,2}^j(i), \dots, \sigma_{q,k}^j(i), j = 1, \dots, m \tag{6}$$

The singular value distance between the i th frame of the reference video sequence $C_r[p : p + N]$, $C_r(p + i)$, $1 \leq i \leq N$ and the i th frame of the query video clip $C_q[i]$ can then be defined as

$$d(C_q[i], C_r[p + i]) = \sum_{j=1}^m \sum_{k=1}^r \left| \sigma_{q,k}^j[i] - \sigma_{r,k}^j[p + i] \right| \tag{7}$$

As such, the dissimilarity between the two sequences $D(C_q, C_r[p : p + N - 1])$ is computed by averaging over N dissimilarities, i.e.

$$D(C_q, C_r[p : p + N - 1]) = \frac{\sum_{i=1}^N d(C_q[i], C_r[p + i])}{N} \tag{8}$$

3.2 Matching Algorithm

To deal with an image size variation, the small sized image is used as the reference and the large sized image is adjusted by down-sampling. Given a query video clip C_q with N frames, the reference video sequence C_r , C_q is compared to the sub sequence $C_r[p : p + N - 1]$. The detailed matching procedure is as follows.

- Step 1) Set query clip length N
- Step 2) Select query clip V_q from random point in query video C_q with length N .
- Step 3) Set p to be 1.
- Step 4) Compute the distance between two sequences, C_q and $C_r[p : p + N - 1]$
- Step 5) Increase p by 1. Repeat step 4 until $p = n - N$, where n is a number of frames in the reference video.

- Step 6) From the sequence of distances, find best match location.
- Step 7) Repeat steps 1-6, 100 times.

4 Experimental Results

This section first introduces ordinal measure which we want to compare with proposed signature, and then we address the matching performance of the proposed method.

4.1 Comparisons with Ordinal Measure for Artificial Images

The ordinal measure was originally proposed by Bhat et al. for computing image correspondence [3], then Mohan applied the ordinal measure to video sequence matching [4]. The ordinal feature is obtained as follows. The video frame is partitioned into $N = N_x \times N_y$ equal-sized blocks (Mohan used $N_x = N_y = 3$) and the average gray level in each block for N sub-image is computed. The set of average intensities is then sorted in an ascending order and a rank assigned to each block using integer 1 to N . The main advantages of the ordinal measure are a short processing time, small memory space requirement, and robustness to intensity changes.

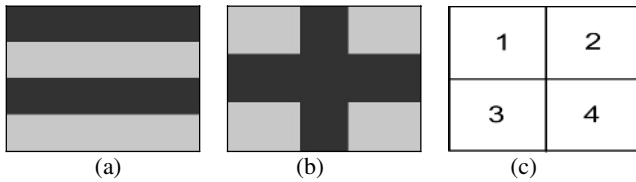


Fig. 5. Sample images for comparison of ordinal and singular values, (a) two horizontal lines image with gray level 200 and 50, (b) cross lines image with gray level 200 and background with gray level 50, (c) sub-block number of images

However there exist many real video or movie scenes with same ordinal feature even though they have entirely different contents or structure of image. Fig. 5(a) and (b) show examples that the ordinal feature fails in artificial images in which they have same average intensity value of block pixels irrespective of different structures or contents, while the proposed system has a good discrimination power due to the different singular values corresponding to different structural information as shown in Table 1.

Table 1. Ordinal and the proposed features for different contents

| #block | Ordinal | | | | Singular values | | | |
|--------|-----------|------|-----------|------|-----------------|------------|------------|------------|
| | Fig. 5(a) | | Fig. 5(b) | | Fig. 5(a) | | Fig. 5(b) | |
| | mean | rank | mean | rank | σ_1 | σ_2 | σ_1 | σ_2 |
| 1 | 125 | 1 | 125 | 1 | 1 | 0 | 0.9325 | 0.0675 |
| 2 | 125 | 2 | 125 | 2 | 1 | 0 | 0.9325 | 0.0675 |
| 3 | 125 | 3 | 125 | 3 | 1 | 0 | 0.9325 | 0.0675 |
| 4 | 125 | 4 | 125 | 4 | 1 | 0 | 0.9325 | 0.0675 |

4.2 The Performance Measure for Matching Results

The performance of the proposed algorithm was plotted based its receiver operating characteristics (ROC) curve, which is a plot of the false positive rate (FNR) versus the false negative rate (FNR). Let N_T be the total number of match tests conducted, with FN the number of false negatives (clips that should have been matched, yet were not) and FP the number of false positives (clips that were matched that were not part of the reference set, as another video was used for the FNR). Thus, the FNR and FPR were as follows:

$$FNR(\tau) = \frac{FN}{N_T}, \quad FPR(\tau) = \frac{FP}{N_T} \quad (8)$$

Where τ is the normalized threshold value varying from zero to one. The ROC curves were computed by varying τ . A good ROC curve lies very close to the axes, while an ideal curve pass through (0,0), i.e. zero FNR and zero FPR.

4.3 Matching Results for Real Video Sequence

An MPEG-1(320×240) video reference sequence with 200,000 frames was used that included a variety of sub sequences: movie, football, golf, CF, and TV drama. The query video sequence was derived from MPEG-1(160×120) encodings of the reference video and a home video composed of 51,000 frames. To adjust the dimensions of the video clip, the reference clip was reduced to 160×120.

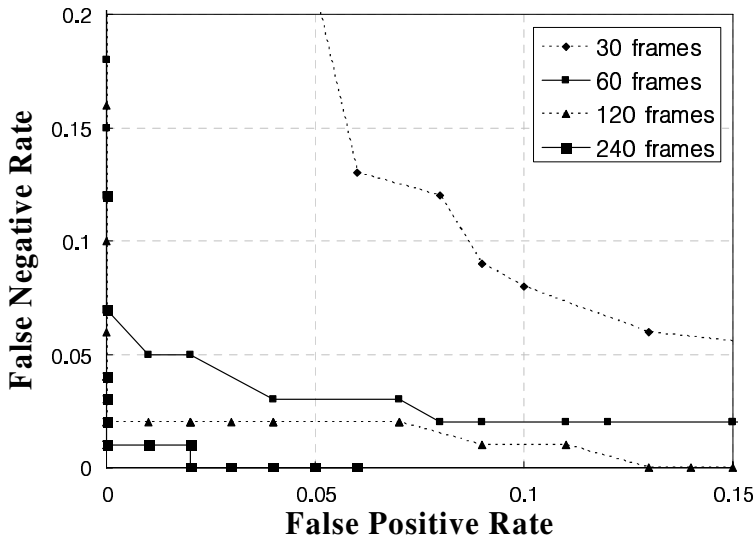


Fig. 6. ROC curve for ordinal signature: FPR vs. FNR

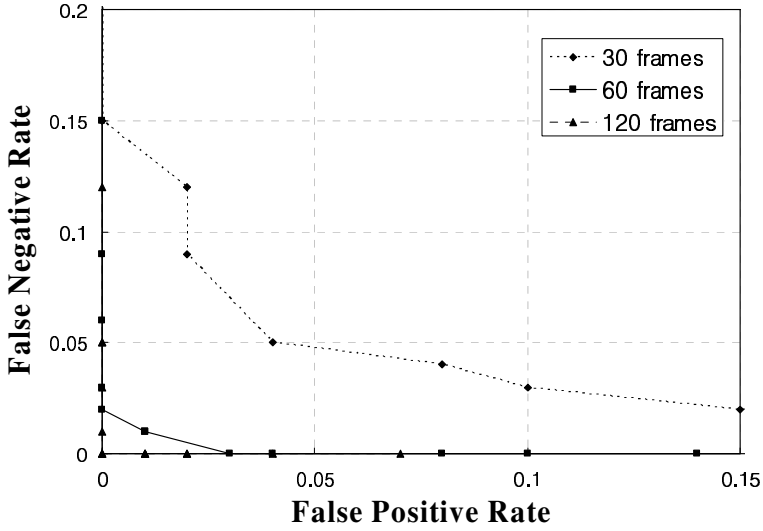


Fig. 7. ROC curve for singular value signature with $r=1$: FPR vs. FNR

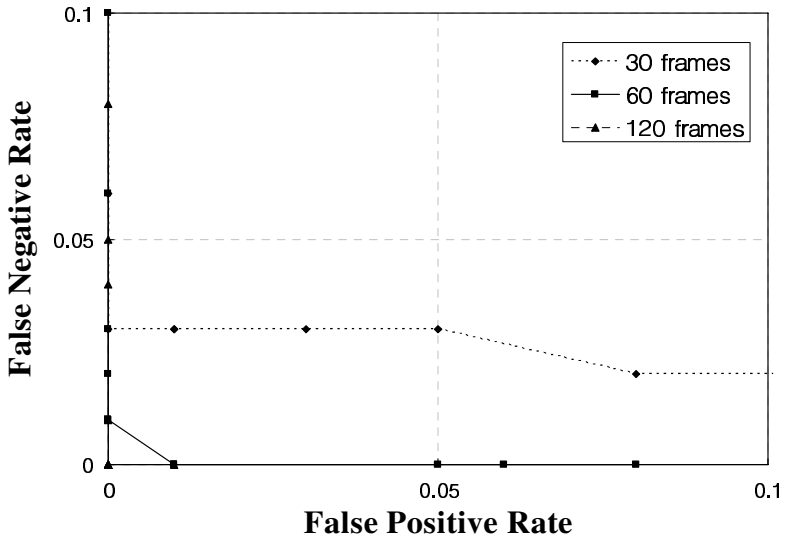


Fig. 8. ROC curve for singular value signature with $r=2$: FPR vs. FNR

For the test, 100 query clips with different clip lengths were sampled from the query video sequence. The ROC curves were then computed when varying τ with an increment of 0.5%.

As shown in Fig. 6, 7 and 8, the proposed system using 2 singular values as signature has superior performance to the ordinal feature irrespective of clip length, and also better one even though when only one singular value is used. And it can reduce the process time by using iterative algorithm for getting singular values because we use only small number of largest value of singular values, even though it needs time-consuming to calculate the SVD decomposition.

4.4 Memory Requirement

The proposed signature is composed of a decimal fraction from zero to one. Thus, when two dominant singular values are used for the video matching process, the memory requirement of the signature is 16 bytes/frame (4 bytes/block). Therefore, the signatures are converted to integers as follows

$$\sigma' = \text{round off}(255 \times \sigma) \quad (9)$$

where $0 \leq \sigma \leq 1$ and $0 \leq \sigma' \leq 255$.

From Eq. 9, the memory requirement is then reduced to 8 bytes/frame.

5 Conclusions

This paper proposed a new image signature using singular values for video sequence matching. When evaluating the proposed signature in comparison with ordinal measure, the proposed signature produced a better performance than the ordinal signature. One of the reasons is that this signature overcomes a weak point of ordinal signature in case that block mean are same but contents are different. If we use only one dominant singular value for video matching process, the memory for storing/indexing signatures is same as ordinal signature with 9 bytes/frame. A selection method for the proper number of singular values is further needed to get higher matching performance in both accuracy and time efficiency.

References

1. A. K. Jain, A. Vailaya, and W. Xiong: Query by clip. *Multimedia Syst. J.*, vol. 7, no. 5, pp. 369–384, 1999.
2. M. Naphade, M. Yeung, and B. Yeo.: A novel scheme for fast and efficient video sequence matching using compact signatures, in *Proc. SPIE Conf. Storage and Retrieval for Media Databases*, vol. 3972, pp. 564–572, Jan. 2000.
3. D. N. Bhat and S. K. Nayar: Ordinal measures for image correspondence, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 4, pp. 415–423, Apr. 1998.
4. R. Mohan: Video sequence matching, in *Proc. Int. Conf. Audio, Speech and Signal Processing (ICASSP)*, vol. 6, pp. 3697–3700, Jan. 1998.
5. A. Hampapur, K. H. Hyun, and R. M. Bolle: Comparison of sequence matching techniques for video copy detection, in *Proc. Storage and Retrieval for Media Databases*, pp. 194–201, Jan. 2002.
6. J. H. Luo and C. C. Chen: Singular value decomposition for texture analysis, In *Proceedings of SPIE- The International Society of Optical Engineering*, vol. 2298, Applications of Digital Image Processing XVII, CA, USA, pp.407-418, 1994.

The Papoulis-Gerchberg Algorithm with Unknown Signal Bandwidth

Manuel Marques, Alexandre Neves, Jorge S. Marques, and João Sanches

Instituto Superior Técnico & Instituto de Sistemas e Robótica
1049-001 Lisboa - Portugal
manuel@isr.ist.utl.pt
alex.filipe@gmail.com
jrm@isr.ist.utl.pt
jmrs@alfa.ist.utl.pt

Abstract. The Papoulis-Gerchberg algorithm has been extensively used to solve the missing data problem in band-limited signals. The interpolation of low-pass signals with this algorithm can be done if the signal bandwidth is known. In practice, the signal bandwidth is unknown and has to be estimated by the user, preventing an automatic application of the Papoulis-Gerchberg algorithm. In this paper, we propose a method to automatically find this parameter, avoiding the need of the user intervention during the reconstruction process. Experimental results are presented to illustrate the performance of the proposed algorithm.

1 Introduction

The reconstruction of signals after a non uniform sampling of the original signal is a key problem in many areas such as communications, medical imaging, geophysics and astronomy. The reconstruction of a signal without a priori information is an ill-posed problem because the observed signals are often incomplete and only some samples are observed. For this reason, we require some information about signal for a successful reconstruction. In this work, we assume that signal is low-pass.

Several methods were proposed [4,5,6,7,8] to reconstruct low-pass signals from a set of non-uniform samples e.g., using Fourier Analysis and Wavelets in order to interpolate the signal. The Papoulis-Gerchberg algorithm (P-G) [1,2,3] is a popular technique. It amounts to alternatively applying the space and frequency information available about the signal until it converges. However, this algorithm requires the knowledge of the signal bandwidth. This parameter must be determined by the user by a manual way. This paper tries to avoid this procedure and provides an algorithm to automatically obtain the bandwidth estimate therefore making the P-G fully automatic.

2 Formulation Problem and Notation

A discrete signal with N samples can be described by a N -dimensional complex vector x . The elements of the vector are denoted by $x[0], x[1], x[2], \dots, x[N-1]$ and correspond to samples of the signal.

The Discrete Fourier Transform (DFT) of the signal $x \in \mathbb{C}^N$ is the vector $X \in \mathbb{C}^N$ given by

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j \frac{2\pi}{N} nk}. \tag{1}$$

Because the DFT is a linear map in \mathbb{C}^N , the expression (1) can be represented in matrix form,

$$X = Fx, \tag{2}$$

where F is a $N \times N$ matrix and each element is given by

$$F_{ab} = e^{-j \frac{2\pi}{N} ab}$$

In this work, we will only consider band-limited signals. The DFT of these signals presents the following property [9]

$$X[i] = 0, i \in S \tag{3}$$

where S is a proper and fixed nonempty subset of $\{0, 1, \dots, N - 1\}$. The set of band-limited signals that verify (3) is a linear subspace of \mathbb{C}^N . The dimension of this subspace is equal to the cardinal of the complement of S and it is often denoted as the signals bandwidth. The signals are low-pass if the complement of S can be written as follows

$$\{0\} \cup \{1, 2, \dots, M\} \cup \{N - M + 1, N - M + 2, \dots, N - 1\}$$

In this case, the signal is characterized by the DC coefficient and by the first M harmonics. If the number of known samples L satisfies (4),

$$L \geq 2M + 1 \tag{4}$$

the signal's reconstruction without error is possible.

The signals considered in this work are low-pass ones. Therefore, they can be written as follows

$$x = Bx \tag{5}$$

where $B = F^{-1} \Gamma F$ and Γ is a $N \times N$ diagonal matrix defined by:

$$\Gamma = \text{diag}[1, \underbrace{1, \dots, 1}_{M1's}, 0, \dots, 0, \underbrace{1, \dots, 1}_{M1's}]$$

The problem we intent to solve is to reconstruct the low-pass signal $x[n]$ knowing a subset of its samples given by

$$y = Dx \tag{6}$$

where D is diagonal matrix with binary diagonal coefficients: $d_{ii} = 1$ if the $i - th$ sample is observed and $d_{ii} = 0$ otherwise. It is important to say that we assume that M is unknown.

3 The Papoulis-Gerchberg Algorithm

Let $x[n]$ be a low-pass signal with $M + 1$ harmonics. The Papoulis-Gerchberg algorithm reconstructs the signal if the condition (4) is satisfied and the M value is known. The algorithm starts with an initial signal estimate $\hat{x}_1 = y$ (6).

The iterative process consists of three steps.

1. Filter \hat{x}_i with a low-pass filter, eliminating the components with frequencies higher than the frequency of the $M - th$ harmonic.

$$z_{i+1} = B_{PG}\hat{x}_i \quad (7)$$

2. Insert known samples in the estimative.

$$\hat{x}_{i+1} = Ds + (I - D)z_{i+1} \quad (8)$$

3. Verify if the process converged or not. If not, return to the first step and consider $i = i + 1$.

The matrix B_{PG} performs the low-pass filtering operation and it is defined by

$$B_{PG} = F^{-1}\Gamma_{PG}F \quad (9)$$

where

$$\Gamma = \text{diag}[1, \underbrace{1, \dots, 1}_{M_{PG}1's}, 0, \dots, 0, \underbrace{1, \dots, 1}_{M_{PG}1's}]$$

is a $N \times N$ diagonal matrix and M_{PG} is the bandwidth estimate.

When the signal bandwidth is known, $M_{PG} = M$ and $B_{PG} = B$.

The algorithm convergence is proved in [1]. In this paper, the stop criterion is based on the L_1 norm between signals of two consecutive estimations.

$$\frac{1}{N} \sum_{n=1}^N |y_{i+1}[n] - y_i[n]| < \delta \quad (10)$$

where δ is a threshold specified by the user.

4 The Estimation of Signal Bandwidth

4.1 Motivation

As mentioned before, a perfect reconstruction of a non uniform sampled signal can be obtained by the Papoulis-Gerchberg algorithm if the number of observed samples is enough (satisfies (4)) and if the M value is known. However, in many situations, this last condition is not true and it becomes important to use an alternative way to solve the problem.

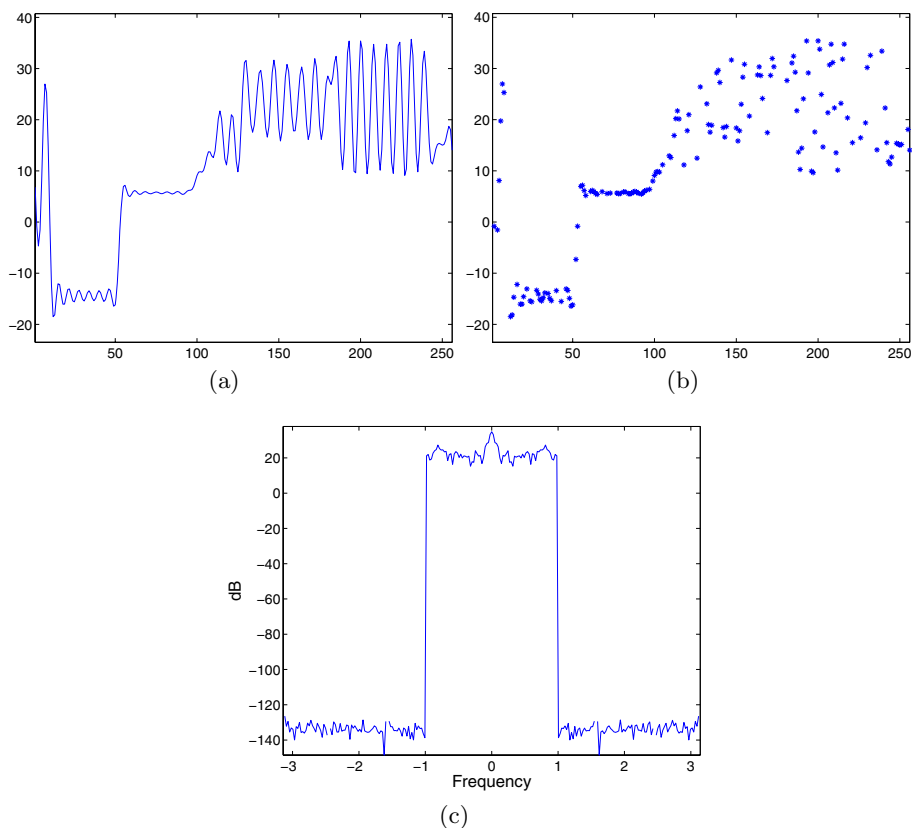


Fig. 1. Synthetic example: 60% of all samples are known. (a) Original signal (b) Known samples (c) DFT.

In order to motivate this problem we shown in Fig. 1(a) a discrete low-pass signal with 256 samples and bandwidth $M = 40$. The Fourier spectrum of the original signal is shown in figure 1(c). The signal was then randomly sampled: 60% of the samples are known and the others remain unknown (see Fig. 1(b)).

Before we describe the new method, it is relevant to discuss the importance of the signal bandwidth M . Figure 2 presents the SNR results of the estimates obtained by the P-G using M_{PG} values ranging between 1 and the largest integer satisfying (4). In this figure, we can see that a correct reconstruction (high SNR) of the original signal is obtained for a small range of M_{PG} values, close to the true value of the bandwidth M . When M_{PG} is lower or much higher than M , the P-G estimate does not converge to the original signal.

When the M_{PG} value (9) is lower than M , the reconstructed signal $\hat{x}[n]$ has a low-pass spectrum

$$\hat{X}[i] = 0, \quad i \in T \quad (11)$$

where $T \supset S$. This means that we wish to find a signal that contains the known samples and less harmonics than the original. Due to this last restriction, the original signal is not in the complement of the subspace T and the mentioned algorithm does not converge to this signal because the step 2 creates discontinuities in the time domain.

When the M_{PG} value is greater than M the reconstructed signal is low-pass with

$$\hat{X}[i] = 0, i \in V \tag{12}$$

where $V \subset S$. In this case, the complement of the subspace V contains the original signal x but the P-G will not probably converge to the original signal x since there is an infinite number of signals in the subspace V satisfying (12) for the known samples.

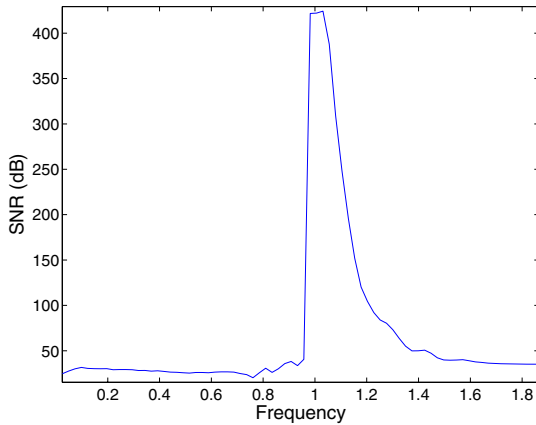


Fig. 2. Reconstruction performance SNR as function of the filter B_{PG} cut frequency

The signal to noise ratio, presented in Fig. 2, provides a valuable information about the signal bandwidth. However it cannot be used in practice because we do not know the original signal. An alternative technique must be used instead as discussed in section 4.2.

4.2 The Proposed Method

The main goal of the proposed method is to reconstruct the original signal from a set of non uniform samples, without knowing the signal bandwidth.

The strategy followed in this paper consists of finding the lowest M that leads to a low-pass signal estimate (12) with the known samples. In other words, we wish to find a signal that contains known samples and is defined by the smallest number of Fourier coefficients $(2M + 1)$.

This is performed by applying the P-G algorithm for all M values such that inequality (4) holds. For each M_{PG} value, the percentage of energy contained in the signal's high frequencies is calculated as well as the log energy ratio

$$g[k] = \log_{10} \left(\frac{E_h(k)}{E_T(k)} \right), \quad k = 0, 1, \dots, \lfloor L/2 \rfloor \tag{13}$$

where E_h and E_T are the high frequency and the total energies, respectively, defined as follows

$$E_T(k) = \sum_{m=0}^{N-1} |\hat{X}^k[m]|^2 \tag{14}$$

$$E_h(k) = \sum_{m=h+1}^{N-h+1} |\hat{X}^k[m]|^2 \tag{15}$$

where \hat{X}^k is Fourier transform of the signal \hat{x}_k . This signal is obtained by the P-G algorithm with $M_{PG} = k$. The log energy ratio function provides very useful information about the signal bandwidth.

The bandwidth estimate proposed in this paper is the value of k which minimizes the difference $g[k] - g[k-1]$. It has been experimentally found that the first difference of the energy ratio $g[k]$ has its largest fall for $k = M$. This estimate is not the unique M_{PG} that allows to obtain a low pass signal but it is the first one. With this strategy, we find the smallest linear subspace (the complement of V) that contains the low pass signal verifying (6).

To illustrate the estimation of M , we show, in Fig. 3, the log energy ratio function obtained for the sampled signal presented in the Fig. 1(b).

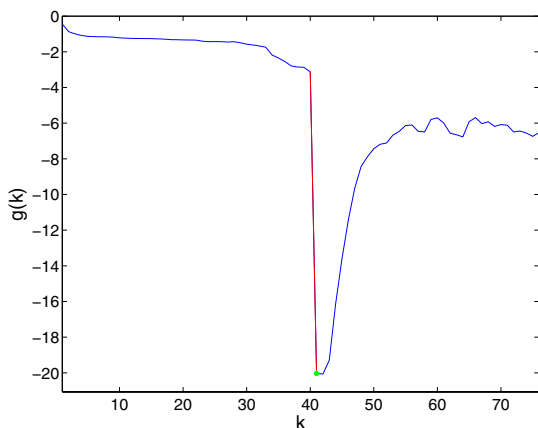


Fig. 3. The log energy ratio function for the example in figure 1(b)

The smallest difference between two consecutive values of $g[k]$ is shown with a bold line. In this case, the bandwidth estimate is equal to the true value $M = 40$.

This method can be extended to 2D low-pass signals in several ways e.g., considering a 2D signal as a set 1D low-pass signals. In the case of images we can independently apply this method to columns or rows or use a joint estimation procedure.

5 Results

To evaluate the proposed method, several experiments were performed with synthetic signals and real signals. Two experiments will be shown in this section. In the first experiment we have generated 2000 random signals with length $N = 256$ and bandwidth $M = 40$ and applied the algorithm to each of them.

The signals were generated as follows. First we split the time domain into a set of intervals with random lengths. Then, for each interval we generated one of the following signals: constant, linear or sinusoid with random parameters. Finally, we eliminated the high frequency components to guarantee that the signal is low pass with bandwidth $M = 40$.

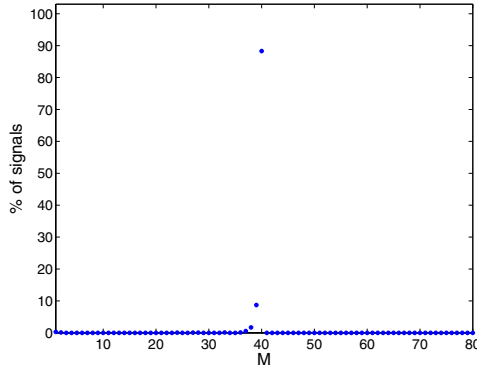


Fig. 4. Histogram of M values estimated with $\delta = 10^{-3}$ and 80% of known samples

Figures 4 and 5 display the histogram of the bandwidth estimates for several values of δ (see (10)) and percentage of unknown samples. The performance of the algorithm depends on both parameters. We observe a smooth degradation of the performance when the number of observations decreases from 80% to 40% of N .

Evaluating results presented in these figures, we can say that the percentage of known samples affects the method's performance, but even when we have a small set of known samples (40% when the lowest percentage to satisfy (4) is 31,6%) the method gives 60% of correct answers (Fig. 5(a)) and most of the errors are small.

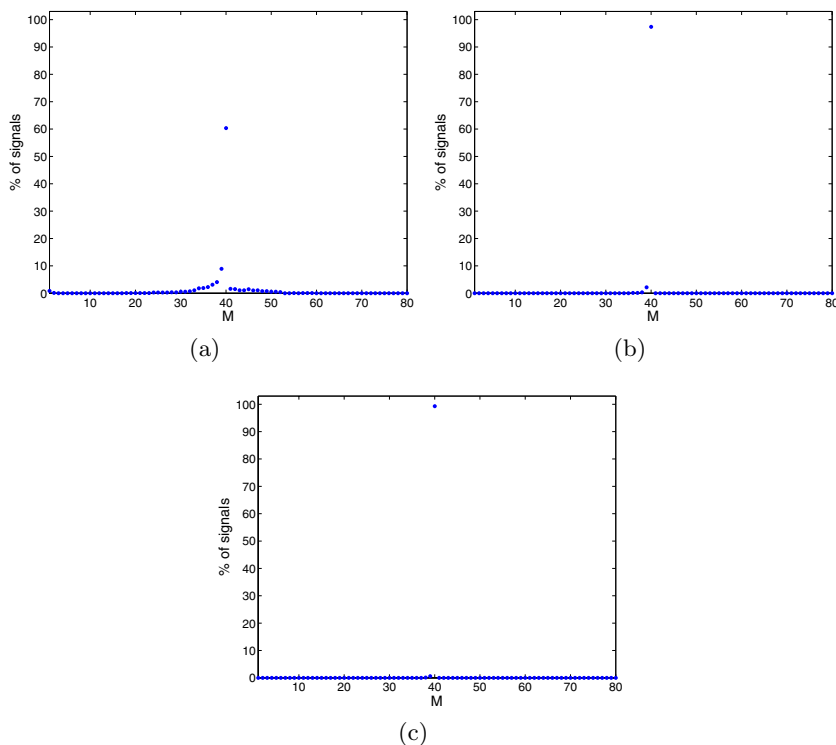


Fig. 5. Histograms of M values estimated with $\delta = 10^{-6}$: (a) 40% of known samples (b) 60% of known samples (c) 80% of known samples

Concerning the parameter δ , small values of δ lead to better estimates. However, the estimation method becomes slower because we have to perform L iterations.

Table 1, shows the mean and standard deviation of SNR results of the estimated signals obtained by the method for unknown M and the P-G algorithm for known M . The two methods lead to indistinguishable results when the percentage of known samples is 50%, 60%, 70%, 80%. Only in the case 40% of known samples we observe a difference of $3dB$ between the reconstructions performed with known and unknown M .

The proposed method was used to reconstruct real signal and images from an incomplete set of samples. To illustrate the method's performance in the case of images we applied the proposed algorithm to the *Lena* image with 15% of unknown samples (see Fig. 6(a)).

We created an enlarged version of the *Lena* image with 512×512 pixels and bandwidth $M = 128$. This was done by padding with zeros an initial version of the *Lena* image with 256×256 and zeroing the high frequency coefficients of the Fourier spectrum. We then applied the proposed algorithm to each row independently.

Table 1. Results of the P-G algorithm for the case of known bandwidth and unknown bandwidth

| Known Samples (%) | M known | | M unknown | |
|----------------------|----------|---------|-----------|---------|
| | Mean(dB) | STD(dB) | Mean(dB) | STD(dB) |
| 40 | 24,2 | 15,3 | 21,5 | 16,3 |
| 50 | 56,9 | 21,7 | 56,0 | 22,9 |
| 60 | 82,9 | 14,7 | 82,9 | 14,9 |
| 70 | 97,3 | 10,3 | 97,3 | 10,3 |
| 80 | 108,0 | 7,7 | 108,0 | 7,8 |

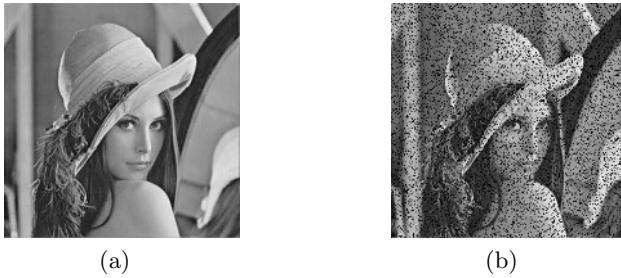


Fig. 6. Non uniform sampling: (a) Original image (b) Sampled image with 85% of known samples

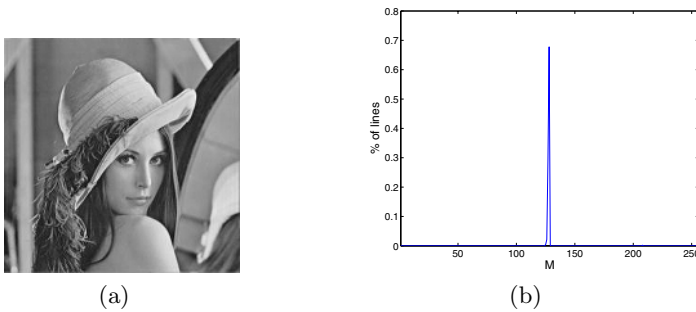


Fig. 7. Method's results: (a) Estimated image (b) Histogram of M values

In this case, the Fig. 6(a) has 512×512 pixels and $M = 128$.

Figure 7(b) shows the histogram of the M estimates for the image's rows. The Fig. 7(a) displays the reconstructed image obtained with the proposed method which is indistinguishable from the original. A SNR=30 dB was obtained.

For a better evaluation of the method's performance, we show a detail of the eye. The original image (Fig. 8(a)) and the reconstructed image (Fig. 8(c)) are almost undistinguishable.

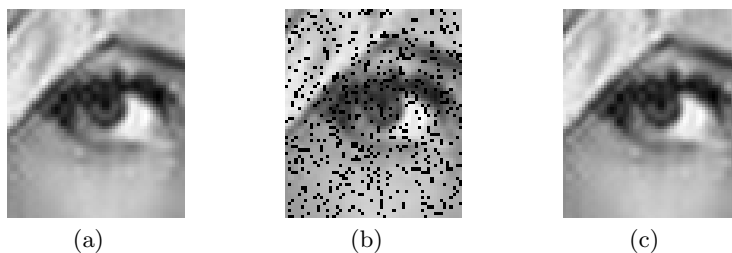


Fig. 8. Eye Reconstruction: (a) Original image (b) Sampled image (c) Estimated image

6 Conclusions

This paper presents a method to estimate the signal bandwidth for non-uniform sampled signal. This allows an automatic reconstruction of non-uniform sampled signal using the Papoulis-Gerchberg algorithm, when the bandwidth is unknown. Very good experimental results are obtained with synthetic and real signals.

References

1. Papoulis, A.: A new algorithm in spectral analysis and band-limited extrapolation. *IEEE Transactions on Circuits Syst.* **19** (1975) 735–742
2. Papoulis, A.: A new method in image restoration. *JSTAC*, Paper VI-3 (1973)
3. Gerchberg, R. W.: Super-resolution through error energy reduction. *Optica Acta* **21-9** (1974) 709–720
4. Wingham, D. J.: The reconstruction of a band-limited function and its Fourier transform from a finite number of samples at arbitrary locations by singular value decomposition. *IEEE Transactions on Circuit Theory* **40** (1992) 559–570
5. Yen, J. L.: On nonuniform sampling of bandwidth-limited signals. *IRE Transactions on Circuit Theory* **CT-3** (1956) 251–259
6. Thomas, J. O. and Yao, K.: On some stability and interpolatory properties of nonuniform sampling expansions. *IEEE Transactions on Circuit Theory* **14** (1967) 404–408
7. Benedetto, J. and Chui, C. K. (ed): Irregular sampling and frames. In: *Wavelets: A Tutorial in Theory and Applications*, Academic Press (1992) 445–507
8. Feichtinger, H. G., Grchenig, K. and Strohmer, T: Efficient numerical methods in non-uniform sampling theory. *Numerische Mathematik* **69** (1995) 423–440
9. Ferreira, P. J. and Marvasti, F. A. (ed.): Iterative and Noniterative Recovery of Missing Samples for 1-D Band-Limited Signals. In: *Sampling Theory and Practice*, Plenum Publishing Corporation, 2001 235–282

Continuous Evolution of Fractal Transforms and Nonlocal PDE Imaging

Edward R. Vrscay

Department of Applied Mathematics
Faculty of Mathematics
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1
ervrscay@uwaterloo.ca

Abstract. Traditional fractal image coding seeks to approximate an image function u as a union of spatially-contracted and greyscale-modified copies of itself, i.e., $u \approx Tu$, where T is a contractive *fractal transform* operator on an appropriate space of functions. Consequently u is well approximated by \bar{u} , the unique fixed point of T , which can then be constructed by the discrete iteration procedure $u_{n+1} = T_n$.

In a previous work, we showed that the evolution equation $y_t = Oy - y$ produces a continuous evolution $y(x, t)$ to \bar{y} , the fixed point of a contractive operator O . This method was applied to the discrete fractal transform operator, in which case the evolution equation takes the form of a nonlocal differential equation under which regions of the image are modified according to information from other regions.

In this paper we extend the scope of this evolution equation by introducing additional operators, e.g., diffusion or curvature operators, that “compete” with the fractal transform operator. As a result, the asymptotic limiting function y_∞ is a modification of the fixed point \bar{u} of the original fractal transform. The modification can be viewed as a replacement of traditional postprocessing methods that are employed to “touch up” the attractor function \bar{u} .

1 Introduction

The well-known examples of fractal sets, e.g., ternary Cantor set, Sierpinski gasket [8], demonstrate the property of *self-similarity* – each piece of the set can be viewed as a geometrically contracted copy of the entire set. Indeed, such self-similarity properties inspired the seminal paper of M. Barnsley and S. Demko on “iterated function systems” and the inverse problem of fractal construction [4] which, in turn, led to intensive investigations into fractal image coding and compression [3,5,9].

It is, however, too much to expect that a general image, viewed as a function or measure, would be well approximated by a union of geometrically contracted and and greyscale modified copies of itself. Following the later idea of A. Jacquin [13], however, images are generally well approximated by a union of spatially-contracted and greyscale modified copies of *subsets* of themselves. This is the

basis of block-based fractal image coding [3,5,9] or “local IFS” methods. Indeed, recent experiments by our group [2,1] confirm that images are generally quite *locally self-similar* – subblocks of an image are typically well approximated by *many* decimated subblocks from elsewhere in the image. In the next section, some evidence to support this statement will be presented.

Mathematically, block-based fractal image coding seeks to approximate an image function u by the unique fixed point \bar{u} of a contractive *fractal transform* operator T . The action of T on an image function is to replace each subblock R_i with a shrunken and greyscale modified copy of a subblock D_j of the image. More details of this procedure will be given in the next section. The fixed point approximation \bar{u} to u is generated by the iteration procedure $u_{n+1} = Tu_n$, where u_0 can be any starting function or “seed.” Convergence of the u_n to \bar{u} is guaranteed by Banach’s fixed point theorem.

In a previous work [6] we introduced the following class of evolution equations associated with contraction mappings T on a Banach space of functions $B(X)$:

$$\frac{\partial u}{\partial t} = Tu - u. \quad (1)$$

The result is that the function $u(x, t)$ approaches the fixed point \bar{u} as $t \rightarrow \infty$ – a *continuous evolution* toward the fixed point \bar{u} of the contraction map T , as opposed to the usual discrete sequence of iterates u_n defined earlier.

We then applied this evolution method to some fractal transform operators on measure and function spaces, in particular, the block-based fractal image coding operator. The original motivation to devise such evolution equations arose from a desire to perform continuous, yet fractal-like “touch-up” operations on images. A fractal-based evolution method could conceivably be used to produce arbitrarily small alterations to an image $u(x)$ in a neighbourhood of a point $x_0 \in X$ that depend upon the behaviour of u at other regions of X . This is an example of a *nonlocal* imaging operation.

In this paper, we show that one can accommodate additional operators into the evolution operator in Eq. (1), for example, the simple diffusion operator,

$$\frac{\partial u}{\partial t} = \epsilon \nabla^2 u + Tu - u. \quad (2)$$

where positive diffusion, i.e., $\epsilon > 0$ corresponds to blurring and negative diffusion $\epsilon < 0$ produces sharpening. Other differential operators, e.g., anisotropic diffusion [15,17] could also be considered. In all cases, we have a *nonlocal* partial differential equation governing the evolution of an image function $u(x, t)$.

As a consequence, there is a competition between the action of the differential operator and the nonlocal fractal transform operator. For at least small perturbations ϵ , the asymptotic limit function u_∞ is a kind of perturbation of the fixed point \bar{u} of the contraction operator T .

At first, the idea of a nonlocal operator might seem to be counterproductive to PDE imaging. In PDE-based enhancement methods such as denoising, the time evolution of an image u at a point x is determined completely by its local properties, i.e. value and derivatives, e.g. Gaussian smoothing [17], anisotropic diffusion

[15], total variation minimization [16]. Recently, however, nonlocal methods have been shown to very effective in image enhancement, e.g. the method of “nonlocal means” in image denoising [7]. It is conceivable that such nonlocal methods could make more inroads into PDE imaging methods. Indeed, one purpose of this paper is to show how the nonlocal fractal transform method can easily be incorporated into a PDE-based approach.

The structure of this paper is as follows. In Section 2, we briefly review the basics of block-based fractal image coding, followed by an application of the continuous evolution method. In Section 3, we examine the effects of adding other differential operators to the evolution equation, specifically on the limiting functions. Some concluding statements are made in Section 4.

2 Block-Based Fractal Image Coding

For simplicity, the support X of an image function u will be considered as $n \times n$ pixel array, i.e., a discrete support. Now consider a partition of X into nonoverlapping subblocks R_i , $1 \leq i \leq N$, so that $X = \cup_i R_i$. Associated with *range block* R_i is a larger *domain block* $D_i \subset X$ so that $R_i = w_i(D_i)$, where w_i is a 1-1 contraction map. (In the pixel case, it will represent the decimation needed to accomplish the spatial contraction.) Assume that the image function $u(R_i) = u|_{R_i}$ supported on each subblock R_i is well approximated by a spatially-contracted and greyscale-modified copy of $u(D_i) = u|_{D_i}$:

$$u(R_i) \approx \phi_i(u(w_i^{-1}(D_i))), \tag{3}$$

where the $\phi_i : \mathbf{R} \rightarrow \mathbf{R}$ are greyscale maps that are usually affine in form:

$$\phi_i(t) = \alpha_i t + \beta_i. \tag{4}$$

Because the range blocks R_i are nonoverlapping, we can write Eq. (3) as

$$u(x) \approx (Tu)(x) = \alpha_i(u(w_i^{-1}(x))) + \beta_i, \quad x \in R_i, \tag{5}$$

where T is the block fractal transform operator defined by the range-domain assignments and the affine greyscale maps ϕ_i . Under suitable conditions on the α_i greyscale coefficients and the \mathbf{R}^2 contraction factors of the w_i , the operator T is contractive in an appropriate image function space $\mathcal{F}(X)$ (typically $L^2(X)$) [10].

It is a well-known result that if the *collage distance* $\| u - Tu \|$ is small, then u is well approximated by the fixed point \bar{u} of T . This is stated more precisely by the *Collage Theorem* [3],

$$\| u - \bar{u} \| \leq \frac{1}{1 - c_T} \| u - Tu \|, \tag{6}$$

where c_T is the contraction factor of T . In this way, the inverse problem of fractal image coding can be reformulated into a more tractable problem. Instead

of searching for an operator T whose fixed point \bar{u} is close to u , we search for a T that maps u close to itself.

In Figure 1 is presented the fixed point approximation \bar{u} to the standard 512×512 pixel *Lena* image (8 bits per pixel) using a partition of 8×8 nonoverlapping pixel blocks ($64^2 = 4096$ blocks). The “domain pool” for each range block was the set of $32^2 = 1024$ 16×16 nonoverlapping pixel blocks. For each 8×8 pixel range block R_i , the 16×16 pixel block that provided the best approximation in Eq. (3) was chosen as the domain block D_i that eventually defined the fractal transform operator T for this image. The fixed point approximation \bar{u} was obtained by starting with the seed image $u_0(x) = 255$ (plain white image) and iterating $u_{n+1} = Tu_n$ to $n = 15$. The iterates u_1 , u_2 and u_3 are also shown in this figure.



Fig. 1. Starting at upper left and moving clockwise: The iterates u_1 , u_2 and u_3 along with the fixed point \bar{u} of the fractal transform operator T designed to approximate the standard 512×512 (8 bpp) *Lena* image. The “seed” image was $u_0(x) = 255$ (plain white).

Finally, we return to the comments made in the introduction regarding the statistical local self-similarity property of images. A series of extensive experiments by our group [2,1] has shown that images are generally quite *locally self-similar*: Using the terminology of fractal block coding developed above, range blocks R_i of an image are typically well approximated by a good number of domain blocks D_j from the image. To illustrate, we return to the *Lena* image of Figure 1 and the fractal coding procedure used to produce the fixed point \bar{u} in that figure. Recall that 4096 8×8 nonoverlapping range blocks and 1024 nonoverlapping 16×16 nonoverlapping domain blocks were used in the coding.

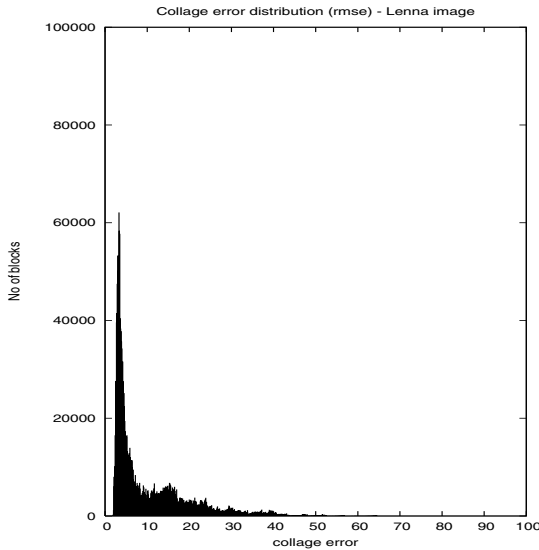


Fig. 2. Histogram distribution of collage errors Δ_{ijk} , cf. Eq. (7) for all domain-range block pairings for *Lena* image

In Figure 2 is presented a histogram plot of the collage errors for *all possible range/domain block pairs* R_j/D_i , with all 8 square-to-square isometries being considered:

$$\Delta_{ijk} = \min_{\alpha, \beta} \| R_i - \alpha u(w_j^{(k)}(D_j)) - \beta \|, \tag{7}$$

$$1 \leq i \leq 1024, \quad 1 \leq j \leq 4096, \quad 1 \leq k \leq 8,$$

for a total of 33,554,432 collage errors. (Of course, for each range block R_i , the domain block $D_j(i)$ and mapping $w_i^{k(i)}$ producing the lowest collage error was chosen for the fractal code that produced Figure 1.) Although there are very few, if any, exact matchings, a very pronounced peak is situated not far from zero collage error. Of course, the histogram distributions vary from image to image.

Images such as *Mandrill*, which is well known in presenting difficulties for any kind of image coding, demonstrate more flattened histogram distributions. A purely random image, i.e., Gaussian noise with variance σ will demonstrate a Gaussian-like distribution of collage errors that peaks at σ .

3 Continuous Evolution of Fractal Transform Operators

We now employ the block-based fractal transform operator T , defined in Eq. (5), in the continuous evolution Eq. (1). The result is a nonlocal partial differential equation in the image function $u(x, t)$:

$$\frac{\partial u(x, t)}{\partial t} = \alpha_i u(w_i^{-1}(x), t) + \beta - u(x, t), \quad x \in R_i. \tag{8}$$

In Figure 3 are shown some steps in the time evolution of images $u(x, t)$ as determined by the above evolution equation, where T is the fractal transform operator used in the construction of Figure 1. In this case, the time evolution proceeds at a slower pace than in Figure 1, starting at $u_0 = 255$ (plain white image) and proceeding in time steps of 0.2. A simple forward Euler scheme using a step size of $h = 0.1$ was used.

3.1 Evolution in the Presence of Diffusion

In this section, we examine the effects of the isotropic diffusion term in Eq. (2) for various values of the diffusion constant ϵ .

Case I: $\epsilon > 0$

In Figure 4 are presented the limiting images for ϵ values of 0.5, 5.0, 10.0 and 20.0. Eq. (2) was integrated to $t = 50$ using a simple Euler scheme with time step $h = 0.01$. As ϵ increases, the blurring effects of the diffusion operator become more pronounced.

In standard, i.e., discrete, fractal image coding, there has always been a concern about the blockiness exhibited by the fixed points of fractal transform operators. This is mostly due to the partitions used to produce the range blocks – most often, there is no attempt to perform any kind of “patching” of neighbouring blocks over their common boundaries. Various methods have been used to reduce such blockiness. One method, that of “postprocessing,” was to blur the final fixed point image, either over its entirety or selectively across the boundaries of the range blocks. Another is to blur the image after each application of the fractal transform operator T , a kind of intermediate processing. The diffusion operator in Eq. (2) essentially performs such an intermediate processing in a continuous manner. The asymptotic images are no longer fixed points of the fractal transform operator T but rather solutions to the partial differential equation

$$\epsilon \nabla^2 y + Ty - y = 0. \tag{9}$$

For small values of ϵ , these asymptotic limiting functions could possibly be viewed as perturbations of the fixed point function \bar{u} of T , a subject of current investigation.

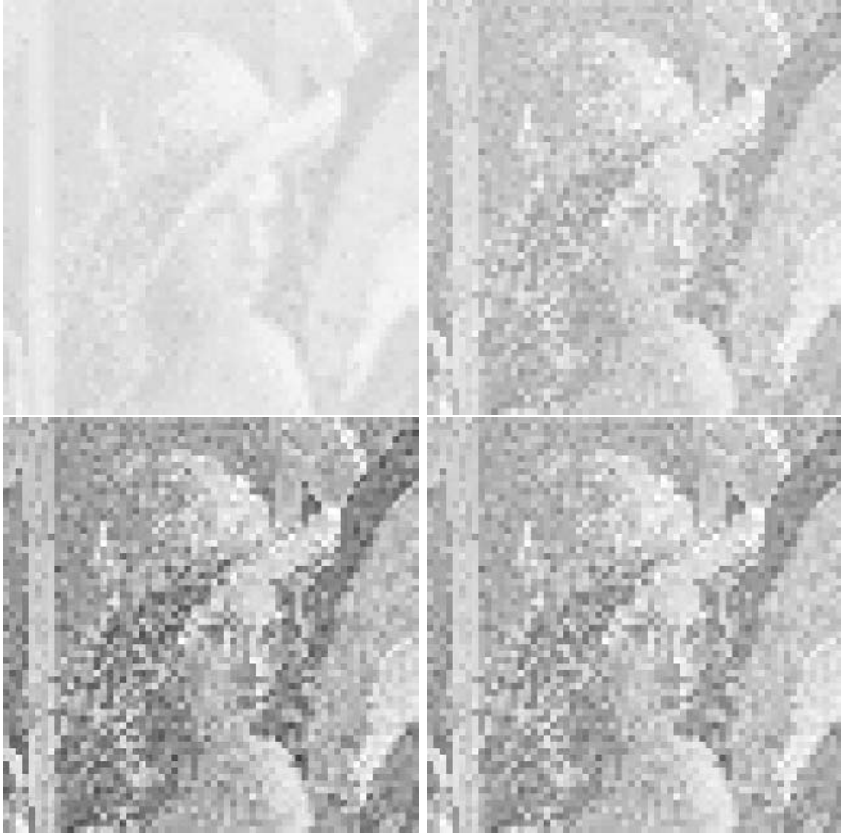


Fig. 3. Starting at upper left and moving clockwise: The images $u(x, t)$ at times 0.2, 0.4, 0.6, 0.8 produced from $u(x, 0) = 255$ (plain white) under evolution by $u_t = Tu - u$ where T is the fractal transform whose discrete iteration was shown in Figure 1

Case 2: $\epsilon < 0$

For this case of “negative diffusion,” we expect the limiting images $u(x, t), t \rightarrow \infty$ to be sharpened. In Figure 5 is presented the limiting image for $\epsilon = -0.1$. This image was obtained by integrating Eq. (2) $t = 50$ using the simple Euler forward scheme with time step $h = 0.001$. Such a small value for the time step was used because of the numerical instability involved with negative diffusion. The image is a somewhat sharpened version of the fixed-point *Lena* image of Figure 1. Unfortunately, the sharpening enhances not only the edges present in the image but also those that lie along the boundaries of the 8×8 range blocks.

Difficulties are encountered in the numerical computations when ϵ is chosen to be lower than -0.1 . For example, when $\epsilon = -0.2$, a great deal of irregularity develops in the image, erupting into virtual instability at $t \approx 7.5$ and virtual



Fig. 4. Starting at upper left and moving clockwise: The limiting images $u(x, t), t \rightarrow \infty$ produced by integrating Eq. (2) for ϵ values of 0.5, 5.0, 10.0 and 20.0, respectively

randomness at $t = 20$. Such a negative diffusion scheme is known to be numerically unstable. At the time of writing, it is not clear whether the $t = 20$ result is due to this instability or the actual lack of a nontrivial asymptotic limiting function because of the nonsmooth nature of the image function $u(x, t)$. If the latter applies, then there may be some some kind of critical value $\epsilon_0 \in [-0.2, -0.1]$ at which a limiting function ceases to exist. The negative diffusion term can be viewed as a process that competes against the term $Tu - u$ which is trying to drive the evolution toward the fixed point \bar{u} of T .

3.2 Evolution for a Convex Combination of Contraction Mappings

Let us now suppose that we have a set of contraction maps driving the evolution in Eq. (1) in parallel. It is natural to consider a convex combination. Let $T_i, i = 1, 2, \dots, n$ be a set of contraction maps with contraction factors $c_i \in [0, 1)$ fixed points \bar{y}_i . Now let $\lambda_i, i = 1, 2, \dots, n$, be a partition of unity, i.e., $\lambda_i \in (0, 1)$



Fig. 5. Left: The limiting image $u(x, \infty)$ produced by integrating Eq. (2) for $\epsilon = -0.1$, corresponding to negative diffusion. Euler method, step-size $h = 0.001$, integrated to $t = 50$. Right: The limiting image for $\epsilon = 0$, presented for comparison.

with $\sum_i^n \lambda_i = 1$, and consider the evolution equation

$$\frac{\partial y}{\partial t} = \sum_{i=1}^n \lambda_i (T_i y - y). \tag{10}$$

This equation may be rewritten as

$$\frac{\partial y}{\partial t} = T y - y, \tag{11}$$

where

$$T = \sum_{i=1}^n \lambda_i T_i. \tag{12}$$

Now, for any $u, v \in \mathcal{F}(X)$,

$$\begin{aligned} \| T u - T v \| &= \left\| \sum_{i=1}^n \lambda_i T_i u - \sum_{i=1}^n \lambda_i T_i v \right\| \\ &\leq \sum_{i=1}^n \lambda_i \| T_i u - T_i v \| \\ &\leq \sum_{i=1}^n \lambda_i c_i \| u - v \|. \end{aligned} \tag{13}$$

Since $c = \sum_{i=1}^n \lambda_i c_i \in [0, 1)$, it follows that T is a contraction mapping with a unique fixed point \bar{y} . From our earlier result associated with Eq. (1), it also follows that \bar{y} is a globally asymptotically stable solution of Eq. (10).

Numerical experiments, where the T_i are fractal transform operators corresponding to different images, have demonstrated convergence to limiting asymptotic functions u_∞ . These fixed points demonstrate characteristics of the different images – a kind of interpolation. (However, the blockiness of the fractal transform operators makes these interpolations rather unattractive.) In fact, “interpolation” is not the most appropriate term. Eq. (10) actually represents a continuous version of the iterative method of *projections onto convex sets* (POCS) that has been used in a variety of applications including imaging [18].

It is interesting to consider the situation where the T_i are *different* fractal transform operators for the *same* image. For example, for each range block R_i we could consider a number of domain blocks, possibly weighting each block inversely according to the collage error that it yields. The idea of using multiple domain blocks in order to improve the accuracy of approximation has been around for some time (see, for example, [9]). However, S. Alexander [1] has recently shown that this idea can be used effectively for purposes other than simply increasing the PSNR, for example, denoising, edge detection and block classification. We simply outline here the idea of how denoising can be accomplished.

Fractal Image Denoising

Firstly, it is well known that lossy compression schemes can denoise an image, which is also the case in fractal image coding. Assuming that the noise is additive, stationary and of zero mean, the spatial contraction/decimation that comprises the mapping of domain to range blocks contributes to the reduction of the variance of the noise. (We have shown [11] that the degree of denoising can be increased by improving the fractal code of the “denoised” attractor.) Alexander’s scheme, however, capitalizes on the fact, mentioned earlier, that there are generally several domain blocks that match a given range block almost as well as the “optimal” block, i.e., the block yielding the lowest collage error. (Most often, the difference is very small.) A number of these lowest-error blocks are then employed to produce the modified range block, a kind of weighted averaging which reduces the noise. Several strategies were investigated and work very well [1]. These methods are easily incorporated into a continuous evolution formulation but will not be discussed any further here.

Finally, we mention that an analogous “nonfractal” strategy has been adopted for denoising, namely, the “patching” of a domain block with a number of blocks of the same size from elsewhere in the image [7]. This method relies on *translational* similarity of an image as opposed to the *scaling similarity* inherent in fractal coding approaches.

4 Concluding Remarks

In a previous work, we showed that the evolution equation $y_t = Oy - y$ produces a continuous evolution $y(x, t)$ to \bar{y} , the fixed point of a contractive operator O . When applied to a fractal block coding operator T , the evolution equation takes the form of a nonlocal differential equation so that a region (range block) of an image is being continuously modified by another region (domain block).

In this paper we have extended the scope of this evolution equation by introducing additional operators, e.g., diffusion, that “compete” with the fractal transform operator. As a result, the asymptotic limiting function y_∞ is a modification of the fixed point \bar{u} of the original fractal transform. In the case of diffusion, the modification can be viewed as a replacement of traditional post-processing methods that are employed to “touch up” the attractor function \bar{u} . Negative diffusion produces a sharpening of \bar{u} . It now remains to establish theoretically the existence of such limiting asymptotic functions.

This continuous evolution method will easily accommodate other operators, e.g., nonlinear anisotropic diffusion operators [15,17]. A possible concern, however, is the artificial blockiness that is introduced by the fractal coding method from the range block boundaries. There are a number of ways to reduce these problems, including (i) overlapping range blocks and (ii) a kind of Gaussian window of the domain-to-range block mapping that extends beyond the usual rigid range block boundaries.

Finally, we mention that the continuous evolution formalism can actually help to suggest other discrete iteration processes that can be employed with fractal image coding. As was pointed out in our earlier work [6], when a simple forward Euler scheme with time step $h > 0$ is employed for the integration of Eq. (1), we obtain the discrete iteration process

$$\begin{aligned} y_{n+1} &= y_n + h(Ty_n - y_n) \\ &= (1 - h)y_n + hTy_n. \end{aligned} \tag{14}$$

For $0 < h < 1$, the y_{n+1} produced by the Euler method is a linear interpolation between y_n and Ty_n . In the case $h = 1$, we obtain the usual iteration procedure $y_{n+1} = Ty_n$. The discussion in Section 3.2 above suggests a discrete formulation for a convex combination of “competing” contraction maps. Indeed, this formulation is quite analogous to the method of projections onto convex sets (POCS) [18]. One may also consider the use of penalty functions in the evolution equation to “steer” the time evolution.

Acknowledgements

This research has been supported in part by the Natural Sciences and Engineering Research Council of Canada, which is hereby gratefully acknowledged.

References

1. S.K. Alexander, *Multiscale methods in image modelling and image processing*, Ph.D. Thesis, Department of Applied Mathematics, University of Waterloo (2005).
2. S.K. Alexander, E.R. Vrscay and S. Tsurumi, An examination of the statistical properties of domain-range block matching in fractal image coding (preprint).
3. M.F. Barnsley, *Fractals Everywhere*, Academic Press, New York (1988).
4. M.F. Barnsley and S. Demko, Iterated function systems and the global construction of fractals, *Proc. Roy. Soc. London* **A399**, 243-275 (1985).

5. M.F. Barnsley and L.P. Hurd, *Fractal Image Compression*, A.K. Peters, Wellesley, Massachusetts (1993).
6. J. Bona and E.R. Vrscay, "Continuous evolution of functions and measures toward fixed points of contraction mappings," in *Fractals in Engineering: New Trends in Theory and Applications*, edited by J. Levy-Vehel and E. Lutton (Springer Verlag, London, 2005), pp. 237-253.
7. A. Buades, B. Coll and J.-M. Morel, A nonlocal algorithm for image denoising, *CVPR* (2), 60-65, 2005.
8. K. Falconer, *The Geometry of Fractal Sets*, Cambridge University Press, Cambridge (1985).
9. Y. Fisher, *Fractal Image Compression*, Springer Verlag, New York (1995).
10. B. Forte and E.R. Vrscay, Theory of generalized fractal transforms, in *Fractal Image Encoding and Analysis*, edited by Y. Fisher, NATO ASI Series F 159, Springer Verlag, New York (1998).
11. M. Ghazel, G. Freeman and E.R. Vrscay, Fractal image denoising, *IEEE Transactions on Image Processing* **12**, no. 12, 1560-1578, 2003.
12. J. Hutchinson, Fractals and self-similarity, *Indiana Univ. J. Math.* **30**, 713-747 (1981).
13. A. Jacquin, Image coding based on a fractal theory of iterated contractive image transformations, *IEEE Trans. Image Proc.*, **1** 18-30 (1992).
14. N. Lu, *Fractal Imaging*, Academic Press, New York (1997).
15. P. Perona and J. Malik, Scale-space and edge detection using anisotropic diffusion, *IEEE Trans. PAMI* **12**, 629-639 (1990).
16. L. Rudin, S. Osher and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D* **60**, 259-268 (1992).
17. G. Sapiro, *Geometric partial differential equations and image analysis*, Cambridge University Press, New York (2001).
18. D. Youla and H. Webb, *Image restoration by the method of convex projections: Part 1-Theory*, *IEEE Transactions on Medical Imaging*, vol. MI-1, no. 2, pp. 81-94, October 1982.

A Fast Algorithm for Macroblock Mode Selection in H.264 Video Coding

Donghyung Kim¹, Jongho Kim¹, Seungjong Kim², and Jechang Jeong¹

¹ Department of Electrical and Computer Engineering, Hanyang University,
17 Haengdang, Seongdong, Seoul 133-791, Korea
{kimdh, angel, jjeong}@ece.hanyang.ac.kr

² Department of Computer Information, Hanyang Woman's College,
17 Haengdang, Seongdong, Seoul 133-793, Korea
jkim@hywoman.ac.kr

Abstract. To improve coding efficiency, the H.264 video coding standard uses new coding tools, such as variable block size, quarter-pixel-accuracy motion estimation, multiple reference frames, intra prediction and a loop filter. Using these coding tools, H.264 achieves significant improvement in coding efficiency compared with existing standards. However, encoder complexity also increases tremendously. Among the tools, macroblock mode selection and motion estimation contribute most to total encoder complexity. This paper focuses on complexity reduction in macroblock mode selection. Of the macroblock modes which can be selected, inter8×8 and intra4×4 have the highest complexity. We propose two methods for complexity reduction of inter8×8 and intra4×4 by using the costs of the other macroblock modes. Simulation results show that the proposed methods save about 53% of total encoding time compared with the H.264 reference implementation, whereas the average PSNR decreases less than 0.05 dB.

1 Introduction

The H.264/AVC standard is a video compression standard that was jointly developed by the ITU-T Video Coding Experts Group and the ISO/IEC Motion Picture Experts Group [1]. To improve coding efficiency, H.264 adopts new coding tools, such as quarter-pixel-accuracy motion estimation (ME), multiple reference frames, a loop filter, variable block size (VBS), etc. [2], [3]. These tools have enabled the standard to achieve higher coding efficiency than prior video coding standards. The encoder complexity, however, increases tremendously.

Several approaches have been proposed to reduce the complexity of the H.264 encoder. Yin et al. proposed a method to alleviate encoder complexity caused by ME and macroblock mode selection [4]. Their low complexity ME algorithm consists of two steps. First, integer-pixel ME is carried out using enhanced prediction zonal search (EPZS). Then, depending on the result of the integer-pixel ME, sub-pixel ME is carried out within some limited areas. To achieve faster macroblock mode selection, their method simply examines limited modes based on the costs of inter16×16, inter8×8, and inter4×4. Huang et al. proposed an algorithm to reduce the time to search the reference frames for ME complexity reduction [5]. For each macroblock, they analyze the available information after intra prediction and ME from the previous frame to

determine whether it is necessary to search more frames. Their method can save about 10-67% of ME computation. Ahmad et al. proposed a fast algorithm for macroblock mode selection based on a 3D recursive search algorithm that takes cost into account as well as the previous frame information [6]. This algorithm leads to a decrease of over 30% in encoding time compared with the H.264 reference implementation. The bitstream length, however, increases by about 15%.

To speed up the H.264 encoding time, we focus on complexity reduction of macroblock mode selection. When an 8x8 DCT is not used, the candidate macroblock modes are SKIP, inter16x16, inter16x8, inter8x16, inter8x8, intra16x16, and intra4x4. An inter8x8 mode can be further partitioned into four sub-macroblock modes: inter8x8, inter8x4, inter4x8, and inter4x4. Among these modes, inter8x8 and intra4x4 modes contribute most to the complexity, especially when rate-distortion optimization (RDO) is used.

In this paper, we propose two algorithms. One is to alleviate inter8x8 complexity. It estimates four sub-macroblock modes within inter8x8 by using the costs of other inter modes with relatively low complexity. The other method reduces intra4x4 complexity, using the similarity between RD costs of two intra modes.

2 Mode Selection Algorithm in the H.264 Reference Software

2.1 Macroblock and Sub-macroblock Modes

The H.264 standard allows the following macroblock modes: SKIP, inter16x16, inter16x8, inter8x16, inter8x8, intra16x16, intra8x8, and intra4x4. Furthermore, each

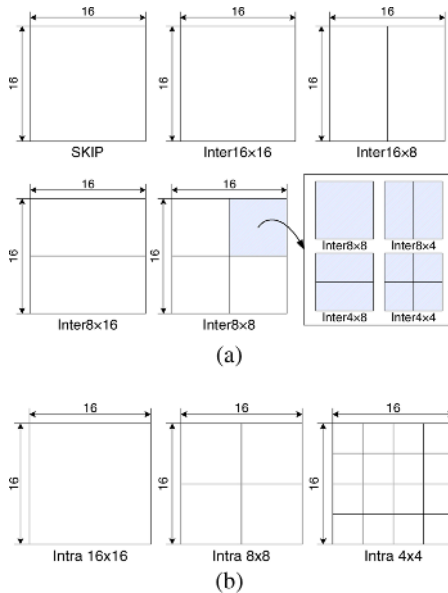


Fig. 1. Macroblock partitions of (a) inter and (b) intra modes

block within inter8x8 can be divided into four sub-macroblock modes. The allowed sub-macroblock modes are inter8x8, inter8x4, inter4x8, and inter4x4. Figure 1 depicts the macroblock partitions of inter and intra macroblock modes, including the SKIP mode.

An inter16x16 mode has only one motion vector, whereas inter16x8 and inter8x16 have two motion vectors. An inter8x8 mode may have 4-16 motion vectors depending on the selected sub-macroblock modes. A SKIP mode refers to the mode where neither motion vector nor residual is encoded. Three intra modes have different prediction modes. Four prediction modes are available in intra 16x16, and nine prediction modes are available in intra8x8 and intra4x4.

2.2 Macroblock Mode Selection in the Reference Software

The reference software, JM9.3 [7], supports three cost calculation criteria: motion vector (MV) cost, reference frame (REF) cost, and rate distortion (RD) cost. The MV cost is calculated using a lambda factor and is defined as:

$$\begin{aligned} \text{MVCost} &= \text{WeightedCost}(f, \text{mvbits}[(cx \ll s) - px] \\ &\quad + \text{mvbits}[(cy \ll s) - py]) \\ \text{where} \\ f &: \text{lambda factor} \\ cx, cy &: \text{candidate } x \text{ and } y \text{ position for ME} \\ px, py &: \text{predicted } x \text{ and } y \text{ position for ME} \end{aligned} \quad (1)$$

The REF cost is also calculated using a lambda factor and is defined as:

$$\begin{aligned} \text{REFcost} &= \text{WeightedCost}(f, \text{refbits}(\text{ref})) \\ \text{where } f &: \text{lambda factor} \end{aligned} \quad (2)$$

In (1) and (2), *WeightedCost*() returns the cost for the bits of motion vector and reference frame, respectively. Finally, the RD cost is defined as:

$$\begin{aligned} \text{RDcost} &= \text{Distortion} + \lambda \cdot \text{Rate} \\ \text{where } \lambda &: \text{Lagrange multiplier} \end{aligned} \quad (3)$$

In (3), the distortion is computed by calculating the SNR of the block and the rate is calculated by taking into consideration the length of the stream after the last stage of encoding.

When RDO and five reference frames are used, using these cost functions, the process of macroblock mode selection in the reference software is as follows:

Step 1. Find reference frames and motion vectors for each block in inter16x16, inter16x8, and inter8x16.

$$\begin{aligned} [\mathbf{MV}_i, \mathbf{REF}_i] &= \arg \min_{\mathbf{MV}, \mathbf{REF}} (\text{MVCost}(\mathbf{MV}) + \text{REFcost}(\mathbf{REF})) \\ \text{where } i &= \text{inter16} \times 16, \text{ inter16} \times 8, \text{ inter8} \times 16. \\ \mathbf{MV} &\in \{\text{Search Range}\}, \mathbf{REF} \in \{0, 1, \dots, 4\} \\ \mathbf{MV}_i &: \text{Motion vector set in } i \text{ mode} \\ \mathbf{REF}_i &: \text{Reference frame set in } i \text{ mode} \end{aligned} \quad (4)$$

Step 2. Calculate the sums of MV cost and REF cost in inter16×16, inter16×8, and inter8×16.

$$\begin{aligned}
J_i &= \text{MVcost}(\mathbf{MV}_i) + \text{REFcost}(\mathbf{REF}_i) \\
&\text{where } i = \text{inter16} \times 16, \text{inter16} \times 8, \text{inter8} \times 16. \\
J_i &: \text{the sum of MV cost and REF cost in } i \text{ mode.}
\end{aligned} \tag{5}$$

Step 3. Find reference frames and motion vectors for the first sub-macroblock in inter8×8.

$$\begin{aligned}
[\mathbf{MV}_i, \mathbf{REF}_i] &= \arg \min_{\text{MV, REF}} (\text{MVcost}(MV) + \text{REFcost}(REF)) \\
&\text{where } i = \text{inter8} \times 8, \text{inter8} \times 4, \text{inter4} \times 8, \text{inter4} \times 4.
\end{aligned} \tag{6}$$

Step 4. Calculate the sums of MV cost and REF cost for the first sub-macroblock in inter8×8.

$$\begin{aligned}
J_i &= \text{MVcost}(\mathbf{MV}_i) + \text{REFcost}(\mathbf{REF}_i) \\
&\text{where } i = \text{inter8} \times 8, \text{inter8} \times 4, \text{inter4} \times 8, \text{inter4} \times 4.
\end{aligned} \tag{7}$$

Step 5. Select the mode for the first sub-macroblock in inter8×8.

$$\begin{aligned}
&\text{Sub-macroblock mode} = \\
&\arg \min_{\text{mode}} (\text{RDcost}(\text{inter8} \times 8), \text{RDcost}(\text{inter8} \times 4) \\
&\quad, \text{RDcost}(\text{inter4} \times 8), \text{RDcost}(\text{inter4} \times 4))
\end{aligned} \tag{8}$$

Step 6. Repeat steps 3 to 5 for the other sub-macroblocks in inter8×8.

Step 7. Select the macroblock mode

$$\begin{aligned}
&\text{Macroblock mode} = \\
&\arg \min_{\text{mode}} (\text{RDcost}(SKIP), \text{RDcost}(\text{inter16} \times 16) \\
&\quad, \text{RDcost}(\text{inter16} \times 8), \text{RDcost}(\text{inter8} \times 16), \text{RDcost}(\text{inter8} \times 8) \\
&\quad, \text{RDcost}(\text{intra16} \times 16), \text{RDcost}(\text{intra4} \times 4))
\end{aligned} \tag{9}$$

In steps 1 and 2, the reference software finds reference frames and motion vectors which minimize the sum of MV cost and REF cost in inter16×16, inter16×8, and inter8×16. Steps 3 to 6 are the process of selecting sub-macroblock modes in inter8×8. The final step decides the macroblock mode by comparing RD costs of all macroblock modes.

3 Proposed Algorithm

3.1 Complexity Reduction of Inter8×8

Since each sub-macroblock within inter8×8 needs additional RD cost computations for the selection of sub-macroblock modes, inter8×8 has the highest complexity among all of the inter macroblock modes. For complexity reduction of inter8×8, we assume that the costs of inter macroblock modes monotonically increase or decrease according to their partitioned direction. Under this assumption, we restrict selectable sub-macroblock modes by using the MV costs and REF costs of inter16×16,

inter16x8, and inter8x16. For example, if the sum of MV and REF costs of inter16x16 is larger than that of inter16x8 and is smaller than that of inter8x16, we consider only inter8x8 and inter8x4 as sub-macroblock modes. Figure 2 depicts the proposed method for the complexity reduction of inter8x8.

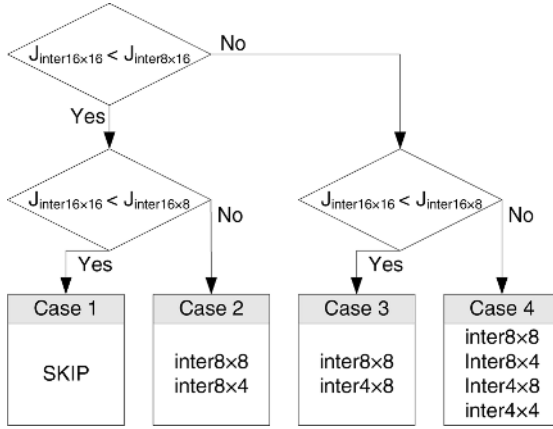


Fig. 2. Block diagram for the restriction of selectable sub-macroblock modes

In case 1, since $J_{inter16x16}$ is smaller than both $J_{inter16x8}$ and $J_{inter8x16}$, neither additional block partition in the horizontal direction nor in the vertical direction is needed. In this case we do not consider any sub-macroblock mode, and step 3 to step 6 in the reference software are skipped. In case 2, since $J_{inter16x16}$ is smaller than $J_{inter8x16}$ and is larger than $J_{inter16x8}$, additional block partitioning is only considered in the vertical direction. In this case, either inter8x8 and inter8x4 is selected as a sub-macroblock mode, and the formulae of steps 3 to 6 in the reference software are modified as follows:

$$[MV_i, REF_i] = \underset{MV, REF}{\operatorname{argmin}} (MVcost(MV) + REFcost(REF)) \tag{10}$$

where $i = inter8x8, inter4x8$.

$$J_i = MVcost(MV_i) + REFcost(REF_i) \tag{11}$$

where $i = inter8x8, inter8x4$.

$$\begin{aligned} \text{Sub-macroblock mode} = \\ \underset{mode}{\operatorname{argmin}} (RDcost(inter8x8), RDcost(inter8x4)) \end{aligned} \tag{12}$$

In case 3, since $J_{inter16x16}$ is smaller than $J_{inter16x8}$ and is larger than $J_{inter8x16}$, only additional block partition is considered, and only in the horizontal direction. In this case, either inter8x8 and inter4x8 is selected as a sub-macroblock mode, and the formulae of steps 3 to 6 in the reference software are modified as follows:

$$[\mathbf{MV}_i, \mathbf{REF}_i] = \underset{MV, REF}{\operatorname{arg\,min}} (MV\text{cost}(MV) + \text{REFcost}(REF))$$

where $i = \text{inter}8 \times 8, \text{inter}4 \times 8$. (13)

$$J_i = MV\text{cost}(\mathbf{MV}_i) + \text{REFcost}(\mathbf{REF}_i)$$

where $i = \text{inter}8 \times 8, \text{inter}4 \times 8$. (14)

$$\text{Sub-macroblock mode} = \underset{\text{mode}}{\operatorname{arg\,min}} (\text{RDcost}(\text{inter}8 \times 8), \text{RDcost}(\text{inter}4 \times 8))$$
(15)

In case 4, since $J_{\text{inter}16 \times 16}$ is larger than both $J_{\text{inter}16 \times 8}$ and $J_{\text{inter}8 \times 16}$, we consider all sub-macroblock modes, as in the reference software.

3.2 Complexity Reduction of Intra4x4

When 8x8 DCT is not used, the allowed intra modes are intra4x4 and intra16x16. Of the two intra modes, intra4x4 has the higher complexity because it has more prediction modes. Since intra16x16, as described in Section 2, has only four prediction modes and intra4x4 has nine prediction modes for finer granularity, intra4x4 generally yields a smaller prediction error than intra16x16. However, most of the macroblocks have only a small difference between the RD costs of intra16x16 and intra4x4. This is because edges directed in vertical or horizontal directions are dominant in natural images, which are considered in intra16x16.

Using this characteristic, we first find the inter mode with a minimum RD cost. Then we compare the RD cost of the selected inter mode with that of intra16x16. If the RD cost of intra16x16 is much larger, that is, if Eq. (16) is true, then the RD cost computation of intra4x4 is skipped:

$$\text{Min}[\text{RDcost}(\text{inter modes})] \cdot K < \text{RDcost}(\text{intra } 16 \times 16)$$
(16)

In (16), K is a constant. Table 1 describes the missing rate of intra4x4. The missing rate indicates the probability that the skipped intra4x4 has the smallest RD cost. As shown in Table 1, the average missing rate is only about 0.7% for K = 1.5. This means that the RD cost difference factor between intra4x4 and intra16x16 is less than 1.5 for 99.3% of the macroblocks.

Table 1. Missing rates of intra4x4 according to K

| Sequences | Missing Rate (%) | | |
|------------|------------------|-------|-------|
| | K=1.3 | K=1.5 | K=1.7 |
| Coastguard | 2.0 | 0.4 | 0.2 |
| Container | 2.7 | 1.0 | 0.9 |
| Mobile | 0.4 | 0.0 | 0.0 |
| News | 5.8 | 0.6 | 0.5 |
| Salesman | 6.6 | 0.4 | 0.0 |
| Silent | 4.7 | 1.7 | 0.4 |
| Stefan | 2.8 | 0.2 | 0.0 |
| Trevor | 9.0 | 0.9 | 0.1 |

4 Simulation Results

Since the proposed methods for complexity reduction of inter8x8 and intra4x4 are uncorrelated, the two methods can be applied independently or simultaneously. We applied the two proposed algorithms simultaneously to encode test sequences. For the purpose of evaluation, the public reference encoder JVT Model (JM) v.9.3 was used. The software was tested on an Intel Pentium-IV based computer with 512 MB RAM under the Windows XP Professional operating system.

We adopted full search for ME, used RDO, and set Quantization Parameter (QP) and K in (16) to 28 and 1.5, respectively. The simulation was performed on eight standard video sequences in QCIF (176x144) format. These included Coastguard, Container, Mobile, News, Salesman, Silent, Stefan, and Trevor. These sequences were selected on the basis of length of encoded streams and degree of motion. The first 100 frames of each of these sequences were used.

For 99 P-frames, Tables 2 and 3 describe the reduction ratios of the number of RD cost computations in inter8x8 and intra4x4. As shown in these results, we can save about 72% and 90% of the RD cost computations, respectively.

Table 2. The number of RD cost computation in inter8x8

| Sequences | Reference Software | Proposed Method | Reduction Ratio (%) |
|------------|--------------------|-----------------|---------------------|
| Coastguard | 156,816 | 59,760 | 61.9 |
| Container | 156,816 | 17,896 | 88.6 |
| Mobile | 156,816 | 65,360 | 58.3 |
| News | 156,816 | 30,256 | 80.7 |
| Salesman | 156,816 | 28,224 | 82.0 |
| Silent | 156,816 | 40,464 | 74.2 |
| Stefan | 156,816 | 56,472 | 64.0 |
| Trevor | 156,816 | 54,800 | 65.1 |

Table 3. The number of RD cost computation in intra4x4

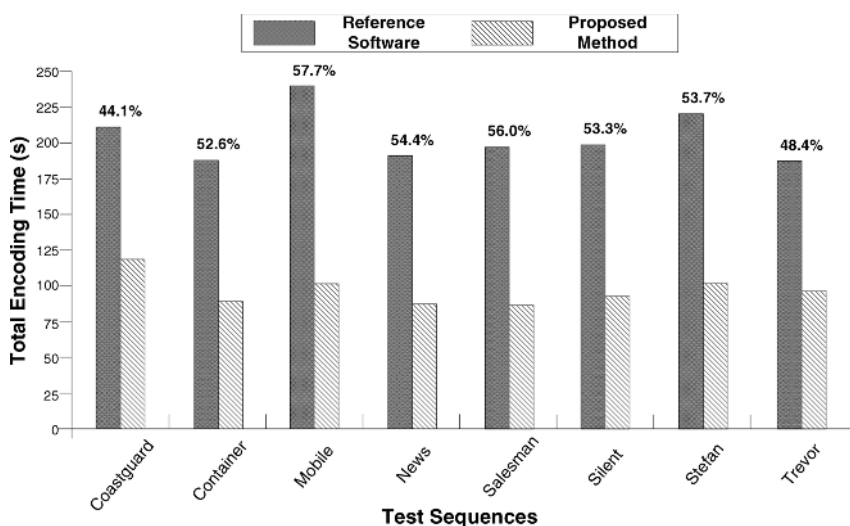
| Sequences | Reference Software | Proposed Method | Reduction Ratio (%) |
|------------|--------------------|-----------------|---------------------|
| Coastguard | 35,343 | 10,838 | 69.3 |
| Container | 35,343 | 6,566 | 81.4 |
| Mobile | 35,343 | 400 | 98.9 |
| News | 35,343 | 2,225 | 93.7 |
| Salesman | 35,343 | 529 | 98.5 |
| Silent | 35,343 | 2,388 | 93.2 |
| Stefan | 35,343 | 3,038 | 91.4 |
| Trevor | 35,343 | 3,463 | 90.2 |

Table 4. Comparison of bitrates (Kbits/sec)

| Sequences | Reference Software | Proposed Method | Increase Ratio (%) |
|------------|--------------------|-----------------|--------------------|
| Coastguard | 249.00 | 251.28 | 0.9 |
| Container | 40.16 | 40.74 | 1.4 |
| Mobile | 496.49 | 497.24 | 0.2 |
| News | 75.84 | 76.75 | 1.2 |
| Salesman | 56.89 | 57.61 | 1.3 |
| Silent | 82.69 | 83.71 | 0.2 |
| Stefan | 379.26 | 380.86 | 0.4 |
| Trevor | 132.49 | 133.58 | 0.8 |

Table 5. Comparison of PSNRs (dB)

| Sequences | Reference Software | Proposed Method | Increase Ratio (%) |
|------------|--------------------|-----------------|--------------------|
| Coastguard | 33.93 | 33.89 | 0.04 |
| Container | 36.07 | 36.06 | 0.01 |
| Mobile | 33.14 | 33.06 | 0.08 |
| News | 36.65 | 36.64 | 0.01 |
| Salesman | 35.57 | 35.54 | 0.03 |
| Silent | 35.84 | 35.81 | 0.03 |
| Stefan | 34.22 | 34.15 | 0.07 |
| Trevor | 36.40 | 36.33 | 0.07 |

**Fig. 3.** Comparison of total encoding time

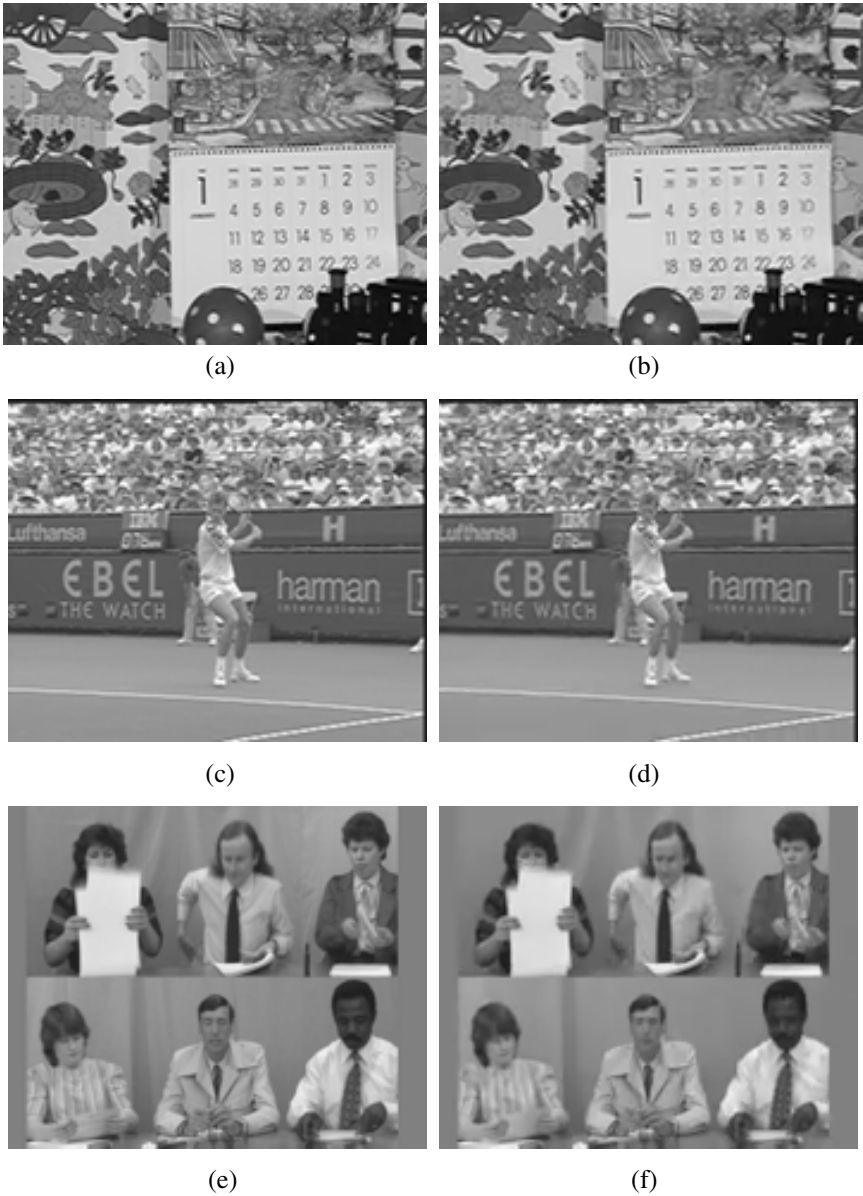


Fig. 4. The 10th reconstructed frames of three sequences when using (a) JM (33.338dB) (b) the proposed method (33.304dB) (c) JM (34.778dB) (d) the proposed method (34.690dB) (e) JM (36.019dB) (f) the proposed method (35.946dB)

Tables 4 and 5 compare the bitrates and PSNRs for each test sequence. Since the reference implementation is an exhaustive search for selecting the macroblock mode, the number of encoded bits is the least for each sequence. Tables 4 and 5 show the

average increase of the total bitrates is only about 0.9%, and the average PSNR drop is only about 0.043 dB when using the proposed method.

Finally, Fig. 3 compares total encoding time from the proposed method with that from the reference software. This result shows a substantial decrease of about 53% in total encoding time compared with the reference implementation.

Figure 4 compares subjective qualities of three sequences which have comparatively higher PSNR drop. Each frame in Fig. 4 is the 10th reconstructed frame, that is to say, the 9th P-frame. As shown in this figure, the difference between subjective qualities of each sequence is nearly unrecognizable.

5 Conclusions

We proposed two simple and effective schemes for the quick selection of macroblock modes in H.264 video coding. Using our methods, the RD cost computations of inter8×8 and intra4×4 were reduced by about 72% and 90%, respectively. Both schemes can be applied independently. When both methods are used simultaneously, simulation results show that our methods can save about 53% of total encoding time regardless of input sequences, yet the average increased rate of the total bits and average PSNR drop are only about 0.9% and 0.043 dB, respectively. This huge reduction of encoder complexity may be useful in real-time implementation of the H.264/AVC standard.

Acknowledgement

This research was supported by Seoul Future Contents Convergence (SFCC) Cluster established by Seoul Industry-Academy-Research Cooperation Project.

References

1. Wiegand, T.: Version 3 of H.264/AVC. Doc. JVT-K051 (2004)
2. Wiegand, T., Sullivan, G. J.: Overview of the H.264/AVC Video Coding Standard. *IEEE Trans. Circuits Syst. for Video Technol.*, Vol. 13. (2003) 560-576
3. Wiegand, T., Schwarz, H., Joch, A., Kossentini, F.: Rate-Constrained Coder Control and Comparison of Video Coding Standard. *IEEE Trans. Circuits Syst. for Video Technol.*, Vol. 13. (2003) 688-703
4. Yin, P., Tourapis, H. C., Tourapis, A. M., Boyce, J.: Fast Mode Decision and Motion Estimation for JVT/H.264. *ICIP'03*, Vol. 3. (2003) 853-856
5. Huang, Y. W., Hsieh, B. Y., Whang, T. C., Chien, S. Y., Ma, S. Y., Shen, C. F., Chen, L. G.: Analysis and Reduction of Reference Frames for Motion Estimation in MPEG-4 AVC/JVT/H.264. *ICASSP'03*, Vol. 3 (2003) 145-148
6. Ahmad, A., Khan, N., Masud, S., Maud, M.A.: Efficient Block Size Selection in H.264 Video Coding Standard. *Electronics Letters*, Vol. 40. (2004) 19-21
7. JM9.3: <http://bs.hhi.de/~suehring/tml/download/jm93.zip>.

Visual Secret Sharing Scheme: Improving the Contrast of a Recovered Image Via Different Pixel Expansions

Ching-Nung Yang and Tse-Shih Chen

Department of Computer Science and Information Engineering
National Dong Hwa University
#1, Da Hsueh Rd, Sec. 2, Hualien, 974, Taiwan
Tel.: +886-3-8634025, Fax: +886-3-8634010
cnyang@mail.ndhu.edu.tw

Abstract. Visual secret sharing (VSS) scheme is an image sharing scheme that divides a secret pixel into several sub pixels (the number of sub pixels is called the pixel expansion) and the secret image is visually revealed without additional computations. However the contrast of recovered image is poor and thus the VSS framework is always a research issue and not a practical scheme. In this paper, we introduce a new framework which prioritizes the pixels with different pixel expansions to reconstruct a high-quality image for practical use. Also, integrating the optical character recognition with the proposed VSS scheme we show a new automated visual authentication scheme.

1 Introduction

In the so-called (k, n) visual secret sharing (VSS) scheme, which was first proposed by Naor and Shamir [1], a secret image is divided into n shadow images (shadows). This (k, n) -threshold scheme satisfies the perfect secrecy that requires at least k shadows for the secret image reconstruction. VSS schemes decode the secret image by the visual sight directly and do not need any computation or cryptography background. VSS schemes share each secret pixel by m sub pixels and thus the shadow size is expanded, where the value m is called the pixel expansion. For example, the pixel expansions of the $(2, 2)$, $(2, n)$, $(3, n)$ and the optimal (k, k) Naor-Shamir VSS schemes are 2 , n , $2n-2$ and 2^{k-1} , respectively [1]. Size-reduced VSS schemes were proposed [2-7]. Some of them even had the non-expandible shadow size (i.e., $m=1$). As proposed in [1-11], all previous VSS schemes used the same pixel expansion for each secret pixel. In this paper, we first introduce a new framework which uses the different number of sub pixels to represent the secret pixel according to its "importance". The different pixel expansion performs the appropriate roles, i.e., the large pixel expansion improves the contrast and the small pixel expansion reduces the shadow size. Finally, we realize a VSS scheme with the high-contrast recovered image and the less shadow size.

Additionally, for measuring the contrast of VSS schemes, some previous definitions of contrasts were defined by means of the whiteness of black and white secret pixels and the pixel expansion but there is no consistency among these definitions due to the bias of human visual system. With the help of machine to

recognize the printed text image, e.g., the optical character recognition (OCR) system, we design a simulated human visual system to measure the contrast and avoid the personal bias. Extending this OCR-like human visual system, a new novel VSS-based visual authentication system is also proposed. In brief, there are two major advantages for our VSS scheme based on different pixel expansions: (1) any certain shadow size (2) the high-contrast recovered image. Both advantages imply the wide applicability of our proposed scheme.

The rest of paper is organized as follows. In Section 2, the previous VSS schemes and the motivation of our work are described. In Section 3, our new framework is proposed. Also, a fairer measurement is defined to compare the recovered printed text image with the help of the OCR technique. Experiments for two types of secret images (the printed text image and binary halftone picture) are given in Section 4. Section 5 shows the practicability of our proposed scheme and describes a new VSS-based authentication system. We draw the conclusions in Section 6.

2 Preliminaries

2.1 Previous VSS Schemes

For the conventional (k, n) VSS scheme [1], the dealer uses two sets, C_1 and C_0 , including matrices which are obtained from the column permutation of black and white $n \times m$ Boolean matrices B_1 and B_0 , respectively. To share a black (resp. white) secret pixel, one matrix in C_1 (resp. C_0) is randomly chosen. Then a secret pixel is divided into m sub pixels according the chosen matrix $S=[s_{ij}]$ where $s_{ij}=1$ is for the j th sub pixel in the i th shadow is black; otherwise white. VSS schemes need to satisfy the contrast condition and security condition. When stacking any k or more shadows, the gray level of the secret pixel can be visually decoded by OR-ing the corresponding rows in S . If the Hamming weight of the OR-ed m -tuple V , $H(V)$, is greater or equal to $(m-l)$, this gray level is interpreted as black and if less or equal to $(m-h)$ the pixel is white, where h and l are the whiteness of white and black pixels. When stacking r (less than k) shadows, the two collections of $r \times m$ matrices obtained by restricting each $n \times m$ matrices in C_1 and C_0 to rows i_1, i_2, \dots, i_r are not visual in the sense that they contain the same matrices with the same frequencies. Different h, l and m result in different resolutions of recovered images.

Some modified VSS schemes were proposed to further reduce the pixel expansion and meantime hold the contrast and security conditions. Kuwakado and Tanaka [2] discovered that there exist the homogeneous columns in the chosen matrices which can be deleted to reduce the pixel expansion. Two VSS schemes based on 2-pixel and m -pixel encoding methods [3, 4] were proposed such that the pixel expansion is reduced to $m/2$ and 1, respectively. A completely different approach from the 'deterministic' VSS scheme [1-4] was the 'probabilistic' method [5, 6]. The probabilistic schemes have no pixel expansion, i.e., the shadow size is the same to the secret image. The dealer randomly chooses the black or white pixel to represent the pixel in the secret image according to the probability matrices [6]. Finally, one can "view" the recovered image due to the appearance frequencies while the edge of recovered image is irregular. Cimato et al. [7] generalized the probabilistic VSS scheme in [6] to adjust the pixel expansion between 1 to m for trading the shadow size

with the contrast. From the above description, for a (k, n) Naor-Shamir VSS scheme with the pixel expansion m_{NS} [1], the modified VSS schemes with m between 1 to m_{NS} can be obtained [2-7].

2.2 Motivation

From the recovered image, it is observed that the large m achieves the high resolution of recovered image but causes the expansible shadows. For example, the Naor-Shamir optimal $(8, 8)$ VSS scheme has an incredible pixel expansion $m=2^{(8-1)}=128$ and this is not practicable for use. The small m results in the small shadow size but distorts the clearness of recovered image. However, no matter what pixel expansion was used all previous VSS schemes used the same expansion in one shadow image. To be possessed of the advantages of both large and small m and avoid their weakness, we use each in its proper purpose. The large m for the more “important” pixels is to keep the resolution of image and the small m for the less “important” pixels is to reduce the shadow size. By using the different pixel expansions for the pixels with the different importance levels, we finally get a VSS scheme with the high-contrast recovered image and the small shadows.

What are the important pixels in a secret image? In general, we may use the pixels located at the edges as the more important pixels. This is due to the fact that the most important and meaningful information in an image is the edge. Figs. 1(a-1) and (b-1) are the black/white printed text image and gray Lena picture with size 300×300 pixels. Figs. 1(a-2) and (b-2) are their edges obtained from the Sobel edge detector [8]. It is evident that the dots in Figs. 1(a-2) and (b-2) carry the important information of Figs. 1(a-1) and (b-1), respectively. By choosing these dots as the more important pixels and other remaining pixels as the less important pixels and assigning different pixel expansions for them, the quality of recovered image is assured and the shadow size is reduced simultaneously.

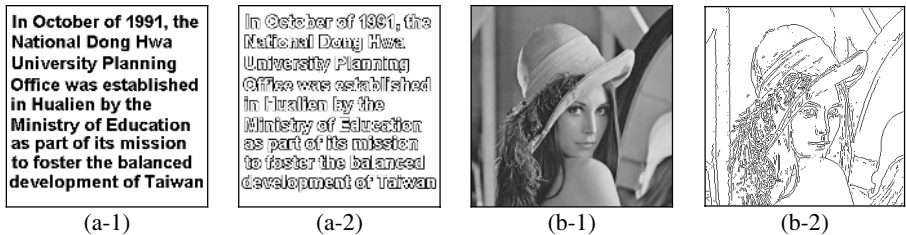


Fig. 1. Results of edge detector for the printed text and Lena picture

3 The Proposed VSS Scheme with Different Pixel Expansions

3.1 Basic Concept

In this new VSS framework, we do not design the new matrices but propose a methodology that uses the existing schemes with different pixel expansions

simultaneously to improve the contrast for a certain shadow size. Our approach is effective regardless of the black/white, gray or colored schemes. Thus, to clarify the concept of proposed framework, we herein only deal with the black/white secret images (the black/white printed text image and the binary halftone picture) for simplicity.

An illustration of using the same and different pixel expansions in one shadow is shown in Fig. 2. A simple example, a secret image with two pixels \blacksquare and \square is shown in Fig. 2(a). When using the same pixel expansion like the conventional VSS scheme, Figs. 2(b-1) and (b-2) are the examples for $m=2$ and 4, respectively. Suppose the pixel \square is more important than the pixel \blacksquare . We use two sub pixels and four sub pixels to represent pixels \blacksquare and \square , respectively. Fig. 2(c) shows four possible patterns to arrange these six sub pixels with the corresponding positions of the associative secret pixels. It is evident that Fig. 2(c-1) is the best arrangement among these four patterns and Fig. 2(c-4) is a bad one because the co-locations between the original secret pixel and the corresponding sub pixels are not consistent and this will distort the clearness.

There is no pixel arrangement problem for using the same pixel expansion while we should carefully arrange the co-locations of sub pixels for our methodology using the different pixel expansion. A measurement of co-location, i.e., the overlapping percentage between a sub pixel relative to its associative secret pixel in [9] can be adopted for our use. Let $W_{(x,y)}^{(u,v)}$ be the overlapping percentage of a sub pixel located at the coordinate (u, v) relative to a secret pixel located at the coordinate (x, y) . For example, to calculate the overlapping percentages for Fig. 2(c-1), we redraw Fig. 2(c-1) pattern in Fig. 3. The values of weighting $W_{(1,1)}^{(u,v)}$ and $W_{(1,2)}^{(u,v)}$, where $u \in [1, 2]$, $v \in [1, 3]$, and $(x, y) = (1, 1)$ and $(1, 2)$ are pixels \blacksquare and \square are shown below:

$$W_{(1,1)}^{(1,1)} = 1.0; W_{(1,1)}^{(1,2)} = 0.0; W_{(1,1)}^{(1,3)} = 0.0; W_{(1,1)}^{(2,1)} = 1.0; W_{(1,1)}^{(2,2)} = 0.0; W_{(1,1)}^{(2,3)} = 0.0;$$

$$W_{(1,2)}^{(1,1)} = 0.0; W_{(1,2)}^{(1,2)} = 0.5; W_{(1,2)}^{(1,3)} = 1.0; W_{(1,2)}^{(2,1)} = 0.0; W_{(1,2)}^{(2,2)} = 0.5; W_{(1,2)}^{(2,3)} = 1.0.$$

For example, the sub pixel $(u, v)=(1, 2)$ is not overlapped with the secret pixel $(x, y)=(1, 1)$ but half overlapped with the secret pixel $(x, y)=(1, 2)$. So, $W_{(1,1)}^{(1,2)} = 0$ and $W_{(1,2)}^{(1,2)} = 0.5$. The average overlapping percentage for Fig. 2(c-1) is then

$$\bar{W} = \left(\sum_{All (u,v)} W_{(x,y)}^{(u,v)} \right) / 6 = (W_{(1,1)}^{(1,1)} + W_{(1,2)}^{(1,2)} + W_{(1,2)}^{(1,3)} + W_{(1,1)}^{(2,1)} + W_{(1,2)}^{(2,2)} + W_{(1,2)}^{(2,3)}) / 6 = 5/6.$$

Using the same approach, the average overlapping percentages for Figs. 2(c-2), (c-3) and (c-4) are determined as $4/6$, $4/6$ and $3/6$, respectively. The higher overlapping percentage implies the less geometrical distortion-like effects in the recovered image [9]. The results show that the \bar{W} can really be used to determine the suitable pattern for our proposed VSS scheme. It is consistent with the immediate observation that Fig. 2(c-1) is a reasonable choice for the less distortion. Meantime,

the important pixel $\boxed{2}$ in Fig. 2(c-1) is represented by four sub pixels same as the Fig. 2(b-2) and the shadow size is less than Fig. 2(b-2). When compared to Fig. 2(b-1), we have the better resolution for pixel $\boxed{2}$. The average pixel expansions of Fig. 2(b-1), Fig. 2(b-2) and Fig. 2(c) are 2, 4 and 3, respectively. Our new framework is provided with the advantages of large and small pixel expansions simultaneously.

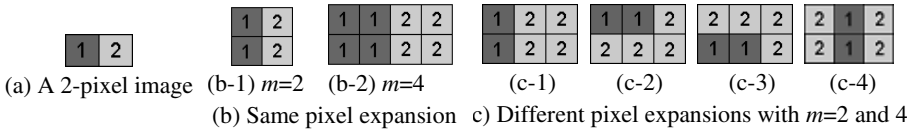


Fig. 2. A 2-pixel secret image and the co-locations of sub pixels in the shadow images using the same and different pixel expansions

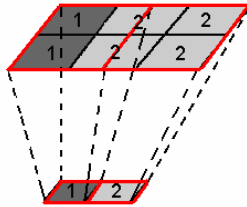


Fig. 3. The overlapping between sub pixels and secret pixels for Fig. 2(c-1) pattern

From this 2-pixel example, using different pixel expansions in the shadow images to improve the contrast and reduce the shadow size is achieved. However, for a large-sized secret image we need to take the following problems into account:

- (1) About “importance”, some concerns need to be considered: how many importance levels for the pixels we need and how to classify the pixels in a secret image?
- (2) How to assign different pixel expansions for the pixels with different importance?
- (3) What pixel expansions in the shadow images are used?
- (4) How to arrange the co-locations of sub pixels for achieving the high-contrast recovered image?
- (5) Because of using different pixel expansion, we should show how to fairly measure the contrast.

3.2 The Proposed VSS Scheme

Our aim in this paper is to solve the above problems. Considering problems (1) and (2), it is not appropriate to assign the same pixel expansion for the pixels with the same importance. When some important pixels are located around, if we assign them with the same pixel expansion then we may not find the sufficient sub pixels to represent the secret pixel in the overlapping areas. Therefore, our encoding algorithm prioritizes the pixels as the most pixel expansion as possible according to their importance, i.e., processes the most important pixels first and then the less important

pixels, and etc. About the importance of pixels, we can use the edge detectors to classify the pixels in the secret image into some categories by setting different thresholds. As the threshold increases the weak edges are removed and the strong edges are remained. For example, we may define a set of threshold levels ($\lambda_0, \lambda_1, \dots, \lambda_{L-1}$) in the Sobel edge detector to classify L -level importance. Because we do not assign the same pixel expansion for the pixels with the same importance level but prioritize the pixels according to their importance, so it is enough to divide the pixels into two categories (important and unimportant pixels). Consider two types of secret images experimented in this paper: the simple black/white printed text image and the binary halftone picture. Using any λ will detect the same edges for a simple black/white printed text image and the important pixels are simply the edges of texts (see Fig. 1(a-2)). For the binary halftone picture, we use a certain λ and the original gray picture to find the edges. The binary halftone picture obtained from the original gray picture is used as the secret image.

Considering problem (3), we first choose a (k, n) VSS scheme with the minimum pixel expansion m_C and the high contrast from the existing conventional VSS schemes, and then obtain other size-reduced VSS schemes with pixel expansions $1 \sim (m_C - 1)$ [2-7]. In the proposed VSS scheme, we may also choose $m=0$, i.e., discard a very unimportant secret pixel.

Considering problem (4), we should find a high average overlapping percentage to recover a high-contrast image and this will be done in the following encoding algorithm. Some notations are defined first:

- \tilde{I} The original gray secret picture with the size $(I_x \times I_y)$ used for finding the edges.
- I The binary halftone picture of \tilde{I} by the *halftoning* technique which is a technology to transfer a gray level picture into a black/white picture. Image I is used as a secret image.
- $E_\lambda(\cdot)$ The Sobel edge detector with a threshold λ .
- m_C The minimum pixel expansion for a VSS scheme with the high contrast chosen from the existing conventional (k, n) VSS schemes.
- $O^{(i)}$ The n output shadow images, $i \in [1, n]$, with the size $(O_u \times O_v)$ where $(O_u \times O_v) \leq (I_x \times I_y \times m_C)$.
- M The mapping pattern, a $(O_u \times O_v)$ matrix which elements are the location coordinates (x, y) in I .

Encoding algorithm:

Input: $\tilde{I}, I, \lambda, m_C$ and $(O_u \times O_v)$.

/* Note: for the black/whit printed text image, $I = \tilde{I}$ and any λ results in the same edges */

Output: $O^{(i)}, i \in [1, n]$.

Mapping:

(M-1): Find the important pixels $(x^{(1)}, y^{(1)})$ using $E_\lambda(\tilde{I})$ and let the remaining pixels $(x^{(2)}, y^{(2)})$ be unimportant pixels.

(M-2): Find the set of location coordinates (u, v) which overlaps the important (resp. unimportant) pixels, i.e., $S_{(x^{(1)}, y^{(1)})} = \{(u, v), \text{ which overlaps } (x^{(1)}, y^{(1)})\}$ (resp. $S_{(x^{(2)}, y^{(2)})} = \{(u, v), \text{ which overlaps } (x^{(2)}, y^{(2)})\}$).

(M-3): For $l = 1$ to 2 do

{ For all $(x^{(l)}, y^{(l)}) \in [I_x, I_y]$ and $(u, v) \in [O_u, O_v]$ do

{ For $m = m_C$ to 0 do

{ if no sufficient $m(u, v)$ exist in $S_{(x^{(l)}, y^{(l)})}$ break;

choose m non-chosen (u, v) with large $W_{(x^{(l)}, y^{(l)})}^{(u,v)}$ from $S_{(x^{(l)}, y^{(l)})}$

and let $(u, v) = (x^{(l)}, y^{(l)})$; };

(M-5): Let the non-chosen $(u, v) = *$ (dummy pixels);

(M-6): Determine the mapping pattern M .

Sharing:

(S-1): For different m , use the corresponding VSS schemes to share the secret pixel in I by m sub pixels.

(S-2): Generate $n(O_u \times O_v)$ -sized shadows according to the mapping pattern M .

Suppose there are N_I important pixels, N_U unimportant pixels in a secret image and $N_S = (I_x \times I_y)$ is the number of secret pixels, where $(N_I + N_U) = N_S$. Let $N_I^{(m)}$, $N_U^{(m)}$ and $N_S^{(m)}$, $0 \leq m \leq m_C$, be the number of pixels using the pixel expansion m in the important, unimportant pixels and the secret image where $N_I^{(m)} + N_U^{(m)} = N_S^{(m)}$. Then, the average

pixel expansions of important and unimportant pixels is $m_I = \left(\sum_{m=0}^{m_C} m \times N_I^{(m)} \right) / N_I$, $m_U = \left(\sum_{m=0}^{m_C} m \times N_U^{(m)} \right) / N_U$, and $m_A = \left(\left(\sum_{m=0}^{m_C} m \times N_S^{(m)} \right) + N_D \right) / N_S$, where N_D is the number of dummy pixels in the shadow image.

An example in Fig.4 illustrates the operation of encoding algorithm for a (2, 3) VSS scheme. Fig. 4(a) is a black/white printed text image “T” with $(I_x \times I_y) = (8 \times 8)$ pixels where there are 10 important pixels: 11~15, 21, 29, 37, 45, 53, and other 54 pixels are unimportant pixels. We want to generate three shadows with the size $(O_u \times O_v) = (10 \times 10)$ pixels. First, choose $m_C = 3$ and three (2, 3) VSS schemes with $m = 1, 2$ and 3: the Naor-Shamir scheme for $m = 3$, the Kuwakado-Tanaka scheme for $m = 2$ [2] and the Yang scheme for $m = 1$ [6]. Fig. 4(b) is the mapping patterns and it is observed that $N_I^{(0)} = 0$, $N_I^{(1)} = 1$, $N_I^{(2)} = 3$, $N_I^{(3)} = 6$ for these ten important pixels. For other 54 unimportant pixels, $N_U^{(0)} = 0$, $N_U^{(1)} = 33$, $N_U^{(2)} = 21$, $N_U^{(3)} = 0$. There is no dummy pixel in Fig. 4(b), i.e., $N_D = 0$.



Fig. 4. The mapping pattern M for the secret image “T” using the proposed encoding algorithm (a) the 8×8-pixel secret image with 10 important pixels (b) the 10×10-pixel mapping pattern

3.3 The Contrast

We had already solved problems (1)-(4) mentioned in Section 3.1. In this sub section, we address the last problem: *how to fairly measure the contrast of recovered image?* Previous theoretical definitions of contrasts were $(h - l)/m$, $(h - l)/(m(h + l))$ and $(h - l)/(m + l)$ in [1], [10] and [11], respectively. However, the contrast measurement is based on the very subjective measurement of human visual sight. Thus the inconsistency among the previous definitions may happen due to the personal subjective opinion.

In this paper, for the black/white printed text, we give a completely different measurement based on the “machine” such that the personal bias can be eliminated. Another care for the contrast measurement is that the more pixel expansion results in the high-contrast recovered image and the less pixel expansion causes the distortion. Therefore, just comparing the contrast but not caring the shadow size is not fair. From these two observations, by using the machine instead of human visual sight and considering the shadow size, the new measurement R (normalized recognition rate) is described as follows.

With the help of machine to recognize the printed text image, e.g., the OCR system, we can avoid the personal bias. When employing OCR, the recovered image is adjusted to the input of the OCR. The adjusting process includes removing the noise-like random dots and rescaling the recovered image to keep the aspect ratio of a character invariant. This adjusting process operates like the human visual system that overlooks the noisy random dots and concatenates the meaningful pixels. At this time, the combination of adjusting process and OCR algorithm simulates the ability of human visual system and shown in Fig. 5.

By using this OCR-like human visual system, a new contrast measurement is defined. Let the number of characters in a test image be N_1 , the number of recognized characters at OCR output be N_2 . Then the recognition rate is (N_2/N_1) . For comparison with the same condition, the recognition rate is normalized by dividing the average pixel expansion as $R = (N_2/N_1)/m_A$. Detail experiments and comparison will show that the normalized recognition rate R is effective for measuring the contrast.

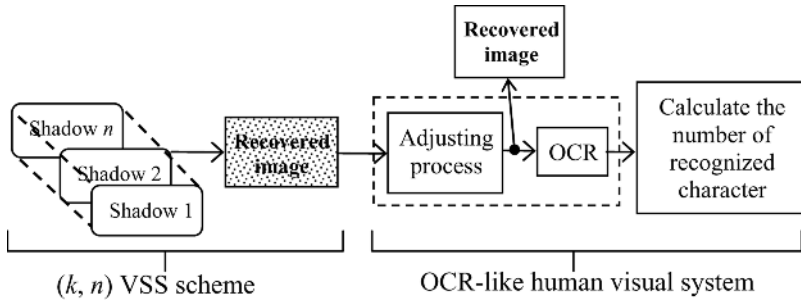


Fig. 5. Simulated human visual system

4 Experiments

4.1 Experimental Results for the Black/White Printed Text Image

For a black/white printed text image, we use the proposed (2, 3) VSS scheme to test the new contrast measurement R . A 300×300 -pixel printed text Fig. 1(a-1) is used as the secret image including 161 English characters. Fig. 1(a-2) shows that there are 12,989 important pixels about 14.43%. Choose the following shadow sizes for test: (300×300) , (600×300) , (900×300) , (425×425) , (520×520) , (240×375) , (750×240) , (750×360) . All recovered images are shown in Fig. 6. (Note: the recovered images are adjusted and rescaled as the original image to keep the aspect ration invariant for OCR.)

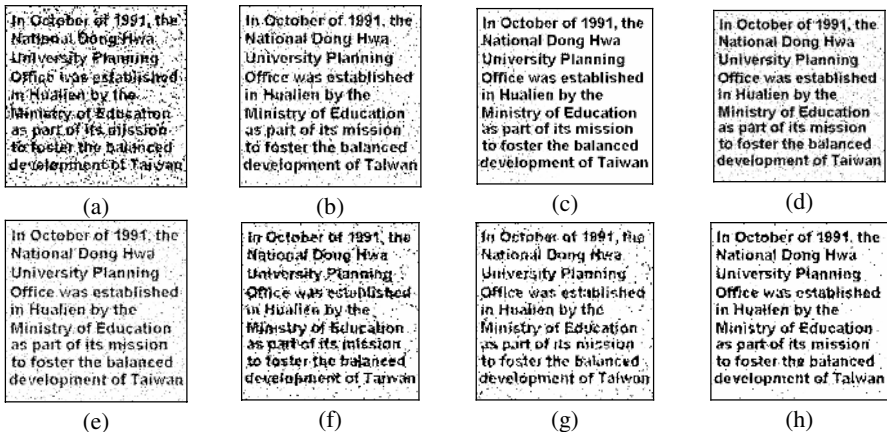


Fig. 6. (a)-(h) obtained from Figs. 7(a)-(h) by adjusting and rescaling

Recognize Fig. 6 by OCR. The numbers of recognized characters are 35, 95, 146, 145, 155, 53, 64 and 133, respectively, and their corresponding normalized recognition rates are 0.2174, 0.2950, 0.3022, 0.4503, 0.3209, 0.3292, 0.1988 and 0.2754. For these eight shadow sizes, the maximum number of recognized characters

is $N_2=155$ (the shadow size $(O_u \times O_v)=(520 \times 520)$) and the maximum number of normalized recognition rate is $R=0.4503$ (the shadow size $(O_u \times O_v)=(425 \times 425)$). The detail analysis for other shadow sizes can refer the full version.

4.2 Experimental Results for the Binary Halftone Picture

Also, we use a binary halftone picture, House, as a secret image, for testing the proposed (2, 3) VSS scheme. Suppose $(I_x \times I_y)=(300 \times 300)$ and $(O_u \times O_v)=(400 \times 400)$, i.e., $m_A=1.778$. A suitable threshold λ in the Sobel edge detector is chosen to achieve the maximum average pixel expansion of important pixels m_I . Fig. 7 shows this optimal case ($\lambda=0.04$ and the maximum $m_I=2.407$). Fig. 7(a) is the original gray secret image used for finding the edges. Fig. 7(b) shows the edges in house obtained from the Sobel edge detector with $\lambda=0.04$ (important pixels: 14250; unimportant pixels: 75750). Fig. 7(c) is the binary halftone picture used as the secret image and Fig. 9(d) is the recovered image (The detail analysis can refer the full version).

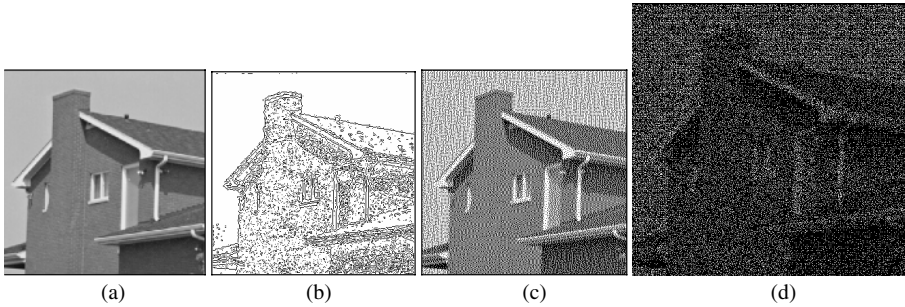


Fig. 7. The proposed (2, 3) VSS scheme (a) the original gray secret picture used for finding the edges (b) the important pixels in House using the threshold $\lambda=0.04$ ($N_I=14250$) (c) the binary halftone picture House used as the secret image (d) the recovered image

5 Applications

5.1 Wide Applicability of the Proposed VSS Scheme

Two major advantages of our proposed VSS scheme are the availability for any certain shadow size and the improvement of contrast. Considering the availability for any certain shadow size $(O_u \times O_v)$, for example a secret image in a VSS scheme with $m=3$ is 100×200 pixels and the recovered image needs to be 100×300 pixels for some certain application. As the conventional VSS scheme, we have to reduce the secret image half, i.e., 100×100 pixels, and then obtain the 100×300 -pixel shadow images for the display need. However, the proposed scheme can be directly used to achieve the 100×300 size. The feature of any shadow size makes our scheme more practicable for applications. For example, our schemes can be applied in the VSS applications in [12, 13] where two shadow sizes are required: the monitor screen size and the cellphone display size. The second advantage of our scheme is the improvement of

contrast with the less shadow size when compared to the conventional VSS scheme. This improvement can be applied in a digital image indexing scheme based on VSS technique [14]. We can use our scheme to share the binary metadata images used in a digital image indexing to reduce the shadow size but meantime have the high contrast. The less shadow size means the less modification of LSBs in a color filter array (CFA) image and results in the high-quality indexed demosaiced image.

5.2 A New VSS-Based Visual Authentication System by Collaborating Our Scheme and OCR Technique

OCR technique used in Section 3.3 for measuring the contrast can also be extended to come up a new VSS-based visual authentication system. In this section, by combining two techniques, VSS and OCR, we introduce a novel visual authentication scheme.

Some practical authentication and identification schemes based on VSS were proposed in [12, 13, 15]. The concept of visual authentication and identification schemes using (2, 2) VSS scheme is described as the following. The client (C) wishes to communicate the server (S) with a message I (a black/white printed text image, e.g., alphanumerical message). First, C creates a random shadow T_1 and gives it to S secretly. When starting the visual authentication protocol, S uses (2, 2) VSS scheme to create another shadow image T_2 given to C like a ticket which the stacked result (T_1+T_2) shows the information I . On receiving T_2 , S stacks these two shadow images and decides whether accepts I or not. This authentication scheme provides a single authentication since (2, 2) VSS scheme is just a one-time pad. For a more practical use, we need a many-times authentication system which can be used for several authentications, i.e., C only sends one shadow and then S can authenticate n messages. Noar and Pinkas [15] used a trivial way to achieve authentications n times by directly using n different areas in the shadow image. However, there are two problems in this many-times authentication scheme for practical application. The first problem is the reduction of resolution due to the small areas used for one authentication. The second problem is that the message I is viewed by a person to determine whether accept or reject and this will induce the personal bias.

Both problems can be solved by combining our VSS (the high-contrast recovered image) and OCR-like human visual system (automated recognition). We can try out the shadow size using our proposed scheme to achieve the 100% successful recognition rate. The “recovered secret printed text” could be a password or an authentication token and even can be divided into several areas for more authentications. Also, we can let the pixels in authentication areas be important pixels to enhance the recognition rate. Finally, we improve the previous authentication systems to retrieve the password automatically instead of viewing by the human visual system. The fusion of OCR and VSS makes the visual authentication system more practicable.

6 Conclusion

We take the lead to use different pixel expansions in a shadow image to achieve a high-contrast and any-size VSS scheme. Using the recognition capability of OCR, we

define a normalized recognition rate to fairly measure the contrast. The new measurement avoids the bias of human visual system and solves the problem that there is no consensus on previous definitions of contrast. In this paper, we also show the VSS scheme is not just an academic research issue but can be suitably modified for applications. For example, we introduce a new VSS-based automated visual authentication system based on OCR technique.

References

1. Naor M. and Shamir A.: Visual Cryptography. *Advances in Cryptology-EUROCRYPT'94*, LNCS 950, Springer-Verlag (1995) 1-12
2. Kuwakado H. and Tanaka H.: Size-Reduced Visual Secret Sharing Scheme. *IEICE Trans. on Funda. of Elect., Comm. and Comp. Sci.*, Vol. E87-A, No. 5 (2004) 1193-1197
3. Yang C. N. and Chen T. S.: New Size-Reduced Visual Secret Sharing Schemes with Half Reduction of Shadow Size. accepted for publication in *IEICE Trans. on Funda. of Elect., Comm. and Comp. Sci.*
4. Hou Y. C. and Tu S. F.: A Visual Cryptographic Technique for Chromatic Images Using Multi-pixel Encoding Method. *Journal of Research and Practice in Information Technology*, Vol. 37, No. 2 (2005) 179-191
5. ITO R., Kuwakado H., and Tanaka H.: Image Size Invariant Visual Cryptography. *IEICE Trans. on Funda. of Elect., Comm. and Comp. Sci.*, Vol. E82-A, No.10 (1999) 2172-2177
6. Yang C. N.: New Visual Secret Sharing Schemes Using Probabilistic Method. *Pattern Recognition Letters*, Vol. 25, Issue 4 (2004) 481-494
7. Cimato S., De Prisco R., and De Santis A.: Probabilistic Visual Cryptography Schemes. *The Computer Journal*, Vol. 49, No. 1 (2006) 97-107
8. Gonzalez R. C. and Woods R. E., *Digital Image Processing*, second edition, International Edition, Prentice Hall (2002)
9. Yang C. N. and Chen T. S.: New Aspect Ratio Invariant Visual Secret Sharing Schemes Using Square Block-wise Operation. *ICIAAR 2005*, Canada, LNCS 3656 (2005) 1167-1174 (the full version to be published at *Pattern Recognition*)
10. Verheul E. R., and Van Tilborg H. C. A.: Constructions and Properties of k out of n Visual Secret Sharing Schemes. *Designs, Codes and Cryptography*, Vol. 11, No. 2 (1997) 179-196
11. Eisen P. A., and Stinson D. R.: Threshold Visual Cryptography Schemes with Specified Whiteness. *Designs, Codes and Cryptography*, Vol. 25, No. 1 (2002)15-61
12. Yamamoto H., Hayasaki Y., and Nishida N.: Securing Information Display by Use of Visual Cryptography. *Optics Letters*, Vol. 28, No. 17 (2003) 1564-1566
13. Tuyls P., Kevenaar T., Schrijf G. J., Staring A. A. M., and van Dijk M.: Visual Crypto Displays Enabling Secure Communications. *Security in Pervasive Computing*, LNCS 2802 (2003) 271-284
14. Lukac R. and Plataniotis K.: Digital image indexing using secret sharing schemes: a unified framework for single-sensor consumer electronics. *IEEE Trans. On Consumer Electronics*, Vol. 51(2005) 908-916
15. Naor M. and Pinkas B.: Visual Authentication and Identification. *Advances in Cryptology-CRYPT'97*, LNCS 1294, Springer-Verlag (1997)322-336

Fast and Efficient Basis Selection Methods for Embedded Wavelet Packet Image Coding

Yongming Yang and Chao Xu*

National Lab. on Machine Perception, Peking University, Beijing, China, 100871
{Yangym, Xuchao}@cis.pku.edu.cn

Abstract. Wavelet packet (WP) image coding algorithms have shown consistent improvement over those based on wavelet transform. However, most of the current work cannot produce an embedded bit stream, since an embedded WP image coding method requires a valid and uniform decomposition for a given image. In this paper, based on the idea of maximizing variance of each subband, we propose two rate-unrelated basis selection criteria for embedded image coding. According to these criteria, efficient decompositions are found by growing the decomposition tree once or by pruning the full decomposition tree once. The experimental results show that, with the same subband coding scheme, the proposed basis selection methods achieve better coding performance than that of previous selection criteria. Comparison with the rate-distortion optimized basis selection scheme shows that, the proposed methods have 0.33dB loss at most, with the advantages of an embedded coding fashion and lower computational complexity.

1 Introduction

Wavelet transform has been successfully adopted for still image coding in recent years for its nice localization properties in both time and frequency domain. On the other hand, for some kind of images (such as image with oscillatory patterns or localized textures), the wavelet transform is not an effective tool, since its fixed space-frequency tiling does not always match the spectrum of these images. To solve this problem, wavelet packet (WP) was developed [1] to adapt the underlying wavelet bases to the frequency content of a given signal.

As a generalization of dyadic wavelet transform, wavelet packet offers a rich set of decomposition structures. To select a valid wavelet packet basis for a given image, i.e. find a WP decomposition structure which makes the input image amenable to subsequent coding, a basis selection criterion (cost function) should be given in advance. If the cost function is additive, then the optimal basis in the sense of this additive function can be found by pruning the full decomposition tree in a bottom-up fashion; otherwise a greedy tree-growing strategy can be adopted to find a basis. In the latter case, the basis selection procedure is simpler than the tree pruning algorithm, yet the selected basis is not optimal.

* Member, IEEE.

Since the best basis search procedure is equivalent to the procedure of minimizing a given cost function, a proper design of the cost function is essential to generating an effective basis. In the previous work [1, 2], many additive cost functions were proposed, such as the entropy based and the L1-norm based cost functions. These cost functions have two advantages. The first one is that since these cost functions have no relation with rate and distortion, the full decomposition tree only needs be pruned once to form the optimal decomposition. The second virtue is that based on these cost functions, the wavelet packet decomposition structure is uniform at various bit rates, making these criteria feasible for embedded wavelet packet image coding. However, since these cost functions do not take into account subsequent subband coding methods, the coding results based on these criteria are not good.

To improve the performance of wavelet packet image coding algorithm, Ramchandran and Vetterli [3] developed a rate-distortion (R-D) based cost function. According to this criterion, the quantization and the best basis choice are jointly optimized in an operational rate-distortion sense. However, the best basis search method is extremely computationally intensive due to its three layer embedded search procedures. Moreover, for a given image, this R-D optimized wavelet packet decomposition structure is variable with different target bit rate, making it difficult for embedded image coding.

To achieve an embedded wavelet packet image coding algorithm with high coding performance and low computational complexity, in this paper, we propose two basis selection criteria based on the simple idea of maximizing the variance of each subband. According to these criteria, efficient decompositions for embedded coding are found by growing the decomposition tree once or by pruning the full decomposition tree once. The first criterion is the variance function of each subband. A gain factor (*GF*) defined by the ratio of the variance function is used to find the final decomposition. The second criterion is deduced from variance. Though this criterion is exactly the already proposed L1-norm, a control factor (*CF*) defined by the ratio of the L1-norm is introduced to improve its efficiency of wavelet packet basis choice.

This paper is organized as follows. Section 2 presents a brief review of wavelet packet decomposition and some previously proposed cost functions. These cost functions are used to assess our basis selection criteria under the circumstance of the same subband coding scheme. In Section 3, after a brief description of the underlying quantization and entropy coding methods for each WP subband, the proposed basis selection criteria are detailed. In Section 4 some experimental results are given. Finally, conclusions are drawn in Section 5.

2 Wavelet Packet

2.1 Wavelet Packet Decomposition

The basic idea of wavelet packet is to allow non-octave subband decomposition to adaptively select the best basis for a particular signal. Different from the wavelet transform, wavelet packet not only iterates a two-channel filterbank over the low

frequency band at each level, but also over the other high frequency bands. This operation leads to a complete quaternary decomposition tree for a 2-D image with each node denoting a subband (the root node is the input image). To achieve the optimal WP decomposition, i.e. find the best subtree of the full tree, decisions should be made for each non-leaf node whether to keep its four sub-branches or not (corresponding to decomposing this node subband or not).

As mentioned above, a basis selection criterion (i.e. cost function) should be given in advance for the purpose of finding the final wavelet packet decomposition. The objective is to select a subtree of the full tree so that the global cost function is minimized. An additive cost function can achieve the optimal decomposition by pruning the full tree from bottom to the root node. During the tree-pruning procedure, the cost of a parent subband is compared with the sum of its four children's costs. If the parent's cost is costly, the four sub-branches are retained, and the parent's cost is updated to the sum of its children's costs, otherwise the four sub-branches are pruned. When this tree-pruning procedure reaches the root node, the best subtree that minimizes the global cost function is found. For a non-additive cost function, the bottom-up tree-pruning method is meaningless, because it's not proper to compare or update the parent's cost with the sum of its four children's costs. In this case, a subband is probed to decide whether to decompose it or not according to the non-additive cost function. This operation leads to a top-down tree-growing procedure.

2.2 Cost Functions

In the previous work [1, 2], many rate-unrelated additive cost functions are proposed, while to our knowledge, none of these cost functions can generate a consistently efficient WP decomposition for embedded image coding. Two noticeable cost functions are the *Shannon* entropy cost function and the L1-norm cost function, which are defined as follows:

Shannon entropy cost function:

$$\text{Cost}_{shannon}(x) = \exp\left(-\sum_i \frac{|x_i|^2}{\|x\|_2^2} \ln \frac{|x_i|^2}{\|x\|_2^2}\right) \quad (1)$$

L1-norm cost function:

$$\text{Cost}_{L1-Norm}(x) = \sum_i |x_i| \quad (2)$$

where x denotes the decomposed coefficients in a wavelet packet subband. In equality (1), $\|x\|_2^2$ denotes the energy of the signal x .

Ramchandran and Vetterli proposed a rate-distortion based cost function [3], which is given by

$$\text{Cost}_{R-D}(\lambda) = D + \lambda \cdot R \quad (3)$$

where λ is the quality factor, which controls the tradeoff between the distortion D and the budget bit rate R . The rate-distortion cost function not only solves the

problem of the best basis selection, but also allocates the target bit rate among the objective subbands optimally, i.e., the given bit budget is distributed among different subbands so that the global distortion is minimized. The disadvantages of this cost function are that the computational complexity is quite high due to its three layer embedded basis searching procedure, and that the rate-distortion optimized WP decomposition is inappropriate for embedded image coding.

3 Embedded WP Image Coding Algorithm

In this section, the subband coding algorithm is first described briefly. Then the proposed basis criteria are detailed. A block-partitioning coding scheme [4] is used to quantize each WP subband in a bitplane-by-bitplane manner. The quantization symbols are entropy encoded by an adaptive arithmetic coder [5] with the JPEG2000 context modeling [6].

After the wavelet packet decomposition, the transformed image is encoded sequentially from its highest bitplane to the lower bitplanes. In each bitplane, the WP coefficients belonging to a subband are coded with the block-partitioning quantization algorithm and the JPEG2000 context modeling entropy coding procedure. The coding process ceases till the budget bit rate exhausts, generating an embedded bit stream.

3.1 Subband Coding Algorithm

The block-partitioning coding algorithm adopts the bitplane coding fashion. In each bitplane, the coding algorithm finds and codes the significant coefficients of current bitplane. To achieve high coding efficiency, it tests the significance of a coefficient block first. If the block is significant, it is partitioned to locate the significant coefficients in it; otherwise it will be retained for the next bitplane revisit. In this way, groups of insignificant coefficients are represented with only one symbol, at the same time the significant coefficients can be located quickly (More details about this algorithm are presented in [7]).

One advantage of the block-partitioning algorithm is that it can achieve similar or better performance compared with the zerotree-based partitioning algorithm (SPIHT [8]) by exploiting the intraband dependencies only. Via coding each WP subband independently with the block-partitioning algorithm, the difficulty in defining hierarchical tree for WP subbands is naturally avoided. In our previous work [7], the block-partitioning coding algorithm is combined with the R-D optimized basis selection criterion [3]. Though this algorithm gives better objective results and visual quality, it is not an embedded coding scheme and the computational complexity is quit high. In next section, the experimental results of [7] (WP-BPO) are also presented to compare our proposed basis selection criteria with the rate-distortion optimized cost function.

3.2 Variance Based Basis Selection Criterion

The main principle of the block-partitioning algorithm is to exploit the energy compaction property of wavelet transform, which means that the major energy is

concentrated in the lowest frequency subband, and in higher frequency subbands, energy is mainly along the image's edges. As shown in Fig. 1, after a level of wavelet transform, the major energy is converged in the LL subband, so we deem that the variance of signal $|y|$, which depicts the dispersion degree of a signal, should be larger than the variance of signal $|x|$. This observation motivates us to use the variance measure as a basis selection criterion.

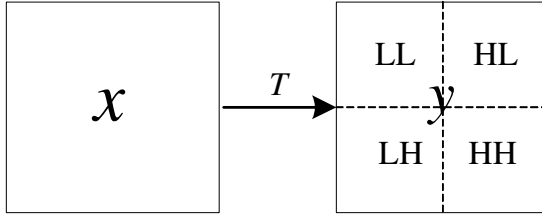


Fig. 1. Decomposition of an input signal x . The decomposition criterion is defined by the relation of the variance of $|x|$ between the variance of $|y|$.

The variance of signal $|x|$ is defined as follows.

$$D(x) = E(|x|^2) - (E(|x|))^2 = \frac{1}{n}(\|x\|_2^2 - \frac{1}{n}(\sum_i |x_i|)^2) \tag{4}$$

where n is the coefficient number of signal x . At the very start, the basis selection criterion can be given as follows: if $D(y)$ is larger than $D(x)$ (i.e. $D(y)/D(x) > 1.0$), the decomposition condition is satisfied, else signal x should not be decomposed to signal y . However, we find that this decomposition criterion is so loose that many subbands which should not be decomposed are further decomposed.

To solve this problem, we define a gain factor (GF) as the ratio of $D(y)$ to $D(x)$, where $D(y)$ and $D(x)$ are the variance of signal $|y|$ and $|x|$ respectively. The gain factor depicts the variance variation between the input signal x and the transformed signal y .

$$GF = D(y)/D(x) \tag{5}$$

The stricter basis selection criterion is brought forward by introducing a constant $GF_0 > 1.0$: if the inequality $GF > GF_0$ is satisfied, signal x is decomposed to signal y ; else signal x is retained.

Table 1 presents the gain factor of wavelet transform at each level for four test images, from which we can obtain two facts. The first one is that for most of the test images (except the Fingerprint image), the lower the transform level is, the larger the gain factor is. This fact validates a principle that a larger gain factor corresponds to a

more desirable decomposition. The second fact is that though the gain factor becomes smaller as the transform level becomes higher, it is still much larger than 1.0. These observations justify the energy clustering property of wavelet transform from the view of variance.

Table 1. Gain factor for images at each level of wavelet transform, using Daubechies 9/7-tap filter for 5-level octave decomposition

| Layer Image | Level I | Level II | Level III | Level IV | Level V |
|----------------|---------|----------|-----------|----------|---------|
| Lena | 2.429 | 2.377 | 2.272 | 2.169 | 1.902 |
| Barbara | 2.595 | 2.500 | 2.408 | 2.237 | 1.860 |
| Fingerprint | 2.559 | 2.362 | 1.989 | 2.483 | 3.055 |
| Goldhill | 2.413 | 2.356 | 2.299 | 2.217 | 2.124 |

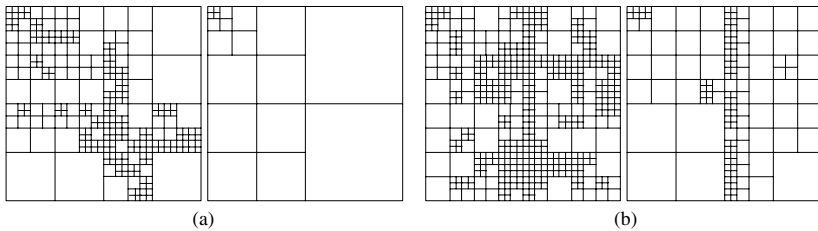


Fig. 2. Decomposition structures for (a) Lena and (b) Barbara images with $GF_0 = 1.0$ and $GF_0 = 1.075$, using Daubechies 9/7-tap filter with maximum 5-level decomposition. (a) Lena: left for 1.0 and right for 1.075. (b) Barbara: left for 1.0 and right for 1.075.

Since the variance based basis selection criterion is not additive, the basis searching procedure should be executed in a top-down fashion. Now we can summarize the whole basis selection procedure as follows:

- For a given input image, determine the maximum decomposition depth N .
- For each subband x (the initial subband is the input image), if the decomposition depth does not reach, decompose it to signal y (Fig. 1), compare the gain factor $D(y)/D(x)$ with GF_0 .
 - If $GF > GF_0$, the subband x is decomposed. Probe the decomposed four subbands of signal y orderly with the same method of subband x .
 - Else, subband x is retained.

Though the given constant GF_0 is related to image and subband, we find that it is not sensitive to different images. The experimental results show that the coding performance becomes better with the augment of GF_0 when it is smaller than 1.08; when

$GF_0 > 1.08$, the larger this constant is, the worse the coding performance is. When GF_0 reaches 1.20, the WP decomposition is exactly the wavelet transform for all the test images. In our experiment, this constant is set to be 1.075, which gives a relatively better coding result. Fig. 2 shows the WP decomposition structures with $GF_0 = 1.0$ and $GF_0 = 1.075$ respectively for Lena and Barbara images. From this figure we can see that the decomposition structure with $GF_0 = 1.0$ is very littery, which causes a worse coding performance. While for $GF_0 = 1.075$, the WP decomposition structure, which is the sub-decomposition of the former since the selection criterion is stricter, is terse and valid. For example, the WP decomposition for Lena image is similar to the optimal decomposition in the R-D sense, which is approximately the wavelet transform.

3.3 L1-Norm Based Basis Selection Criterion

The basis selection criterion proposed in last subsection is based on the idea of maximizing the variance of a signal $|x|$, i.e. if the inequality $D(y)/D(x) > GF_0$ is satisfied, then signal x is decomposed to signal y . From equality (4) we can see that, if the transform T is orthonormal, then $D(y) > D(x)$ is equivalent to $\sum_i |y_i| < \sum_i |x_i|$, since the energy of a signal after an orthonormal transform is conservative, i.e. $\|x\|_2^2 = \|y\|_2^2$. This discussion gives an explanation for the already proposed L1-norm cost function from the view of variance. Still, we find that the L1-norm based decomposition condition (i.e. if $\sum_i |y_i| < \sum_i |x_i|$, x is decomposed to y .) is too loose, causing many unnecessary decompositions. Similar to the variance based basis selection criterion, the decomposition condition can be restricted by introducing a control factor (CF), which is defined as the ratio of $\sum_i |y_i|$ to $\sum_i |x_i|$.

$$CF = \sum_i |y_i| / \sum_i |x_i| \quad (6)$$

Thus the new basis selection criterion can be described as follows: if the inequality $CF < CF_0$ (where $CF_0 < 1.0$ is a given constant) is satisfied, then signal x is decomposed to signal y ; else signal x is retained.

Table 2 presents the control factor of wavelet transform at each level for the same images as those in Table 1. From this table we can see that with the rise of the transform level, the control factor becomes larger for most of the test images, furthermore, these control factors are much lesser than 1.0 and none of them is close to 1.0. This observation implies that the L1-norm basis selection criterion (i.e. $CF_0 = 1.0$) is too loose.

Due to the additive property of the L1-norm, the basis searching procedure can be executed in a bottom-up fashion, leading to an optimal decomposition. The whole basis selection procedure based on the L1-norm cost function is summarized as follows:

- Grow a full decomposition tree to a predetermined decomposition depth N for the input image.
- Prune the full tree recursively, starting from the leaf nodes to the root node. At each non-leaf node x , calculate $\sum_i |x_i|$ and $\sum_i |y_i|$, where signal y is the four children subbands of the subband x (Fig. 1) in the full decomposition tree. Compare the control factor $\sum_i |y_i| / \sum_i |x_i|$ with CF_0 .
 - If $CF < CF_0$, the four children subbands (i.e. signal y) is retained, and the L1-norm cost of subband x is updated to $\sum_i |y_i|$.
 - Else, the four children subbands are pruned.
- When this tree-pruning procedure reaches the root node, the final WP decomposition based on the L1-norm selection criterion is found.

Table. 2. Control factor for images at each level of wavelet transform, using Daubechies 9/7-tap filter for 5-level octave decomposition

| Layer Image | Level I | Level II | Level III | Level IV | Level V |
|----------------|---------|----------|-----------|----------|---------|
| Lena | 0.550 | 0.563 | 0.589 | 0.624 | 0.687 |
| Barbara | 0.582 | 0.591 | 0.586 | 0.631 | 0.732 |
| Fingerprint | 0.540 | 0.620 | 0.753 | 0.668 | 0.607 |
| Goldhill | 0.570 | 0.584 | 0.601 | 0.632 | 0.662 |

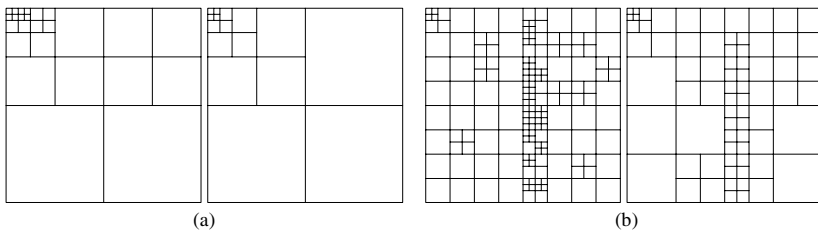


Fig. 3. Decomposition structures for (a) Lena and (b) Barbara images with $CF_0 = 1.0$ and $CF_0 = 0.935$, using Daubechies 9/7-tap filter with maximum 5-level decomposition. (a) Lena: left for 1.0 and right for 0.935. (b) Barbara: left for 1.0 and right for 0.935.

Similar to the choosing method of constant GF_0 , the constant CF_0 for all the test images is determined via plentiful experiments, though the optimal CF_0 for different image is dissimilar. The experimental results show that when CF_0 is near 0.88, the WP decomposition degenerates into the wavelet transform for all the test images. The relatively better coding results are achieved when CF_0 is between 0.93 and 0.94. For

$CF_0 < 0.93$, the smaller the constant is, the worse the coding performance is; while for $CF_0 > 0.94$, a larger CF_0 often causes a worse coding result. In our experiment, CF_0 is set to be 0.935. The WP decomposition structures with $CF_0 = 1.0$ and $CF_0 = 0.935$ for Lena and Barbara images are illustrates in Fig. 3 respectively. It is easy to deduce that the decomposition structure with $CF_0 = 0.935$ is the sub-decomposition of the former, since a smaller CF_0 corresponds to a stricter decomposition condition. Comparison of Fig. 3 and Fig. 2 shows that this L1-norm based basis selection criterion produces much succincter decomposition structures than the variance based selection criterion.

4 Experimental Results

In this section, some experimental results are presented to evaluate the proposed basis selection criteria for embedded wavelet packet image coding. The experiments are conducted on three $512 * 512$ 8-bit grayscale images: Lena, Barbara and Fingerprint, using the popular biorthogonal Daubechies 9/7-tap filter [9] with the maximum 5-depth decomposition. The measure used to describe the coding performance of each basis selection criterion is the peak-signal-to-noise ratio (PSNR).

Table 3 summarizes the coding results at various bit rates for the three test images. To compare with a state-of-the-art wavelet image coding algorithm, the coding result of the new still image compression standard JPEG2000 is given. In this table, the experimental result of the wavelet transform based embedded block-partitioning coding algorithm (denoted by “Wavelet”) is also listed for the purpose of assessing the performance difference between wavelet transform and the WP decomposition. Table 3 also lists the coding results of the entropy cost function (denoted by “Entropy”) and rate-distortion optimized cost function (denoted by “WP-BPO” [7]). For the proposed variance based basis selection criterion, the coding results corresponding to $GF_0 = 1.0$ and $GF_0 = 1.075$ are presented; while for the L1-norm based basis selection criterion, the coding results with $CF_0 = 1.0$ and $CF_0 = 0.935$ are given. When CF_0 equals 1.0, it corresponds to the already proposed L1-norm cost function.

For all the test images, WP-BPO achieves the best coding performance among all the basis selection criteria. However, it is not an embedded coding algorithm since it uses the R-D optimized cost function. For Lena image, WP-BPO outperforms “ $GF_0 = 1.075$ ” and “ $CF_0 = 0.935$ ” by about 0.07dB and 0.08dB averagely. These insignificant improvements can be explained by the fact that the optimal decomposition for Lena image is approximately the wavelet transform. The coding results of “ $GF_0 = 1.0$ ” and “ $CF_0 = 1.0$ ” is worse than the coding result of “Wavelet”, which reinforces the fact that these decomposition conditions are too loose. For Lena image, the coding result of “Entropy” is also inferior to that of “Wavelet”.

For Barbara image, the coding results of “ $GF_0 = 1.075$ ” and “ $CF_0 = 0.935$ ” are worse than WP-BPO by about 0.31dB and 0.20dB respectively on the average, at the

Table 3. Rate-distortion (PSNR) performance at various bit rates for different basis selection criterion, using maximum 5-depth decomposition

| Lena | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|-------|
| Rate(bpp) | 1.0 | 0.8 | 0.5 | 0.4 | 0.25 | 0.2 | 0.125 |
| JPEG2000 | 40.33 | 39.24 | 37.28 | 36.13 | 34.14 | 33.01 | 31.00 |
| Wavelet | 40.39 | 39.35 | 37.34 | 36.28 | 34.23 | 33.19 | 31.23 |
| WP-BPO | 40.50 | 39.41 | 37.41 | 36.35 | 34.37 | 33.31 | 31.24 |
| Entropy | 40.35 | 39.33 | 37.27 | 36.23 | 34.09 | 33.05 | 31.10 |
| $GF_0 = 1.0$ | 39.62 | 38.75 | 36.53 | 35.72 | 33.43 | 32.55 | 30.39 |
| $GF_0 = 1.075$ | 40.37 | 39.36 | 37.36 | 36.33 | 34.25 | 33.20 | 31.22 |
| $CF_0 = 1.0$ | 40.24 | 39.30 | 37.22 | 36.25 | 34.20 | 33.16 | 31.11 |
| $CF_0 = 0.935$ | 40.39 | 39.35 | 37.34 | 36.28 | 34.23 | 33.19 | 31.23 |

| Barbara | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|-------|
| Rate(bpp) | 1.0 | 0.8 | 0.5 | 0.4 | 0.25 | 0.2 | 0.125 |
| JPEG2000 | 37.18 | 35.29 | 32.30 | 30.86 | 28.36 | 27.33 | 25.42 |
| Wavelet | 37.14 | 35.13 | 31.97 | 30.52 | 28.24 | 26.79 | 25.24 |
| WP-BPO | 37.87 | 36.22 | 33.24 | 31.86 | 29.53 | 28.37 | 26.60 |
| Entropy | 37.42 | 35.55 | 32.65 | 31.35 | 29.19 | 28.02 | 26.34 |
| $GF_0 = 1.0$ | 36.09 | 34.70 | 31.53 | 30.61 | 28.16 | 27.44 | 25.77 |
| $GF_0 = 1.075$ | 37.60 | 35.78 | 32.83 | 31.53 | 29.26 | 28.11 | 26.38 |
| $CF_0 = 1.0$ | 37.52 | 35.77 | 32.80 | 31.54 | 29.25 | 28.19 | 26.40 |
| $CF_0 = 0.935$ | 37.69 | 35.93 | 33.02 | 31.64 | 29.38 | 28.19 | 26.42 |

| Fingerprint | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|-------|
| Rate(bpp) | 1.0 | 0.8 | 0.5 | 0.4 | 0.25 | 0.2 | 0.125 |
| JPEG2000 | 36.77 | 34.72 | 31.76 | 30.23 | 27.55 | 26.19 | 24.22 |
| Wavelet | 37.07 | 34.93 | 31.95 | 30.48 | 27.64 | 26.34 | 24.39 |
| WP-BPO | 37.52 | 35.62 | 32.30 | 30.83 | 28.22 | 27.06 | 25.05 |
| Entropy | 36.77 | 34.75 | 31.86 | 30.44 | 27.50 | 26.45 | 24.30 |
| $GF_0 = 1.0$ | 36.40 | 34.75 | 31.29 | 30.09 | 27.37 | 26.29 | 23.97 |
| $GF_0 = 1.075$ | 37.25 | 35.22 | 31.87 | 30.54 | 27.84 | 26.77 | 24.56 |
| $CF_0 = 1.0$ | 37.10 | 35.14 | 31.84 | 30.48 | 27.71 | 26.62 | 24.44 |
| $CF_0 = 0.935$ | 37.40 | 35.37 | 32.12 | 30.59 | 27.90 | 26.67 | 24.69 |

**Fig. 4.** Decoded images for Lena at 0.25bpp. (a) Variance criterion with $GF_0 = 1.075$, PSNR = 34.25dB. (b) L1-norm criterion with $CF_0 = 0.935$, PSNR = 34.23dB. (c) R-D optimized cost function (WP-BPO), PSNR = 34.37dB.

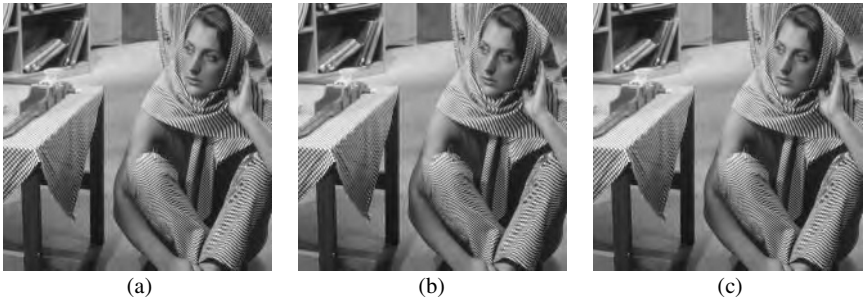


Fig. 5. Decoded images for Barbara at 0.25bpp. (a) Variance criterion with $GF_0 = 1.075$, PSNR = 29.26dB. (b) L1-norm criterion with $CF_0 = 0.935$, PSNR = 29.38dB. (c) R-D optimized cost function (WP-BPO), PSNR = 29.53dB.

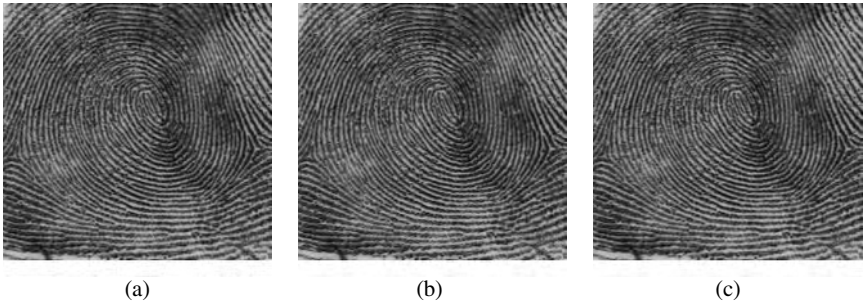


Fig. 6. Decoded images for Fingerprint at 0.25bpp. (a) Variance criterion with $GF_0 = 1.075$, PSNR = 27.84dB. (b) L1-norm criterion with $CF_0 = 0.935$, PSNR = 27.90dB. (c) R-D optimized cost function (WP-BPO), PSNR = 28.22dB.

same time, “ $GF_0 = 1.075$ ” and “ $CF_0 = 0.935$ ” outperform JPEG2000 by 0.69dB and 0.80dB on the average. Though the coding results of “Entropy” and “ $CF_0 = 1.0$ ” are better than JPEG2000, comparison of “ $CF_0 = 0.935$ ” and “ $CF_0 = 1.0$ ” shows that the coding performance of the latter criterion can still be improved by restricting its basis selection condition.

For Fingerprint image, WP-BPO outperforms “ $GF_0 = 1.075$ ” and “ $CF_0 = 0.935$ ” by about 0.33dB and 0.26dB averagely. Similar to the Lena image, the coding result of “Entropy” is inferior to the result of “Wavelet”.

From the PSNR results presented in Table 3, we can summarize the coding performance of each basis selection criterion as follows. The variance criterion with $GF_0 = 1.075$ and the L1-norm criterion with $CF_0 = 0.935$ are both efficient methods for embedded WP image coding. Although the variance criterion is inferior to the L1-norm criterion in PSNR, however, the variance based basis selection procedure is executed in a top-down fashion, thus the computational complexity is lower than the

L1-norm criterion, which induces a bottom-up basis searching procedure. Comparison of both these methods with the rate-distortion optimized cost function shows that the performance decline is within 0.33dB on the average, with the advantage of an embedded coding fashion. From the table we can also see that the basis selection conditions with $GF_0 = 1.0$ and $CF_0 = 1.0$ are both too loose, inducing a worse coding performance. At last, the already proposed entropy based cost function does not always produce a valid basis (such as for the case of Lena and Fingerprint image).

The decoded images at 0.25bpp for the three test images are shown in Fig. 4, Fig.5 and Fig. 6 respectively. For Lena and Fingerprint image, the proposed basis selection criteria have no obvious visual difference with the rate-distortion optimized cost function. For Barbara image, we notice that the tablecloth of the WP-BPO decoded image is clearer than the other two decoded images, though the difference is not distinct.

5 Conclusions

In this paper, we propose two rate-unrelated basis selection criteria for embedded WP image coding. The first criterion is based on the idea of maximizing the variance of a subband, while the second is the L1-norm based basis selection criterion. By introducing two factors (i.e. gain factor and control factor), the efficiency of wavelet packet basis selection for embedded image coding is improved significantly. The experimental results show that the image coding algorithm based on the proposed basis criteria achieves similar objective result and visual quality to those of the R-D optimized selection criterion, while the bit stream is embedded and the computational complexity is lower due to the simple searching procedure for a valid decomposition. In this work, the constants GF_0 and CF_0 are given in advance by plentiful experiments, an interesting future work is how to decide the constants GF_0 and CF_0 dynamically when determining whether to decompose a subband or not.

Acknowledgements

This paper was supported by China 973 program NKBRPC under Grant 2004CB318005. The author would like to thank Yu Zheng in our laboratory for the helpful discussion for the design of the cost functions.

References

1. R. Coifman, Y. Meyer, S. Quake, and V. Wickerhauser, "Signal processing and compression with wave packets", Numerical Algorithms Research Group, New Haven, CT: Yale University, 1990.
2. R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Inform. Theory, Special Issue on Wavelet Transforms and Multires. Signal Anal.*, vol. 38, pp. 713-718, Mar. 1992.
3. K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Processing.*, vol. 2, pp. 160-175, Apr. 1993.

4. Z. Liu, Lina J. Karam, "An efficient embedded zerotree wavelet image codec based on intraband partitioning," in *IEEE Int. Conf. Image Processing*, vol. 3, pp. 162-165, 2000.
5. I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Commun. ACM.*, vol. 30, pp. 520-540, June 1987.
6. D. Taubman, and M. W. Marcellin, "JPEG2000 Image Compression Fundamentals, Standards and Practice," *Kluwer Academic Publishers*, 2002.
7. Y. Yang and C. Xu, "A wavelet packet based block-partitioning image coding algorithm with rate-distortion optimization," in *IEEE Int. Conf. Image Processing (ICIP)*, 2005.
8. A. Said and W. A. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243-250, June 1996.
9. M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*. vol. 1, pp. 205-220, Apr. 1992.

Fractal Image Coding as Projections Onto Convex Sets

Mehran Ebrahimi and Edward R. Vrscay

Department of Applied Mathematics
Faculty of Mathematics
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1
m2ebrahi@uwaterloo.ca, ervrscay@uwaterloo.ca

Abstract. We show how fractal image coding can be viewed and generalized in terms of the method of projections onto convex sets (POCS). In this approach, the fractal code defines a set of spatial domain similarity constraints. We also show how such a reformulation in terms of POCS allows additional constraints to be imposed during fractal image decoding. Two applications are presented: image construction with an incomplete fractal code and image denoising.

1 Introduction

In this paper we show how fractal image coding can be formulated in terms of a powerful method known as projections onto convex sets (POCS) [15].

In standard fractal image coding [1,3,7,11], images are normally considered as elements of a complete function space \mathcal{F} , e.g., $L^p(X)$, for $X \subset R^2$. Given an image function u , we seek a contractive operator $T : \mathcal{F} \rightarrow \mathcal{F}$ such that its fixed point \bar{u} is a good approximation to u . This comprises *fractal coding*. At the *decoding* stage, the *fractal transform* T is applied iteratively to an initial “seed” image u_0 . Banach’s contraction mapping theorem guarantees the convergence of the iterates defined by $u_{n+1} = Tu_n$ to \bar{u} .

One may, however, consider this decoding process to be too restrictive since any additional knowledge about the original image (e.g., bounding constraints on the pixel values) cannot be applied simultaneously with the fractal transform. In practice, therefore, additional constraints have usually been applied to the fixed point of the contraction map T . Such post-processing is usually equivalent to the application of projections on the fixed point. For example, thresholding the data in order to enforce certain bounds is simply a projection of the the data onto the space of images that lie within those bounds.

In our POCS approach, a fractal code is considered as a set of similarity constraints in the spatial domain. Fortunately these constraints are convex and closed, and therefore satisfy the requirements in setting up a POCS approach.

In what follows, we shall show how POCS can provide the opportunity to apply the fractal code of an image simultaneously with additional constraints

during the decoding stage. These constraints may include other spatial, statistical, spectral and pattern properties of the unknown image, e.g., bounds on pixel values, bounded energy, smoothness, similarity to the observed data, etc..

A most interesting consequence of such a POCS-type reformulation allows us to bypass the traditional view of fractal coding as simply a mapping of domain image subblocks onto range image subblocks. Instead, we consider the application of the projections on a closed and convex set that is constructed using the similarity constraint between the domain and range subblocks. This type of projection translates to a *simultaneous* alteration of domain-range block pairs. In this paper, we introduce such projections explicitly and examine some interesting implications and applications.

Let us qualify that we are not interested in the compression capabilities of fractal image coding here. Our study picks up upon the thesis of H. Puh on set-theoretic coding [13]. In that work, there were no explicit details on how the self-similarity properties of an image can be translated into POCS-type inequality constraints.

2 Some Basics of Fractal Image Coding

More details on fractal image coding can be found in many places [1,3,7,11], including one of our recent papers on fractal-based denoising [9].

2.1 Fractal Image Encoding

Fractal image coding seeks to approximate an image by a union of spatially-contracted and greyscale-modified copies of subblocks of itself. If we let the image of interest I be represented by an image function $u(x, y)$, then the result of the coding procedure is a contractive mapping T , the so-called *fractal transform* operator, the fixed point \bar{u} of which provides an approximation to u . In other words,

$$u \cong \bar{u} = T\bar{u}. \quad (1)$$

To obtain T , the image I is first partitioned (e.g., uniform, quadtree) into a set of nonoverlapping range blocks R_i . For each range block R_i , one searches for a larger domain block D_i (from an appropriate “domain pool” \mathcal{D} that is often common to all range blocks of the same size) such that $u(R_i)$ is approximated as well as possible by a modified copy of $u(D_i)$, i.e.,

$$u(R_i) \cong \phi_i(u(D_i)) = \phi_i(u(w_i^{-1}(R_i))), \quad (2)$$

where $\phi_i : \mathbf{R} \rightarrow \mathbf{R}$ is a greyscale map that operates on pixel intensities and w_i denotes the 1-1 contraction/decimation that maps pixels of D_i onto pixels of R_i . The *fractal code* defining T consists of the maps ϕ_i as well as the domain-range assignments determined during the coding procedure.

In the calculations reported below, we employ the simplest form of partitioning, namely a uniform scheme. The range blocks R_i will be $N \times N$ square nonoverlapping pixel blocks. They will share a common domain pool \mathcal{D} that

consists of all nonoverlapping $2N \times 2N$ pixel blocks. This choice of domain pool is, of course, nonoptimal. The set of all $2N \times 2N$ pixel blocks obtained by single pixel-shifts would be better but search times would be enormous. In any case, the purpose of this paper is to show the basics of a method that can be adapted to any partitioning scheme.

Finally, there are 8 ways – four rotations and four inversions – by which a larger square domain block D_j can be contracted/decimated and then mapped onto a smaller range block R_i . We shall index these possible transformations by means of a k superscript in the spatial mapping, i.e., $w_i^{(k)} : D_i \rightarrow R_i$, $k = 1, 2, \dots, 8$.

Let us now assume that we have fractally coded an image function u according to Eq (2). Because of the nonoverlapping nature of the partition, we may write

$$u(x, y) \cong (Tu)(x, y) = \sum_i \phi_i(u(w_i^{-1}(x, y))). \tag{3}$$

The image function u is thus approximated as a union of spatially-contracted (w_i) and greyscale-distorted (ϕ_i) copies of itself. This union of modified copies (a trivial consequence since the copies do not overlap) defines a special kind of *fractal transform* operator T . If the above approximation is a good one, then the so-called *collage distance* $\|u - Tu\|$ is small. From the ‘‘Collage Theorem’’ [2],

$$\|u - \bar{u}\| \leq \frac{1}{1 - c_T} \|u - Tu\|, \tag{4}$$

it then follows that if u is ‘‘close’’ to Tu , then u is also close to \bar{u} , the fixed point of T . Here, $c_T \in [0, 1)$ denotes the contraction factor of T . The quantity $\|u - \bar{u}\|$ is the error of approximation of u by \bar{u} .

In practice, one normally assumes the greyscale maps ϕ_i to be affine, i.e.,

$$\phi_i(t) = \alpha_i t + \beta_i. \tag{5}$$

For a given domain-range block pair D_j/R_i , the optimal value of the α and β parameters is usually accomplished by means of least-squares fitting. If we let x_m and y_m , $m = 1, 2, \dots, N^2$ denote the pixel greyscale values of the $w^{(k)}$ -decimated domain block D_j and range block R_i , respectively, then α and β are determined so that that the squared L^2 ‘‘collage distance’’ over R_j ,

$$\Delta^{(k)} = \sum_{m=1}^{N^2} [y_m - \alpha x_m - \beta]^2, \tag{6}$$

is minimized. In the construction of the fractal transform operator T , we choose, for each range block R_i , the domain block $D_j \in \mathcal{D}$ and geometric transformation $w^{(k)}$ which yield the minimum collage distance $\Delta^{(k)}$. In this way, the total collage distance $\|u - Tu\|$ in Eq. (4) is minimized.

2.2 Fractal Image Decoding

Given a contractive fractal transform T , we may generate its fixed point \bar{u} by the iteration procedure $u_{n+1} = Tu_n$, starting with an arbitrary “seed” image u_0 .



Fig. 1. The fractal transform operator designed to approximate the 256×256 (8 bpp) “Lena” image (Left). The “seed” image was $u_0(x) = 255$ (plain white). The fractal transform T was obtained by “collage coding” 8×8 nonoverlapping pixel range blocks. The domain pool was comprised of nonoverlapping 16×16 pixel blocks.

In the decoding procedure, the image subblocks $u_n(R_i)$ of u_n are replaced by modified copies $\phi_i(u_n(D_i))$ according to Eq. (2). Banach’s contraction mapping theorem guarantees that the sequence of images u_n converges to \bar{u} .

In this scheme, the range blocks $R_{n+1,i}$ comprising image u_{n+1} are simply modified versions of appropriate domain blocks $D_{n,i}$ of u_n . More precisely, $u_{n+1}(R_i) = \phi_i(w_i^{(k)}(u_n(D_i)))$.

In Figure 1 is presented the fixed point approximation \bar{u} to the standard 256×256 Lena image (8 bits/pixel) using a partition of 8×8 nonoverlapping pixel blocks ($64^2 = 4096$ in total). The domain pool for each range block was the set of $32^2 = 1024$, 16×16 non-overlapping pixel blocks. (This is clearly not optimal.) This image was obtained by starting with the seed image $u_0(x) = 255$ (plain white image) and iterating $u_{n+1} = Tu_n$ to $n = 15$.

3 Set Theoretic Image Coding and Restoration

The method of projections onto convex sets (POCS) has attracted much attention in a multitude of image restoration and reconstruction applications. The main reasons are the simplicity, flexibility, and powerful inclusion of *a priori*

information. It is generally simple to define convex constraint sets which incorporate desired solution characteristics. These sets may impose restrictions such as positivity or bounded energy which are difficult to represent in terms of cost functionals. The only potential source of difficulty in applying POCS is to determine the projection operators. If the the convex and closed sets are constructed and the projections are found, a point in the intersection these sets can be found if the intersection is not empty.

Assume that x is known to lie in m given sets C_i , $i = 1, 2, \dots, m$ where each of the sets represents a constraint on the image. If the sets C_i are closed and convex, we associate projection operators P_i , $i = 1, 2, \dots, m$, to each C_i . The projection of h onto C_i is defined as $g = P_i h$, with $g \in C_i$ and

$$\|g - h\| = \inf_{y \in C_i} \|y - h\|. \quad (7)$$

Bregman [4] proved that if the sets C_i , $i = 1, 2, \dots, m$ are closed and convex and $\bigcap_i C_i \neq \emptyset$, then the sequence $\{X^{(n)}\}$ defined recursively as

$$X^{(n+1)} = P_m \dots P_2 P_1 X^{(n)}, \quad (8)$$

converges to a fixed point in the intersection of all C_i s. In other words, $X^{(n)}$ converges to a point X , where $X \in \bigcap_i C_i$. This sequence of projections is applied repeatedly to yield an updated estimate of the image. It is imperative to note that the point X is, in general, nonunique. Its only distinguishing feature is that it lies on the intersection of all constraint sets. In general, it will be dependent on the initial guess X^0 . Detailed theoretical discussions of the POCS method can be found in [4,10,15].

4 POCS and Fractal Image Coding

In this section we consider a number of self-similarity constraints along scales which can then be used to reformulate fractal image coding as a POCS method. It is assumed that an image has a continuous representation as a function in $L^p(X)$ where $X \subset \mathbf{R}^2$. The theoretical basis for this discussion (e.g., proofs of propositions) has been presented elsewhere [6].

4.1 Self-similarity Constraints Using Collage Distances

The collage distance is an important measure of self-similarity. We shall consider three different types of collage distances from which similarity constraints can be built. These are (i) the total collage distance, (ii) the collage distance for a fixed domain-range block pair and (iii) the pointwise collage distance.

Definition 1. For a given image $u \in L^p(X)$, where $1 \leq p \leq \infty$ we introduce the following collage distance definitions.

a) The total collage distance is defined as

$$\Delta(u) = \left\| (Tu) - u \right\|_p \quad (9)$$

$$= \left(\int_{(x,y) \in X} \left| \sum_i \phi_i(u(w_i^{-1}(x,y))) - u(x,y) \right|^p dx dy \right)^{\frac{1}{p}},$$

when $1 < p < \infty$. Appropriate definitions are used for $p = 1$ and $p = \infty$.

b) The collage distance for a fixed pair of domain and range blocks is

$$\begin{aligned} \Delta_{(i,j)}^{(k)}(u) &= \left\| \phi_i \left(u(w_{i,j,(k)}^{-1}(R_i)) \right) - u(R_i) \right\|_p \\ &= \left(\int_{(x,y) \in R_i} \left| \phi_i(u(w_i^{-1}(x,y))) - u(x,y) \right|^p dx dy \right)^{\frac{1}{p}}, \end{aligned} \tag{10}$$

where $1 < p < \infty$. Again, appropriate definitions are used for $p = 1$ and $p = \infty$. As before R_i is a range block. $w_{i,j,(k)}^{-1}$, is the inverse of contraction from D_j to R_i and $(k) = K_{(i,j)} \in \{0, \dots, 7\}$.

c) The pointwise collage distance is defined as

$$\Delta_{(i,j)(x,y)}^{(k)}(u) = \left| \phi \left(u(w_{i,j,(k)}^{-1}(x,y)) \right) - u(x,y) \right|, \tag{11}$$

where $(x,y) \in R_i$.

By its nature, fractal image coding implies perfect self-similarity between elements in the range and domain pool of an attractor image \bar{u} . In natural images, however, perfect self-similarity generally does not hold. Therefore, a model that is more consistent with the physical world would allow collage distances, as measures of similarity, to deviate from zero but not beyond some threshold value, say $\delta > 0$. Such threshold values can be determined based on the application. Such a framework provides more flexibility to incorporate additional knowledge about the image and to relax the perfect self-similarity constraint in order to obtain a better approximation of an image in the reconstruction process. In the perfect self-similarity assumption case, this threshold may be set to zero, representing perfect consistency with traditional fractal image coding.

Each of the three collage distances described above may be used in the procedure. Associated with each collage distance are the following self-similarity constraints in $L^p(X)$ for $1 \leq p \leq \infty$. (i) Based on the total collage distance define

$$\Psi = \left\{ u \in L^p(X) : \Delta(u) \leq \delta \right\}. \tag{12}$$

(ii) Based on range based collage distance define

$$\Psi_{(i,j)}^{(k)} = \left\{ u \in L^p(X) : \Delta_{(i,j)}^{(k)}(u) \leq \delta_{(i,j)}^{(k)} \right\}. \tag{13}$$

(iii) Finally define the set $\Psi_{(i,j)(x,y)}^{(k)}$ based on the pointwise collage distance as

$$\Psi_{(i,j)(x,y)}^{(k)} = \left\{ u \in L^p(X) : \Delta_{(i,j)(x,y)}^{(k)}(u) \leq \delta_{(i,j)(x,y)}^{(k)} \right\}. \tag{14}$$

Closedness and convexity of these sets can be verified easily under some trivial assumptions [6]. In practice, when dealing with digital images we need to construct new convex and closed sets based on the image greyvalues at discrete points. In the following section, we study the discrete counterpart of $\Psi_{(i,j)(x,y)}^{(k)}$.

4.2 Discrete Pointwise Collage Constraints and Associated Projections

When we are working with discrete digital images, we may assume any digital image $U \in l^2(X)$, where X is a bounded subset of \mathbf{Z}^2 . Note that in the continuous case, we considered the set $\Psi_{(i,j)(x,y)}^{(k)}$ for $(x, y) \in R_i$. In the same fashion, for the discrete case, we define $\Psi_{(i,j)(m,n)}^{(k)}$, where (m, n) is a point with integer coordinates over a fixed R_i .

$$\begin{aligned} \Psi_{(i,j)(m,n)}^{(k)} &= \left\{ U \in l^2(X) : \Delta_{(i,j)(m,n)}^{(k)}(U) \leq \delta_{(i,j)(x,y)}^{(k)} \right\} \\ &= \left\{ U \in l^2(X) : \left| \phi \left(U(w_{i,j,(k)}^{-1}(m, n)) \right) - U(m, n) \right| \leq \delta_{(i,j)(m,n)}^{(k)} \right\} \end{aligned} \tag{15}$$

In the discrete case (m, n) corresponds to four points (s, t) , $(s, t + 1)$, $(s + 1, t)$, and $(s + 1, t + 1)$ under the inverse mapping $w_{i,j,(k)}^{-1}(m, n)$. Here $U(w_{i,j,(k)}^{-1}(m, n))$ can be written as

$$U(w_{i,j,(k)}^{-1}(m, n)) = \frac{1}{4} \left\{ U(s, t) + U(s, t + 1) + U(s + 1, t) + U(s + 1, t + 1) \right\}. \tag{16}$$

Hence, replacing $\phi(t) = \alpha t + \beta$ yields

$$\begin{aligned} \Psi_{(i,j)(m,n)}^{(k)} &= \left\{ U \in l^2(X) : \left| \frac{\alpha}{4} \left\{ U(s, t) + U(s, t + 1) + U(s + 1, t) \right. \right. \right. \\ &\quad \left. \left. \left. + U(s + 1, t + 1) \right\} + \beta - U(m, n) \right| \leq \delta_{(i,j)(m,n)}^{(k)} \right\}. \end{aligned} \tag{17}$$

Proposition 1. [6] $\Psi_{(i,j)(m,n)}^{(k)}$ is a closed and convex set on the Hilbert space $l^2(X)$ provided that $\phi(t) = \alpha t + \beta$.

Now that the convex and closed constraints are constructed we have to find the projection operators on these sets.

Proposition 2. [6] Assume an element U is given in the Hilbert space $l^2(X)$, where X is a bounded subset of \mathbf{Z}^2 . Let V be the projection of U on $\Psi_{(i,j)(m,n)}^{(k)}$. This projection can be computed in the following manner. Assume that the four points in the set

$$\mathcal{A} = \left\{ (s, t), (s, t + 1), (s + 1, t), (s + 1, t + 1) \right\} \tag{18}$$

are those that are mapped to (m, n) under the contraction $w_{i,j,(k)}$. Set $\delta = \delta_{(i,j)(m,n)}^{(k)} \geq 0$ and

$$r = \frac{\alpha}{4} \left[U(s, t) + U(s, t + 1) + U(s + 1, t) + U(s + 1, t + 1) \right] + \beta - U(m, n). \tag{19}$$

In the case that $(m, n) \notin \mathcal{A}$, the values $V(p, q)$ for $p, q \in \mathbf{Z}$ can be computed as

$$U(p, q) + \left\{ \begin{array}{l} \text{Case } (p, q) \in \mathcal{A} \\ \frac{-\alpha/4}{1+4(\alpha/4)^2}(r - \delta), r > \delta \\ 0, \quad |r| \leq \delta \\ \frac{-\alpha/4}{1+4(\alpha/4)^2}(r + \delta), r < -\delta \\ \\ \text{Case } (p, q) = (m, n) \\ \frac{1}{1+4(\alpha/4)^2}(r - \delta), r > \delta \\ 0, \quad |r| \leq \delta \\ \frac{1}{1+4(\alpha/4)^2}(r + \delta), r < -\delta \\ \\ \text{Case } (p, q) \notin \mathcal{A} \cup \{(m, n)\} \\ 0. \end{array} \right. \quad (20)$$

In the case that $(m, n) \in \mathcal{A}$, the values $V(p, q)$ for $p, q \in \mathbf{Z}$ can be computed as

$$U(p, q) + \left\{ \begin{array}{l} \text{Case } (p, q) \in \mathcal{A} \setminus \{(m, n)\} \\ \frac{-\alpha/4}{(1-\alpha/4)^2+3(\alpha/4)^2}(r - \delta), r > \delta \\ 0, \quad |r| \leq \delta \\ \frac{-\alpha/4}{(1-\alpha/4)^2+3(\alpha/4)^2}(r + \delta), r < -\delta \\ \\ \text{Case } (p, q) = (m, n) \\ \frac{1-\alpha/4}{(1-\alpha/4)^2+3(\alpha/4)^2}(r - \delta), r > \delta \\ 0, \quad |r| \leq \delta \\ \frac{1-\alpha/4}{(1-\alpha/4)^2+3(\alpha/4)^2}(r + \delta), r < -\delta \\ \\ \text{Case } (p, q) \notin \mathcal{A} \\ 0. \end{array} \right. \quad (21)$$

The above projection can be found using KKT conditions for solving constrained optimization problems. Another approach to prove this proposition is to view the optimization problem as a deblurring of a signal in presence of spatially-varying blur [14]. Now that the constraints $\Psi_{(i,j)(m,n)}^{(k)}$ and its associated projections are in hand we may apply these constraints in a POCS sequence along with other consistent constraints at the decoding stage.

5 Applications

5.1 Restoration of an Image with an Incomplete Fractal Code

Assume that we are given an incomplete fractal code of an image, i.e., some of the domain-range block assignments and corresponding greyscale map coefficients are missing. In such a case, the usual fractal decoding scheme, in which an arbitrary “seed” image is employed, will collapse since the range blocks of the image for which the fractal code is missing cannot be modified. These blocks will simply remain identical to the corresponding subblocks of the seed image.

To illustrate, let us consider the fractal code associated with the Lena image in Figure 1. Suppose that the fractal code corresponding to all range blocks in the bottom half of the image are missing. Using a random image as the seed, the limiting image produced by the fractal decoding procedure is shown in Figure 2(b). Only the domain blocks in the top-half of the image have been modified – those in the lower half are identical to their counterparts in the seed image. Likewise, if the fractal code is missing for the range blocks depicted in black in Figure 2(d), then fractal coding produces the limiting image in Figure 2(e). We now show how these two situations can be improved using POCS.

The incomplete fractal code problem is an underdetermined, hence ill-posed, inverse problem. There are many possible solutions. We now consider a POCS-type method that will employ the pointwise collage constraints $\Psi_{(i,j)(m,n)}^{(k)}$ with $\delta = 0$, associated with the partially known fractal code. For the problem associated with Figure 2(b), where the fractal code for the bottom half of image is missing, we obtain Figure 2(c) with this POCS method, having once again started with the random seed image. Figure 2(c) is seen to represent a significant improvement over Figure 2(b).

Along with the collage constraints, however, we can impose additional constraints as desired. For example, consider the missing fractal code problem of Figure 2(d). Standard fractal decoding yields Figure 2(e). The POCS method with an additional smoothness constraint produces Figure 2(f), a significant improvement over Figure 2(e). The smoothness constraint was imposed by means of a low-pass filtering operation in the frequency domain.

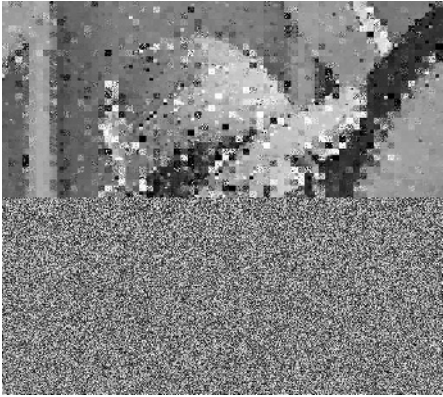
Let us now explain why the POCS method can yield better approximations to the original image in the case of missing fractal codes, with particular reference to Figures 2(b) and 2(c). As mentioned above, the usual fractal decoding iteration scheme of mapping domain blocks to range blocks does not change the range blocks for which the fractal code is missing, e.g., the bottom part of the image in Figure 2(b). On the other hand, the POCS method alters domain and range blocks simultaneously through projections. Because of these projections, it is possible that portions of the lower part of the image are modified since some *domain blocks* used in the fractal coding procedure come from that region. The more domain blocks that lie in the region of missing fractal code (hence missing range blocks), the “fuller” the attractor that is generated by the fractal coding procedure, hence the better the approximation to the original image.



(a)



(d)



(b)



(e)



(c)



(f)

Fig. 2. (a) Lena original. (b) Decoded as bottom half of the fractal code is missing, random seed. (c) Proposed using POCS. (d) A quarter of the code related to the range blocks in black is missing. (e) Decoded image starting from black seed, a quarter of the fractal code is randomly missing. (f) Proposed using POCS.

Here we come to a very important point regarding the POCS formulation of fractal coding. The usual fractal coding procedure involves mapping domain blocks onto range blocks. The POCS method of projections actually translates to a *simultaneous alteration of domain-range block pairs*. This accounts for the superior results of the POCS method in the incomplete fractal code problem.

5.2 Fractal Image Denoising

It is well known that subjecting a noisy image to a lossy compression scheme, e.g., JPEG, can produce some denoising of the image. This is also observed in the case of fractal image coding. If a noisy image is fractally coded, with little or no regard for compression, then the attractor produced by the fractal code often represents a significantly denoised version of the original image. (In fact, one can go some steps further and improve this procedure – see [9].) In such fractal image denoising algorithms, however, one performs the usual iteration procedure after the fractal code is obtained. Once again, regardless of the starting seed image, the procedure converges to the attractor of the fractal transform defined by the code. However, the POCS-based reconstruction algorithm explained in this paper can provide added flexibility. For example, if we choose the input noisy image as the starting point of the iteration, it is possible that the procedure converges to a noise-free image that is closer to the original noisy image.

As discussed earlier, the solution of the POCS-based reconstruction method – the point X of Section 3 – can also depend on the starting point of the iteration if the solution space is non-unique. A larger solution space for POCS-based fractal coding can be produced if the strict similarity constraints are relaxed to inequality constraints. Furthermore, if the original noisy image is chosen as the starting point of the POCS-based iteration, the information in this image is actually used in the reconstruction process, even after it has been used to determine the fractal code which determines the similarity/inequality constraints. Based on experiments, we believe that by choosing appropriate δ tolerance values in the constraints, one may obtain denoised images that are visually more pleasing because they are “closer” to the original image.

Figure 3 presents the result of an experiment that agrees with this claim. The original and a noisy version of a 256×256 image of Lena are shown in Figures 3(a) and 3(b). The fractal transform of the noisy image in Figure 3(b) corresponding to 8×8 pixel range blocks and 16×16 pixel domain blocks was then computed. Traditional fractal decoding produces the attractor shown in Figure 3(c). In spite of its blockiness, this image looks less noisy than the one in Figure 3(b).

Using the same fractal code as above, we then applied the following POCS-based reconstruction method. In order to enlarge the solution space we used $\delta = |r|/3$ at every point of the image in this experiment, where r is defined in Eq. (19). Finally, the starting point of the POCS iterations was the original noisy image shown in Figure 3(b). The result of this POCS-based procedure is shown in Figure 3(d). Although no postprocessing has been employed here, the blockiness plaguing Figure 3(c) is not visible in Figure 3(d). As well, it seems that more denoising has been achieved with Figure 3(d) than with Figure 3(c).



Fig. 3. (a) Lena original. (b) The noisy Image. (c) The IFS attractor of the noisy image, using 8×8 range blocks. (d) Proposed using POCS, the starting point is the noisy image, and the same fractal transform with 8×8 range blocks is used, with $\delta = |r|/3$.

6 Conclusions

In this paper, we have described a reformulation of traditional fractal image decoding in terms of the framework of projections onto convex subsets (POCS). In the case that all the constraints defined by the fractal code are applied to the problem then the solution, defined by the intersection of all constraints, is unique and corresponds to the fixed point of the contractive fractal transform. There is a difference, however, between the POCS method and the fractal coding method regarding the nature of the respective iteration procedures and the convergence

toward the solution. In the POCS method, the convergence is accomplished in a set-theoretic framework. The projections associated with POCS method involve simultaneous alteration of domain-range block pairs, unlike the case of fractal coding in which domain blocks are mapped onto range blocks, which may be viewed as a “greedy process.”

The principal advantage of the POCS framework is that it provides the flexibility to incorporate constraints and possibly additional knowledge about the reconstructed image. In this framework, it is also possible to replace the strict equality constraints associated with traditional fractal coding with inequalities that allow similarities to within some designated threshold. The latter represent more feasible and realistic conditions encountered in the real world. And, in this way, the set of feasible solutions is extended.

A POCS-based approach also provides the opportunity of solving underdetermined inverse problems in fractal coding as we have shown for the case of the incomplete fractal code problem.

Since the POCS framework allows fractal coding to be employed along with additional knowledge about the image/signal, some potential applications for future consideration include, but are not limited to, the elimination of post-processing in fractal decoding using POCS-based reconstruction and fractal coding with overlapping domain and range pools.

Finally, recall that if all similarity constraints defined by the fractal code are applied strictly, then the solution is unique, namely, the fixed point of the associated fractal transform operator. If additional constraints are applied in a POCS-based framework, then the entire set of constraints may be inconsistent, i.e., there is no “solution” that satisfies all constraints. In this interesting situation, one may need to employ the more recent POCS formulations for inconsistent feasibility problems as studied by P. Combettes [5].

Acknowledgements

This research was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) in the form of a Discovery Grant (ERV) which is hereby gratefully acknowledged.

References

1. M.F. Barnsley, *Fractals Everywhere*, Academic Press, New York, 1988.
2. M.F. Barnsley, V. Ervin, D. Hardin and J. Lancaster, Solution of an inverse problem for fractals and other sets, *Proc. Nat. Acad. Sci. USA* **83**, 1975-1977, 1985.
3. M.F. Barnsley and L.P. Hurd, *Fractal Image Compression*, A.K. Peters, Wellesley, Mass., 1993.
4. L. M. Bregman, The method of successive projection for finding a common point of convex sets, *Soviet Math. Dokl.*, vol. 6, pp. 688-692, May 1965.
5. P. L. Combettes, *Inconsistent signal feasibility problems: Least-squares solutions in a product space*, *IEEE Transactions on Signal Processing*, vol. 42, no. 11, pp. 2955-2966, November 1994.

6. M. Ebrahimi and E.R. Vrscay, *Generalized fractal image coding using projections onto convex sets*, (preprint).
7. Y. Fisher, *Fractal Image Compression, Theory and Application*, Springer Verlag, New York, 1995.
8. B. Forte and E.R. Vrscay, Theory of generalized fractal transforms, in *Fractal Image Encoding and Analysis*, Y. Fisher, Ed., NATO ASI Series F 159, Springer Verlag, New York, 1998.
9. M. Ghazel, G. Freeman and E.R. Vrscay, *Fractal image denoising*, *IEEE Transactions on Image Processing* **12**, no. 12, 1560-1578, 2003.
10. L.G. Gubin, B.T. Polyak and E.T. Raik, *The Method of Projections for finding the Common point of Convex Sets*, *USSR Comput. Math. and Phys.*, vol. 7, no. 6, pp. 1-24, 1967.
11. N. Lu, *Fractal Imaging*, Academic Press, NY, 1997.
12. H. Puh and P. L. Combettes, *Operator Theoretic Image Coding*, Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 4, pp. 1863-1866. Atlanta, GA, 1996.
13. H. Puh, *Set Theoretic Image Coding*, Ph.D. Thesis, Courant Institute, NY, 1996.
14. M.K. Ozkan, A.M. Tekalp, and M.I. Sezan, *POCS-based Restoration of Space-Varying Blurred Images*, *IEEE Transactions on Image Processing*, Volume 3, Issue 4, Page(s):450-454, July 1994.
15. D. Youla and H. Webb, *Image restoration by the method of convex projections: Part 1-Theory*, *IEEE Transactions on Medical Imaging*, vol. MI-1, no. 2, pp. 81-94, October 1982.

DCT-Domain Predictive Coding Method for Video Compression*

Kook-yeol Yoo

School of EECS, Yeungnam University, 214-1 Dae-Dong, Kyungsan-City,
Kyungpook 712-749, South Korea
kyoo@yu.ac.kr

Abstract. The H.263 Annex I method for the intraframe coding is based on the prediction in DCT domain, unlike JPEG, MPEG-1, and MPEG-2 where the intraframe coding uses block DCT, independent of the neighboring blocks. In this paper, we show the ineffectiveness of H.263 Annex I prediction method by mathematically deriving the spatial domain meaning of H.263 Annex I prediction method. Based on the derivation, we propose a prediction method which is based on the spatial correlation property of image signals. From the experiment and derivation, we verified the proposed method.

1 Introduction

With the broad deployment of Internet and broadband mobile network, the industrial demand on the visual communication over such networks is highlighted nowadays. These visual communications are fairly different from the traditional one over ISDN (Integrated Service Digital Network), in which the QoS(Quality of Service) is guaranteed [1]. For the Internet, the network bandwidth has time-varying feature and the packet can be lost due to network congestion [1,2]. On the other hand, the bit and burst errors in the mobile network cause the undesirable corruption in the video bitstream [3]. In general, the video bitstream is coded based on the temporal prediction, called motion compensation, for the high coding efficiency. This excellent method for video compression poses the serious quality degradation in temporal direction, i.e., the corruption in one frame is propagated into subsequent frames, simply temporal error propagation [3]. The error recovery is inevitable process in the video transmission over Internet and mobile networks.

The most effective method to cut off such temporal error propagation is very important thing for visual communication to have commercial competence, especially in the visual quality. Two methodologies are reported in the literatures. In [3], the resynchronization codeword, which is unique codeword not occurring in the video bitstream, is inserted intermittently in the video bitstream. The correct decoding can be resumed after detecting the resynchronization codeword. The resynchronization method itself is not sufficient for temporal error propagation problem, since it can

* This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

remove the spatial error propagation only. So resynchronization method only reduces the temporal error propagation, but still errors are propagated in temporal direction. The intraframe coding, independent of the previous frames can completely refresh the temporal errors in video sequences [4,5]. The intraframe coding, however, produces too much bits compared with interframe coding, resulting in great burden in network [6]. Therefore, the efficient intraframe coding is very important thing in the visual communication over Internet and mobile network [7].

For the standard codecs, such as JPEG, MPEG-1, and MPEG-2, the intraframe coding uses the block DCT, independent of neighboring blocks, followed by quantization, zig-zag scanning, RLC (Run-Length Coding), and VLC coding [8,9,10]. The intraframe coding method in the recently standardized H.263++ Annex I uses the DPCM (Differential PCM), i.e., the coefficients of current block DCT are predicted from those in the neighboring blocks and the difference DCT coefficients are coded, providing the better coding efficiency over the conventional independent block DCT coding.

In this paper, we mathematically analyzed the DCT-domain predictor of H.263++ Annex I to derive the physical meaning of the predictor. From the analysis, we show that the current DCT-domain predictor corresponds to the spatial prediction of a pixel from the pixel with 8 pixel distance. For the image signal, the spatial correlation drastically decreases when the pixel distance increases. To improve this inefficiency in H.263 Annex I predictor, we modified the DCT-domain predictor whose spatial domain meaning is the prediction of the pixel using the nearest pixels.

This paper is organized as follows: in Section 2, we briefly describe the H.263 Annex I intraframe coding method. The analysis of DCT-domain predictor in H.263 Annex I is provided and the proposed method is derived from the analysis in Section 3. The simulation results are given in Section 4 and our conclusion is drawn in Section 5.

2 H.263 Annex I Intraframe Coding Method

The Fig. 1 shows the overall block diagram of H.263 Annex I intraframe coding method. The input image is segmented into macroblocks as shown in Fig. 2-(a). Each macroblock consists of four luminance blocks and two chrominance blocks. Each block has 256 pixels, i.e., 8 pixels per line and 8 lines. Note that the basic unit of coding mode decision is macroblock. First, the pixels block is transformed into DCT coefficients using DCT as follows:

$$C(u, v) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} c(i, j) \cos\left(\frac{2\pi}{N} ui\right) \cos\left(\frac{2\pi}{N} vj\right) \quad (1)$$

where $c(i, j)$ and $C(u, v)$ represent the pixel value in the block coordinate $[i, j]^T$, $i, j \in [0, N-1]$ and DCT coefficient in the frequency $[u, v]^T$ $u, v \in [0, N-1]$, respectively. u and v represent the horizontal and vertical frequencies in DCT domain, respectively. Note that N is equal to eight since block consists of 8x8 pixels.

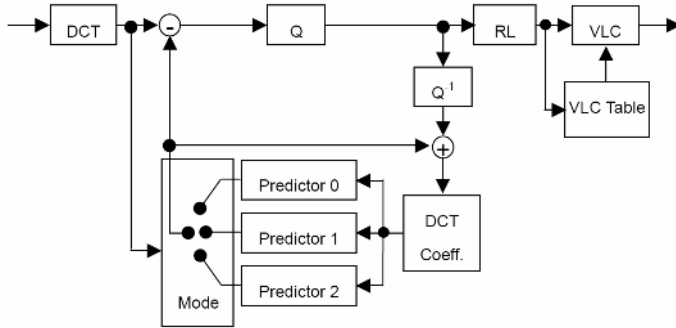


Fig. 1. H.263 Annex I encoder

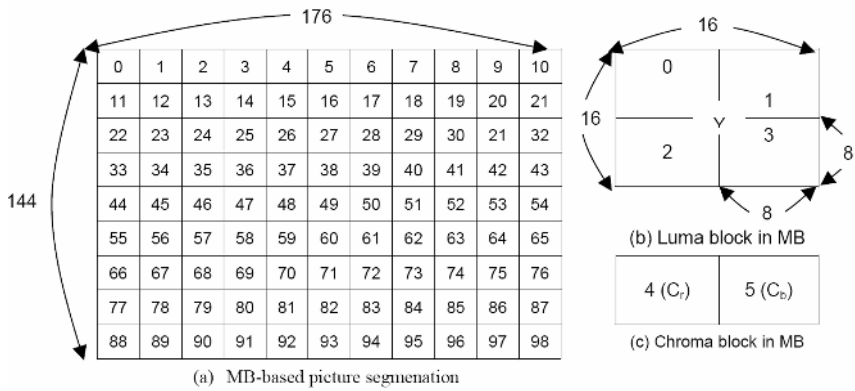


Fig. 2. MB-based image segmentation and MB configuration for QCIF image

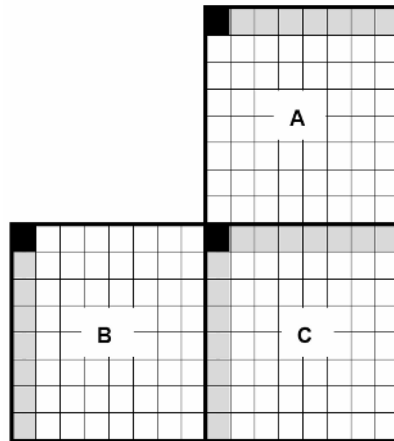


Fig. 3. The configuration of DCT block used in H.263 Annex I DCT coefficient prediction

Before coding of DCT coefficients, the DCT coefficients of current block are predicted using those of the neighboring blocks. Three prediction methods are used in H.263 Annex I. The configuration of blocks used in the prediction is shown in Fig. 3. The current block is denoted as c . The upper and left blocks to the current block are denoted as a and b , respectively. Also, the DCT coefficients of blocks, a , b and c are denoted as A , B , and C , respectively.

The prediction methods of the modes are as follows:

Mode 0: DC prediction only

$$\hat{C}(0,0) = (A(0,0) + B(0,0)) / 2 \tag{2}$$

$$\hat{C}(u,v) = 0, \text{ for } u, v \neq 0 \tag{3}$$

where $\hat{C}(u,v)$ is the predicted version of DCT coefficients of current block, $C(u,v)$. Notice that all the coefficients except DC component are coded without using DCT coefficient prediction.

Mode 1: DC and AC prediction using upper block

$$\hat{C}(u,0) = A(u,0), \text{ } u = 0, \dots, N - 1 \tag{4}$$

$$\hat{C}(u,v) = 0, \text{ for } v \neq 0 \tag{5}$$

Mode 2: DC and AC prediction using left block

$$\hat{C}(0,v) = B(0,v), \text{ } v = 0, \dots, N - 1 \tag{6}$$

$$\hat{C}(u,v) = 0, \text{ for } u \neq 0 \tag{7}$$

$$M = \arg \min_{M \in \{0, 1, 2\}} \text{SAD}_M \tag{8}$$

$$\text{SAD}_M = \sum_{i=0}^6 |C_i(0,0) - \hat{C}_i(0,0)| + \sum_{j=1}^{N-1} \left\{ |C_i(u,0) - \hat{C}_i(u,0)| + |C_i(0,v) - \hat{C}_i(0,v)| \right\} \tag{9}$$

where subscript i represents i -th block, and block with block index i is depicted in Fig. 2-(b). SAD_M represents the SAD when using prediction mode M . For instance, when M is equal to 0, it represents the Mode 0, DC prediction only.

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 11 | 12 | 13 | 14 |
| 5 | 6 | 9 | 10 | 18 | 17 | 16 | 15 |
| 7 | 8 | 20 | 19 | 27 | 28 | 29 | 30 |
| 21 | 22 | 25 | 26 | 31 | 32 | 33 | 34 |
| 23 | 24 | 35 | 36 | 43 | 44 | 45 | 46 |
| 37 | 38 | 41 | 42 | 47 | 48 | 49 | 50 |
| 39 | 40 | 51 | 52 | 57 | 58 | 59 | 60 |
| 53 | 54 | 55 | 56 | 61 | 62 | 63 | 64 |

(a) Mode 1

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 1 | 5 | 7 | 21 | 23 | 37 | 39 | 53 |
| 2 | 6 | 8 | 22 | 24 | 38 | 40 | 54 |
| 3 | 9 | 20 | 25 | 35 | 41 | 51 | 55 |
| 4 | 10 | 19 | 26 | 36 | 42 | 52 | 56 |
| 11 | 18 | 27 | 31 | 43 | 47 | 57 | 61 |
| 12 | 17 | 28 | 32 | 44 | 48 | 58 | 62 |
| 13 | 16 | 29 | 33 | 45 | 49 | 59 | 63 |
| 14 | 15 | 30 | 34 | 46 | 50 | 60 | 64 |

(b) Mode 2

Fig. 4. Alternative DCT scanning patterns for H.263 Annex I

After mode decision, the predicted error coefficients are quantized and are scanned into one-dimensional coefficients sequence. The scanning pattern for Mode 0 is conventional zig-zag scanning. The scanning patterns for Mode 1 and Mode 2 are depicted in Fig. 4-(a) and Fig.4-(b), respectively.

The scanned DCT coefficients are mapped into (RUN, LEVEL, EOB) by using RLC (Run Length Coding) where EOB represents End Of Block. These 3-D symbols are entropy coded by using VLC table. Note that the VLC table used in H.263 Annex I is different table used in H.263 codec not using Annex I coding mode, since the symbol statistics using Annex I coding mode is fairly different from the ones not using Annex I coding mode.

3 Proposed DCT Coefficient Prediction Method

To know the physical meaning of H.263 Annex I predictor in the spatial domain, we first investigate the Mode 1 predictor. By applying Eq. (1) into Eq. (4), we get the following equation:

$$E(u,0) = C(u,0) - \hat{C}(u,0) \tag{10a}$$

$$= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \{c(i, j) - a(i, j)\} \cos\left(\frac{2\pi}{N} ui\right) \cos\left(\frac{2\pi}{N} vj\right) \tag{10b}$$

$$= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \{c(i, j) - a(i, j)\} \cos\left(\frac{2\pi}{N} ui\right) \tag{10c}$$

where $E(u,0)$ represent the prediction error DCT coefficients. If the position vector in the current block is transformed into the position vector in the current image, we can represent it as Eq. (11). Similarly we can transform the block position vector of the upper block into the image position vector of Eq. (12).

$$I(x, y) = c(i, j) \tag{11}$$

$$I(x, y - N) = a(i, j) \tag{12}$$

where the position vector, $[x, y]^T$ in the current image corresponds to the position vector, $[i, j]^T$ in the current block. By substituting Eqs. (11) and (12) into Eq. (10), we can represent as follows:

$$E(u,0) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \{I(x+i, y+j) - I(x+i, y-N+j)\} \cos\left(\frac{2\pi}{N} ui\right) \tag{13}$$

Eq. (13) represents that the predictor in the H.263 Annex I corresponds to the spatial prediction of a pixel from the pixel with 8 pixels distance. Generally, the spatial correlation of image with respect to pixel distance is modeled as co-variance model of follows [12, 13]:

$$r(d) = \rho^{|d|} \tag{14}$$

where ρ is a image dependent parameter and $|d|$ is the pixel distance between two pixels to measure the correlation. The correlation to the pixel distance is depicted in Fig. 5. For the sample value of $\rho = 0.95$ presented in [14], the correlation using Mode 1, i.e., $d = 8$ is 0.663, resulting in very low spatial correlation. From this fact, we can easily know that the current DCT coefficient predictor of H.263 Annex I is not efficient predictor due to very low correlation.

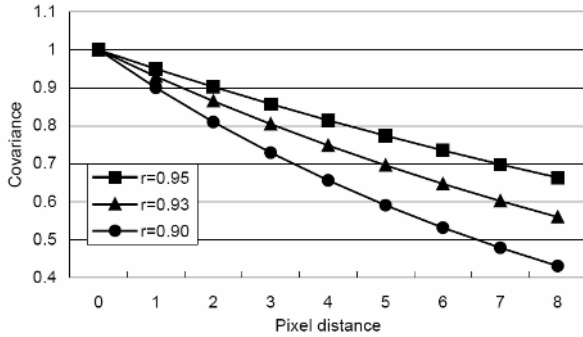


Fig. 5. Correlation with respect to pixel distance and ρ value in covariance image model

To increase the correlation between the pixels used in prediction, we propose to use the nearest pixel among the upper block to the current pixel, $c(i, j)$. In other words, the nearest pixel in the upper block, $a(i, j)$, $i, j \in [0, N - 1]$ is $a(i, N - 1)$ and we use this pixel, $a(i, N - 1)$ in the prediction of Eq. (10) instead of the pixel, $a(i, j)$ as follows:

$$\begin{aligned}
 E(u,0) &= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (c(i, j) - a(i, N - 1)) \cos\left(\frac{2\pi}{N} ui\right) \\
 &= C(u,0) - N \sum_{i=0}^{N-1} a(i, N - 1) \cos\left(\frac{2\pi}{N} ui\right)
 \end{aligned}
 \tag{15}$$

According to Eq. (15), the distances of pixels in the current are ranged 1 pixel to 8 pixels. So the average correlation of proposed prediction method is as follows:

$$r = \frac{1}{N} \sum_{d=1}^N \rho^d = \frac{\rho}{N} \frac{(1 - \rho^N)}{(1 - \rho)}
 \tag{16}$$

To know the difference in correlations of both methods, we compute the following:

$$\rho^N - \frac{1}{N} \sum_{d=1}^N \rho^d = \frac{1}{N} \sum_{d=1}^N (\rho^N - \rho^d)
 \tag{17}$$

For the magnitude comparison, we take the logarithm with base ρ into each term of summation, which is monotonically increasing function as follows:

$$\log_{\rho}(\rho^N) - \log_{\rho} \rho^d = N - d \geq 0
 \tag{18}$$

The above inequality and Eq. (17) shows that the proposed method gives always larger correlation than the H.263 Annex I prediction method. For the sampled value, $\rho = 0.95$, the correlation for the proposed method is larger by 0.136 than the H.263 Annex I prediction method.

Since the derivation of Mode 2 is the same with Mode 1, it is not described in this paper. For the DC component, we can easily know that the pixels line contingent current block has closer DC value of current block by considering the above derivation.

Based on the above derivation, we propose the following prediction method:

Mode 0: DC prediction only

$$\hat{C}(0,0) = (A(0,0) + B(0,0)) / 2 \quad (19)$$

$$\hat{C}(u,v) = 0, \text{ for } u, v \neq 0 \quad (20)$$

where

$$A(0,0) = \frac{1}{N} \sum_{i=0}^{N-1} a(i, N-1) \quad (21)$$

$$B(0,0) = \frac{1}{N} \sum_{j=0}^{N-1} b(N-1, j) \quad (22)$$

Mode 1: DC and AC prediction using upper block

$$\hat{C}(u,0) = N \sum_{i=0}^{N-1} a(i, N-1) \cos\left(\frac{2\pi}{N} ui\right), u = 0, \dots, N-1 \quad (23)$$

$$\hat{C}(u,v) = 0, \text{ for } v \neq 0 \quad (24)$$

Mode 2: DC and AC prediction using left block

$$\hat{C}(0,v) = N \sum_{j=0}^{N-1} b(N-1, j) \cos\left(\frac{2\pi}{N} vj\right), v = 0, \dots, N-1 \quad (25)$$

$$\hat{C}(u,v) = 0, \text{ for } u \neq 0 \quad (26)$$

4 Simulation Results

The evaluation of the proposed method is based on the common test condition used in the ITU-T SG16/Q.15 H.263++ standardization [15]. The test sequences used in simulation are described in Table 1.

Table 1. Test sequence used in the simulation

| Sequence | Spatial resolution | Temporal resolution |
|-----------|--------------------|---------------------|
| Foreman | | |
| Container | QCIF (176x144) | 10Hz |
| News | | |
| Paris | CIF(352x288) | |

The main investigating factor in this paper is the correlation between the pixel and its predicted version. We measure the correlations produced by the proposed method and H.263 Annex I prediction method for the test sequences of Table 1. Table 2 shows that the proposed method improves correlation for all the test sequences, implicating that the proposed method can give better prediction results and thereby, the proposed method has better coding efficiency.

According to the common test condition, the fixed quantization step size, i.e., QP=4, 5, 7, 10, 15, 25, is used for all the sequences. To obtain the averaged performance, the coding method is confined to intraframe coding for all the pictures. Tables 2

and 3 show that the proposed method gives about 2-3% reduction in total bitrate with marginal improvement in average PSNR. The ‘News’ and ‘Paris’ sequences whose correlations are highly increased than other sequences in Table 1 reduce more bits as shown in Table 3. Notice that the gain in bitrate reduction is decreased as the quantization step size is increased. This is originated from it that the proposed method uses only one pixels line left or up to the current block, thereby it is more sensitive to the coding noises. Note that the conventional H.263 Annex I uses 8x8 block DCT coefficients of upper or left block for the coefficient prediction.

The reason for the lower improvement, by the proposed method, compared with its increased correlation lies in VLC table. As we described in the Section II, the H.263 uses dedicate VLC table for H.263 Annex I, which is statistically tuned to H.263

Table 2. Comparison of correlation coefficients of the proposed and H.263 Annex I method

| Sequence | H.263 Annex I | | Proposed Method | |
|-----------|---------------|--------|-----------------|--------|
| | Mode 1 | Mode 2 | Mode 1 | Mode 2 |
| Foreman | 0.940 | 0.800 | 0.951 | 0.924 |
| Container | 0.754 | 0.786 | 0.793 | 0.797 |
| News | 0.585 | 0.551 | 0.726 | 0.641 |
| Paris | 0.620 | 0.596 | 0.735 | 0.651 |
| Mean | 0.725 | 0.683 | 0.801 | 0.753 |

Table 3. Average PSNR comparison

| QStep | Foreman | | Container | | News | | Paris | |
|-------|---------|-------|-----------|-------|-------|-------|-------|-------|
| | H.263 | Prop | H.263 | Prop | H.263 | Prop | H.263 | Prop |
| 4 | 40.45 | 40.45 | 40.57 | 40.58 | 41.07 | 41.07 | 40.12 | 40.15 |
| 5 | 38.39 | 38.40 | 38.359 | 38.60 | 38.94 | 38.96 | 37.97 | 37.98 |
| 7 | 36.36 | 36.37 | 36.56 | 36.57 | 36.82 | 36.85 | 35.76 | 35.78 |
| 10 | 34.52 | 34.53 | 34.65 | 34.68 | 34.83 | 34.83 | 33.72 | 33.73 |
| 15 | 31.95 | 31.97 | 32.07 | 32.10 | 32.02 | 32.07 | 30.89 | 30.91 |
| 25 | 29.19 | 29.22 | 29.01 | 29.06 | 28.92 | 29.00 | 27.81 | 27.85 |
| Mean | 35.14 | 35.16 | 35.24 | 35.27 | 35.43 | 35.46 | 34.38 | 34.40 |

Table 4. Bit reduction in percentage (%); H.263 in kbits/frame, Prop in % bit reduction relative to H.263

| QStep | Foreman | | Container | | News | | Paris | |
|-------|---------|------|-----------|------|-------|------|-------|------|
| | H.263 | Prop | H.263 | Prop | H.263 | Prop | H.263 | Prop |
| 4 | 51.0 | 2.38 | 47.5 | 2.19 | 50.7 | 3.20 | 255.8 | 2.94 |
| 5 | 41.4 | 2.49 | 38.6 | 2.45 | 42.1 | 3.40 | 212.9 | 2.95 |
| 7 | 31.6 | 2.36 | 29.5 | 2.60 | 32.9 | 3.21 | 167.0 | 2.88 |
| 10 | 23.7 | 2.01 | 22.5 | 2.18 | 25.3 | 2.71 | 128.7 | 2.82 |
| 15 | 15.8 | 1.70 | 15.6 | 1.37 | 17.6 | 2.52 | 89.5 | 2.74 |
| 25 | 9.2 | 0.52 | 9.5 | 0.17 | 10.9 | 2.19 | 54.5 | 1.92 |
| Mean | 28.8 | 1.91 | 27.2 | 1.83 | 29.9 | 2.87 | 151.4 | 2.83 |

Annex I prediction method. Since the statistics of symbol, (RUN, LEVEL, EOB) produced by proposed method is different from the that of H.263 Annex I, the VLC table used in comparison is not optimized to the proposed method. The further study in this work is, therefore, to design the VLC table suitable to the proposed method.

5 Conclusion

In this paper, we analytically showed that the DCT-domain predictor used in H.263 Annex I is inefficient by investigating the spatial correlation. Based on the analysis, we proposed the modified prediction method, in which the pixel with highest correlation to the predicted pixel is used in the prediction. From the simulation, we verified the proposed method and its improvements over the H.263 Annex I. For the further study, we leave the VLC table design suitable to proposed method.

References

1. Yoo, K.-Y.: New packetization method for error resilient video communications. Springer LNCS 3046 (2004), 329-323
2. Conklin, G.J., Greenbaum, G.S., et. al.: Streaming video over the Internet: approaches and directions. *IEEE Trans. on Circ. Syst. Video Technol.* 11 (2001) 282-300
3. Yoo, K.-Y.: Adaptive resynchronisation marker positioning method for error resilient video transmission. *IEE Electronics Letters* 34 (1998) 2084-2085
4. Liao, J. Y., Villasenor, J.: Adaptive intra block update for robust transmission of H.263. *IEEE Trans. on Circ. Syst. Video Technol.* 10 (2000) 30-35
5. Cote, G., Kossentini, F.: Optimal intra coding of blocks for robust video communication over Internet. *Signal Processing: Image Communication* 15 (1999) 25-34
6. Roh, B.H., Kim, J.K.: Starting time selection and scheduling methods for minimum cell loss ratio of superposed VBR MPEG video traffic. *IEEE Trans. on Circ. Syst. Video Technol.* (1999) 920-928
7. Kim, H.C, Yoo, K.-Y.: Complexity-scalable DCT-based video coding algorithm for computation limited terminals. *IEICE Trans. Communications* (2005) 2868-2871
8. Pennebaker, W.B., Mitchell, J. L: *JPEG: still image data compression standard*, New York, Van Nostrand (1993)
9. MPEG Editorial Group: Coding of moving pictures and associated audio for DSM at upto about 1.5Mbits/s: ISO/IEC 11172-2 Video IS. ISO/IEC JTC1/SC29/WG11 (1993)
10. MPEG Editorial Group: Generic coding of moving pictures and associated audio: ISO/IEC 13818-2 Video IS. ISO/IEC JTC1/SC29/WG11 (1995)
11. ITU-T: ITU-T Recommendation H.263, Video coding for low bit rate communication," ITU-T (1998)
12. Signal Process. Multimed. Lab., Univ. British Columbia: TMN 8 (H.263+) encoder/decoder, version 3.1.3," TMN 8 (H.263+) Codec (1998)
13. Lee, S.H., Yoo, K.-Y., Kim, J.-K.: A convergence analysis of a differential method for 2-D motion parameter estimation. *Journal of KICS* 23 (1998)1869-1882
14. Jain, A.K: *Fundamentals of digital image processing*, Prentice Hall Inc. (1989)
15. Sullivan, G.: Draft Meeting Report of the Eighth Meeting (Meeting H) of the ITU-T Q.15/16," ITU-T Q.15/16, Doc. # Q15-H37d1, Berlin, Germany (1999)

Simple Detection Method and Compensation Filter to Remove Corner Outlier Artifacts

Jongho Kim, Donghyung Kim, and Jechang Jeong

Department of Electrical and Computer Engineering, Hanyang University,
17 Haengdang, Seongdong, Seoul 133-791, Korea
{angel, kimdh, jjeong}@ece.hanyang.ac.kr

Abstract. We propose a simple detection method and compensation filter in order to remove corner outlier artifacts. The corner outlier artifacts are detected based on the directions of edges going through a block corner and the characteristics of blocks around the edges. According to the detection results, we compensate the stair-shaped discontinuities, the so-called corner outlier artifacts, using the neighboring pixels of the artifacts in spatial domain. Simulation results show that the proposed method improves, when combined with a deblocking filter in particular, both objective and subjective visual quality.

Keywords: Corner outlier artifact, deblocking filter, MPEG-4 video, low bit-rate video.¹

1 Introduction

The block-based processing plays an essential role in most video coding standards that have a hybrid structure including the motion compensated prediction and the block-based transform. Consequently, it produces undesired artifacts such as blocking artifacts, ringing noise, and corner outlier artifacts, particularly when the video is very highly compressed. Blocking artifacts are the grid noise along block boundaries in relatively flat areas; ringing noise is the Gibb's phenomenon owing to truncation of high-frequency coefficients by quantization; and corner outlier artifacts are a special case of blocking artifacts appearing at the cross-point of a block corner and a diagonal edge. To reduce blocking artifacts and ringing noise, a number of studies have been carried out in spatial [1, 2, 5] and transform domain [3, 4], respectively.

However, the corner outlier artifacts are still visible in some video sequences, since a deblocking filter is not applied to areas that include a large difference at a block boundary in order to avoid undesired blurring [1]. The corner outlier artifacts degrade visual quality remarkably, since the Human Visual System (HVS) is sensitive to variations of apparent edges. Nevertheless, only a few studies have been carried out on removing the corner outlier artifacts because the corner outlier artifacts appear in limited areas and the peak signal-to-noise ratio (PSNR) improvement is somewhat small [1]. Therefore, we propose a simple and effective method, which contains

¹ This work was supported by Seoul Future Contents Convergence (SFCC) Cluster established by Seoul Industry-Academy-Research Cooperation Project.

detection of the corner outlier artifacts and compensation filter to remove the artifacts, mainly in low bit-rate video. The proposed method can be used in combination with various deblocking [1-4] and deringing filters [5] for more improvement of visual quality.

The remaining parts of the paper are as follows. We present the detection method of corner outlier artifacts based on the pre-defined patterns, i.e., the direction of an edge going through a block corner, in Section 2. Section 3 describes, in detail, the proposed compensation filter to remove the corner outlier artifacts. Simulation results and conclusions are given in Section 4 and Section 5, respectively.

2 Detection of Corner Outlier Artifacts

Consider the original and reconstructed frames as illustrated in Fig. 1, where a diagonal edge **E** goes through a block corner. In Fig. 1, the edge divides into two areas represented in dark-shaded areas and white areas. The edge occupies large areas in block **B** and **C**, whereas it occupies very small areas (d_0 in Fig. 1) in block **D**. If block **D** is flat except d_0 , the AC coefficients of DCT of block **D** are mainly related to d_0 , and their values are small. Since most small AC coefficients are truncated by quantization in very low bit-rate coding, the area of d_0 , which represents the edge in block **D**, cannot be reconstructed shown as dashed line and area in Fig. 1(b). As a result, a visually annoying stair-shaped artifact is produced around the block corner. Such artifacts are called as corner outlier artifacts.

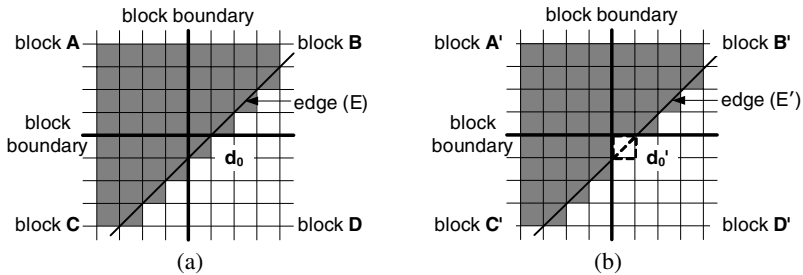


Fig. 1. Concept of a corner outlier artifact, (a) an edge going through a block corner in an original frame, (b) a corner outlier artifact located in d_0' in a reconstructed frame

Based on the two major observations, we propose a simple and effective detection method of the corner outlier artifacts. First, there is a large difference between boundary values of the block including the corner outlier artifact and the other three blocks around the cross-point. For example, the difference between d_0' and its upper pixel or between d_0' and its left pixel is large as shown in Fig. 1(b). Second, corner outlier artifacts are more noticeable in homogeneous areas, that is, each block around the cross-point is relatively flat. We investigate the characteristics of the blocks in terms of these observations around every cross-point in order to detect corner outlier artifacts appropriately. In addition, since the corner outlier artifacts are prominent when a diagonal edge occupies block areas unequally around a cross-point, we deal with four types, as depicted in Fig. 2, based on the edge direction. The corner outlier artifact

satisfying the proposed detection conditions among each detection type is regarded as an actual corner outlier artifact. Dealing with the pre-defined detection types instead of detecting an edge accurately has an advantage in reduction of computational complexity.

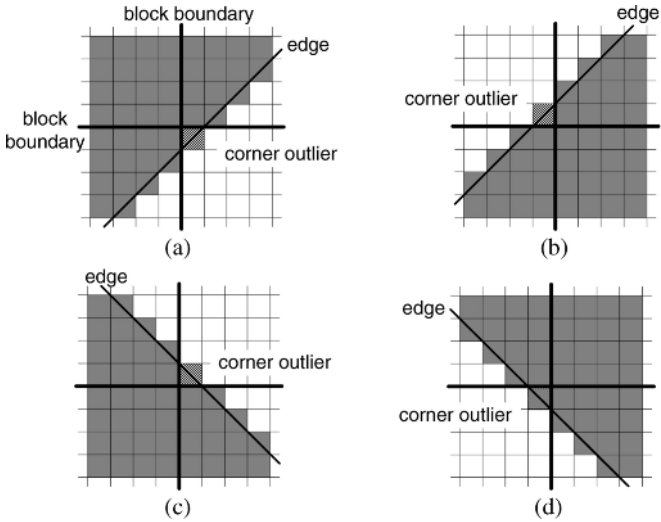


Fig. 2. Detection types based on the edge direction, (a) **D** vs. **{A, B, C}** with edge direction of lower 45°, (b) **A** vs. **{B, C, D}** with edge direction of upper 45°, (c) **B** vs. **{A, C, D}** with edge direction of upper 135°, (d) **C** vs. **{A, B, D}** with edge direction of lower 135°

To detect the corner outlier artifacts based on the first observation, we obtain the average values of the four pixels around the cross-point, which are represented as the shaded areas in Fig. 3, using

$$A_{avg} = \frac{1}{4} \sum_{i=1}^4 a_i, \quad B_{avg} = \frac{1}{4} \sum_{i=1}^4 b_i, \quad C_{avg} = \frac{1}{4} \sum_{i=1}^4 c_i, \quad D_{avg} = \frac{1}{4} \sum_{i=1}^4 d_i \quad (1)$$

where each capital and small letter denotes blocks and pixels, respectively, that is, A_{avg} is an average from a_1 to a_4 of block **A**, B_{avg} is an average from b_1 to b_4 of block **B**, and so on. Then, to select the detection type among the four pre-defined types, we examine the differences between the average values of the corner-outlier-artifact candidate block and its neighboring blocks, using the equations listed in Table 1.

Table 1. Detection criterion according to the detection types

| Candidate block | Detection type | Detection criterion |
|-----------------|-------------------------------|---|
| A | A vs. {B, C, D} | $ A_{avg} - B_{avg} > 2QP$ and $ A_{avg} - C_{avg} > 2QP$ |
| B | B vs. {A, C, D} | $ B_{avg} - A_{avg} > 2QP$ and $ B_{avg} - D_{avg} > 2QP$ |
| C | C vs. {A, B, D} | $ C_{avg} - A_{avg} > 2QP$ and $ C_{avg} - D_{avg} > 2QP$ |
| D | D vs. {A, B, C} | $ D_{avg} - B_{avg} > 2QP$ and $ D_{avg} - C_{avg} > 2QP$ |

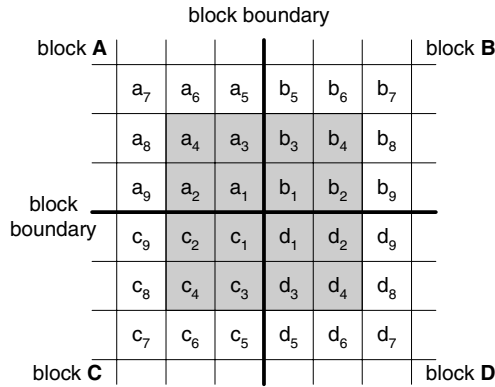


Fig. 3. Arrangement of pixels and their indices around the cross-point for detecting and filtering corner outlier artifacts. Shaded pixels are used for detection with the first observation and all pixels in this figure are used for detection with the second observation and filtering.

By using the average values around the cross-point instead of each pixel value, we can reduce detection errors in relatively complex areas. In Table 1, QP denotes the quantization parameter used as a threshold for deciding the corner outlier artifact in terms of the first observation. When block **A** has a corner outlier artifact, the difference between A_{avg} and B_{avg} , and the difference between A_{avg} and C_{avg} are larger than a quality-dependent value, $2QP$, thereby we consider the block **A** has a corner outlier artifact by the first observation. Practically, since we do not know the corner-outlier-artifact candidate block, four detection criteria listed in Table 1 should be investigated. If two corner outlier artifacts appear at one cross-point, the artifacts are arranged on diagonally opposite sides because the corner outlier artifacts cannot be placed vertically or horizontally. In this case, the proposed method can detect both artifacts without additional computations, since we examine the four detection types independently.

To detect the corner outlier artifacts based on the second observation, we examine whether the block satisfying the condition in Table 1 is flat or not. That is, when the candidate block is block **A**, we examine flatness of block **A** with respect to the pixel including the corner outlier artifact using

$$A_{flat} = \sum_{i=2}^9 |a_1 - a_i| \leq QP \tag{2-1}$$

where a_i represents the corner-outlier-artifact candidate pixel. In case where other block is determined as a candidate block, we examine the following conditions, respectively.

$$B_{flat} = \sum_{i=2}^9 |b_1 - b_i| \leq QP, \quad C_{flat} = \sum_{i=2}^9 |c_1 - c_i| \leq QP, \quad D_{flat} = \sum_{i=2}^9 |d_1 - d_i| \leq QP \tag{2-2}$$

where each capital and small letter denotes blocks and pixels, respectively, and QP is the quantization parameter. We regard the block that satisfies both conditions of Table 1 and (2) as a corner-outlier-artifact block, and then the proposed filter is applied to the block in order to remove the artifact.

3 Compensation Filter to Remove Corner Outlier Artifacts

To remove the corner outlier artifacts, we propose a simple filter that compensates the stair-shaped discontinuity using its neighboring pixels. To reduce computational complexity, we apply the proposed filter under the assumption that the edge has a diagonal direction instead of detecting the actual edge direction in neighboring blocks. This assumption is reasonable because:

1. Corner outlier artifacts created by a diagonal edge are more noticeable than artifacts created by a horizontal or vertical edge at a cross-point.
2. Various deblocking filters can remove corner outlier artifacts created by a horizontal or vertical edge.

According to this assumption and the detection results obtained in Section 2, when block **A** includes a corner outlier artifact as shown in Fig. 2(b), the pixels in block **A** are replaced by

$$\begin{cases} a_1' = (b_1 + b_2 + c_1 + d_1 + d_2 + c_3 + d_3 + d_4) / 8 \\ a_2' = (a_2 + a_1' + b_1 + c_2 + c_1 + d_1 + c_4 + c_3 + d_3) / 9 \\ a_3' = (a_3 + b_3 + b_4 + a_1' + b_1 + b_2 + c_1 + d_1 + d_2) / 9 \\ a_4' = (a_4 + a_3' + b_3 + a_2' + a_1' + b_1 + c_2 + c_1 + d_1) / 9 \\ a_5' = (a_5 + b_5 + b_6 + a_3' + b_3 + b_4 + a_1' + b_1 + b_2) / 9 \\ a_9' = (a_9 + a_2' + a_1' + c_9 + c_2 + c_1 + c_8 + c_4 + c_3) / 9 \end{cases} \quad (3)$$

where each index follows that of Fig. 3. This filter compensates the stair-shaped discontinuity using the average value of neighboring pixels along the diagonal edge. For block **B**, **C**, and **D**, the filtering methods are similar to (3) and actual equations are listed in Appendix.

4 Simulation Results

The proposed method is applied to ITU test sequences at various bit-rates. We use MPEG-4 verification model (VM) for low bit-rate DCT-based video compression [6]. Test sequences are coded with two coding structures: IPPP structure, in which all frames of a sequence are inter-frame coded except the first frame, and I-only structure, in which all frames are intra-frame coded. In IPPP structure, the H.263 quantization method with fixed QP is adopted, and motion search range is [-16.0, 15.5]. All 300 frames of each sequence are tested with given frame rates, bit-rates, and frame sizes. To arrive at a certain bit-rate, an appropriate quantization parameter is chosen and kept constant throughout the sequence. This can avoid possible side effects from typical rate control methods. The MPEG-4 decoded frames are obtained at the bit-rates of 10, 24, 48, and 64kbps, respectively. Test sequences for the simulation are listed in Table 2. To evaluate the proposed method, we apply the proposed filter to the MPEG-4 decoded frames with two cases: a case where all coding options are switched off, and a case where the deblocking filter option is turned on.

The simulation results in terms of PSNR for IPPP coded sequences are summarized in Table 3. It can be seen that PSNR results for the luminance component are increased by up to 0.02dB throughout the sequences. Just a slight improvement in PSNR is obtained due to limitation of the area satisfying the filtering conditions. However, the proposed filter improves subjective visual quality considerably, in particular for sequences with apparent diagonal edges such as *Hall Monitor*, *Mother & Daughter*, *Foreman*, etc. For emphasizing the effect of the proposed filter, partially enlarged frames for the first frame of *Hall Monitor* sequence under each method, i.e., the result of the proposed method without and with the deblocking filter, respectively, are shown in Fig. 4. Fig. 5 shows the result frames for the first frame of *Foreman* sequence with CIF size in order to observe the performance of the proposed filter in relatively large frame sizes.

Table 2. Test sequences for IPPP structure in the simulation

| Sequence | Bit-rate (kbps) | Frame rate (Hz) | Frame Size |
|-------------------|-----------------|-----------------|------------|
| Hall Monitor | 10 | 7.5 | QCIF |
| | 24 | 10 | QCIF |
| | 48 | 7.5 | CIF |
| Mother & Daughter | 10 | 7.5 | QCIF |
| | 24 | 10 | QCIF |
| Container Ship | 10 | 7.5 | QCIF |
| | 24 | 10 | QCIF |
| Foreman | 48 | 10 | QCIF |
| | 64 | 10 | QCIF |
| News | 48 | 7.5 | CIF |
| | 64 | 10 | CIF |
| Coastguard | 48 | 10 | QCIF |

Table 3. PSNR results for IPPP structure

| Sequence | QP | Bit-rate (kbps) | PSNR_Y (dB) | | | |
|-------------------|----|-----------------|--------------|--------------------------------|------------|------------------------------|
| | | | No filtering | No filtering + proposed filter | Deblocking | Deblocking + proposed filter |
| Hall Monitor | 18 | 9.39 | 29.8728 | 29.8851 | 30.1957 | 30.2090 |
| Mother & Daughter | 16 | 9.45 | 32.2256 | 32.2469 | 32.3684 | 32.3894 |
| Container Ship | 17 | 9.8 | 29.5072 | 29.5082 | 29.7281 | 29.7291 |
| Hall Monitor | 9 | 24.29 | 34.0276 | 34.0325 | 34.1726 | 34.1835 |
| Mother & Daughter | 8 | 23.83 | 35.3303 | 35.3316 | 35.2686 | 35.2708 |
| Container Ship | 10 | 21.61 | 32.5701 | 32.5711 | 32.6003 | 32.6013 |
| Foreman | 14 | 46.44 | 30.6237 | 30.6252 | 31.0727 | 31.0739 |
| Coastguard | 14 | 44.68 | 29.0698 | 29.0763 | 29.1562 | 29.1586 |
| Hall Monitor | 12 | 47.82 | 33.8216 | 33.8236 | 34.0921 | 34.0948 |
| News | 19 | 47.21 | 31.2192 | 31.2202 | 31.3516 | 31.3528 |
| Foreman | 12 | 64.65 | 31.4786 | 31.4798 | 31.7587 | 31.7598 |
| News | 16 | 63.14 | 32.0648 | 32.0658 | 32.1233 | 32.1243 |

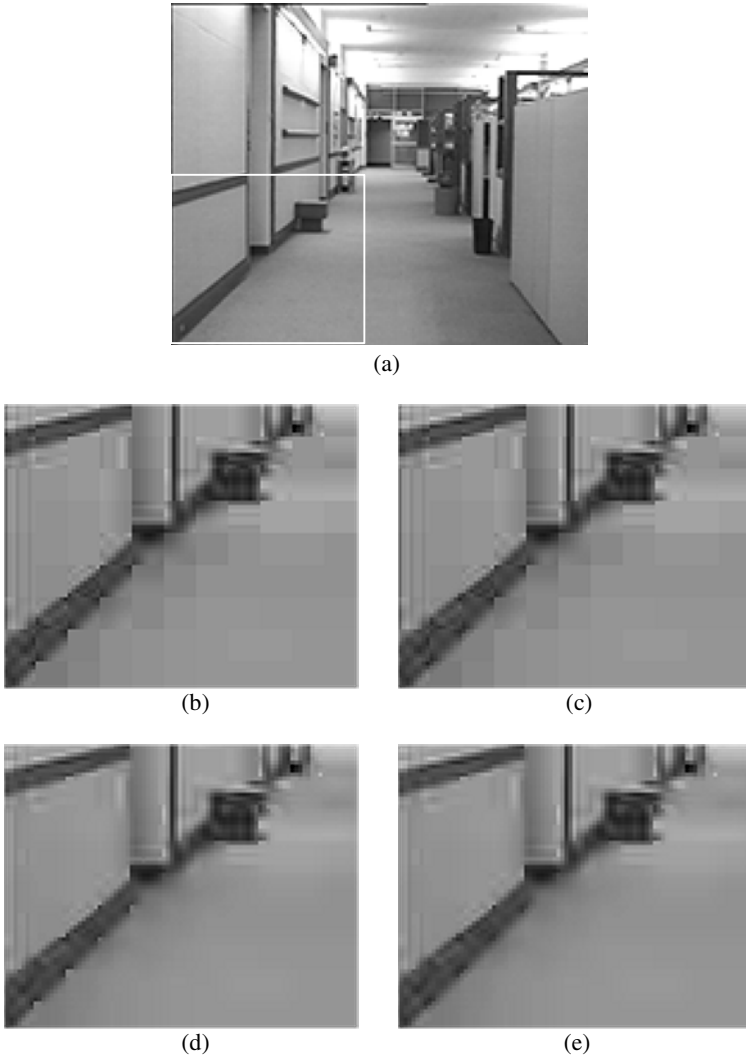


Fig. 4. Result images for *Hall Monitor* sequence, (a) Original sequence (QCIF, QP=17), (b) and (d) Partially enlarged images of MPEG-4 reconstructed images without and with the MPEG-4 deblocking filter, respectively, (c) and (e) Partially enlarged images of the proposed method without and with the MPEG-4 deblocking filter, respectively

The test conditions, coding options, and evaluation methods of I-only structure are similar to the case of IPPP structure. In this case, we pay attention to the filtering results along with variation of QP instead of bit-rates. Table 4 shows the results for I-only coded sequences of *Hall Monitor* and *Foreman*. The results are similar to the IPPP coded case. The proposed filtering conditions are satisfied at low QP, since each frame is relatively flat originally for sequences with low spatial details such as *Hall Monitor*. On the other hand, the proposed filtering conditions are satisfied at relatively

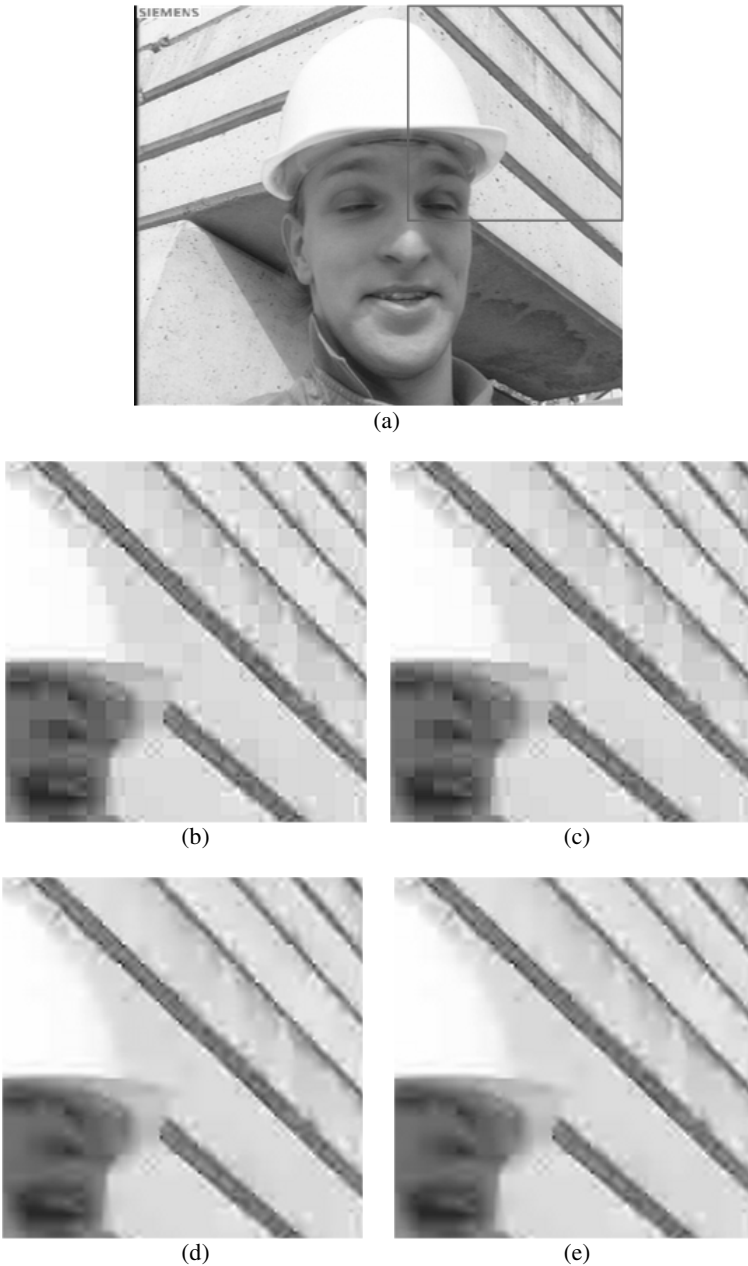


Fig. 5. Result images for *Foreman* sequence (a) Original sequence (CIF, QP=22), (b) and (d) Partially enlarged images of MPEG-4 reconstructed images without and with the MPEG-4 deblocking filter, respectively, (c) and (e) Partially enlarged images of the proposed method without and with the MPEG-4 deblocking filter, respectively

high QP, since each frame is flattened at high QP for sequences with medium or high spatial details such as *Foreman*. This tendency maintains in the sequences with or without the deblocking filter.

Table 4. PSNR results for I-only structure

| Sequence | Method | PSNR_Y (dB) | | | |
|--------------|---------------------|-------------|---------|---------|---------|
| | | QP=12 | QP=17 | QP=22 | QP=27 |
| Hall Monitor | No filtering | 32.8170 | 30.5284 | 28.9326 | 27.6323 |
| | No+proposed | 32.8396 | 30.5381 | 28.9374 | 27.6344 |
| | Deblocking | 33.2223 | 30.9624 | 29.4025 | 28.1188 |
| | Deblocking+proposed | 33.2422 | 30.9848 | 29.4052 | 28.1230 |
| Foreman | No filtering | 32.1830 | 30.0531 | 28.6406 | 27.5515 |
| | No+proposed | 32.1862 | 30.0606 | 28.6540 | 27.5684 |
| | Deblocking | 32.5741 | 30.5267 | 29.1635 | 28.1405 |
| | Deblocking+proposed | 32.5760 | 30.5316 | 29.1745 | 28.1526 |

5 Conclusions

The corner outlier artifacts are very annoying visually in low bit-rate video although they appear in limited areas. To remove the corner outlier artifacts, we have proposed a simple and effective post-processing method, which includes a detection method and a compensation filter. The proposed method, particularly in combination with deblocking and deringing filters, more improves both objective and subjective visual quality.

References

1. Park, H., Lee, Y.: A Postprocessing Method for Reducing Quantization Effects in Low Bit-rate Moving Picture Coding. *IEEE Trans. Circuits and Syst. Video Technol.*, Vol. 9. (1999) 161-171
2. Kim, S., Yi, J., Kim, H., Ra, J.: A Deblocking Filter with Two Separate Modes in Block-based Video Coding. *IEEE Trans. Circuits and Syst. Video Technol.*, Vol. 9. (1999) 156-160
3. Zhao, Y., Cheng, G., Yu, S.: Postprocessing Technique for Blocking Artifacts Reduction in DCT Domain. *IEE Electron. Lett.*, Vol. 40. (2004) 1175-1176
4. Wang, C., Zhang, W.-J., Fang, X.-Z.: Adaptive Reduction of Blocking Artifacts in DCT Domain for Highly Compressed Images. *IEEE Trans. Consum. Electron.*, Vol. 50. (2004) 647-654
5. Kaup, A.: Reduction of Ringing Noise in Transform Image Coding Using Simple Adaptive Filter. *IEE Electron. Lett.*, Vol. 34. (1998) 2110-2112
6. Text of MPEG-4 Video Verification Model ver.18.0. ISO/IEC Doc. N3908, (2001)

Appendix: Compensation Filters for Other Blocks that Include a Corner Outlier Artifact

In case where block **B** includes a corner outlier artifact, as seen in Fig. 2(c), the pixels in block **B** are replaced by

$$\begin{cases} b_1' = (a_1 + a_2 + d_1 + c_1 + c_2 + d_3 + c_3 + c_4) / 8 \\ b_2' = (b_2 + b_1' + a_1 + d_2 + d_1 + c_1 + d_4 + d_3 + c_3) / 9 \\ b_3' = (b_3 + a_3 + a_4 + b_1' + a_1 + a_2 + d_1 + c_1 + c_2) / 9 \\ b_4' = (b_4 + b_3' + a_3 + b_2' + b_1' + a_1 + d_2 + d_1 + c_1) / 9 \\ b_5' = (b_5 + a_5 + a_6 + b_3' + a_3 + a_4 + b_1' + a_1 + a_2) / 9 \\ b_9' = (b_9 + b_2' + b_1' + d_9 + d_2 + d_1 + d_8 + d_4 + d_3) / 9 \end{cases} \quad (\text{A-1})$$

in case where block **C** includes a corner outlier artifact, as seen in Fig. 2(d), the pixels in block **C** are replaced by

$$\begin{cases} c_1' = (d_1 + d_2 + a_1 + b_1 + b_2 + a_3 + b_3 + b_4) / 8 \\ c_2' = (c_2 + c_1' + d_1 + a_2 + a_1 + b_1 + a_4 + a_3 + b_3) / 9 \\ c_3' = (c_3 + d_3 + d_4 + c_1' + d_1 + d_2 + a_1 + b_1 + b_2) / 9 \\ c_4' = (c_4 + c_3' + d_3 + c_2' + c_1' + d_1 + a_2 + a_1 + b_1) / 9 \\ c_5' = (c_5 + d_5 + d_6 + c_3' + d_3 + d_4 + c_1' + d_1 + d_2) / 9 \\ c_9' = (c_9 + c_2' + c_1' + a_9 + a_2 + a_1 + a_8 + a_4 + a_3) / 9 \end{cases} \quad (\text{A-2})$$

and in case where block **D** includes a corner outlier artifact, as seen in Fig. 2(a), the pixels in block **D** are replaced by

$$\begin{cases} d_1' = (c_1 + c_2 + b_1 + a_1 + a_2 + b_3 + a_3 + a_4) / 8 \\ d_2' = (d_2 + d_1' + c_1 + b_2 + b_1 + a_1 + b_4 + b_3 + a_3) / 9 \\ d_3' = (d_3 + c_3 + c_4 + d_1' + c_1 + c_2 + b_1 + a_1 + a_2) / 9 \\ d_4' = (d_4 + d_3' + c_3 + d_2' + d_1' + c_1 + b_2 + b_1 + a_1) / 9 \\ d_5' = (d_5 + c_5 + c_6 + d_3' + c_3 + c_4 + d_1' + c_1 + c_2) / 9 \\ d_9' = (d_9 + d_2' + d_1' + b_9 + b_2 + b_1 + b_8 + b_4 + b_3) / 9 \end{cases} \quad (\text{A-3})$$

In each equation, the indices follow those of Fig. 3.

Rate Control Algorithm for High Quality Compression of Static Test Image in Digital TV System

Gwang-Soon Lee¹, Soo-Wook Jang², Chan-Ho Han², Eun-Su Kim³,
and Kyu-Ik Sohng²

¹ Electronics and Telecommunications Research Institute,
161 Gajeong-Dong, Yuseong-Gu, Daejeon 305-350, Korea
gslee@etri.re.kr

² School of Electrical Engineering and Computer Science, Kyungpook National University,
1370 San-Gyuk Dong, Buk-Gu, Daegu, 702-701, Korea
{jjang, chhan, kiso_hng}@ee.knu.ac.kr

³ Department of Electronic Engineering, Sunmoon University
eunsu.kim@sunmoon.ac.kr

Abstract. In this paper, we propose a new algorithm for generating test bitstream in high quality to evaluate the static image quality of DTV receiver, based on a target bit allocation in the process of rate control. In order to allocate the number of target bits, we consider the normalized complexities, which are updated or maintained according to GOP picture qualities. The proposed rate control method is suitable for the compression of static test pattern while MPEG-2 Test Model 5 is suitable for moving picture. To evaluate the performance of proposed algorithm, the test bitstream are generated by a MPEG-2 software encoder using the proposed algorithm. Experimental results show that average PSNR of the proposed method is higher than those of the conventional case. With experiment in DTV system, we have confirmed that the proposed algorithm has a stable bit rate and good video quality and it is suitable for evaluation of the DTV receiver.

1 Introduction

As the service using the digital TV (DTV) increases, how to measure a picture quality becomes the main issue in digital TV manufacturers. When an original image for a test has been encoded and decoded, there will be differences between the original and decoded images. So, generating the reference test stream to guarantee the high picture quality and stable bit rate is necessary for the test of DTV receiver [1]-[4]. Moreover, the test bitstream must satisfy MPEG-2 [5], DVB [6], and ATSC [7] standards.

The MPEG-2 Test Model 5 (TM5) [8] algorithm is widely used for bit rate control. In TM5, however, the target number of bits and the number of actual coding bits do not match well at static video compression, such as test patterns. To perform the high quality video compression for static test patterns such as color bar, pulse & bar, and multi-burst, we propose a method to generate test pattern in high quality by considering the partition of DCT block. Specially, we propose a new target bit allocation method using normalized complexities in the process of rate control, which are updated or

maintained by means of GOP picture qualities. The proposed method is using the fact that the generated bits and average quantization value have almost identical distribution per each GOP, and is suitable for compression of the static test pattern while the target bit allocation method in MPEG-2 TM5 is suitable for moving picture.

To evaluate the proposed algorithm, we generate the test bitstream about three static test patterns by using the proposed algorithm and TM 5 respectively. Experimental results showed that average PSNR of the proposed method is higher than those of the conventional case. And we have tested the generated test stream in experimental DTV broadcasting system, and confirmed that the proposed algorithm has a stable bit rate and good video quality.

2 Rate Control Algorithm of MPEG-2 TM5

The amount of bits being generated in MPEG-2 can be controlled by complexity of images and by some setting variables of an encoder. Therefore, the encoder can controls the amount of bits after estimating bits generation with a virtual buffer. The MPEG-2 TM5 (test model 5) rate control scheme comprises of three steps, target bits allocation, bit rate control and adaptive quantization [8], [9].

After encoding each picture, a global complexity measure, X_i , X_p , or X_b is updated as follows.

$$X_{[i,p,b]} = S_{[i,p,b]} \times Q_{[i,p,b]}, \quad (1)$$

where S_i , S_p , and S_b are the number of bits generated by encoding a picture type I, P, and B respectively, and Q_i , Q_p , and Q_b are their average quantization parameters over all the macroblocks in the picture. A target number of bits, T_i , T_p , or T_b is then assigned for the next picture of the same type in the GOP as follows.

$$T_i = \max \left\{ \frac{R}{1 + \frac{N_p X_p}{X_i K_p} + \frac{N_b X_b}{X_i K_b}}, \frac{\text{bit_rate}}{8 \times \text{picture_rate}} \right\}, \quad (2)$$

$$T_p = \max \left\{ \frac{R}{N_p + \frac{N_b K_p X_b}{K_b X_p} + \frac{N_b X_b}{X_i K_b}}, \frac{\text{bit_rate}}{8 \times \text{picture_rate}} \right\}, \quad (3)$$

$$T_b = \max \left\{ \frac{R}{N_b + \frac{N_p K_b X_p}{K_p X_b} + \frac{N_b X_b}{X_i K_b}}, \frac{\text{bit_rate}}{8 \times \text{picture_rate}} \right\}, \quad (4)$$

where K_p and K_b are constant parameters, R is the remaining number of bits assigned to the current GOP, N_p and N_b are the number of P and B pictures

remaining in the current GOP, respectively. *bit_rate* is the rate at which the coded bitstream is delivered from encoder to decoder, and *picture_rate* is the frame rate at which pictures are reconstructed from the decoding process.

In the process of the bit rate control, reference quantization variables are computed for each macroblock based on the status of virtual buffer fullness. Finally, the adaptive quantization makes the buffer status be even by controlling the quantization steps of the pictures being encoded, based on both the current buffer status and spatial divergence.

3 Generation of Test Pattern in High Quality

As a quantization process in encoder is lossless transform, the image compression using this technique brings the quantization error inevitably. This quantization error results in degrading image quality, so that a special method to reduce it is necessary to implement video test pattern in high quality for DTV system. In order to reduce the quantization error, the coefficients generated by DCT transform should be minimized and grouped at the low frequency area of DCT block. In case of generating a static test pattern for DTV, what we should consider is that patches on test image are made by 8×8 block which is quantization process unit. It means that patches on the test image are divided at the boundary of blocks as possible as it can. Moreover, one more thing we have to take into account is a block size at color images. Macroblock (MB) in color image can be comprised at the format of 4:4:4, 4:2:2 or 4:2:0 according to the rate of between luminance and chrominance signal, where 4:2:0 format is widely used in case of DTV broadcasting. So, in order to consider 8×8 block size at even chrominance images, Cb, Cr of 4:2:0 format, the patches on the test image should be made by 16×16 MB unit as much as possible.

We performed the computer simulation as shown in Fig. 1 to investigate the amount of bit generation according to patch's position on an image of one MB. Test MBs for this simulation is as shown in Fig. 2, where MB (a) is luminance level 128 and chrominance level 64, MB (b) is composed of a patch that is 8×8 block size and its luminance level 255, (c) and (d) are MB whose patch moves to the right. As a result, in case any border of patch is not included in the MB as shown in (a), quantization coefficients are generated to the least. On the other hand, in case the patch is included in the MB, quantization coefficients are generated more and more. Further, the more quantization coefficients are generated, the more quantization error which lead to degrade image quality, so that the compression ratio is decreased.

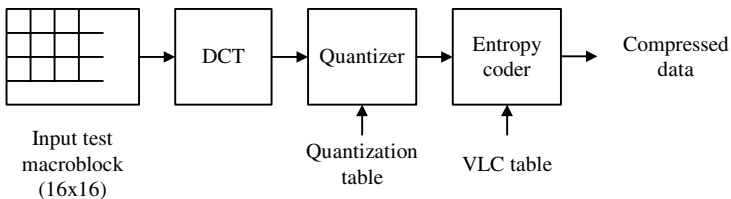


Fig. 1. Block diagram for measuring the generated bit at each MB patterns

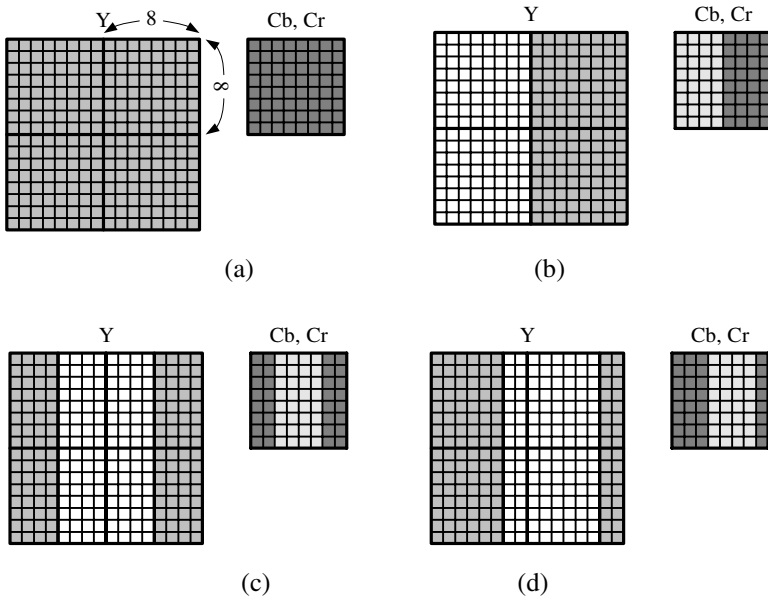


Fig. 2. MB patterns with variation of the patch position: (a) Patch 0, (b) Patch 1, (c) Patch 2 and (d) Patch 3

The amount of bits generated at this simulation is shown in Table 1. From the result of this table, we can see that the amount of bit generation is the smallest in case there are no borders of patches in MB as shown in Fig.1 (a), being the secondary smallest in case of patches having border of 8×8 blocks as shown in Fig. 1(b).

Table 1. Generated bits of MB patterns with variation of the patch position

| | Patch 0 | Patch 1 | Patch 2 | Patch 3 |
|----------------|---------|---------|---------|---------|
| Generated Bits | 34 | 148 | 272 | 453 |

4 Proposed Rate Control Algorithm for Static Images

In this paper, we propose new algorithm for target bit allocation during rate control to encode the static test images in high quality at MPEG-2 vide encoder. We could see that target bits in MPEG-2 TM 5 are allocated variably according to each I, P and B pictures in order to be suitable for compression of the moving picture, as shown in equation (2), (3), (4). On the other hand, our method for target bit allocation is suitable for static images like test images that can be used for the DTV receiver.

When the images composed of little moving objects, like static test patterns, are encoded, the generated bits and average quantization variables are almost identical per each GOPs. In this condition, it is reasonable to allocate the target bits by GOP unit because information used at the previous GOP can be reference to the complexity

estimation of the current GOP. Therefore, we propose the algorithm to maintain the optimum picture quality by checking the picture quality per GOP, as shown in Fig. 3. The rate control at initial GOP of image sequences is performed by the same way as the TM 5. The proposed algorithm at first calculates an average picture quality per GOP, if it is bigger than the maximum picture quality that is also updated per each GOP, then distribution of bit allocation in GOP are updated, or it is maintained as previous one.

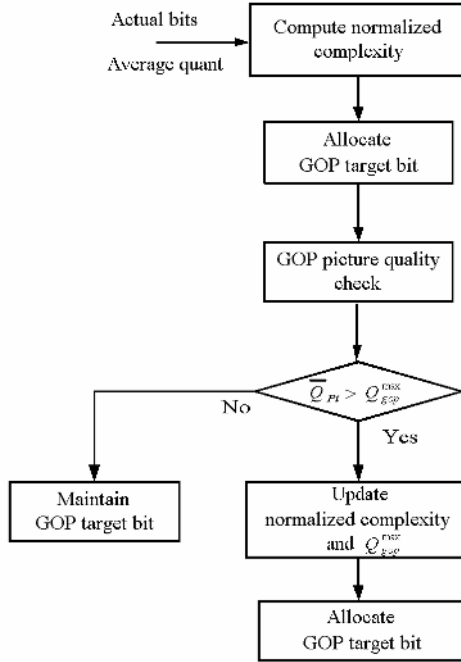


Fig. 3. The proposed algorithm for allocating the target bits suitable for encoding of static image

4.1 Target Bit Allocation by GOP Unit

In our algorithm, the initial values for bit allocation are same with TM 5. From the second GOP, in order to use the normalized complexity of previous GOP and to estimate that of current GOP, we define the normalized complexity of *i*-th GOP, $X_i^{norm}(k)$, as

$$X_i^{norm}(k) = \frac{\alpha(k) \cdot X_i(k)}{X_{total}} \text{ for } k = 0, 1, 2, \dots, N-1, \tag{5}$$

where $X_{total} = \sum_{k=0}^{N-1} \alpha(k) X_i(k)$, and $X_i(k) = S_i(k) \times Q_i(k)$.

$X_i(k)$ is k -th picture complexity of i -th GOP, and $\alpha(k)$ is the ratio coefficient of picture type, I, P, and B. Therefore, the ratio of target number of bits, T_I , T_P , and T_B can be expressed as

$$T_I : T_P : T_B = X_I^{norm} : X_P^{norm} : X_B^{norm} \quad (6)$$

Finally, the number target of bits k -th picture of i -th GOP $T_i(k)$ is decided as

$$T_i(k) = X_i^{norm}(k) \times B_i^{total} \quad \text{for } k = 0, 1, 2, \dots, N-1, \quad (7)$$

where $B_i^{total} = R_{i-1} + N \times B_{pic}$, and $B_{pic} = \frac{R_{bit}}{R_{pic}}$. R_{i-1} is remaining bits of previous GOP, and B_{pic} is uniformly allocated bits per each picture. Consequently, the number of target bits, $T_i(k)$, is rationally allocated by means of normalized complexities, $X_i^{norm}(k)$. Besides the number of target bits, $T_i(k)$, is updated or maintained by means of GOP picture qualities.

4.2 Update of Target Bits by GOP Unit

In the proposed algorithm, in order to obtain the maximum image quality, we update or maintain the normalized complexities and the number of target bits for each GOP. The relationship between the GOP and proposed normalized complexity is shown in Fig. 4. In this paper, the target bits being calculated by equation (7) is updated or maintained according to average picture quality of GOP. At first, average picture quality of the present GOP, \bar{Q}_{Pi} , is calculated as

$$\bar{Q}_{Pi} = \frac{1}{N} \sum_{k=0}^{N-1} Q_{Pi}(k) \quad (8)$$

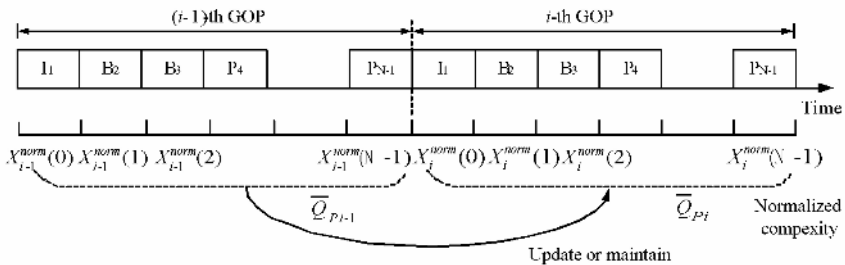


Fig. 4. Relationship between the GOP and proposed normalized complexity

where $Q_{Pi}(k)$ is picture quality of k -th picture in i -th GOP. Then, \bar{Q}_{Pi} is compared with Q_{GOP}^{max} which is maximum average picture quality up to the previous GOP.

If \bar{Q}_{Pi} is larger than Q_{GOP}^{\max} , then normalized complexity at the present GOP and the target bits using this are updated, or else the target bits at each picture inside GOP are maintained. As a result, the proposed algorithm makes sure the best picture quality by continuously updating the normalized complexity and target bits so that the average picture quality per GOP, \bar{Q}_{Pi} , can be the maximum.

5 Experiments and Results

To evaluate the performance of proposed algorithm, the test bitstream are generated by an MPEG-2 software encoder using either the proposed algorithm or MPEG-2 TM5 with three test patterns, multiburst, crosshatch, and KNU composite test pattern which is modified TCM test image [11], as shown in Fig. 5.

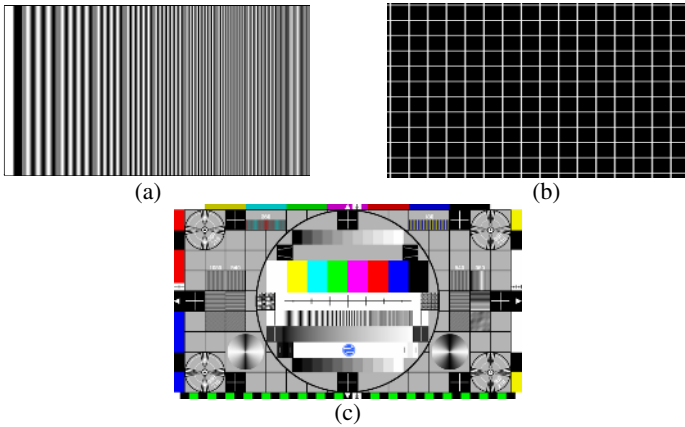


Fig. 5. (a) Multiburst (1280x720), (b) crosshatch (1920x1080), and (c) KNU composite test pattern (1920x1080)

The KNU composite test pattern is designed to be suitable by us for evaluating the static picture quality in DTV. The borders of patterns on these test images are aligned by 16 pixel macroblock or 8 pixel units to be encoded effectively. Further, the overflow problem at VBV buffer, which may cause in encoding process of the static images, was resolved by using a zero stuffing method [6]. In order to use the proposed method, we set $\alpha(k)$ in equation (5) as 1, 0.4 and 0.4 for I, P and B pictures respectively. In case $\alpha(k)$ becomes lower, the quantization scale factor become the minimum value, so that picture quality was no longer enhanced.

Table 1 shows their corresponding PSNR values of the three test streams. From this table, the PSNR of the proposed algorithm is about 3 to 5dB more efficient than that of MPEG-2 TM5 for luminance signal and about 1 to 2dB for chrominance signal. Fig. 6 shows variation of PSNR values at the luminance signal and chrominance signal when KNU composite video test pattern is encoded by 12 Mbps. From these figures, we can see the variation of picture quality become stable less than 1dB in

case of luminance signal. Fig. 7 compares a part of the decoded KNU test images and we can confirm image enhancement at the proposed method as well.

Table 1. The average PSNR of conventional and proposed method for static test patterns

| Test images | Video rate [Mbps] | PSNR [dB] | | | | | |
|---------------|-------------------|--------------|------|------|----------|------|------|
| | | Conventional | | | Proposed | | |
| | | Y | Cb | Cr | Y | Cb | Cr |
| Multiburst | 4 | 49.9 | INF | INF | 52.2 | INF | INF |
| Crosshatch | 6 | 44.4 | INF | INF | 59.6 | INF | INF |
| KNU Composite | 8 | 37.7 | 52.0 | 52.3 | 40.1 | 54.3 | 54.4 |
| KNU Composite | 12 | 49.2 | 58.8 | 59.1 | 54.0 | 59.1 | 59.3 |

INF : Infinity

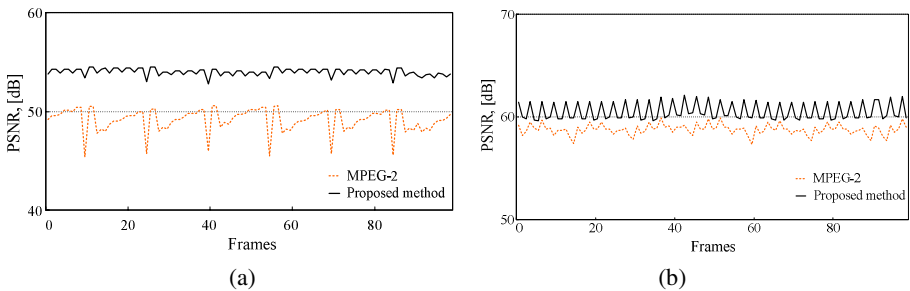


Fig. 6. PSNR values of KNU composite video test pattern encoded by 12 Mbps: (a) Luminance signal and (b) chrominance signal

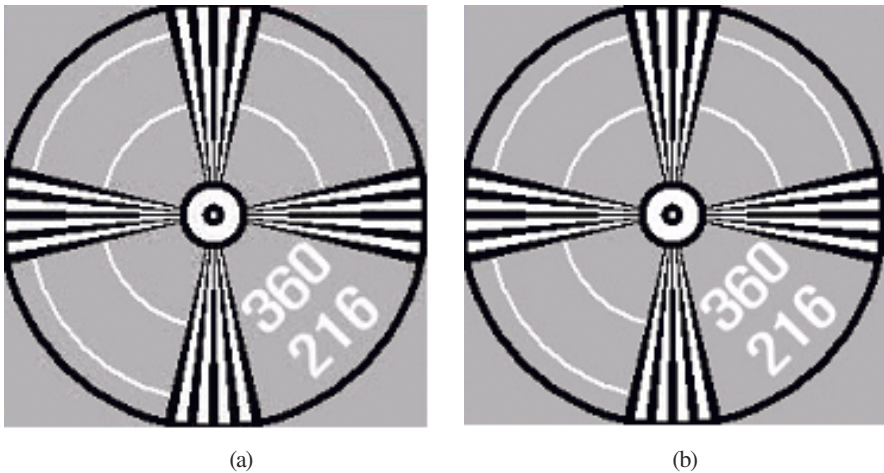


Fig. 7. Picture quality comparison of KNU composite video test pattern encoded by 12 Mbps; (a) MPEG-2 and (b) proposed method

6 Conclusions

This paper proposed a new algorithm for generating a high quality test bitstream to evaluate the static image quality of DTV receiver based on the rate control improvement. To evaluate the performance of proposed algorithm, the test bitstreams were generated by the MPEG-2 software encoder using the proposed algorithm. We can see that the proposed algorithm is about 2 to 5dB more efficient than that of MPEG-2 TM5. The result showed that the proposed bitstream has a good picture quality and stable bit rate, making it suitable to evaluate the picture quality of DTV. The proposed test bitstreams are currently being used in LG Electronics DTV production without any problems.

References

- [1] N. Raul, "Automated Test System for Digital TV Receivers," *IEEE International Conference on Consumer Electronics*, Los Angeles, USA, pp. 13-15, June 2000.
- [2] C.-M. Kim, B.-U. Lee, and R.-H. Park "Design of MPEG-2 Video Test Bitstreams," *Consumer Electronics, IEEE Transactions on*, vol. 45, pp.1213-1220, Nov. 1999.
- [3] K. D. McCann, "Testing and Conformance Checking in the Digital Television Environment," *IEE International Broadcasting Convention*, vol. 428, pp. 331-336, Sep. 1996.
- [4] P. Kavanagh, "Patterns for all Format," *IEE International Broadcasting Convention*, no. 414, pp. 63-69, Sep. 1995.
- [5] ISO/IEC JTC1, *Information technology - Generic coding of moving pictures and associated audio information, Part 2: Video International Standard (IS)*, ISO/IEC 13818-2, March 1996.
- [6] DVB, *Digital Broadcasting Systems for Television, Sound and Data Service; Specification for Service Information in Digital Video Broadcasting (DVB) systems*, ETSI 300 468, 1996.
- [7] ATSC, *ATSC Digital Television Standard, Rev. B*, ATSC Standard A/53B with Amendment 1, Aug. 2001.
- [8] ISO/IEC JTC1/SC29/WG11, *Test Model 5*, Draft, April 1993.
- [9] J. W. Lee and Y. S. Ho, "Target bit matching for MPEG-2 video rate control," *IEEE Region 10 International Conference on Global Connectivity in Energy, Computer, Communication and Control*, vol. 1, pp. 66-69, Sep. 1998.
- [10] Snell & Wilcox Internet Site: <http://www.snellwilcox.com>
- [11] S.-W. Jang and G.-S. Lee, "MPEG-2 Test Stream with Static Test Patterns in DTV System," *International Conference on Image Analysis and Recognition (ICIAR) 2005*, pp. 375-382, Sep. 2005.

JPEG2000-Based Resolution- and Rate-Constrained Layered Image Coding

Jing-Fung Chen¹, Wen-Jyi Hwang^{2,*}, and Mei-Hwa Liu³

¹ Digital Media Center, National Taiwan Normal University, Taiwan

² Graduate Institute of Computer Science and Information Engineering,
National Taiwan Normal University, Taipei, 117, Taiwan
`whwang@csie.ntnu.edu.tw`

³ Department of Electrical Engineering,
Chung Yuan Christian University, Chungli, 32023, Taiwan

Abstract. This paper presents a novel JPEG2000-based algorithm for realizing a layered image coding (LIC) system. In the algorithm, the resolution and rate associated with each layer of the LIC system can be pre-specified. The algorithm encodes an image one layer at a time using the modified JPEG2000 technique. The encoding process at each layer only covers the subbands having resolution level lower than the designated resolution at that layer subject to the pre-specified incremental rate budget. The encoding results at the previous layers will be used in the current layer to accelerate the encoding process. Simulation results show that the algorithm outperforms its counterparts for constructing the rate- and resolution-constrained LIC systems.

1 Introduction

In many multimedia applications over networks, users may have different demands on the quality of images. Therefore, it may be difficult for a server to provide an encoded bit stream satisfying all the requirements. One solution to this problem is to use the simulcast technique where an image is encoded and stored independently for each specific request. This approach requires more resources to be used in the encoder in terms of disk space and management overhead.

To eliminate this drawback, a number of layered image coding (LIC) [1] design algorithms have been proposed. In the LIC scheme, an encoded bit stream is delivered in more than one layer. By decoding bit streams accumulated up to different upper layers from the base layer, we reconstruct the transmitted image at different rates and/or resolutions. Users with various rate and resolution requirements therefore share the same LIC system for efficient image transmission. Many techniques such as embedded zerotree wavelet (EZW) [4], set partitioning in hierarchical trees (SPIHT) [3], and JPEG2000 [5] can be employed for realizing LIC systems. The EZW and SPIHT are only SNR-scalable since they always

* To whom all correspondence should be sent.

display reconstructed images in full resolution. The JPEG2000 supports both the SNR and resolution scalabilities. However, in the algorithm, the rate associated with each resolution level cannot be independently controlled. Therefore, the JPEG2000 may not be effective for realizing the LIC systems where the rate and resolution associated with each layer are desired to be pre-specified. In [2], a modification of the SPIHT, termed layered SPIHT (LSPIHT), is proposed for the realization of a resolution- and rate-constrained LIC. Nevertheless, since the JPEG2000 has better rate-distortion performance as compared with the SPIHT, it is expected that a modification of the JPEG2000 may outperform LSPIHT for the implementation of LIC systems.

This paper employs a layered JPEG2000 (LJPEG2000) technique for LIC design. In the LJPEG2000, the rate and resolution associated with each layer can be pre-specified before encoding. The transmitted images are reconstructed with rate and resolution identical to those of the highest layer accumulated by the decoder. Both the SNR and resolution scalabilities therefore can be achieved. The LJPEG2000 adopts a bottom-up design approach to satisfy both rate and resolution constraints at each layer. Starting from the base layer, the algorithm encodes one layer at a time using JPEG2000 subject to the pre-specified incremental rate budget at that layer until the design of the top layer is completed. The encoding process at each layer covers only the subbands with resolution lower than the resolution constraint at that layer. To enhance the performance of JPEG2000 at each layer, the encoding results of the previous layers are used. Simulation results show that the LJPEG2000 outperforms LSPIHT and simulcast systems subject to the same rate and resolution constraints at each layer.

2 Problem Formulation

Let \mathbf{x} be an image to be transmitted over the LIC system. The dimension of \mathbf{x} is assumed to be $2^p \times 2^p$. Let \mathbf{X} be the p -stage wavelet transform matrix of \mathbf{x} . Then, as shown in Figure 1, \mathbf{X} is also a $2^p \times 2^p$ matrix containing subbands \mathbf{x}_{L0} and $\mathbf{x}_{V_k}, \mathbf{x}_{Hk}, \mathbf{x}_{Dk}, k = 0, \dots, p - 1$, each with dimension $2^k \times 2^k$. Note that, in the wavelet transform matrix, the subbands \mathbf{x}_{Lk} (lowpass subbands at resolution level k), and $\mathbf{x}_{V_k}, \mathbf{x}_{Hk}, \mathbf{x}_{Dk}$ (V, H and D orientation selective highpass subbands

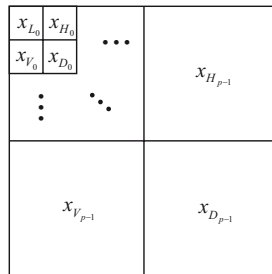


Fig. 1. The wavelet transform coefficients of an image \mathbf{x}

at resolution level k), $k = 0, \dots, p - 1$, are obtained recursively from $\mathbf{x}_{L(k+1)}$ with $\mathbf{x}_{Lp} = \mathbf{x}$, where the resolution level p is also referred to as the full resolution. The decomposition of $\mathbf{x}_{L(k+1)}$ into four subbands $\mathbf{x}_{Lk}, \mathbf{x}_{Vk}, \mathbf{x}_{Hk}, \mathbf{x}_{Dk}$ can be carried out using a simple quadrature mirror filter (QMF) scheme as shown in [6].

A typical implementation of LIC is shown in Figure 2, where the encoded bit streams for reconstructing images in different resolutions and rates are transmitted via more than one layer for decoding. Each layer is associated with a

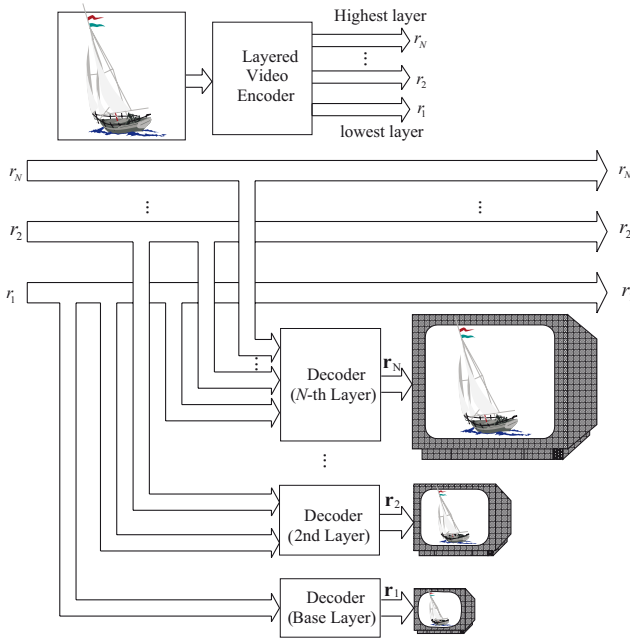


Fig. 2. Basic structure of a LIC

resolution level. The layers are arranged in such a way that layers having lower resolution are placed in lower positions in the LIC. Starting from the base layer (the layer in the lowest position, or the layer 1), the receivers can decode the bit streams up to any layer depending on their requirements for the reconstructed image. The resolution of the reconstructed image after decoding is the resolution of the layer in the highest position among the layers decoded by the receiver.

Suppose there are N layers in the LIC system. Let I_i be the resolution level associated with layer i . In addition, we assume $I_j \leq I_i$ for $j \leq i$. Given an image \mathbf{x} for LIC, the objective of encoding in the layer i is to code the lowpass subband \mathbf{x}_{LI_i} with resolution level I_i and rate budget r_i . To reconstruct \mathbf{x}_{LI_i} in the receiver, the bit streams from layer 1 (the base layer) up to layer i will be decoded. Let \mathbf{r}_i be the accumulated rate up to layer i . The accumulated rate for decoding at layer i , \mathbf{r}_i , is then related to the rate budget at layer i , r_i , by $\mathbf{r}_{i-1} + r_i = \mathbf{r}_i$, where we assume $\mathbf{r}_0 = 0$.

Assume the image to be encoded is with dimension $2^p \times 2^p$, then the bit budget at layer i and accumulated bits up to stage i are $2^p \times 2^p \times r_i$ and $2^p \times 2^p \times \mathbf{r}_i$, respectively. In the design of a LIC system, I_i and $r_i, i = 1, \dots, N$, in general are desired to be pre-specified and controlled to achieve both resolution and SNR scalabilities. We can determine the values of I_i and r_i from the requirement of codecs or from applications. The image encoding for the LIC system is then executed subject to these constraints.

3 LJPEG2000 Algorithm

The basic JPEG2000 uses the post-compression rate-distortion optimization (PCRD-opt) [5] algorithm for determining the rate allocated to each subband. Therefore, the rate used for the encoding of the low-pass subband \mathbf{x}_{LI_i} at each resolution level i cannot be prespecified/controlled. Consequently, the implementation of the rate- and resolution-constrained LIC using the JPEG2000 may be difficult. The LJPEG2000 technique can be used to solve this problem.

Starting from the base layer, the LJPEG2000 algorithm constructs one layer at a time until the design of the top layer is completed. The LJPEG2000 for the construction of each layer i is a JPEG2000 process covering the subbands having resolution level lower than I_i with the rate budget r_i . Since $I_j \leq I_i$ for $j \leq i$, the subbands encoded at each lower layer j are also encoded at layer i . Therefore, the LJPEG2000 at layer i can use the results of LJPEG2000 at layer $j, j = 1, \dots, i - 1$, to improve the coding efficiency. To see how the encoding results at the lower layers can be exploited, we first provide a brief overview of the JPEG2000. As shown in Figure 3, an image to be encoded by JPEG2000 is transformed using a wavelet transform and quantized. The quantization indices in each subband are divided into rectangular codeblocks. The quantization indices within each codeblock are compressed using a bitplane coder. The individual codeblock streams are then grouped together to form the JPEG2000 codestream.

Since the LJPEG2000 is an extension of the basic JPEG2000, it encodes each layer of the LIC on bitplane-by-bitplane basis. Suppose the encoding of layer $i - 1$ is completed, and the encoding of layer i is to be done. Let \mathcal{S}_i be the set of codeblocks in the subbands constituting the lowpass subband \mathbf{x}_{LI_i} . Consider a codeblock $B \in \mathcal{S}_i$. Suppose B also belongs to \mathcal{S}_{i-1} , and some most significant bitplanes of B have already been encoded at layer $i - 1$ and/or below. The encoding process at layer i does not have to encode these bitplanes because their encoded bitstreams can be found from the bitstreams at the lower layers. We call the removal of these bitplanes, the bitplane reduction of B at layer i . In the LJPEG2000, the encoding process at each layer can use the bitplane reduction technique for exploiting the encoding results at the lower layers.

To employ the bitplane reduction at each layer i , the bitplanes which have been encoded at the previous layers should be identified. This can be accomplished by obtaining the reconstructed codeblocks \hat{B}_{i-1} at layer $i - 1$. Note that

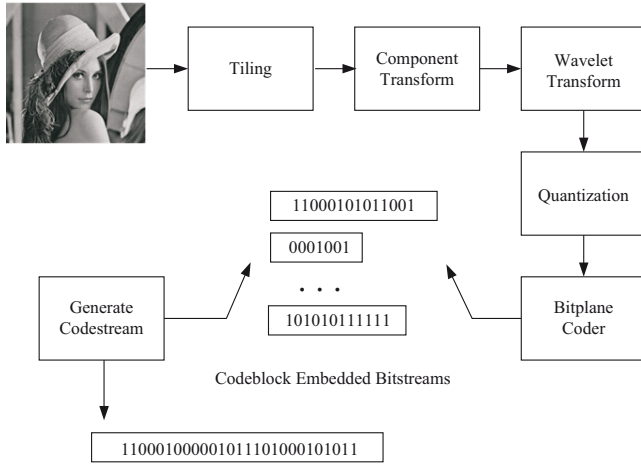


Fig. 3. Overview of the JPEG2000 algorithm

\hat{B}_{i-1} contains all the most significant bitplanes which have been encoded up to the layer $i - 1$. Let

$$b_i(B) = B - \hat{B}_{i-1}. \tag{1}$$

Therefore, $b_i(B)$ contains the bitplanes which have not been encoded in B . The bitplane encoder at layer i encodes the $b_i(B)$ for $\forall B \in \mathcal{S}_i$. The encoding of each bitplane here follows the same procedure as that of the basic JPEG2000. Let $\hat{b}_i(B)$ be the reconstructed $b_i(B)$ obtained from the decoder of the LJPEG2000 at the layer i . The reconstructed codeblock B at layer i is then given by

$$\hat{B}_i = \hat{B}_{i-1} + \hat{b}_i(B). \tag{2}$$

The encoding process at layer i is subject to the rate budget r_i . Therefore, an optimal rate allocation is necessary for determining the rate allocated to each codeblock $B \in \mathcal{S}_i$ at layer i . Let $d(B, \hat{B}_i)$ be the distortion of B at layer i , where $d(X, Y)$ is the squared distance between vectors X and Y . Let $r_i(B)$ be the rate allocated to B for the bitplane encoding of $b_i(B)$. The objective of the optimal rate allocation at layer i is to find an optimal set of rates $\{r_i^*(B), B \in \mathcal{S}_i\}$ such that

$$\begin{aligned} \{r_i^*(B), B \in \mathcal{S}_i\} = \arg \min_{\{r_i(B), B \in \mathcal{S}_i\}} & \sum_{B \in \mathcal{S}_i} d(B, \hat{B}_i), \\ \text{subject to} & \sum_{B \in \mathcal{S}_i} r_i(B) \leq r_i. \end{aligned} \tag{3}$$

The PCRD-opt technique in the basic JPEG2000 can be used to solve this problem. The cost function for the PCRD-opt at layer i is given as $\sum_{B \in \mathcal{S}_i} d(B, \hat{B}_i)$. After the optimal rate allocation, the bitstream for each codeblock $B \in \mathcal{S}_i$ are

grouped together to form the codestream of the LJPEG2000 at layer i . The same procedure is repeated for each layer until the codestream of the highest layer is formed. The complete flowchart of the LJPEG2000 algorithm is shown in Figure 4.

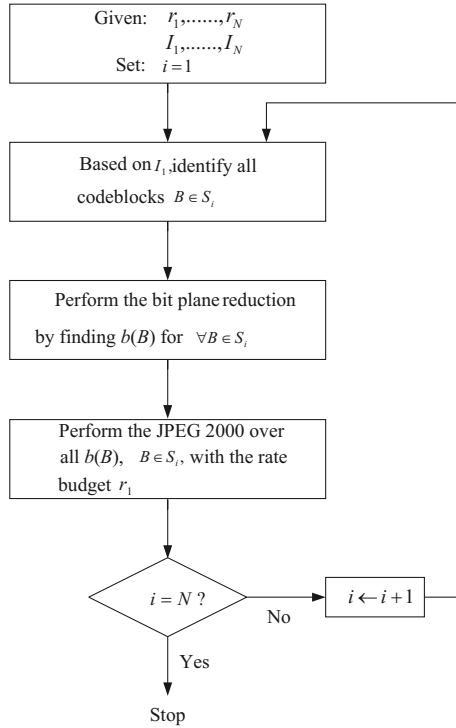


Fig. 4. The flowchart of LJPEG 2000 algorithm

4 Simulation Results

This section presents some simulation results of the LJPEG2000 technique for the applications of LIC systems. Table 1 shows the performance of the LIC realized by LJPEG2000 and LSPIHT for various test images. The dimension of the test images is 512×512 . Therefore, the full resolution level is $p = 9$. The wavelet transform matrices of the test images are obtained by the 9/7-tap filter. The number of layers in the LIC is $N = 3$. The resolution level of layers 1, 2 and 3 of the LIC are $I_1 = 7, I_2 = 8$, and $I_3 = 9$, respectively. The accumulated rate up to layers 1, 2 and 3 of the LIC are $\mathbf{r}_1 = 0.015, \mathbf{r}_2 = 0.03$ and $\mathbf{r}_3 = 0.083$, respectively. Therefore, the rate for the encoding of layers 1, 2 and 3 are $r_1 = 0.015, r_2 = 0.015$ and $r_3 = 0.053$, respectively. The performance for each layer is measured with PSNR. Let \mathbf{x} be the image to be encoded, then

Table 1. Performance of the LIC system realized by LSPIHT and LJPEG2000 for various test images

| System | LSPIHT-based LIC[2] | | | LJPEG2000-base LIC | | | |
|--------------------------------|---------------------|-------|-------|--------------------|-------|-------|-------|
| i | 1 | 2 | 3 | 1 | 2 | 3 | |
| I_i | 7 | 8 | 9 | 7 | 8 | 9 | |
| r_i | 0.015 | 0.015 | 0.053 | 0.015 | 0.015 | 0.053 | |
| \mathbf{r}_i | 0.015 | 0.03 | 0.083 | 0.015 | 0.03 | 0.083 | |
| PSNR _{i} | Barbara | 23.34 | 24.79 | 30.64 | 34.06 | 30.22 | 31.79 |
| | House | 30.22 | 32.82 | 38.39 | 40.19 | 38.48 | 40.46 |
| | Pepper | 27.12 | 30.37 | 34.42 | 35.24 | 34.25 | 34.42 |
| | Tree | 24.99 | 27.54 | 30.81 | 31.56 | 29.48 | 31.91 |

the lowpass subband \mathbf{x}_{LI_i} is the desired image to be reconstructed in the layer i . Let $\hat{\mathbf{x}}_{LI_i}$ be the reconstructed image in the layer i , and D_i be the mean squared distance between \mathbf{x}_{LI_i} and $\hat{\mathbf{x}}_{LI_i}$. Then the PSNR in the layer i , denoted by PSNR_i , is defined as $\text{PSNR}_i = 10 \log(255^2/D_i)$. In Table 1, we also note that r_i and \mathbf{r}_i indicates the required rates for encoding and reconstructing lowpass subband \mathbf{x}_{LI_i} in the layer i , respectively.

It is observed from Table 1 that LJPEG2000 outperforms LSPIHT at each layer. This is because LJPEG2000 and LSPIHT are based on a sequence of JPEG2000 and SPIHT operations, respectively. In addition, the JPEG2000 algorithm has been found to be more effective for image coding over the SPIHT algorithm. Figure 5 shows the original and reconstructed versions of the image “Barbara” coded by LSPIHT and LJPEG2000 at each layer. Note that the images in the layers 1 and 2 are of lower resolution, and therefore are smaller in size. However, in Figure 5, they are zoomed in/up to have same size image as in layer 3 for easy comparison. From Table 1 and Figure 5, it is observed that the LJPEG2000 technique provides high PSNR and excellent visual quality in each layer.

For comparison purpose, we also implement a JPEG2000-based simulcast system for the coding of $\mathbf{x}_{LI_i}, i = 1, \dots, 3$. In the system, the lowpass subimages $\mathbf{x}_{LI_i}, i = 1, \dots, 3$, are encoded independently using the basic JPEG2000 technique with the rate r_i shown in Table 1. Consequently, the rates for encoding each \mathbf{x}_{LI_i} in both Tables 1 and 2 are the same, and are equal to r_i . Table 2 shows the resulting PSNR of the reconstructed lowpass subbands $\mathbf{x}_{LI_i}, i = 1, \dots, 3$, for various images. From Tables 1 and 2, for the same test image, we observe that the LJPEG2000-based LIC system has higher PSNR than that of the JPEG2000-based simulcast system at each layer (except the lowest layer, where the PSNRs for both systems are the same because identical procedure is used for the coding at that layer). The LJPEG2000-based LIC enjoys better performance because less coding redundancy is observed in the system. To illustrate this fact in more detail, we first note that, the set of wavelet coefficients of image \mathbf{x}_{LI_j} is contained in that of \mathbf{x}_{LI_i} for $j < i$. In addition, in the JPEG2000-based simulcast system, the lowpass subbands \mathbf{x}_{LI_i} are independently encoded. Therefore, the common coefficients are repeatedly encoded in the system, and the overhead for the coding can be very high. By contrast, in the LIC system using our



Fig. 5. The reconstructed versions of the image “Barbara” at each layer of LIC systems shown in Table1 : (a): LSPIHT-based LIC system, (b): LJPEG2000-based LIC system

Table 2. Performance of the JPEG2000-based simulcast system for various test images. The rate for encoding $\mathbf{x}_{LI_i}, i = 1, \dots, 3$ is r_i , where the value of r_i is shown in Table 1.

| i | 1 | 2 | 3 |
|--|-------|-------|-------|
| I_i | 7 | 8 | 9 |
| PSNR _{i} Barbara | 34.06 | 26.26 | 29.81 |
| House | 40.19 | 35.32 | 38.69 |
| Pepper | 35.24 | 30.26 | 33.30 |
| Tree | 31.56 | 25.68 | 30.44 |

LJPEG2000 technique, the encoding of each layer effectively utilizes the results of JPEG2000 in the previous layers via bitplane reduction to enhance the coding efficiency. Hence, the redundancy for encoding the coefficients which are common to every layer in the LIC system can be effectively reduced. Figure 6 shows the reconstructed “Barbara” at each layer of the LJPEG2000-based LIC system and JPEG2000-based simulcast system subject to the same rate for encoding. As compared with the JPEG2000-based simulcast, the LJPEG2000-based LIC has better visual quality at layers 2 and 3. Consequently, based on the same rate for encoding each lowpass subimage \mathbf{x}_{LI_i} , our LJPEG2000 method outperforms the original JPEG2000 method.

To further assess the performance of LJPEG2000 technique, we also compare the LJPEG2000-based LIC system with the JPEG2000-based simulcast system subject to the same rate for decoding each lowpass subband. In the LIC system, the rates for the encoding and decoding in each layer i are different, and are given as r_i and \mathbf{r}_i , respectively. In the simulcast system, however, the rates for encoding and decoding are the same. Now, based on the same rate \mathbf{r}_i shown in Table 1 for reconstructing \mathbf{x}_{LI_i} , the rate required for encoding in the JPEG2000-based simulcast systems is therefore also \mathbf{r}_i . Table 3 shows the resulting PSNR of the reconstructed \mathbf{x}_{LI_i} of various test images encoded with rate \mathbf{r}_i . Figure 7 shows

Table 3. Performance of the simulcast system for various test images. The rate for encoding $\mathbf{x}_{LI_i}, i = 1, \dots, 3$ is \mathbf{r}_i , where the value of \mathbf{r}_i is shown in Table 1.

| i | 1 | 2 | 3 |
|--|-------|-------|-------|
| I_i | 7 | 8 | 9 |
| PSNR _{i} Barbara | 34.06 | 30.43 | 32.48 |
| House | 40.19 | 39.00 | 40.95 |
| Pepper | 35.24 | 34.43 | 34.67 |
| Tree | 31.56 | 29.68 | 32.41 |

the reconstructed “Barbara” for this case. Since the rate for encoding of \mathbf{x}_{LI_i} in the simulcast system is \mathbf{r}_i ; whereas, the rate for the encoding of \mathbf{x}_{LI_i} is only r_i in the LIC system, from Tables 1 and 3, we find that the JPEG2000-based simulcast system has slightly higher PSNR for each $\mathbf{x}_{LI_i}, i = 2, 3$. Nevertheless, as shown in Figure 7, the visual quality of both systems are similar. Although the simulcast system based on the JPEG2000 technique performs better in this case, it requires

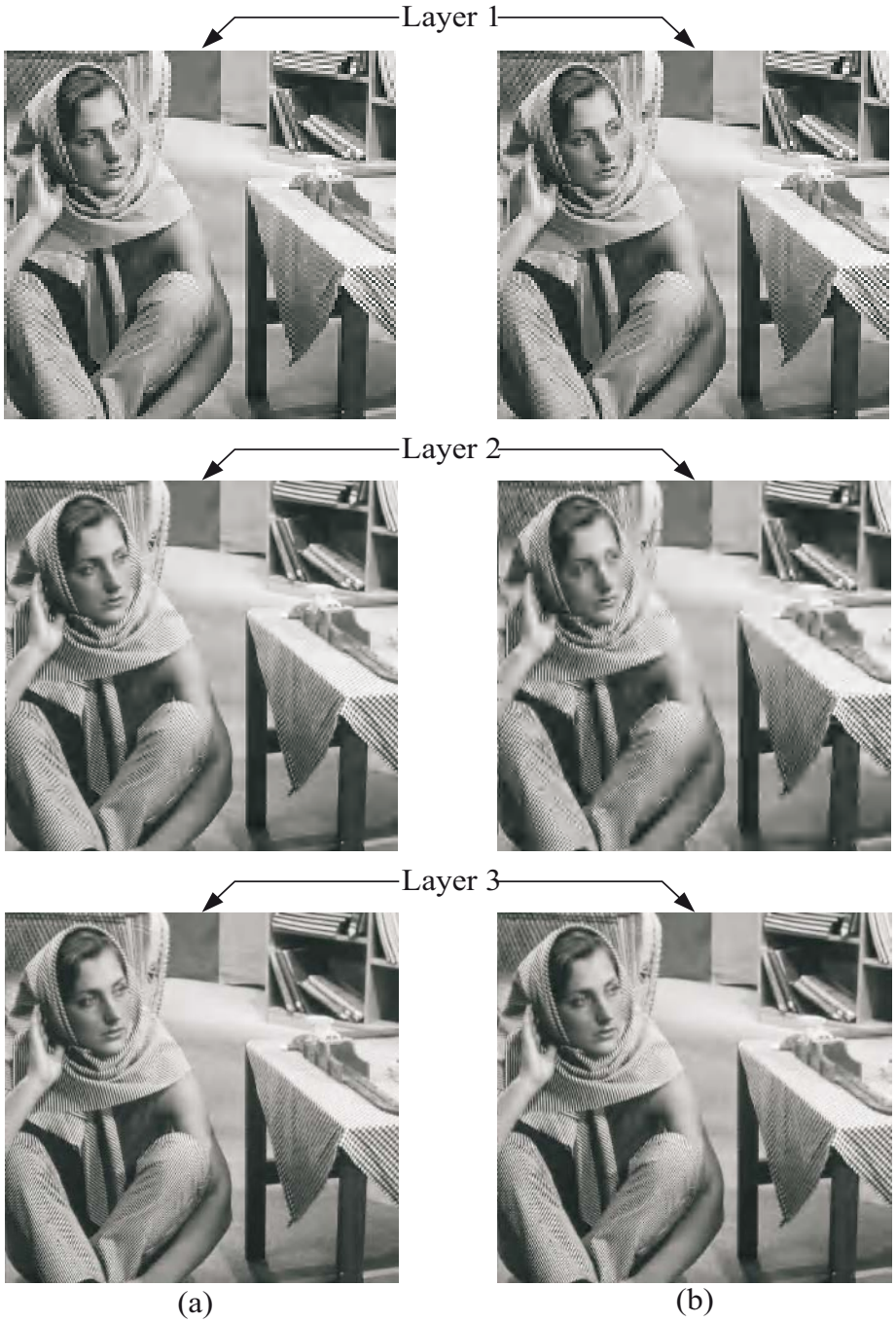


Fig. 6. Reconstructed “Barbara” at each layer i of the LJPEG2000-based LIC system and JPEG2000-based simulcast system subject to the same rate for encoding x_{LI_i} : (a) LJPEG 2000 , (b) JPEG 2000

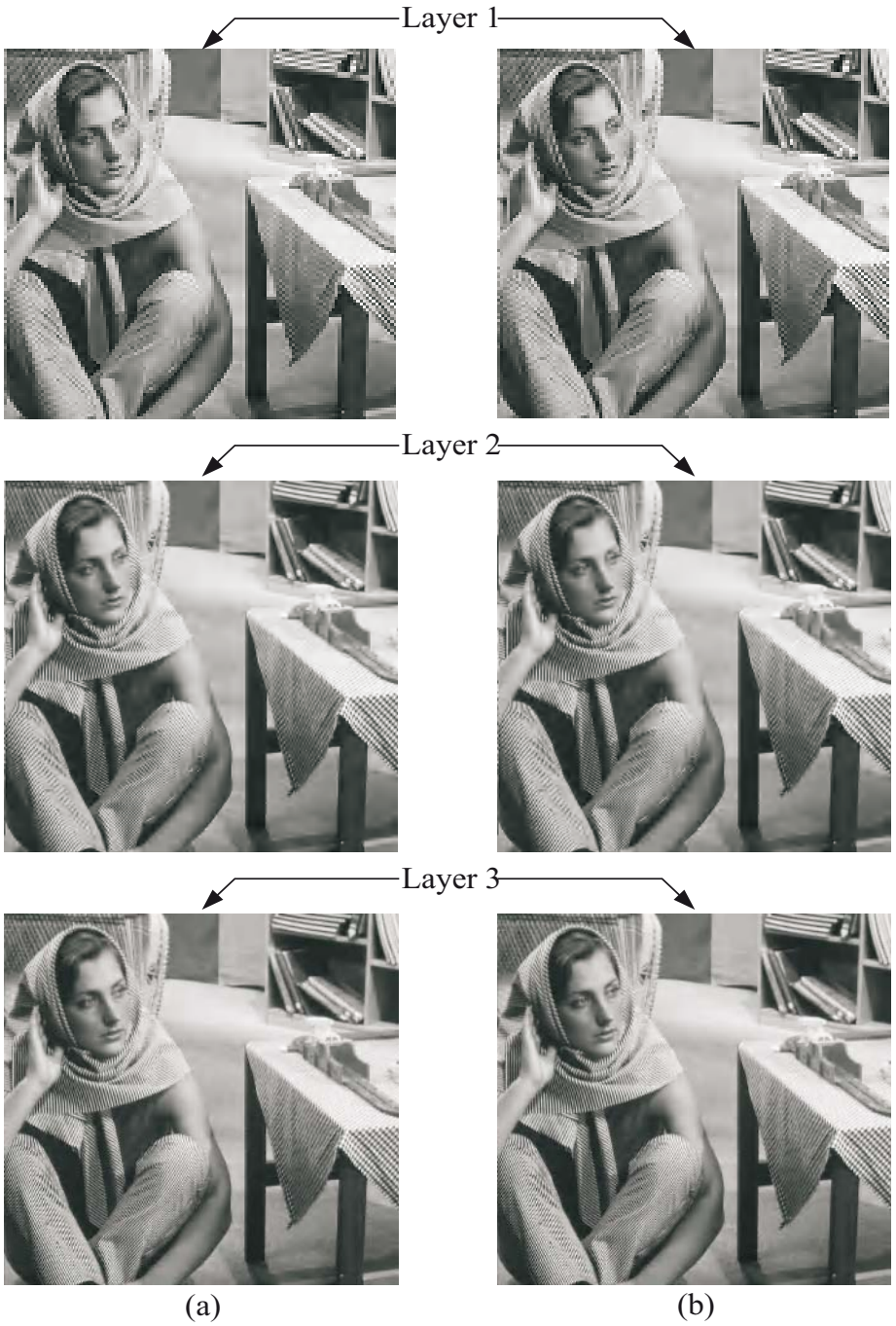


Fig. 7. Reconstructed “Barbara” at each layer i of the LJPEG2000-based LIC system and JPEG2000-based simulcast system subject to the same rate for decoding \mathbf{x}_{LI_i} : (a) LJPEG 2000 , (b) JPEG 2000

much higher rate for encoding. In fact, the rate for encoding *all* the images $\mathbf{x}_{LI_i}, i = 1, \dots, N$, in the system is $\sum_{i=1}^N \mathbf{r}_i$. However, the required rate in the LIC is only \mathbf{r}_N . That is, as compared with JPEG2000-based simulcast system, the LJPEG2000-based LIC system can achieve almost comparable performance with much less rate for image encoding. These facts demonstrate the effectiveness of the LJPEG2000 algorithm.

5 Conclusion

Our experiments have demonstrated that the LJPEG2000-based LIC system outperforms both the LSPiHT-based LIC system and JPEG2000-based simulcast system subject to the same resolution and rate constraints at each layer. In addition, subject to the same rate for decoding the lowpass subband at each layer, the LJPEG2000-based LIC system attains comparable performance to that of the JPEG2000-based simulcast system. In this case, the simulcast system requires significantly higher rates for image encoding/transmission at each layer. The LJPEG2000 algorithm therefore can be an effective alternative for the design of LIC systems where the resolution and rate associated with each layer are desired to be pre-specified.

References

1. Bosveld, F., Lagendijk, R.L., Biemond, J.: Hierarchical Coding. Chap.9 in: Hang, H. M., Woods, J. W. (eds.): Handbook of Visual Communications. Academic Press (1995) 299–340
2. Hwang, W.J., Chine, C.F., Li, K.J.: Scalable Medical Data Compression and Transmission Using Wavelet Transform for Telemedicine Applications. IEEE Trans. Information Technology in Biomedicine, Vol. 7. (2003) 54–63
3. Said, A., Pearlman, W.: A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees. IEEE Trans. Circuits and Systems for Video Technology (1996) 243–250
4. Shapiro, J.M.: Embedded Image Coding Using Zerotrees of Wavelet Coefficients. IEEE Trans. Signal Processing, Vol. 41. (1993) 3445–3462
5. Taubman, D., Marcellin, M.W.: JPEG2000 image compression fundamentals. standards and practice. Kluwer Academic Publishers (2002)
6. Vetterli, M., Kovacevic, J.: Wavelets and Subband Coding. Prentice Hall (1995)

A Permutation-Based Correlation-Preserving Encryption Method for Digital Videos

Daniel Socek¹, Hari Kalva¹, Spyros S. Magliveras²,
Oge Marques¹, Dubravko Čulibrk¹, and Borko Furht¹

¹ Department of Computer Science and Engineering, and
² Department of Mathematical Sciences,
Florida Atlantic University, Boca Raton FL 33431, USA

Abstract. In this work we present and analyze a conceptually novel encryption method for digital videos. This type of encryption has several advantages over currently available video encryption algorithms. We analyze both security and performance aspects of our method, and show that the method is efficient and secure from a cryptographic point of view. Even though the approach is currently feasible only for a selected class of video sequences and video codecs, the method is promising and future investigations might reveal its broader applicability.

1 Introduction

In most digital video security architectures, video encryption plays a key role in ensuring the confidentiality of the video transfer. However, conventional, general-purpose encryption algorithms (such as AES) are not suitable for video encryption in many digital video applications [10,5,3,7], mainly because these algorithms do not conform to various video application requirements that we discuss next. In order to overcome this problem, a significant number of video encryption algorithms specifically designed for digital videos have been proposed (e.g. [8,15,10,4,3]).

In general, applying a well-established, general-purpose, symmetric-key encryption algorithm to a video sequence is a good idea from a security point of view. However, there are applications with requirements that are not supported with conventional encryption methods. Thus, encryption algorithms specifically designed to support these requirements are desirable. These aspects include the following: (1) *level of security and perception*, (2) *format-compliance*, (3) *degree of bitstream expansion*, and (4) *error-tolerance*.

To identify an optimal level of security, we have to carefully compare the cost of the multimedia information to be protected versus the cost of the protection itself. Light-weight encryption, often called *degradation*, may be sufficient for distributing multimedia content of low value. Often, degradation intentionally preserves some perceptual information with visual quality that is unacceptable for entertainment purposes. This type of encryption is referred to as *perceptual encryption*. If the video contains sensitive industrial, governmental or military

information, then the cryptographic strength must be substantial and no perceptual information should be preserved after encryption.

In many applications it is desired that the encryption algorithm preserves the video compression format. In other words, after encrypting the encoded video, ordinary decoders should still be able to decode it without crashing. This is an important feature in digital video broadcasting applications where an encrypted video is broadcast to all system users. This property of an encryption algorithm is often called *format-compliance* (also called *transparency*, *transcodability* or *syntax-awareness*). When feeding the decoder with the format-compliant encrypted data the produced output appears either perceivable but distorted or non-perceivable and random, depending on the type of encryption (see Fig. 1).

In many applications, it is required that the encryption transformation preserves the size of a bitstream. This is known as the *constant bitrate* requirement. However, more often than not, it is simply preferred that the output produced by an encryption-equipped encoder and the output produced by an ordinary encoder have similar sizes (a near-constant bitrate). A near-constant bitrate is likely to occur when a block cipher is used for encryption, since in that case the encrypted output is always a constant multiple of the blocksize.

Finally, for many multimedia systems *error-tolerance*, or *error-resilience*, is usually of high importance. Advanced video coding systems (e.g. H.264) have their own error correcting mechanisms. Hence, a video encryption algorithm that preserves these mechanisms is favorable for video systems with noisy channels.

For the most part, modern cryptography is designed for a generic bitstream, and as such, it disregards the aforementioned properties of a digital video and the requirements of a typical digital video application. In the next section, we present an overview of the video encryption algorithms proposed in the past mainly to overcome some of these application-related problems.

The rest of this paper is organized as follows. Section 2 provides a brief overview of existing video encryption algorithms, and Section 3 introduces our proposed method. In Section 4 we present a security analysis of our method, while in Section 5 we provide its performance analysis with some experimental results. Finally, Section 6 holds our conclusions and suggestions for further research.

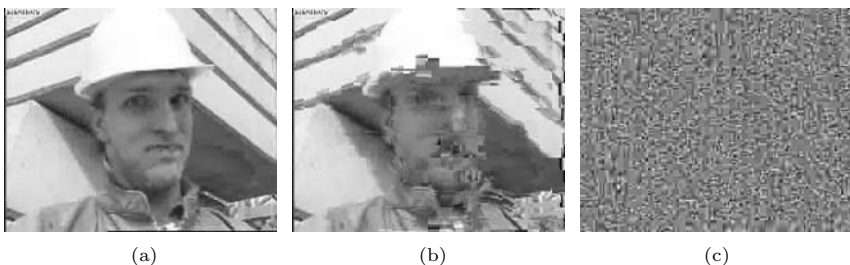


Fig. 1. Decoded video produced by an ordinary decoder for (a) video without encryption, (b) video encrypted with format-compliant perceptual encryption, and (c) video encrypted with high-security format-compliant encryption

2 Overview of Video Encryption Algorithms

In general, there are two basic research methodologies regarding the encryption of digital videos: *selective encryption* approaches and *full encryption* approaches. While full encryption approaches are designed to encrypt the entire video bitstream (raw or compressed), selective encryption approaches perform encryption only on certain, carefully selected parts of the video bitstream. Most of the proposed methods belong to the category of selective encryption approaches since, in many instances, encrypting the entire bitstream introduces an expensive performance overhead.

The idea of selective video encryption was introduced independently by Meyer and Gadget (SECMPEG) [8], and by Maples and Spanos (Aegis) [15]. Since then, a significant number of selective encryption proposals appeared in the scientific literature. A good reference for selective image and video encryption methods along with some security analysis is given in [3,7].

As Fig. 2a depicts, a number of selective video encryption algorithms require a modification of the codec, which is a drawback for systems with pre-installed hardware codecs. Also, many selective approaches apply encryption after compression in which case many application-related requirements are lost (Fig. 2b). Another common problem with selective encryption approaches is the lack of standard methods for proper security evaluation. Many selective encryption approaches were broken shortly after they were proposed because scientists found ways to exploit the information from the remaining, unencrypted bits. For example, Agi and Gong [1] showed weaknesses in SECMPEG [8] and Aegis [15] a year later. Also, a year after Tang proposed an image/video encryption method [16] based on permuting the zig-zag reordering after a DCT transformation, Qiao and Nahrstedt [11] discovered serious weaknesses against attacks derived from statistical analysis of the unencrypted DCT coefficients. Seidel et al. [12] show some weaknesses in the video encryption algorithms by Shi et al. soon after the original proposal [13]. The true correlation between encrypted and unencrypted bits is often hard to properly estimate in terms of required security levels. Hence, the selective encryption approaches are much easier to design and evaluate against the performance aspects, than to properly evaluate in terms of security.

Full encryption approaches essentially encrypt the entire multimedia data. The full encryption approach is not to be confused with the so-called *naïve approach*, which is itself in the category of full encryption. By a naïve approach one understands a video encryption method that uses a conventional, general-purpose modern cryptosystem to encrypt the entire video bitstream. By a “conventional cryptosystem” we mean a modern symmetric-key cryptosystem that is either one of the encryption standards, or a well-established general-purpose cryptosystem that was designed and evaluated by reputable cryptographic experts, companies, or agencies. Full approaches that are not considered naïve are mostly systems that are specifically designed to encrypt a specific multimedia type of data. These approaches, which we refer to as *alternative full approaches*, take into consideration the performance and security aspects needed for an effective multimedia encryption.

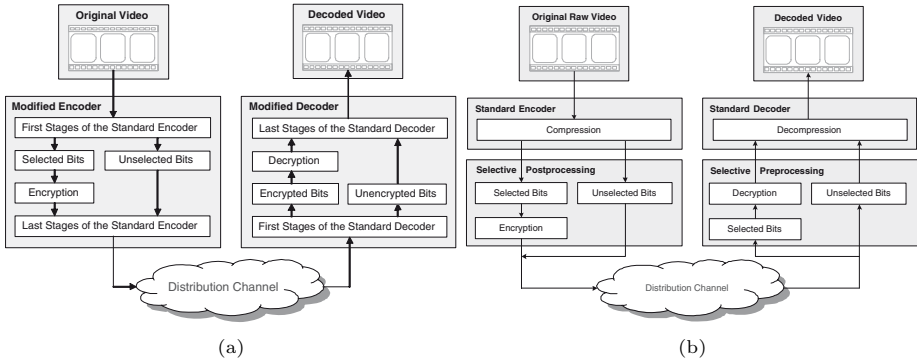


Fig. 2. The usual architectures for selective video encryption: (a) both encoder and decoder must be modified since the encryption/decryption is performed during the compression/decompression stages, and (b) selected parts of the compressed video bitstream are encrypted where in most cases the resulting bitstream is not application-friendly

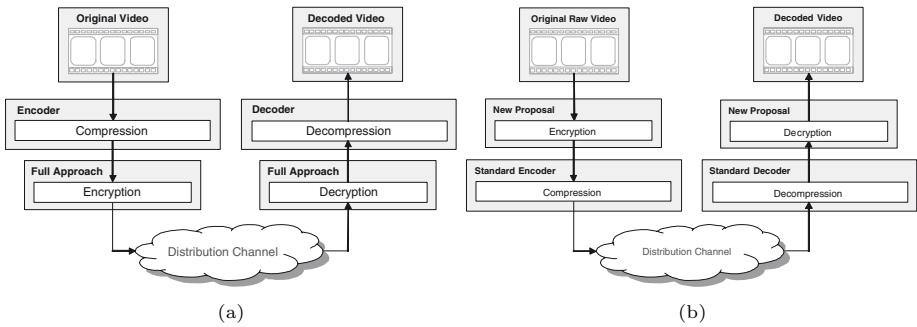


Fig. 3. The architecture for: (a) full (and naïve) video encryption where the entire compressed video bitstream is encrypted using a standard cryptosystem (e.g. AES) or an alternative fast cryptosystem (e.g. chaos-based), and (b) the proposed algorithm where the encryption and decryption are performed outside the current video system which results in no modification to the codec and fully compliant, application-friendly video outputs

A notable path that the research community took for the full alternative video encryption approaches was the use of fast chaotic maps to achieve encryption. Chaos-based methods are apparently promising due to their fast performance. Although many chaotic encryption approaches were shown to be insecure, there are chaotic encryption algorithms that, up to date, remain unbroken (e.g. [4]). An excellent overview of these approaches, along with their comparative security analysis is presented in [5] and [3]. There are a few recently proposed fast, hardware-friendly, full encryption methods that are based on a class of neural networks [2]. However, these methods were later shown to be less secure than originally anticipated [14]. Yi *et al.* proposed a new fast encryption algorithm

for multimedia (FEA-M), which bases the security on the complexity of solving nonlinear Boolean equations [17]. The scheme was shown insecure against several different attacks [18,9].

In addition to questionable security, the full alternative encryption approaches are not application-friendly when applied after compression, and application requirements such as format-compliance or error-resilience are not supported. Fig. 3a shows the typical structural design of full encryption approaches, where encryption is performed after compression and where no modification to the codec is needed.

3 Our Model

Strong encryption is a process that produces randomized data. On the other hand, compression efficiency is directly dependent on the presence of source data redundancy. The more the data is correlated, the better the compression, and vice versa. One may ask the following important question: Is it possible to design an encryption mechanism of reasonable security that preserves, or perhaps even increases the compressibility of data? To our knowledge, no such system has ever been proposed in the published scientific literature. Such system is here referred to as the *Correlation-Preserving Encryption* (CPE). CPE for digital videos encoded with spatial-only coding is indeed possible to achieve with permutation-based transformations. In this section we present the details of our CPE approach. The architecture of this approach is depicted in Fig. 3b.

Let \mathcal{V} be a video sequence consisting of m frames denoted by I_1, I_2, \dots, I_m . For our model we assume that all frames in \mathcal{V} are part of a single scene with a relatively low movement, captured with a static camera, so that the differences between adjacent frames are relatively small. Furthermore, we assume that each frame has a dimension of $w \times h$ and up to 2^n different pixel values (colors), such that $w \times h$ is much larger than 2^n . By the pigeonhole principle this condition favors higher frequencies of the pixel values within a frame. Finally, let σ_i denote a canonical sorting permutation of I_i , and $\sigma_i(I_i)$ the image with sorted pixels from I_i . For a given frame I there are a large number of sorting permutations for I . By a *canonical* sorting permutation of I we mean a *unique* sorting permutation σ that any two distant parties can compute solely by knowing I , which is the case when the parties utilize the same computational method.

We describe two versions of our algorithm. The first one is designed for lossless spatial-only codecs, such as Animated GIF (A-GIF), Motion PNG (M-PNG), or Motion Lossless JPEG (M-JLS), while the second one is targeted for lossy spatial-only codecs, such as Motion JPEG (M-JPEG).

Suppose Alice wishes to securely transmit a video sequence $\mathcal{V} = I_1, I_2, \dots, I_m$ to Bob. We assume that if Alice would like to transmit \mathcal{V} to Bob non-securely, she would normally use video compression algorithm C (the encoder) and decompression algorithm D (the decoder). She opens two channels with Bob, the regular, non-secure multimedia distribution channel R , and a second, secure channel S where transmission data is encrypted using some standard method (e.g. AES-based protocol).

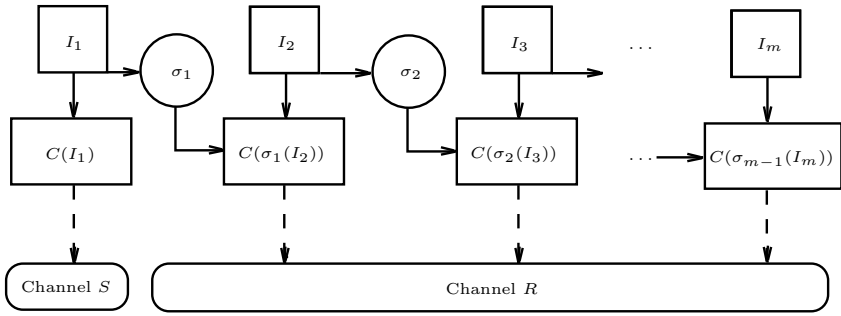


Fig. 4. Diagram of our encryption algorithm for lossless spatial-only video codecs

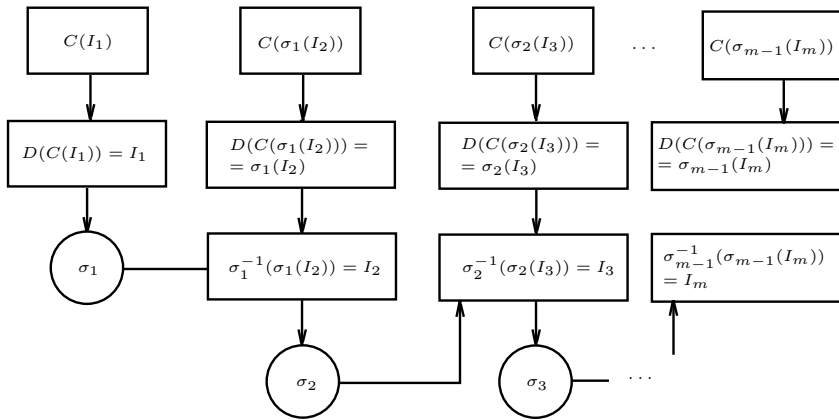


Fig. 5. Diagram of our decryption algorithm for lossless spatial-only video codecs

The Lossless Case. The following is the proposed algorithm for lossless codecs:

1. Given a video sequence $\mathcal{V} = I_1, \dots, I_m$, Alice computes σ_1 .
2. Alice calculates $C(I_1)$ and transmits it through channel S . This is the secret part (the key) of the algorithm.
3. For each subsequent frame $I_i, i = 2, \dots, m$, Alice does the following:
 - (a) She computes the frame $\sigma_{i-1}(I_i)$ and the permutation σ_i ;
 - (b) Alice then applies the standard encoder to the frame $\sigma_{i-1}(I_i)$ and transmits the encoded frame $C(\sigma_{i-1}(I_i))$ to Bob via the regular, non-secure multimedia channel R .

At the other end, Bob performs the following decryption algorithm in order to recover the original video sequence \mathcal{V} :

1. Bob decodes $C(I_1)$ into I_1 and obtains a canonical sorting permutation σ_1 .
2. For each received frame $C(\sigma_{i-1}(I_i)), i = 2, \dots, m$, Bob does the following:

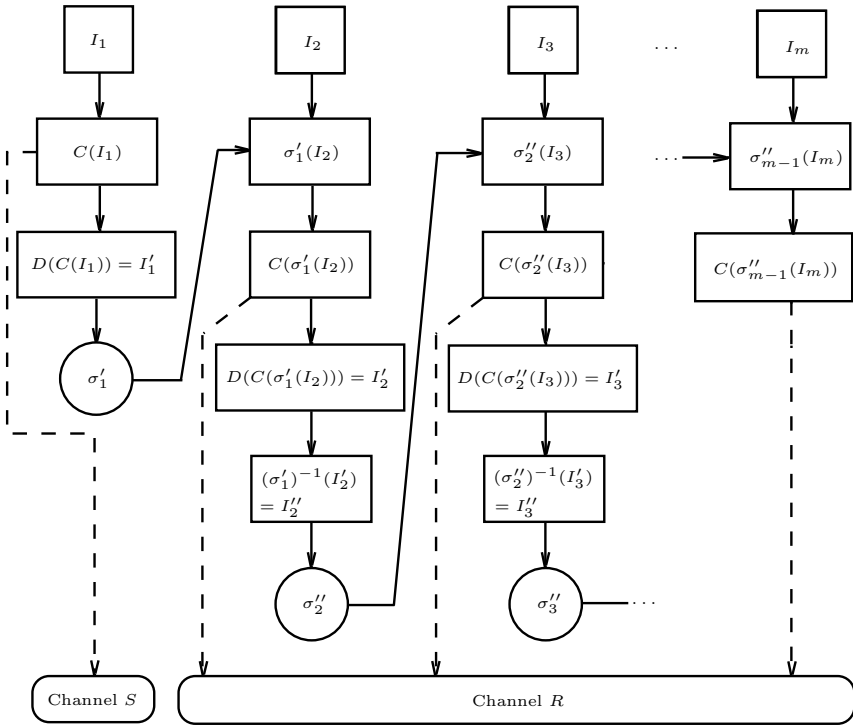


Fig. 6. Diagram of our encryption algorithm for lossy spatial-only video codecs

- (a) Decodes $C(\sigma_{i-1}(I_i))$ into $\sigma_{i-1}(I_i)$ and calculates $I_i = \sigma_{i-1}^{-1}(\sigma_{i-1}(I_i))$ where σ_{i-1}^{-1} is the inverse permutation of σ_{i-1} ;
- (b) Calculates the canonical sorting permutation σ_i of I_i .

The Lossy Case. The proposed algorithm for lossy codecs is as follows:

1. Given a video sequence $\mathcal{V} = I_1, \dots, I_m$, Alice first computes $C(I_1)$ and then $I'_1 = D(C(I_1))$ from which she obtains the canonical sorting permutation σ'_1 .
2. Alice sends $C(I_1)$ via secure channel S to Bob.
3. She applies σ'_1 to I_2 , computes $C(\sigma'_1(I_2))$, and sends it via R to Bob.
4. Next, she computes $I'_2 = D(C(\sigma'_1(I_2)))$ and then $I''_2 = (\sigma'_1)^{-1}(I'_2)$ from which she calculates the canonical sorting permutation σ''_2 .
5. For each subsequent frame $I_i, i = 3, \dots, m$, Alice does the following:
 - (a) Applies σ''_{i-1} to I_i , computes $C(\sigma''_{i-1}(I_i))$, and sends it to Bob via R ;
 - (b) Computes $I'_i = D(C(\sigma''_{i-1}(I_i)))$;
 - (c) Applies $(\sigma''_{i-1})^{-1}$ to get $I''_i = (\sigma''_{i-1})^{-1}(I'_i)$;
 - (d) Calculates the canonical sorting permutation σ''_i .

At the receiver’s side, Bob performs the following decryption algorithm to recover the approximation of the original video sequence \mathcal{V} :

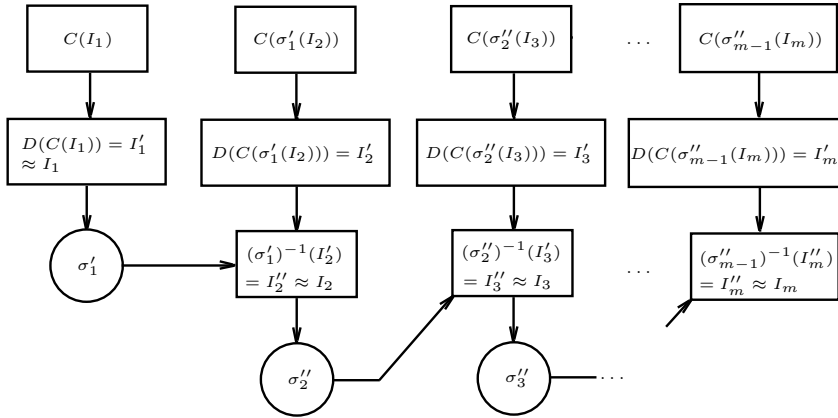


Fig. 7. Diagram of our decryption algorithm for lossy spatial-only video codecs

1. Bob calculates $D(C(I_1)) = I'_1 \approx I_1$ calculates sorting permutation σ'_1 .
2. From $C(\sigma'_1(I_2))$ he computes $I'_2 = D(C(\sigma'_1(I_2)))$.
3. Bob approximates $I_2 \approx I''_2 = (\sigma'_1)^{-1}(I'_2)$.
4. He then recovers the canonical sorting permutation σ''_2 of I''_2 .
5. For each received frame $C(\sigma''_{i-1}(I_i))$, $i = 3, \dots, m$, Bob does the following:
 - (a) Decodes $C(\sigma''_{i-1}(I_i))$ into $I'_i = D(C(\sigma''_{i-1}(I_i)))$;
 - (b) Approximates $I_i \approx I''_i = (\sigma''_{i-1})^{-1}(I'_i)$;
 - (c) If $i < m$ he calculates a sorting permutation σ''_i of I''_i .

4 Security Analysis of Our Model

Key Space. In order to grasp the true complexity of an exhaustive search of the key space related to our method, one should answer the following question: Given a color histogram of an $w \times h$ image I , how many different images can be formed out of the histogram color values? Note that I is just one of these images. We use some known group theoretic properties to provide the answer to this question.

If T is a set, $|T|$ denotes the size of T . Let S_n denote the symmetric group of all permutations on $X = \{1, 2, \dots, n\}$. Then, any subgroup G of S_n , including $G = S_n$, acts on X by permuting the elements of X .

If $x \in X$, and G a subgroup of S_n , then the G -orbit of x , is the subset of X defined by:

$$G(x) = \{\pi(x) : \pi \in G\} \subset X.$$

Moreover, the stabilizer in G of x , denoted by G_x , is the subgroup of all elements of G fixing x . That is, G_x is defined by:

$$G_x = \{\pi \in G : \pi(x) = x\} \leq G.$$

A proposition of elementary group theory states that $|G(x)| = |G|/|G_x|$, a useful fact as we shall see below. In the spirit of our earlier notation, $\sigma_I \in S_{w \times h}$. Furthermore, if \mathbb{Z}_n^m denotes the m -dimensional set of integers modulo n , then $I \in \mathbb{Z}_n^{w \times h}$. We note that if I is a $w \times h$ image with pixel values from \mathbb{Z}_n , then the number of different images that can be formed by permuting the pixels in I is exactly the size of the $S_{w \times h}$ -orbit of I , under the group action of $S_{w \times h}$ on $\mathbb{Z}_n^{w \times h}$. This will help us answer our previously posted question. In our case, $|G| = |S_{w \times h}| = (w \times h)!$. Therefore, if we can determine $|G_I|$, then we would know $|G(I)|$. Let $U(I) \subset \mathbb{Z}_n$ denote the set of unique pixel values that appear in I . Furthermore, if $u \in U(I)$, let $N(u)$ denote the number of times pixel value u appears in I . Then, the following theorem holds:

Theorem 1. *If $G = S_{w \times h}$ acts on $\mathbb{Z}_n^{w \times h}$, $I \in \mathbb{Z}_n^{w \times h}$, and $U(I) = \{u_1, \dots, u_k\}$, then the size of the stabilizer of G_I is*

$$|G_I| = \prod_{i=1}^k N(u_i)!$$

Proof. Let $u \in U(I)$. Then, there are $N(u)$ pixels in image I that have the value of u . Thus, there are $N(u)!$ ways of permuting these values among themselves without changing I . Doing this for all elements of $U(I)$ results in a total number of $N(u_1)! \times \dots \times N(u_k)!$ ways of permuting the pixels of I without changing I . This is exactly the size of the stabilizer of I .

Using the result of Theorem 1 we have that the number of different images that can be formed out of pixels from I , where $U(I) = \{u_1, \dots, u_k\}$, is

$$|S_{w \times h}(I)| = \frac{(w \times h)!}{\prod_{i=1}^k N(u_i)!}$$

Thus, there are exactly $(w \times h)! / \prod_{i=1}^k N(u_i)!$ different images, with exactly the same frequency distribution of $\{N(u_i) : i = 1, \dots, k\}$. These distinct images form the effective key space of our method. Therefore, the size of the key space depends on the color histogram of the encrypted frame. As one can see, this number is extremely large when considering natural images.

Known/Chosen-Plaintext and Chosen-Ciphertext Attacks. Permutation-only video encryption is considered weak against known/chosen-plaintext attack, and a chosen-ciphertext attack [6]. However, all of the previously proposed methods rely on generating the secret permutation using a secret key. Under this scenario, all of the aforementioned attacks are trying to recover the secret key (or a part of it) that was used for the current or future encryptions. Our scheme does not rely on such a principle, and there is no secret key upon which a permutation is generated. Our method relies on the sorting permutation of the *previous* frame, and thus, a key is directly dependant of the plaintext. Under a chosen-plaintext attack, the adversary can compute the sorting permutation for the chosen frame, but this gives no information about the sorting permutations

for the unknown frames. Under a chosen-ciphertext attack, the adversary can recover the unsorting permutation for the chosen encrypted frame, but this gives no information regarding other unknown ciphertexts. In fact, all frames with the same histogram encrypt into the same ciphertext, so when the adversary encounters the same ciphertext again, the previous unsorting permutation may give a completely inaccurate plaintext. A limited known-plaintext attack is applicable to our method, because the adversary can recover all frames that follow the known frame until the scene changes and key frame is updated. However, in our system it is not possible to re-use the secret key for future scenes or sequences.

Known Weaknesses. If one frame of a video scene is known, all frames that follow within that scene are computable by the adversary. In addition, if the adversary has the information on the possible videos to be encrypted, he or she may be able to recognize which video sequence is being transmitted from Alice to Bob by observing the publicly given pixel value histograms of frames. These two attacks are unavoidable in the proposed scheme, and the scheme should not be used when conditions are such that these attacks are possible by an adversary.

5 Performance Analysis and Experimental Results

To evaluate the performance of our method in terms of compression, we run experiments on several fairly static greyscale sequences in CIF/QCIF formats. As

Table 1. Performance of our method for lossless spatial-only codecs

| Sequence | Codec | Codec w/o Encryption | | Codec w/ Encryption | |
|---|-------|----------------------|--------------|---------------------|--------------|
| | | Size[MB] | Avg.Frm.[KB] | Size[MB] | Avg.Frm.[KB] |
| Akiyo CIF, 300 frms | A-GIF | 21.53 | 73.51 | 9.24 | 31.54 |
| | M-PNG | 19.09 | 65.15 | 7.96 | 27.17 |
| | M-JLS | 10.70 | 36.53 | 7.39 | 25.23 |
| Mother and Daughter CIF, 300 frms | A-GIF | 21.68 | 73.99 | 15.93 | 54.38 |
| | M-PNG | 19.23 | 65.64 | 14.97 | 51.11 |
| | M-JLS | 11.31 | 38.62 | 12.63 | 43.12 |
| Monitor Hall CIF, 300 frms | A-GIF | 24.88 | 84.91 | 18.50 | 63.14 |
| | M-PNG | 21.67 | 73.97 | 17.55 | 59.91 |
| | M-JLS | 13.13 | 44.83 | 14.62 | 49.89 |
| Grandma QCIF, 100 frms | A-GIF | 2.14 | 21.91 | 1.45 | 14.84 |
| | M-PNG | 1.90 | 19.46 | 1.30 | 13.34 |
| | M-JLS | 1.18 | 12.13 | 0.89 | 9.09 |
| Claire QCIF, 100 frms | A-GIF | 1.52 | 15.59 | 1.17 | 11.98 |
| | M-PNG | 1.35 | 13.86 | 1.02 | 10.48 |
| | M-JLS | 0.75 | 7.68 | 0.71 | 7.27 |
| Miss America QCIF, 100 frms | A-GIF | 1.54 | 15.80 | 1.33 | 13.57 |
| | M-PNG | 1.44 | 14.79 | 1.27 | 13.05 |
| | M-JLS | 0.81 | 8.34 | 0.91 | 9.37 |

Tables 1 and 2 show, our method preserves, and in many instances even improves the compression performance of the original codec without encryption. Table 2 also compares the loss of quality (in terms of PSNR) when our method is applied to the lossy M-JPEG with quality parameter Q that controls the quantization level in M-JPEG. Q ranges from 0 (the worst quality) to 100 (the best quality).

Table 2. Performance of our method for lossy spatial-only codecs

| Sequence | Q | M-JPEG w/o Encryption | | | M-JPEG w/ Encryption | | |
|---|----|-----------------------|--------------|-------|----------------------|--------------|-------|
| | | Size[MB] | Avg.Frm.[KB] | PSNR | Size[MB] | Avg.Frm.[KB] | PSNR |
| Akiyo CIF, 300 frms | 90 | 3.96 | 15.93 | 45.29 | 3.71 | 12.72 | 41.63 |
| | 70 | 2.05 | 8.24 | 40.39 | 1.92 | 6.59 | 36.13 |
| | 50 | 1.55 | 6.24 | 38.20 | 1.37 | 4.70 | 33.93 |
| Mother and Daughter CIF, 300 frms | 90 | 4.21 | 16.96 | 45.25 | 4.98 | 17.07 | 39.97 |
| | 70 | 2.24 | 9.04 | 40.91 | 2.48 | 8.48 | 34.68 |
| | 50 | 1.70 | 6.84 | 38.88 | 1.72 | 5.91 | 32.70 |
| Monitor Hall CIF, 300 frms | 90 | 5.55 | 22.35 | 43.21 | 6.05 | 20.71 | 38.89 |
| | 70 | 3.02 | 12.15 | 38.12 | 2.93 | 10.03 | 33.49 |
| | 50 | 2.25 | 9.05 | 35.95 | 2.01 | 6.89 | 31.40 |
| Grandma QCIF, 100 frms | 90 | 0.57 | 5.94 | 41.04 | 0.35 | 3.58 | 41.55 |
| | 70 | 0.32 | 3.28 | 36.47 | 0.19 | 1.93 | 36.35 |
| | 50 | 0.24 | 2.48 | 34.81 | 0.14 | 1.44 | 34.06 |
| Claire QCIF, 100 frms | 90 | 0.41 | 4.22 | 45.19 | 0.31 | 3.19 | 42.23 |
| | 70 | 0.25 | 2.62 | 39.86 | 0.17 | 1.80 | 36.60 |
| | 50 | 0.20 | 2.08 | 37.51 | 0.13 | 1.36 | 34.36 |
| Miss America QCIF, 100 frms | 90 | 0.35 | 3.64 | 45.63 | 0.36 | 3.69 | 41.55 |
| | 70 | 0.19 | 1.98 | 41.52 | 0.19 | 1.98 | 35.97 |
| | 50 | 0.15 | 1.56 | 39.71 | 0.14 | 1.46 | 33.83 |

A modest loss of quality occurs by performing the proposed encryption with lossy codecs.

Finally, we note that the computational complexity of the proposed method is very low at the decoder side for both lossless and lossy codecs, since the only additional computation that has to be performed involves the calculation of a sorting permutation. A standard sorting algorithm with complexity of $\mathcal{O}(N \log N)$ can be used to calculate a sorting permutation of the given frame. Inverting and/or applying a permutation is equivalent to a table lookup.

6 Conclusions and Future Research

In this work we proposed a novel video encryption algorithm designed for both lossless and lossy low-motion spatial-only video codecs. The algorithm preserves (or even improves) the spatial correlation of the source data, and can thus be performed before compression at the encoder side, and after decompression at the decoder side, a unique and desirable feature. In effect, the algorithm produces fully application-friendly output, and requires no modification to the codec modules. We have presented both security and performance analysis of our method, and have shown that the algorithm is computationally efficient and resistant to typical cryptanalytic attacks.

Future directions should include an investigation of extending this principle to achieve efficiency in exploiting temporal correlation as well, in order to achieve applicability to more advanced video codecs such as H.26x and MPEG-x.

References

1. I. Agi and L. Gong, "An Empirical Study of Secure MPEG Video Transmission", in Proc. IEEE Symposium on Network and Distributed System Security (SNDSS '96), San Diego, CA, February 22-23, IEEE Computer Society, pp. 137-144, 1996.

2. C.-Ku. Chan, C.-Kw. Chan, L.-P. Lee, and L.M. Cheng, "Encryption System Based On Neural Network", Communications and Multimedia Security Issues of the New Century, Kluwer Academic Publishers, Boston, MA, pp. 117–122, 2001.
3. B. Furht, E.A. Muharemagic, and D. Socek, "Multimedia Security: Encryption and Watermarking", vol. 28 of Multimedia Systems and Applications, Springer, 2005.
4. S. Li, X. Zheng, X. Mou, and Y. Cai, "Chaotic Encryption Scheme for Real-Time Digital Video", In Real-Time Imaging VI, Proc. SPIE, vol. 4666, pp. 149-160, 2002.
5. S. Li, G. Chen, and X. Zheng, "Multimedia Security Handbook", eds. B. Furht and D. Kirovski, vol. 4 of Internet and Communications Series, ch. 4 "Chaos-Based Encryption for Digital Images and Videos", CRC Press, 2004.
6. S. Li, C. Li, G. Chen, and N.G. Bourbakis, "A General Cryptanalysis of Permutation-Only Multimedia Encryption Algorithms", IACR's Cryptology ePrint Archive: Report 2004/374, 2004.
7. X. Liu and A.M. Eskicioglu, "Selective Encryption of Multimedia Content in Distribution Networks: Challenges and New Directions", IASTED International Conference on Communications, Internet and Information Technology (CIIT 2003), Scottsdale, AZ, November 17-19, 2003.
8. J. Meyer and F. Gadegast, "Security Mechanisms for Multimedia Data with the Example MPEG-1 Video", Project Description of SEC MPEG, Technical University of Berlin, Germany, May 1995, <http://www.gadegast.de/frank/doc/secmeng.pdf>.
9. M.J. Mihaljević and R. Kohno, "Cryptanalysis of Fast Encryption Algorithm for Multimedia FEA-M", IEEE Trans. Comm. Letters, vol. 6, no. 9, pp. 382-384, 2002.
10. L. Qiao and K. Nahrstedt, "A New Algorithm for MPEG Video Encryption", in Proc. First International Conference on Imaging Science, Systems and Technology (CISST '97), Las Vegas, NV, June 30-July 3, pp. 21–29, 1997.
11. L. Qiao, K. Nahrstedt, and M.-C. Tam, "Is MPEG Encryption by Using Random List Instead of Zigzag Order Secure?", in Proc. IEEE Intl. Symposium on Consumer Electronics, Singapore, Dec. 2-4, IEEE Comp. Soc., pp. 226–229, 1997.
12. T.E. Seidel, D. Socek, and M. Sramka, "Cryptanalysis of Video Encryption Algorithms", Tatra Mountains Mathematical Publications, vol 29, pp. 1–9, 2004.
13. Bhargava, B., Shi, C., Wang, S.-Y.: MPEG Video Encryption Algorithms. Multimedia Tools and Applications, Kluwer Academic Publishers, 24(1):57-79, 2004.
14. D. Socek and D. Culibrk, "On the Security of a Clipped Hopfield Neural Network Cryptosystem", in Proc. of the ACM Multimedia and Security Workshop (ACM-MM-Sec'05), New York City, NY, August 1-2, pp. 71–75, 2005.
15. G.A. Spanos and T.B. Maples, "Security for Real-Time MPEG Compressed Video in Distributed Multimedia Applications", in Proc. IEEE 15th Intl. Phoenix Conf. on Computers and Communications, Scottsdale, AZ, March 27-29, pp. 72–78, 1996.
16. L. Tang, "Methods for Encrypting and Decrypting MPEG Video Data Efficiently", in Proc. Fourth ACM International Multimedia Conference, Boston, MA, November 18-22, pp. 219–230, 1996.
17. X. Yi, C.H. Tan, C.K. Siew, M.R. Syed, "Fast Encryption for Multimedia", IEEE Trans. Consumer Eletronics 47 (1), pp. 101-107, 2001.
18. A.M. Youssef and S.E. Tavares, "Comments on the Security of Fast Encryption Algorithm for Multimedia (FEA-M)", IEEE Trans. Consumer Eletronics 49 (1), pp. 168-170, 2003.

A Fast Scheme for Converting DCT Coefficients to H.264/AVC Integer Transform Coefficients

Gao Chen, Shouxun Lin, and Yongdong Zhang

Institute of Computing Technology, Chinese Academy of Sciences
Beijing, 100080, P.R. China
{chengao, sxlin, zhyd}@ict.ac.cn

Abstract. Converting MPEG-2 8-tap discrete cosine transform (DCT) coefficients to H.264/AVC 4-tap integer transform (IT) coefficients is one of the indispensable operations for MPEG-2 to H.264 transform domain transcoding. In this paper, we propose a novel transform kernel matrix based on the factorization of DCT to perform this operation directly in transform domain. Furthermore, additional reduction in computation can be obtained by taking advantage of the fact that most of the 8×8 DCT blocks in real video sequences only have nonzero coefficients in the upper left 4×4 quadrant. Relative to Jun Xin's method, the proposed method saves 16 operations in the worst case and 440 operations in the best case for each 8×8 DCT block. Experimental results also show that the proposed method efficiently reduces the computational cost up to 42.9% with negligible degradation in video quality. Hence, the proposed method can be efficiently used in the real-time and high capacity MPEG-2 to H.264 transform domain transcoding.

1 Introduction

The newest video-coding standard, known as H.264/AVC [1], jointly developed by the Joint Video Team of ISO/IEC MPEG and ITU-T VCEG, can offer perceptually equivalent quality video at about 1/3 to 1/2 of the bit-rates offered by the MPEG-2 format [2]. Due to its superior compression efficiency, it is expected to replace MPEG-2 over the next several years, but the complete migration to H.264 will take several years given the fact that MPEG-2 has been widely used in many applications today, including DVD and digital TV. With the deployment of H.264, e.g., for mobile devices, there is a need to develop transcoding technologies to convert videos in the MPEG-2 format into the videos in the H.264 format and vice versa [3]. Furthermore, there is a clear industry interest in technologies facilitating the migration from MPEG-2 to H.264 [4].

In the implementing of MPEG-2 to H.264 transcoder, reducing the computational complexity is a major issue, especially for real-time implementing [3]. It is well known that transform domain transcoding techniques may be simpler than the conventional pixel domain transcoding techniques, since the former avoid the complete decoding and re-encoding which are computationally expensive [3]. For this reason, there has been a great effort in recent time to develop fast algorithms that conduct MPEG-2 to H.264 transcoding in transform domain. Unlike previous transform

domain transcoding, such as H.263 to MPEG-2 transcoding, the decoded DCT coefficients in the MPEG-2 to H.264 transcoder can not be reused directly and have to be converted to IT coefficients. This is because that MPEG-2 uses 8-tap DCT to produce the transform coefficients, while H.264 uses a 4-tap IT to do so. So, one of the indispensable steps in the transform domain MPEG-2 to H.264 transcoding is to convert MPEG-2 DCT coefficients to IT coefficients i.e., DCT-to-IT coefficients conversion.

Based on the fact that the DCT transform matrix is orthonormal and separable, Jun Xin [5] and Bo Shen [6] have derived two different transform kernel matrices and have shown their transform domain conversion methods outperform the pixel domain approach. Although Jun Xin's method needs less 64 shift operations and only two calculating stages compared with Bo Shen's method, there are 19 non-trivial elements in Jun Xin's transform kernel matrix, which cause one matrix multiplication (8×8 with 8×8) to have a total of 352 operations. The computational complexity of Jun Xin's method can be reduced through the factorization of the transform kernel matrix. Furthermore, both of the two existing methods do not utilize the property that a large percentage of 8×8 DCT blocks in real video sequences, we refer as low pass block thereafter, only have nonzero DCT coefficients in the upper left 4×4 quadrant.

The role of DCT-to-IT coefficients conversion in MPEG-2 to H.264 transcoder is equal to the transform, such as DCT, in one of video encoder. There are many fast DCT algorithms have been proposed to implement the video encoder efficiently. In order to implement the MPEG-2 to H.264 transcoder efficiently in the transform domain, we should try our best to speed up the operation of DCT-to-IT conversion. In this work, we propose a fast scheme to speed up this operation. The rest of the paper is organized as follows. Section 2 describes our proposed fast scheme for DCT-to-IT coefficients conversion. A performance evaluation of the proposed algorithm in terms of computational complexity and distortion result is carried out in Section 3. Finally, Section 4 concludes this paper.

2 Fast Scheme for DCT-to-IT Coefficients Conversion

Since the DCT-to-IT coefficients conversion can be represented as multiplication by a fixed matrix, fast multiplication by this fixed matrix is possible if it can be factorized into a product of sparse matrices whose entries are mostly 0, 1 and -1 . We will demonstrate that this can be done efficiently by the factorizing of Jun Xin's transform kernel matrix. Furthermore, since the energy distribution of DCT blocks obtains from the real MPEG-2 video sequences mainly concentrate on the low frequency region, it is beneficial to exploit this property to speed up the DCT-to-IT coefficients conversion in the case of transcoding. The detail process of our proposed fast DCT-to-IT coefficients conversion scheme is described in the following.

2.1 Factorization of the Transform Kernel Matrix

Let T_8 be the transform kernel matrix of DCT, H be the IT transform matrix:

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix}, K = \begin{pmatrix} H & 0 \\ 0 & H \end{pmatrix}. \text{ Jun Xin's transform kernel matrix}$$

(we refer as S or S-transform thereafter), which is shown in (1) and (2), is first adopted as our initial transform kernel matrix, since it requires less 64 operations and only two calculating stages compared with Bo Shen’s one.

$$S = \begin{pmatrix} H & 0 \\ 0 & H \end{pmatrix} \times T_8^T = K \times T_8^T \tag{1}$$

Where the superscript T denotes matrix transposition.

$$S = \begin{pmatrix} a & b & 0 & -c & a & d & 0 & -e \\ 0 & f & g & h & 0 & -i & -j & k \\ 0 & -l & 0 & m & a & n & 0 & -o \\ 0 & p & j & -q & 0 & r & g & s \\ a & -b & 0 & c & 0 & -d & 0 & e \\ 0 & f & -g & h & 0 & -i & j & k \\ 0 & l & 0 & -m & a & -n & 0 & o \\ 0 & p & -j & -p & 0 & r & -g & s \end{pmatrix} \tag{2}$$

Where the values a...s are (rounded off to four decimal places): a= 1.4142, b= 1.2815, c=0.45, d= 0.3007, e= 0.2549, f= 0.9236, g= 2.2304, h=1.7799, i=0.8638, j=0.1585, k=0.4824, l=0.1056, m=0.7259, n=1.0864, o=0.5308, p=0.1169, q=0.0922, r=1.0379, s=1.975. For the proof and more details of the S, please refer to [5].

Then, we shall focus on efficient factorization of the S. A factorization of DCT that is the fastest existing algorithm for 8-point DCT due to Arai, Agui, and Nakajima [7] [8], is exploited to perform the factorizations of the transform kernel matrix S. According to this factorization, T₈ is represented as T₈=DPB₁B₂MA₁A₂A₃, which we state without proof.

Thus, we have

$$S = K \times T_8^T = \overbrace{K \times A_3^T A_2^T A_1^T M^T} B_2^T B_1^T P^T D^T \tag{3}$$

Where D is a diagonal matrix and P is a permutation matrix. We observe that D=D^T, P=P^T. So

$$S = \overbrace{K \times A_3^T A_2^T A_1^T M^T} \times (B_1 B_2)^T \times P \times D \tag{4}$$

After calculating and comparing all possible combinations of this sequence of matrix multiplications, we find that the product of the matrices within the over braces renders the sparsest matrices. Then defining: S^d = K × A₃^T A₂^T A₁^T M^T, we have

$$S = S^d \times (B_1 B_2)^T \times P \times D \tag{5}$$

The DCT-to-IT coefficients conversion now can be carried out multiplication by (B₁B₂)^T and S^d in turn (for simplicity, we refer as S^d or S^d-transform thereafter). The multiplication with D can be ignored since it can be absorbed in MPEG-2 inverse quantization matrix without any change in arithmetic complexity of the dequantizer.

The matrix P causes only changes in the order of the components it can be ignored as well. The matrix of (B_1B_2) only contains 0, 1, and -1 and only S^d contains non-trivial elements as shown in the following.

$$B_1B_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 1 \end{pmatrix} \tag{6}$$

$$S^d = \begin{pmatrix} 4 & 0 & 0 & 0 & a & b & c & 1 \\ 0 & 0 & d & 4 & -e & 0 & f & 2 \\ 0 & 4 & 0 & 0 & 0 & -b & 0 & 1 \\ 0 & 0 & -b & 2 & g & 0 & -h & 1 \\ 4 & 0 & 0 & 0 & -a & -b & -c & -1 \\ 0 & 0 & -d & -4 & -e & 0 & f & 2 \\ 0 & 4 & 0 & 0 & 0 & b & 0 & -1 \\ 0 & 0 & b & -2 & g & 0 & -h & 1 \end{pmatrix} \tag{7}$$

Where the values a ... h are (rounded off to four decimal places): a= 1.0824, b= 1.4142, c=2.6132, d= 4.2426, e= 3.9198, f= 1.6236, g= 1.3066, h= 0.5412.

Let us count the number of operations that are required for performing the above calculating process. Just like the 2D S-transform, the 2D S^d -transform is also separable. For the sake of simplicity, let us confine attention first to the one-dimensional case. The two-dimensional case will be a repeated application for very row and then for very column of each 8×8 DCT block. Let z be an 8-dimensional column vector, and a vector Z be the 1D transform of z. The sparseness and symmetry of S^d can be exploited to perform fast computation of the S^d -transform. The following steps describe the calculating process, which is depicted in Fig. 1 as a flow-graph.

First step, multiplication by $(B_1B_2)^T$:

$$\begin{aligned} x[1] &= z[1] \\ x[2] &= z[2] \\ x[3] &= z[3] - z[4] \\ x[4] &= z[3] + z[4] \\ x[5] &= z[5] - z[8] \\ x[6] &= z[6] + z[7] - z[5] - z[8] \\ x[7] &= z[6] - z[7] \\ x[8] &= z[5] + z[6] + z[7] + z[8] \end{aligned}$$

Second step, multiplication by S^d to get the transform results:

$$\begin{aligned} m_1 &= 4 \times x[1] \\ m_2 &= a \times x[5] + b \times x[6] + c \times x[7] + x[8] \end{aligned}$$

$$\begin{aligned}
 m_3 &= f \times x[7] + 2 \times x[8] - e \times x[5] \\
 m_4 &= d \times x[3] + 4 \times x[4] \\
 m_5 &= 4 \times x[2] \\
 m_6 &= x[8] - b \times x[6] \\
 m_7 &= g \times x[5] - h \times x[7] + x[8] \\
 m_8 &= b \times x[3] - 2 \times x[4]
 \end{aligned}$$

$$\begin{aligned}
 Z[1] &= m_1 + m_2 \\
 Z[2] &= m_3 + m_4 \\
 Z[3] &= m_5 + m_6 \\
 Z[4] &= m_7 - m_8 \\
 Z[5] &= m_1 - m_2 \\
 Z[6] &= m_3 - m_4 \\
 Z[7] &= m_5 - m_6 \\
 Z[8] &= m_7 + m_8
 \end{aligned}$$

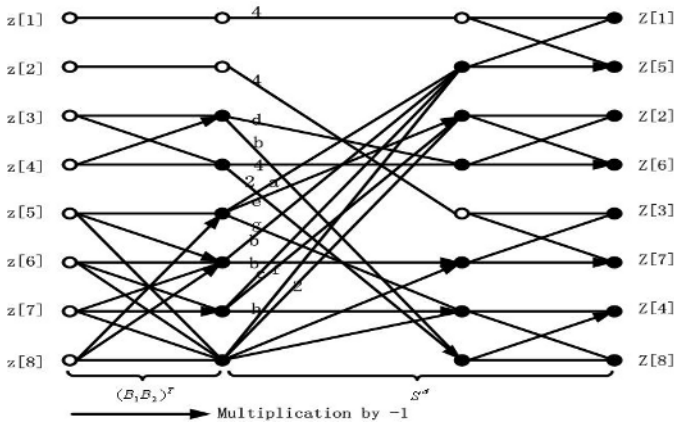


Fig. 1. Fast implementing for S^d -transform

The overall computational requirement of the one dimension calculating process is 15 multiplications and 28 additions. It follows that 2D $(B_1 B_2)^T$ needs 160 ($=16 \times 10$) additions and 2D S^d needs 240 ($=16 \times 15$) multiplications and 288 ($=16 \times 18$) additions, for total of 688 operations. Thus, when compared with the S-transform, the S^d -transform saves 16 operations, where it saves 112 multiplications but increases 96 additions. In addition, our proposed S^d -transform requires three calculating stages, where the $(B_1 B_2)^T$ needs one and the S^d needs two. S^d can be used as a novel transform kernel matrix to perform DCT-to-IT coefficients conversion.

2.2 Simplified S-Transform for the Low Pass Block

The key to reducing the video transcoding complexity is to reuse the information gathered during MPEG-2 decoding stage [3]. Either the S-transform or the S^d -transform can be explicitly used for a low pass block. However, this is not efficient since both of them do not utilize the coefficients distribution property of a low pass

block. In the following, we will demonstrate that a significant saving of computation can be got by taking advantage of the fact that only the low frequency coefficients are nonzero.

We first select to adopt the S-transform as the foundations to simplify the calculating process. Note that one can argue that why not use the S^d -transform to replace the S-transform in processing the low pass block, since the former requires less operations in computation. The reason is that before multiplications by S^d , the input low pass block first needs to be multiplied by $(B_1B_2)^T$ which resulting the intermediate resulting block is no longer a low pass block. So, adopting the S^d -transform cannot exploit the low pass assumption for the computation of the matrix multiplication.

For the same reasons mentioned above, we still first discuss the one dimension case. Let z be an 8-dimensional column vector, and a vector Z be the 1D transform of z . Considering that elements $z[5]-z[8]$ equal to zero for a low pass column vector, the calculating steps described in [5] can be simplified as:

$$\begin{aligned}
 m_1 &= a \times z[1] \\
 m_2 &= b \times z[2] - c \times z[4] \\
 m_3 &= g \times z[3] \\
 m_4 &= f \times z[2] + h \times z[4] \\
 m_6 &= m \times z[4] - i \times z[2] \\
 m_7 &= j \times z[3] \\
 m_8 &= p \times z[2] - q \times z[4] \\
 \\
 Z[1] &= m_1 + m_2 \\
 Z[5] &= m_1 - m_2 \\
 Z[2] &= m_3 + m_4 \\
 Z[6] &= m_4 - m_3 \\
 Z[3] &= m_6 \\
 Z[7] &= -m_6 \\
 Z[4] &= m_7 - m_8 \\
 Z[8] &= m_7 + m_8
 \end{aligned}$$

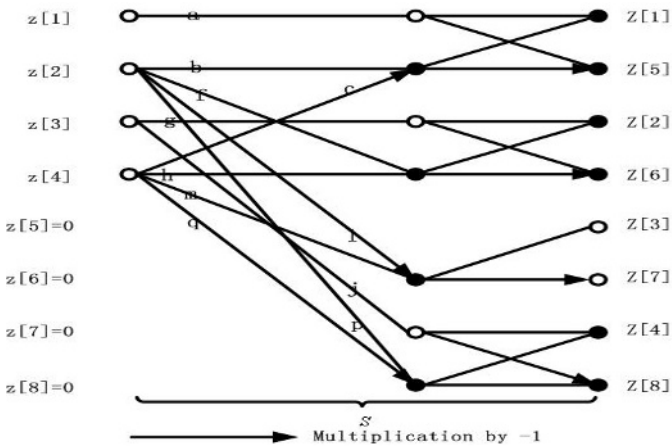


Fig. 2. Simplified S transform for low pass block

Where the const values $a \dots q$ have the same value in equation (2) and the zero variables are eliminated from the computing steps. The corresponding flow-chart is depicted in Fig. 2. This simplified calculating process is referred as the simplified S-transform hereafter.

The simplified S-transform totally needs only 11 multiplications and 11 additions. Furthermore, considering that the 2D transform of a low pass 8×8 DCT block needs only four 1D column transforms and eight 1D row transforms due to that the last 4 column vectors in a low pass block are all zero vector. So, It follows that 2D transform needs 132 ($=12 \times 11$) and 132 ($=12 \times 11$) additions, for total of 264 operations. Thus, the simplified S-transform saves 62.5% of operations (i.e., 440 operations) than the S-transform for a low pass 8×8 DCT block.

2.3 Summary of the Proposed Scheme

The operations and calculating stages needed in different approaches for DCT-to-IT coefficients conversion are tabulated in Table 1. Note that the simplified S-transform is only used for the low pass block.

Table 1. The number of operations and calculating stages for DCT-to-IT coefficients conversion

| Method | Pixel domain | Bo Shen | S-transform | S ^d -transform | Simplified-S |
|--------------------|--------------|---------|-------------|---------------------------|--------------|
| Add | 672 | 352 | 352 | 448 | 132 |
| Mul | 256 | 352 | 352 | 240 | 132 |
| Shift | 64 | 64 | 0 | 0 | 0 |
| Sum of operations | 992 | 768 | 704 | 688 | 264 |
| Calculating Stages | 6 | 4 | 2 | 3 | 2 |

In order to reduce the computation complexity as much as possible, we propose a fast scheme that operates in two steps.

Step 1, the input MPEG-2 8×8 DCT blocks are classified as two types, that are the low pass blocks and the non-low pass blocks.

Usually, the MPEG-2 to H.264 transcoder need to perform an entropy decoding to extract the motion vectors, quantized DCT coefficients from the input MPEG-2 video bit stream. At the same time, the raster scan order of each nonzero DCT coefficients within an 8×8 DCT block is also available to the transcoder. Exploiting such information, if no one of the nonzero DCT coefficients of a 8×8 DCT block is located in the upper left 4×4 quadrant, this 8×8 DCT block is classified as a low pass block, otherwise, it is classified as a non-low pass block. It is well known that only the nonzero coefficients are encoded in MPEG-2 bit-stream. The computation for judging the type of block is very little compared with the computation of DCT-to-IT coefficients conversion and can be ignored.

Step 2, We adopt the optimum algorithm to perform DCT-to-IT coefficients conversion according to the type of DCT block, that is using the S^d-transform for the non-low pass blocks and using the simplified S-transform for the low pass block to perform DCT-to-IT coefficients conversion respectively.

The block diagram of our proposed fast DCT-to-IT coefficients conversion scheme is depicted in Fig. 3.

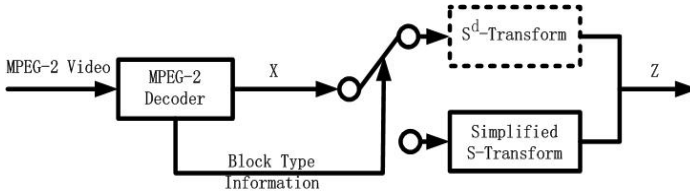


Fig. 3. The proposed fast scheme for DCT-to-IT coefficients conversion

In the worst case, where all input 8×8 DCT blocks are non-low pass blocks, our proposed scheme can save 16 operations for each 8×8 DCT block, and in the best case, where all input 8×8 DCT blocks are low pass blocks, our proposed scheme can save 440 operations for each 8×8 DCT block compared with Jun Xin's method. The practical reduction of complexity depends on the percentage of low pass blocks in video sequences. The distortion and speed performance of the scheme are quantitatively evaluated in following section.

3 Experimental Results

In order to evaluate our proposed method, we adopt MPEG Software Simulation Group (MSSG) MPEG-2 software decoder [9] to decode the input MPEG-2 test videos. A flag, i.e., low pass block flag, which denote whether an 8×8 DCT block is a low pass block or not, is adopted in simulations. The decoded 8×8 DCT block (X) and the associated low pass block flag are sent to three processing systems, which adopting the pixel domain method, the Jun Xin's method, and our proposed method respectively. Each of the three processing systems converts X to the IT coefficients. In order to avoid the influences of other H.264 coding tools, such as intra prediction and variable block size motion compensation, the IT coefficients are directly subjected to H.264 quantization, inverse quantization and inverse IT transform process, which are the same processes as in the reference software H.264/AVC JM8.2 [10], to get the reconstructed images. The H.264 re-quantization QP factors ranging from 0 to 51, corresponding to the full H.264 QP range. The block diagram of simulation setting is shown in Fig. 4.

The first 90 frames of each of test sequences in CIF resolution (352×288) are pre-coded to four different MPEG-2 test videos at 2.5Mbps/s, 3Mbps/s, 3.5Mbps/s and 4Mbps/s respectively. These MPEG-2 test videos are all intra-coded at a frame rate of 30 fps. All simulations are performed using Windows XP, Intel P4 2.8GHz CPU and 1GB memories and Visual C++ 6.0 Compiler. The performance is measured using mean Peak Signal to Noise Ratio (PSNR) between the reconstructed frame sequence and the corresponding original uncompressed frame sequence. The computational complexity is measured using the mean runtime of the all DCT block in one frame needed to convert to IT coefficients. Furthermore, the mean runtime is obtained from the average value of three repeated simulations.

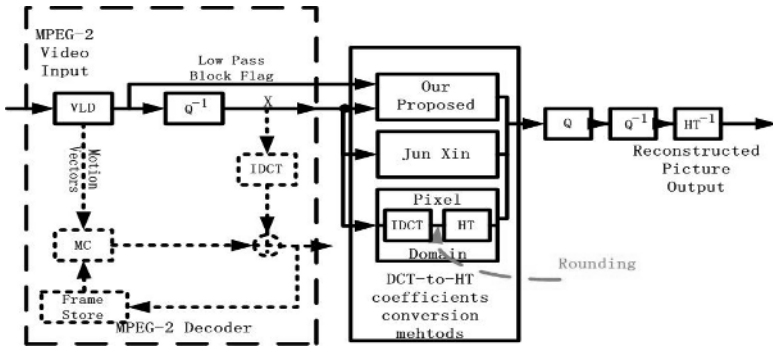


Fig. 4. Simulation setting

A series of experiments processing a great deal of video sequences containing different amounts of motion and spatial details have been made. The results of different sequences are similar except that the percentages of low pass block are different, but due to the limit of page, we only show the results of two sequences: STEFAN and TEMPLATE.

3.1 Our Proposed Method vs. Pixel Domain Method

Fig. 5 shows the PSNR difference between our proposed method and pixel domain method. For all of the four bit-rate, the relative PSNR difference is very small, with the maximum gain around 14×10^{-3} dB and maximum loss around 5×10^{-3} dB. The on the whole superior quality of our proposed method is due to the rounding operation is taken place for the result IT coefficients. On the contrary, for pixel domain method, the saturation and mismatch control (the standard MPEG-2 decoding steps following inverse DCT to reconstruct pixel data), which act as the rounding operation, are taken place for the intermediate results.

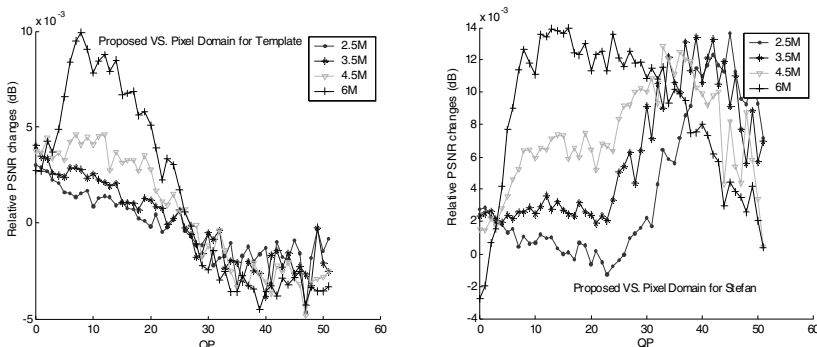


Fig. 5. Relative PSNR difference of our proposed method vs. pixel domain method for (a) TEMPLATE and (b) STEFAN

3.2 Our Proposed Method vs. Jun Xin’s Method

Fig. 6 shows the PSNR difference between our proposed method and Jun Xin’s method. Though our proposed method is equivalent to Jun Xin’s method in terms of mathematics, we can see that our proposed approach produces a little quality degradation compared with Jun Xin’s method. The factor is that there exists the rounding error in the implementing of absorbing the diagonal matrix D to MPEG-2 de-quantization process. However, the minus PSNR gain is very small, with the maximum value lower than 6×10^{-3} dB, which is negligible in practice, especially for practical transcoding application, where the bit-rate reduction and spatial/temporal resolution downscale transcoding are usually considered as the goal. So we can say that our method products almost identical results as Jun Xin’s.

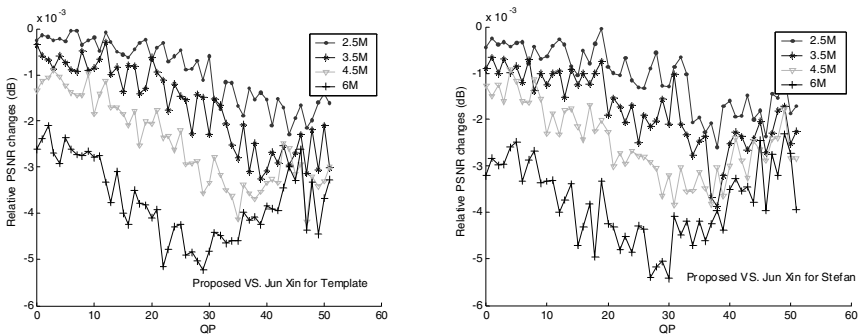


Fig. 6. Relative PSNR difference of our proposed method vs. Jun Xin’s method for (a) TEMPLATE and (b) STEFAN

3.3 Complexity Comparison

The overall percentages of low pass blocks for each sequence at different bit-rate are shown in Table 2. As expected, the lower the bit-rate is, the more of the low pass blocks are in video sequences. This is due to that in order to meet the requirement of target bit-rate, the video encoder has to adopt high quantization parameter, which may cause more DCT coefficients to be quantized to zero. The actual runtime requirements (in terms of microseconds used) for each method at different bit-rates are also shown in Table 2. The relative improvements of our proposed method compared with Jun Xin’s approach are shown inside the parenthesis. Our proposed method achieves the lowest runtime compared with the other methods and saves about 34.3%–42.9% computation compared with Jun Xin’s method. The complexity reduction is proportional to the percentage of the low pass blocks in one real video sequence.

It is interesting to note that the reduction of computational complexity of Jun Xin’s method compared with the pixel domain method is not obvious. This is because that Jun Xin’s method increases 96 multiplication operations compared with the pixel domain method as shown in Table 1 and the real-arithmetic multiplication operation is usually three to four times higher time consuming than the real-arithmetic addition operation in our simulation setting.

Table 2. Runtime for DCT-to-IT coefficients conversion

| Test Sequence | Bit-Rate (Mb/s) | Low pass block percentage | Runtime for DCT-to-IT coefficients conversion (ms) | | |
|---------------|-----------------|---------------------------|--|---------|--------------|
| | | | Pixel Domain | Jun Xin | Proposed |
| TEMPLATE | 2.5 | 60.6% | 12.82 | 12.25 | 7.29 (40.5%) |
| | 3.5 | 44.2% | 13.03 | 12.38 | 7.47 (39.7%) |
| | 4.5 | 36.0% | 12.86 | 12.22 | 7.60 (37.8%) |
| | 6 | 27.1% | 13.05 | 12.12 | 7.84 (35.3%) |
| STEFAN | 2.5 | 59.7% | 13.21 | 12.18 | 6.95 (42.9%) |
| | 3.5 | 49.4% | 13.20 | 12.39 | 7.16 (42.2%) |
| | 4.5 | 41.6% | 13.11 | 12.33 | 7.50 (39.2%) |
| | 6 | 28.2% | 13.24 | 12.26 | 8.06 (34.3%) |

4 Conclusion

In this paper, we derive a fast scheme to reduce the computational complexity of the DCT-to-IT coefficients conversion. We analyze the operations involved in our proposed fast scheme. The efficiency of our proposed fast scheme is also quantitatively evaluated. Our simulation results show that the proposed method leads to about 34.3%–42.9% saves in computational complexity, with negligible impact in PSNR of less 6×10^{-3} dB. In our simulations, only the intra-coded MB is used to test. Considering that the DCT coefficients decoded from inter-coded MBs of MPEG-2 video are computed from the motion compensated residual, there is more low pass block in inter-coded frames than in intra-coded frames. Our proposed method can be more efficiently used in inter-coded frames transcoding. Due to its efficiency, the proposed approach has been applied to our on-going real-time MPEG-2 to H.264 transcoder.

Acknowledgement

This work is supported by the National Nature Science Foundation of China (60302028), by the Key Project of International Science and Technology Cooperation (2005DFA11060), by the Key Project of Beijing Natural Science Foundation (4051004).

References

1. Thomas Wiegand, Gary Sullivan. "Draft ITU-T recommendation and final draft international standard of joint video specification (ITU Rec. H.264 | ISO/IEC 14496-10 AVC)," JVT-G050, Pattaya, Thailand, Mar. 2003.
2. ISO/IEC JTC11/SC29/WG11, "Generic coding of moving pictures and associated audio information: video", ISO/IEC 13818-2. May 1994. A. N. Netravali and B. G. Haskell, Digital Pictures, 2nd ed., Plenum Press: New York, 1995, pp. 613-651
3. A. Vetro, C. Christopoulos, and H. Sun. "Video transcoding architectures and techniques: an overview". IEEE Signal Processing Magazine, Vol 20, no.2, pp.18-29, March. 2003

4. Hari Kalva. "Issues in H.264/MPEG-2 video transcoding". CCNC 2004. First IEEE, 5-8 Jan. 2004
5. J. Xin, A. Vetro and H. Sun, "Converting DCT coefficients to H.264/AVC transform coefficients," IEEE Pacific-Rim Conference on Multimedia (PCM), Lecture Notes in Computer Science, ISSN: 0302-9743, Vol. 3332/2004, pp. 939, November 2004
6. Bo Shen, "From 8-tap DCT to 4-tap integer-transform for MPEG to H.264 transcoding," Proc. IEEE International Conf. on Image Processing (ICIP04), Oct. 2004
7. Y. Arai, T. Agui and M. Nakajima, "A fast DCT-SQ scheme for images," Trans. Of the IEICE, E 71(11): 1095, November 1988
8. W. B. Pennebaker and J. L. Mitchell, JPEG still image data compression standard, Van Nostrand Reinhold, 1993, pp. 53-57
9. MPEG-2 video decodec v12, available online at <http://www.mpeg.org/MPEG/MSSG>
10. 10.H.264/AVC reference software JM8.2, available online at <http://bs.hhi.de/~suehring/tml/download>

A Fast Full Search Algorithm for Motion Estimation Using Priority of Matching Scan

Taekyung Ryu¹, Gwang Jung², and Jong-Nam Kim¹

¹ Division of Electronic Computer and Telecommunication Engineering

Pukyong National University, Busan, 608-737 Korea

² Department of Mathematics and Computer Science

Lehman College, The City University of New York, Bronx, NY 10468, USA

{toydev, jongnam}@pknu.ac.kr, gwang.jung@lehman.cuny.edu

Abstract. Full search motion estimation in real-time video coding requires large amount of computations. Reducing computational cost for full search motion estimation is critical research issue for enabling fast real-time video coding such as MPEG-4 advanced video coding. In this paper, we propose a novel fast full search block matching algorithm which significantly reduces unnecessary computations without affecting motion prediction quality. The proposed algorithm identifies computational matching order from initial calculation of matching differences. According to the computational order identified, matching errors are calculated based on partial distortion elimination method. The proposed algorithm could reduce about 45% of computational cost for calculating block matching errors compared with the conventional algorithm without degrading any motion prediction quality. The proposed algorithm will be particularly useful for realizing fast real-time video coding applications, such as MPEG-4 advanced video coding, that require large amount of computations.

1 Introduction

In video coding, full search (FS) algorithm based on block matching finds optimal motion vectors which minimize the matching differences between reference blocks and candidate blocks in search area. FS algorithm has been widely used in video coding applications because of its simple and easy hardware implementation. However, high computational cost of the FS algorithm with very large search area has been considered as a serious problem for realizing fast real-time video coding.

Several fast motion estimation algorithms have been studied in recent years in order to reduce the computational cost required. These algorithms can be classified into two main groups. One group of algorithms is based on lossy motion estimation technique with degradation of prediction quality compared with the conventional FS algorithm. The other group of algorithms is based on lossless estimation technique that does not degrade the prediction quality. The lossy group of algorithms includes unimodal error surface assumption algorithm, multi-resolution algorithm, variable search range algorithm with spatial/temporal correlation of the motion vectors, half-stop algorithm using threshold of matching distortion, and others [1]. Lossless group of

algorithms includes successive elimination algorithm (SEA), modified SEAs [2]-[8], fast algorithm using a fast 2-D filtering method [9], massive projection algorithm, [10], partial distortion elimination (PDE) algorithm and modified PDE algorithms [11]-[13].

The PDE algorithm as a lossless motion estimation technique has been known as a very efficient algorithm in the sense that it could reduce unnecessary computations required for matching error calculations. To further reduce unnecessary computational cost in calculating matching errors, Kim et al. proposed fast PDE algorithms based on adaptive matching scan [11]-[12]. Obtaining adaptive matching scan order however requires additional computational overhead.

In this paper, we propose a novel lossless fast full search motion estimation algorithm by reducing unnecessary computations which do not affect prediction quality. To achieve this, we divide the matching blocks into sub-square blocks and determine the computational order for the sub-blocks from initial calculation of matching errors. Instead of using conventional top-to-bottom matching ordering, we calculate the matching errors adaptively according to the pre-determined computational order. The proposed algorithm requires very little additional computational overhead for identifying computational order, which makes our algorithm efficient enough to be used with other fast algorithms such as SEA or Multilevel SEA (MSEA). The proposed novel algorithm reduces approximately 45% of computational cost for block matching error calculations compared with the known PDE algorithm without any loss of prediction quality.

This paper is organized as follows. In Section 2, conventional fast full search algorithms are described. Section 3 presents the motivation and the description of the proposed algorithm. In Section 4, experimental results showing the performance of the proposed algorithm are presented. Section 5 concludes the paper.

2 Conventional Fast Full Search Algorithms

SEA algorithm is well known lossless FS motion estimation algorithm. It removes impossible candidate motion vectors by using the sum of the current block, the sum of the candidate blocks and the minimum sum of absolute difference (SAD) [2]. At first, the algorithm computes sum of the rows or the columns in the reference and candidate blocks. After calculating initial matching error of the search origin in the search area, the algorithm removes impossible candidate motion vectors by the Eq. (1). In the Eq. (1), R means the norm of the reference block in the current frame and $C(x,y)$ represents the summation of the norms of the candidate blocks with the motion vector (x,y) in the previous frame. SAD_{min} means the sum of absolute differences as a distortion measure at that checking time. By the Eq. (1), useless computations required for impossible candidates can be eliminated without any degradation of predicted images. If the summation of the candidate blocks is satisfied with the Eq. (1), the candidate blocks are calculated for matching errors with SAD, otherwise the candidate blocks are removed and next candidate blocks will be checked.

$$R - SAD_{min} \leq C(x, y) \leq R + SAD_{min} \quad (1)$$

A few modified algorithms based on SEA have been reported. The performance of the various modified SEA algorithms depends on the way how to calculate the initial matching errors. Oliveira et al. [3] proposed the modified algorithm with less initial matching distortion from the adjacent motion vectors. Lu and others [4] could reduce impossible candidate vectors further by using hierarchical structure of Minkowski's inequality with pyramid of 5 levels. Coban and others [5] used the concept of the Eq. (1) to determine motion vector with optimized rate-distortion. They extended the Eq. (1) by adding the weighted rate term to avoid unnecessary computations. Wang et al. [6] used the Eq. (1) by adding PDE, square root, and square term in order to reduce computational cost. Meanwhile, Gao et al. [7] proposed an algorithm for reducing impossible candidate vectors by using tight boundary levels shown in Eq. (2) and Eq. (3).

$$SSAD_L = \sum_{u=0}^{2^L-1} \sum_{v=0}^{2^L-1} |R_L^{(u,v)}(i, j) - C_L^{(u,v)}(i+x, j+y)| \tag{2}$$

$$SSAD_0 \leq SSAD_1 \leq \dots \leq SSAD_l \leq \dots \leq SSAD_L \leq SAD_{\min} \tag{3}$$

Another algorithm to reduce the computational cost is the PDE approach [11]-[13]. The algorithm uses the partial sum of matching distortion to eliminate impossible candidates before completing calculation of matching distortion in a matching block. That is, if an intermediate sum of matching error is larger than the minimum value of matching error at that time, the remaining computations for matching errors is abandoned. The k th partial SAD can be expressed by the Eq. (4),

$$\sum_{i=1}^k \sum_{j=1}^N |f_{t+1}(i, j) - f_t(i+x, j+y)| \quad k = 1, 2, \dots, N \tag{4}$$

where N represents matching block size. The term, $f_{t+1}(i, j)$, means image intensity at the position (i, j) of the $(t+1)$ th frame. The variables x and y are the pixel coordinate of a candidate vector. If the partial sum of matching distortion exceeds the current minimum matching error at k , then we can abandon the remaining calculation of matching error ($k+1$ to N th rows) by assuring that the checking point is an impossible candidate for the optimal motion vector. Kim et al. [11] calculated block matching errors to reduce unnecessary calculations with the four-directional scan order based on the gradient magnitude of images instead of the conventional top-to-bottom matching scan order. Block matching errors are calculated to further reduce unnecessary computations with adaptive matching scan [12]. While these approaches could reduce unnecessary computations for getting block matching errors, they need additional computations to determine the matching scan order.

3 Proposed Algorithm

Modified PDE algorithms, such as spiral search algorithm and cascaded algorithms, have used adjacent motion vectors [11]-[12]. Ability to reject impossible candidate vectors in the PDE algorithm depends on the search strategy, which makes minimum matching errors can be detected faster. Because PDE algorithm with spiral search

rejects impossible candidate vectors faster than simple PDE, we employ the spiral search in the proposed matching scan algorithm. The relationship between matching error and image gradient of the matching block can be summarized by Taylor series expansion [11]-[12]. Let the image intensity at the position (x,y) of the $(t+1)th$ frame be $\{f_{t+1}(p), p=(x,y)\}$, and the motion vector of the position p be $mv=(mvx, mvy)$. We can describe the relationship between the reference frame and the candidate frame as shown in the Eq. (5).

$$f_{t+1}(p) = f_t(p + mv) \tag{5}$$

$$\begin{aligned} d_{t+1}(p) &= |f_{t+1}(p) - f_t(p + cmv)| \\ &= |f_t(p + mv) - f_t(p + cmv)| \\ &\approx \left| \frac{\partial f_t(p+mv)}{\partial x} (cmvx - mvx) + \frac{\partial f_t(p+mv)}{\partial y} (cmvy - mvy) \right| \\ &\approx \left| \frac{\partial f_{t+1}(p)}{\partial x} (cmvx - mvx) \right| + \left| \frac{\partial f_{t+1}(p)}{\partial y} (cmvy - mvy) \right| \end{aligned} \tag{6}$$

By using modified form of the Taylor series expansion, we can express the relationship between the matching distortion and the gradient magnitude of the reference block as shown in the Eq. (6). Here, $cmv=(cmvx, cmvy)$ represents candidate motion vector corresponding to the matching distortion. From the Eq. (6), we can find out an important fact that the matching distortion at pixel p is proportional to the gradient magnitude of the reference block in the current frame, which corresponds to the complexity of the image data. By localizing the image complexity well, we can further reduce unnecessary computations. In general, image complexity is well localized in a block rather than the whole span of an image. In this paper, we calculate the matching error using 4x4 square sub-blocks instead of the conventional 1x16 row vector to measure the complexity of the matching block.

Efficiently identifying complex square sub-block needs to be performed at early stage of the algorithm. In the previous algorithms, a complex square sub-block was found by calculating gradient magnitude in the reference matching block, where the additional computation for calculating the gradient magnitude can be avoided with large candidates. The previous algorithms [11]-[12] can increase computational load more than the original PDE algorithm when both of them are cascaded to other fast algorithms such as SEA [2]. Instead of calculating the gradient magnitude of the matching blocks, we find complex sub-blocks from initial SAD computation at center point of search area. At first, we calculate block matching error by the unit of square sub-block, and then accumulate the sum of sub-blocks. According to the accumulated sum of sub-blocks, we determine the matching order for the following candidates. The idea further reduces unnecessary computations as explained below. Our proposed algorithm uses the matching order for sub-blocks in all candidates of the search range. The Eq. (7) shows our modified PDE algorithm which employs sub-blocks and the matching order of the sub-blocks from initial computation of SAD.

$$\begin{aligned} &\sum_s^k \sum_{i=1}^{N/s} \sum_{j=1}^{N/s} |f_{t+1}(i, j) - f_t(i + x, j + y)| \\ &k = 1, 2, \dots, N / s * N / s, \\ &s \in matching_order[] \end{aligned} \tag{7}$$

In the Eq. (7), $matching_order[]$ is obtained from the initial partial SAD value for each block. We put the pixel number of $N/s * N/s$ square sub-block as N . The matching order of N candidate sub-blocks is calculated for every reference block of the search range. Thus, we increase the probability of scan order that have larger matching earlier. The computational cost required for sorting $N/s * N/s$ sub-block is small and negligible compared to the overall computational cost of the block matching algorithm. Figure 1 shows the block diagram of the proposed algorithm with the adaptive matching scan using the partial SAD.

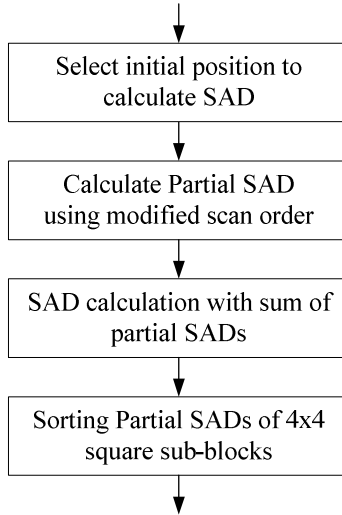


Fig. 1. Block diagram to determine the adaptive matching order

If we cascade the SEA algorithm to our proposed algorithm, we can further remove unnecessary computation. In the MSEA algorithm, block size is 16×16 and sub-block size for adaptive matching scan is 4×4 . If we employ the multilevel SEA instead of the original SEA, we can further reduce computations.

4 Experimental Results

To compare the performance of the proposed algorithm with the conventional algorithms, we use 100 frames of 'foreman', 'car phone', 'trevor', 'clair', 'akio' and 'grand mother' image sequences. In these sequences, 'foreman', and 'car phone' have higher motion variance than the other image sequences. 'clair', 'akio' and 'grand mother' are rather inactive sequences compared with the first two sequences. 'trevor' sequence has intermediate level of motion variance. Matching block size is 16×16 pixels and the search window is ± 7 pixels. Image format is QCIF(176×144) for each sequence and only forward prediction is used.

The experimental results shown in Figures 2 and 3 and Tables 1 and 2 are presented in terms of average number of checking rows with reference to that of full

search without any fast operation. Table 3 shows the peak-to-peak-signal-to-noise ratio (PSNR) performance of the proposed algorithm. All the algorithms employed spiral search scheme to make use of the distribution of motion vectors.

Figure 2 and Figure 3 show the reduced computation of average checking rows using 4x4 square sub-blocks based on PDE algorithm. The adaptive matching scan algorithm significantly reduces unnecessary calculations compared with the conventional sequential scan algorithm. We apply the proposed algorithm to SEA [2] and MSEA [7] to further demonstrate the performance. From the experimental results, we can see that the proposed matching algorithm can reduce unnecessary computations efficiently for PDE itself and cascaded algorithms with SEA and MSEA. Note that the computational reduction is different among three algorithms. In PDE, all candidates in search area are involved in calculating partial matching distortion. However fewer candidates are involved in calculating distortion by SEA or MSEA because many candidates are filtered out by the Eq. (1) and the Eq. (3). MSEA has smaller candidates in calculating partial matching distortion than SEA because of tighter boundary.

Table 1 and Table 2 summarize average numbers of checking rows computed for various algorithms in all sequences of 30Hz and 10Hz, respectively. The average number of checking rows of the conventional full search algorithm without any fast operation is 16.

The importance and efficiency of spiral scan in PDE algorithm was shown in [11]-[12]. From the Table 1, we can see that the computational reduction ratios from the proposed adaptive matching scan combined with SEA and MSEA are 37% and 16% compared with the conventional sequential matching. The results of PSNR are all the same for all algorithms as shown in Table 3 because they are all lossless algorithms.

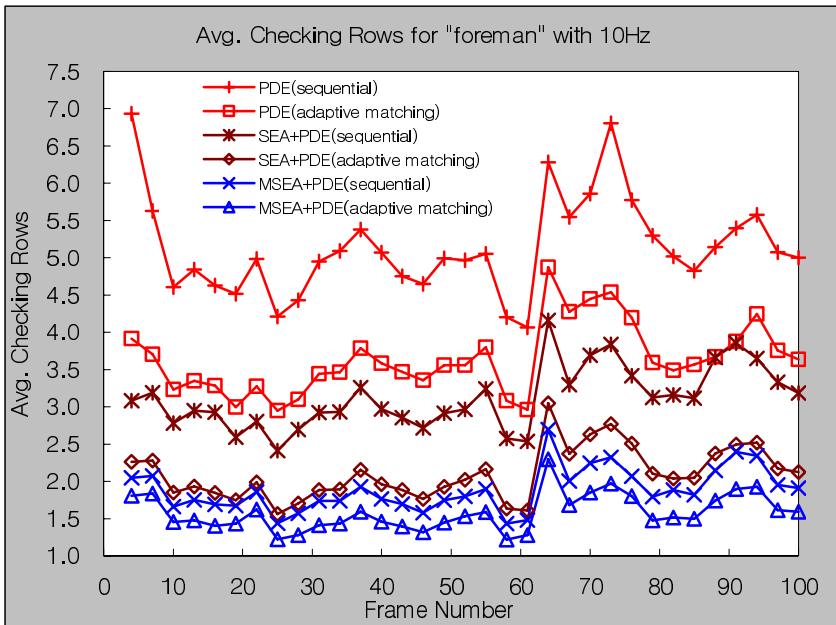


Fig. 2. Average number of rows computed for “foreman” sequence of 10Hz

With the experimental results, we can conclude that our adaptive matching scan algorithm can reduce computational cost significantly without any degradation of prediction quality and any additional computational cost for obtaining matching order.

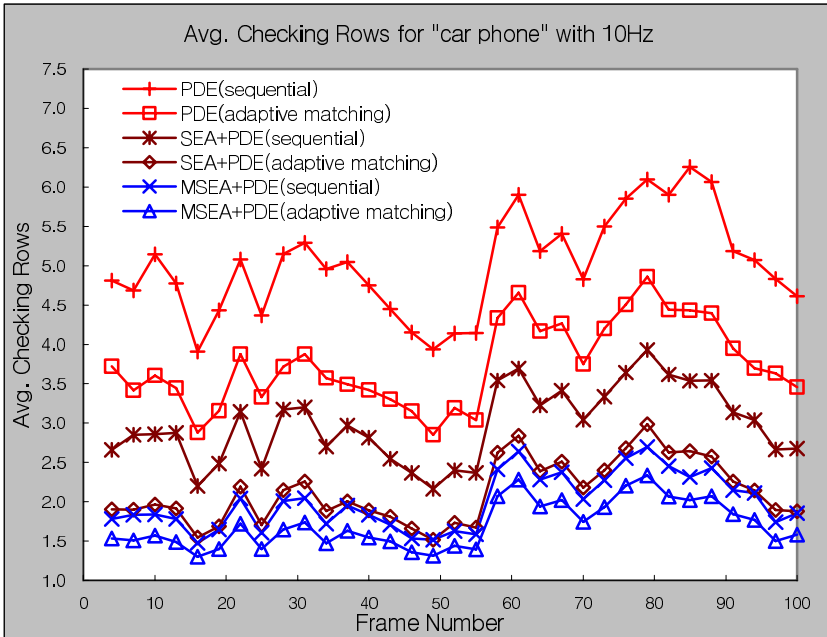


Fig. 3. Average number of rows computed for “carphone” sequence of 10Hz

Table 1. Average number of rows computed for all sequences of 30Hz

| Algorithms | Fore-man | Car phone | Trevor | Claire | Akio | Grand |
|-------------------------------|----------|-----------|--------|--------|-------|-------|
| Original FS | 16.00 | 16.00 | 16.00 | 16.00 | 16.00 | 16.00 |
| PDE (sequential) | 4.05 | 4.23 | 3.18 | 3.94 | 1.71 | 4.18 |
| PDE (adaptive matching) | 2.86 | 3.16 | 2.43 | 3.26 | 1.36 | 3.45 |
| SEA + PDE (sequential) | 1.90 | 2.03 | 1.29 | 1.31 | 0.62 | 1.99 |
| SEA + PDE(adaptive matching) | 1.55 | 1.72 | 1.12 | 1.24 | 0.56 | 1.85 |
| MSEA + PDE(sequential) | 1.32 | 1.56 | 0.91 | 1.05 | 0.58 | 1.77 |
| MSEA + PDE(adaptive matching) | 1.19 | 1.43 | 0.85 | 1.03 | 0.57 | 1.71 |

Table 2. Average number of rows for all sequences of 10Hz

| Algorithms | Fore- man | Car phone | Trevor | Claire | Akio | Grand |
|----------------------------------|--------------|--------------|--------|--------|-------|-------|
| Original FS | 16.00 | 16.00 | 16.00 | 16.00 | 16.00 | 16.00 |
| PDE (sequential) | 4.81 | 4.87 | 4.51 | 4.45 | 2.09 | 4.66 |
| PDE (adaptive matching) | 3.64 | 3.75 | 3.53 | 3.54 | 1.54 | 3.80 |
| SEA + PDE (sequential) | 2.50 | 2.48 | 2.17 | 1.57 | 0.83 | 2.30 |
| SEA + PDE(adaptive matching) | 2.10 | 2.12 | 1.84 | 1.44 | 0.69 | 2.10 |
| MSEA + PDE(sequential) | 1.75 | 1.88 | 1.46 | 1.21 | 0.68 | 1.92 |
| MSEA + PDE(adaptive matching) | 1.58 | 1.71 | 1.32 | 1.16 | 0.63 | 1.83 |

Table 3. Average PSNR of all sequences for the frame rates 30Hz and 10 Hz

| Frame rate | Foreman | Car phone | Trevor | Claire | Akio | Grand |
|------------|---------|--------------|--------|--------|-------|-------|
| 30 Hz | 32.85 | 34.04 | 34.05 | 42.97 | 44.14 | 43.44 |
| 10 Hz | 29.50 | 31.54 | 28.63 | 37.50 | 38.66 | 39.01 |

5 Conclusions

In this paper, we propose a new block matching algorithm by sorting square sub-blocks according to the initial matching distortion. The proposed algorithm reduces unnecessary computations for motion estimation while keeping the same prediction quality compared with the conventional full search algorithm. Unlike the conventional fast PDE algorithms, the proposed algorithm does not require additional computations to identify matching order. The proposed algorithm can be efficiently cascaded to other fast algorithms such as MSEA or SEA. Experimental results show that the proposed algorithm could reduce about 45% of computational cost compared with the conventional PDE algorithm without any degradation of prediction quality. The proposed algorithm will be particularly useful for realizing fast real-time video coding applications, such as MPEG-4 advanced video coding, that require large amount of computations.

Acknowledgement. This work was supported by The Regional Research Centers Program (Research Center for Logistics Information Technology), granted by the Korean Ministry of Education & Human Resources Development.

References

1. F. Dufaus and F. Moscheni, "Motion estimation techniques for digital TV: A review and a new contribution," *IEEE Proceeding*, vol. 83, pp. 858-876, Jun. 1995.
2. W. Li and E. Salari, "Successive elimination algorithm for motion estimation," *IEEE Trans. Image Processing*, vol. 4, pp. 105-107, Jan. 1995.
3. G. de Oliveira and A. Alcaim, "On fast motion compensation algorithms for video coding," *Proc. PCS*, pp. 467-472, 1997.
4. J. Lu, K. Wu, and J. Lin, "Fast full search in motion estimation by hierarchical use of Minkowski's inequality (HUMI)," *Pattern Recog.*, vol. 31, pp. 945-952, 1998.
5. M. Coban and R. Mersereau, "A fast exhaustive search algorithm for rate-constrained motion estimation," *IEEE Trans. Image Processing*, vol. 7, pp. 769-773, May 1998.
6. H. Wang and R. Mersereau, "Fast algorithms for the estimation of motion vectors," *IEEE Trans. Image Processing*, vol. 8, pp. 435-438, Mar. 1999.
7. X. Gao, C. Duanmu, and C. Zou, "A multilevel successive elimination algorithm for block matching motion estimation," *IEEE Trans. Image Processing*, vol. 9, pp. 501-504, Mar. 2000.
8. T. Ahn, Y. Moon and J. Kim, "Fast full-search motion estimation based on Multilevel Successive Elimination Algorithm," *IEEE Trans. Circuits System for Video Technology*, vol. 14, pp. 1265-1269, Nov. 2004.
9. Y. Naito, T. Miyazaki, and I. Kuroda, "A fast full-search motion estimation method for programmable processors with a multiply-accumulator," *Proc. ICASSP*, pp. 3221-3224, 1996.
10. V. Do and K. Yun, "A low-power VLSI Architecture for full-search block-matching motion estimation," *IEEE Trans. Circuits System for Video Technology*, vol. 8, pp. 393-398, Aug. 1998.
11. J. Kim, "Adaptive matching scan algorithm based on gradient magnitude for fast full search in motion estimation," *IEEE Trans. Consumer Electronics*, vol. 45, pp. 762-772, Aug. 1999.
12. J. Kim, S. Byun, Y. Kim and B. Ahn, "Fast Full Search Motion Estimation Algorithm Using Early Detection of Impossible Candidate Vectors", *IEEE Trans. Signal Processing*, vol. 50, pp. 2355-2365, Sept. 2002.

Using Space-Time Coding for Watermarking Color Images

Mohsen Ashourian

Islamic Azad University, Majlesi Branch, P.O. Box 86315-111, Isfahan, Iran
mohsena@ieee.org

Abstract. In this paper, we propose a new scheme for watermarking color images. We use space-time coding for watermark data before embedding in the image. The information of the two generated bitstream of space-time coder are embedded in red and blue component of the image. At the receiver, the space-time decoder combines the information of each bitstream to reconstruct the watermark data. Furthermore, this scheme requires no knowledge of the original image for the recovery of the watermark data. We experiment the proposed scheme for embedding 128 bits in the spatial domain of a color host image of 512×512 pixels. Simulation results show that watermarking scheme based on space-time coding has low visible distortions in the host image and robustness to various signal processing and geometrical attacks, such as addition of noise, compression, cropping and down-sampling.

1 Introduction

Various digital data hiding methods have been developed for multimedia services, where a significant amount of signature data is embedded in the host signal. The hidden data should be recoverable even after the host data has undergone some signal processing operations, such as image compression. It should also be retrieved only by those authorized [1, 2].

Watermarking a multimedia signal can be modeled as transmission of a message (watermark signal) through a communication channel (the multimedia signal). With this view, many of the methods used in digital communication system can be extended for watermarking problem.

Based on this analogy, we propose a space-time block coding system for data embedding in a color image. Space-time coding is popularly used for communication system with deep fading. If we consider host image as the communication channel for the watermark data, fading could be happened due to cropping the image, or its filtering and smoothing. Alamouti pioneered a remarkable space-time block coding scheme over two transmit antennas [3]. This scheme support simple decoding complexity based on linear processing at the receiver. The code provides diversity with half of the maximum possible transmission rate. Fig.1 shows an overview of the space-time coder. In Alamouti's scheme the transmitter sends out data in groups of 2 symbols. The scheme uses two transmit antennas and one receive antenna. At the receiver, we can decode only one channel, when data on the other channel is highly corrupted; otherwise we can combine the received information from both channels.

In the following sections at first we explain how we used space-time coding in the proposed watermarking system in Section 2. Section 3 explains the watermark insertion, and the watermark extraction operations. Finally in Section 4 we present experimental results of the proposed scheme and make conclusions in Section 5.

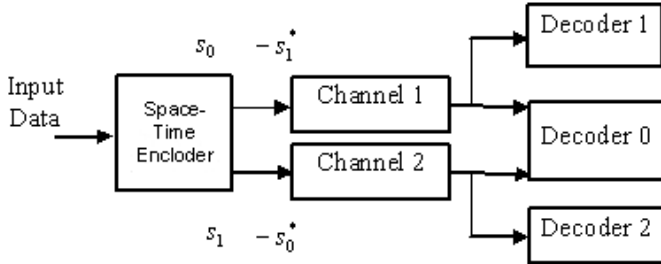


Fig. 1. Overview of the Proposed Scheme

2 Data Embedding and Extraction

The data embedding in the host image could be in the spatial or frequency domain [4, 5, 6]. While data embedding in the spatial domain is more robust to geometrical attacks, such as cropping and down-sampling, data embedding in the frequency domain usually has more robustness to signal processing attacks, such as addition of noise, compression and lowpass filtering.

In this paper, we use data embedding in the spatial domain using the proposed algorithm in [7]. In this method, the watermark data are embedded in one color component of the image based on its luminance. Since human eye is less sensitive to change in blue color, the method suggests modification of the blue component [16]. However the method fails when the image has very low density of blue component. We use space-time coding and embed in both red and blue components, so that it can survive even for images with low density of blue component. We expect that high resilience of space-time coding of the watermark can help the data embedding scheme to survive various attacks.

In order to embed the watermark into pixel values at point (x, y) of the host image, we scramble and arrange these indices as a binary sequence: $\mathbf{D} = \mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n$, where \mathbf{d}_k is a binary variable.

We calculate the luminance value of a pixel at (x, y) using its three color components $r(x, y)$, $g(x, y)$ and $b(x, y)$ as

$$l(x, y) = 0.299r(x, y) + 0.587g(x, y) + 0.114b(x, y) \quad (1)$$

We select a group of pixel with higher value of blue components, and change the blue color component using this equation:

$$b'(x, y) = \begin{cases} b(x, y) + \alpha.l(x, y) & \text{if } d_i = 0 \\ b(x, y) - \alpha.l(x, y) & \text{if } d_i = 1 \end{cases} \quad (2)$$

Where α is the embedding modulation factor. We do a similar process for embedding value in red component of the pixels in the red dominated area of the host image.

For data extraction, we do not need the original host image. At first we calculate an estimate of the blue component of the modified pixel using its neighboring pixels in a window size of 3 by 3 pixels ($C=1$).

$$\hat{b}''(x, y) = \frac{1}{4C} (-2b''(x, y) + \sum_{i=-C}^{+C} b''(x+i, y) + \sum_{i=-C}^{+C} b''(x, y+i)) \tag{3}$$

Here we use of 3 by 3 pixels ($C=1$) around each pixel. Now assuming that we embed each bit M times, we can estimate the average of difference using

$$\sigma_k'' = \frac{1}{M} \sum_{l=1}^M [\hat{b}''(x_l, y_l) - b''(x_l, y_l)] \tag{4}$$

The bit value is determined by looking at the sign of the difference between the pixel under inspection and the estimated original. In order to increase robustness, each signature bit is embedded several times, and to extract the embedded bit the sign of the sum of all differences is used.

We do a similar process for extracting watermark data in the red component of the pixels in the red dominated area of the host image.

4 Experimental Results and Analysis

We use the two images “Birds” and “Boat” which are shown in Fig. 2 and Fig. 3 as host image. Both images are 512x512 pixels and we embed a watermark 128 bits in each one. We set the embedding factors such that their PSNR values stays above 37 dB after data embedding.



Fig. 2. Host Image- Bird

In order to evaluate the system robustness for copyright protection, we should make a binary decision for the presence or absence of the watermark data. We define the similarity factor between the recovered watermark $\hat{s}(m)$ and the original watermark $s(m)$ by



Fig. 3. Host Image – Boat

$$\rho = \frac{\sum_m \hat{s}(m)s(m)}{\sum_m (\hat{s}(m))^2}$$

Based on the value of ρ , we make a decision on the presence ($\rho=1$), or absence of the watermark signal ($\rho=0$).

Resistance to JPEG Compression: The JPEG lossy compression algorithm with different quality factors (Q) is tested. Fig. 4 shows the similarity factor variation for different Q factors. The similarity factor stays above 0.70 for Q larger than 40.

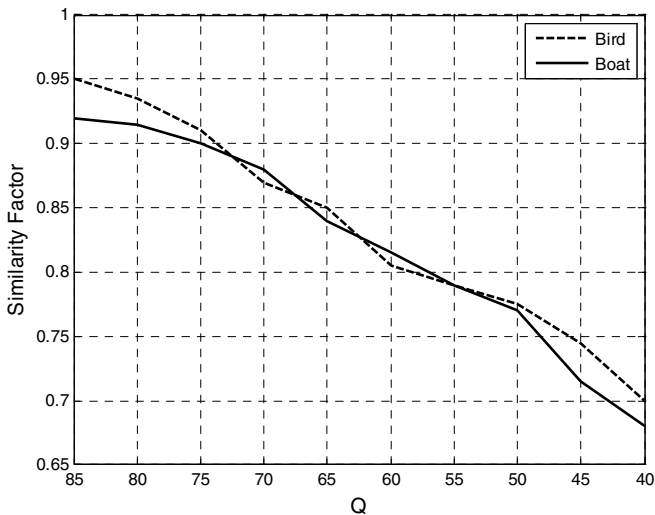


Fig. 4. Similarity factor variation due to JPEG compression

Robustness to Gaussian Noise: We add a Gaussian noise with a different variance to the normalized host signal after embedding the watermark. Fig. 5 shows the variation of similarity factor for additive noise with different variances. From Fig. 5, we conclude that for certain range of noise, our strategy shows good performance in resisting Gaussian noise for data hiding applications.

Resistance to Median and Gaussian Filtering: Median and Gaussian filters of 3×3 mask size are implemented on the host image after embedding the watermark. Similarity factors of the recovered watermark after filtering are listed in Table 1.

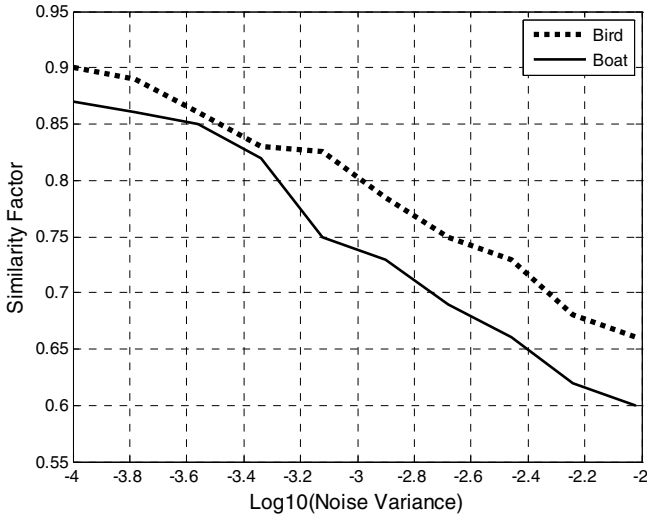


Fig. 5. Similarity factor variation due to additive Gaussian noises

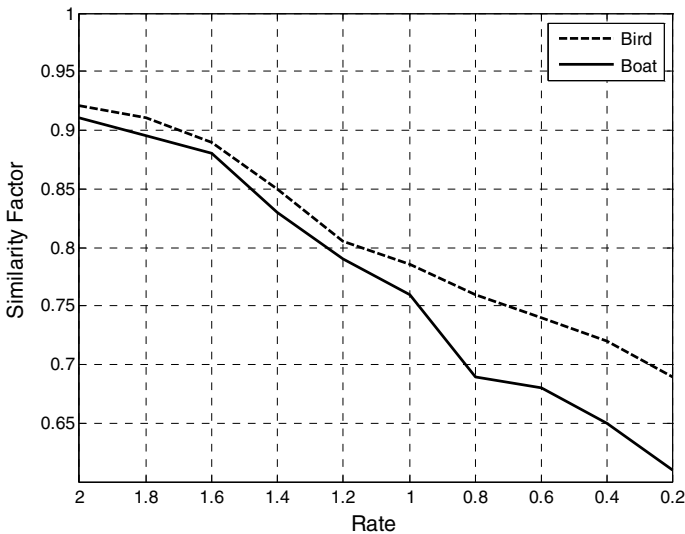


Fig. 6. Similarity factor variation due to JPEG2000 compression

Table 1. Similarity factor after implementing median and Gaussian filters on the host image

| | Median Filter | Gaussian Filter |
|---------------------------|---------------|-----------------|
| Extracted from Bird Image | 0.74 | 0.78 |
| Extracted from Boat Image | 0.70 | 0.73 |

Resistance to JPEG2000 Compression: The JPEG2000 lossy compression algorithm with different output bit rates is tested on the host image. Fig. 6 shows the Similarity factors variation of the recovered watermark.

Resistance to Cropping: Table 2 shows similarity factor values when some parts of the host image corners are cropped. Fig. 7 shows the host image after 10% cropping. Considerably good resistance is due to the existence of two bitstreams in the image

Table 2. Similarity factor for different percentage of cropping the host image

| | 5% | 10% | 15% | 20% |
|---------------------------|------|------|------|------|
| Extracted from Bird Image | 0.82 | 0.69 | 0.64 | 0.55 |
| Extracted from Boat Image | 0.76 | 0.67 | 0.58 | 0.51 |

Table 3. Similarity factor after different amount of down-sampling the host image

| | 1/2 | 1/4 | 1/8 |
|---------------------------|------|------|------|
| Extracted from Bird Image | 0.68 | 0.55 | 0.46 |
| Extracted from Boat Image | 0.70 | 0.58 | 0.51 |

**Fig. 7.** Sample of the host image after 10% cropping

and scrambling of embedded information, which makes it possible to recover the watermark information in the cropped area from the available bitsream in the non-cropped area.

Resistance to Down-sampling: Due to loss of information in the down-sampling process, the host image cannot be recovered perfectly after up-sampling. However, it is possible to recover the watermark data from those pixels available in the host image. Table 3 lists similarity factor values after several down-sampling processes.

5 Conclusions

We have presented a new scheme for embedding watermark data into a color host image. The watermark data are encoded using a space-time coder and the two generated bitstreams are embedded in red and blue components of the host image in the spatial domain. The proposed system does not need the original host image for recovering the watermark data at the receiver. Since the system uses embedding in both red and blue components, it can work well for a variety of images with different distribution of colors. We evaluate the system robustness when the host undergoes various signal processing and geometrical attacks. The results show the system has good robustness. The developed system has low implementation complexity and can be extended for watermark embedding in video in real time.

References

1. Petitcolas F.A.P., Anderson R.J., and Kuhn, M.G.: Information Hiding—a Survey. *Proceedings of the IEEE*, Vol. 87, No.7, (1999) 1062-1078.
2. Wolfgang R.B., Podilchuk C.I., and Delp E.J.: Perceptual watermarks for digital images and video. *Proceedings of the IEEE*, Vol. 87, No. 7, (1999) 1108-1126.
3. Alamouti, S.: A simple transmitter diversity scheme for wireless communication. *IEEE Journal of Selected Areas in Communication*, vol 17, 1998.
4. Bas, P.; Le Bihan, N.; Chassery, J.-M.: Color image watermarking using quaternion Fourier transform. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol.3, (2003)521-524.
5. Barni, M.; Bartolini, F.; Piva, A.: Multichannel watermarking of color images, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 12, Issue 3, (2002) 142 – 156.
6. Ahmadi, N. and Safabakhsh, R.: A novel DCT-based approach for secure color image watermarking. *Proceedings Information Technology, Coding and Computing*, Vol. 2 , (2004)709 – 713.
7. Kutter, M. and Winkler S.: A Vision-Based Masking Model for Spread-Spectrum Image Watermarking. *IEEE Transactions on Image Processing*, Vol. 11, No. 1, (2002) 16-25.

Blind PSNR Estimation of Video Sequences, Through Non-uniform Quantization Watermarking

Tomás Brandão^{1,3} and Paula Queluz^{2,3}

¹ ISCTE, Av. Forças Armadas, 1649-026, Lisboa, Portugal

² IST, Av. Rovisco Pais, 1049-001, Lisboa, Portugal

³ Instituto de Telecomunicações, Torre norte, piso 10,
Av. Rovisco Pais, 1049-001, Lisboa, Portugal
{Tomas.Brandao, Paula.Queluz}@lx.it.pt

Abstract. This paper describes a watermark-based technique that estimates the PSNR of encoded video without requiring the original media data. Watermark embedding is performed in the block-based DCT domain using a new non-uniform quantization scheme that accounts for human vision characteristics. At the extraction, the square distances between received coefficients and nearest quantization points are computed, giving a frame-by-frame estimative for the mean square error (and consequently, the PSNR). Since these distances may be underestimated if distortion is greater than the quantization step used for watermark embedding, it is also proposed the use of the watermark extraction bit error rate, together with image statistics, for PSNR estimation weighting. Results show that PSNR estimation closely follows the true PSNR for a set of video sequences subject to different encoding rates.

1 Introduction

Automated quality monitoring of multimedia data has become an important issue, especially due to the increasing transmission of multimedia contents over the internet and 3G mobile networks. This will allow content providers to track the quality of the received media content, in order to bill users according to their perceived quality of service and to optimize streaming services.

The evaluation of the perceived quality of multimedia contents received by end-users can be achieved by *subjective metrics*, whose scores result from an evaluation performed directly by human viewers. Although these are the most realistic measurements, they require organized tests using a large number of viewers, invalidating its use in real-time applications. An alternative to subjective metrics is the use of *objective metrics*, which aim to automatically compute quality scores that correlate well with the human perception of quality. Ideally, an objective metric should give the same quality score as a human observer. According to the amount of information that is available at the receiver, objective metrics can be classified as [6,10]:

- *Full reference metrics (FR)* – the original media is fully available.
- *Reduced reference metrics (RR)* – side information related to the original media is available.
- *No-reference metrics (NR)* – no information about the original media is available.

In the context of video distribution scenario, it is desirable to perform quality evaluation at the receiving side without accessing the original media data. Thus, most of the effort has been placed in the development of *RR* and *NR* metrics systems. Since the original data is not accessible, watermarking techniques can be potentially used in the development of new *RR/NR* algorithms. The idea consists of embedding an imperceptible reference signal – the watermark – into the original media – the host. During transmission, both the watermark and the host will be subject to the same type of distortions. At the receiver, the watermark is extracted and a quality score is computed by comparing the reference (which is assumed to be known at the receiver) and extracted watermarks.

Watermarking-based approaches have characteristics that resemble *RR* metrics, since additional information is transmitted, although usually independent of the host signal. On the other hand, no additional bandwidth is required to transmit the watermark information. In this sense, watermark-based techniques for quality evaluation can also be placed close to *NR* metrics.

The main goal of the work described throughout this paper is to provide an accurate estimation, for compressed video sequences, of a widely used objective metric, the peak signal-to-noise ratio (PSNR), based on the received watermarked media. Although research in the area is recent, some algorithms have already been proposed: in [9], it is suggested to use watermark extraction error rate as means of scoring the quality of received images; other authors score quality by using the mean square error (MSE) [2,5,8], or the correlation between reference and extracted watermarks [1,6]; as for the embedding techniques, [8,9] use quantization-based approaches [3], while [1,2,5] use spread-spectrum like algorithms [4]; both strategies are analyzed in [6].

In this paper, the watermark is embedded frame-by-frame in the 8×8 block-based DCT domain of the host video, using a quantization-based technique. In order to control watermark imperceptibility, it is proposed the use of non-uniform and frequency adapted quantization functions that consider characteristics of the human visual system (HVS). PSNR estimation is attained through the measurement of the watermark MSE. When compared with previous proposals that also use this metric [2,5,8], in the proposed scheme the differences between extracted and reference watermark signals directly give an estimation for the image MSE. Furthermore, it is proposed to weight these differences as a function of the extraction bit error rate and image statistics. This weighting strategy aims to compensate watermark MSE underestimation when image distortion is larger than the quantization step used for watermark embedding.

Results shown in this paper are focused on frame-by-frame PSNR estimation for video sequences subject to H.264/AVC video encoding at different rates. The experiments were performed by using a sequence set that consists of known video sequences.

This paper is organized as follows: after the introduction, section 2 describes the watermark embedding and extraction schemes as well as the new non-uniform quantization scheme that considers HVS characteristics. Section 3 describes how PSNR estimation is computed from the watermark. Results are shown in section 4 and conclusions are given in section 5.

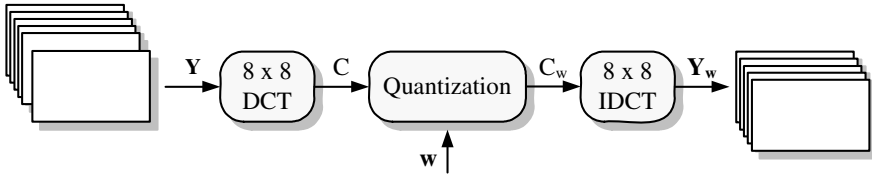


Fig. 1. Watermark embedding

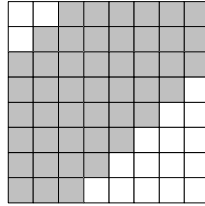


Fig. 2. Watermarking area of an 8x8 block of DCT coefficients

2 Watermarking Scheme

2.1 Watermark Embedding

Consider that a binary watermark message w is to be embedded frame-by-frame into the luminance component Y of the host video. Watermark embedding and extraction are performed in the 8x8 block-based DCT domain of Y , using a quantization-based approach [3].

Figure 1 depicts the embedding scheme. Let $C(i, j, k)$ represent the DCT coefficient at position (i, j) of the k -th block and let Q_l represent the quantizer's output value at level $l = -L, \dots, L-1$. Each coefficient used for embedding (fig. 2) is modified to the nearest quantization level whose least significant bit is equal to the watermark bit to be embedded. Formally, assuming that Q_n is the nearest quantization value to $C(i, j, k)$, the watermarked coefficient – $C_w(i, j, k)$ – is obtained by:

$$C_w(i, j, k) = \begin{cases} Q_n & \text{if } \text{mod}(n, 2) = w(i, j, k) \\ Q_{n+r} & \text{otherwise} \end{cases}, \quad (1)$$

where $w(i, j, k)$ is the watermark bit to embed, $\text{mod}(x, y)$ is the remainder of the integer division of x by y and r is defined as:

$$r = \text{sgn}(C(i, j, k) - Q_n), \quad (2)$$

with $\text{sgn}(x) = -1$ if $x < 0$ and $\text{sgn}(x) = 1$, otherwise.

To complete the embedding process, the inverse DCT transform is computed, resulting in the watermarked frame Y_w .

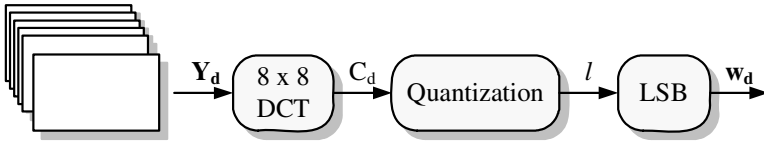


Fig. 3. Watermark extraction

2.2 Watermark Extraction

At the extraction (fig. 3), a watermarked and possibly distorted video frame Y_d is subject to analysis. The estimated watermark, w_d , is obtained by inspecting the LSB of the quantization levels. The extracted watermark bit at position (i, j) of the k -th block, can thus be defined as:

$$w_d(i, j, k) = \text{mod}(l, 2), \quad (3)$$

where l is computed after quantizing the received coefficients $C_d(i, j, k)$ with the same quantization function used at the embedding process.

2.3 Quantization Function

It is important to guarantee that the same quantization function is used both at the embedding and at the extraction phases. A simple solution is to use uniform quantization, as proposed in [8]. However, it presents a serious drawback: the quantization step must be too small in order to attain imperceptibility of the watermark signal. Since the watermark signal is lost if distortion is greater than the quantization step, this approach will only succeed in the presence of relatively small distortions. In order to minimize this problem, a non-uniform quantization scheme, that considers the main features of a perceptual model proposed by Watson [11], was derived. The goal is to assign larger quantization steps to coefficients that allow greater modifications, while keeping the imperceptibility of the watermark signal. Although Watson's model has been extensively used in DCT-based watermarking schemes, its explicit use in the derivation of non-uniform and frequency-adapted quantization functions is, to our knowledge, new.

Watson proposes to estimate the perceptibility of modifications in individual DCT coefficients in terms of *just noticeable differences (JNDs)*, whose threshold values are called *slacks*. The model comprises two masking components accounting for luminance and contrast. The luminance masking threshold for a given coefficient, $T_{lum}(i, j, k)$ is given by:

$$T_{lum}(i, j, k) = T(i, j) \left(\frac{C(0, 0, k)}{C_{00}} \right)^{\alpha_T}, \quad (4)$$

where $T(i, j)$ is the frequency sensitivity of the coefficient at position (i, j) (given in [11]), C_{00} is the average of the DC coefficients in the image, and α_T is a constant with a suggested value of 0.649. *Slack* values – $s(i, j, k)$ – are computed by also considering the effect of contrast masking:

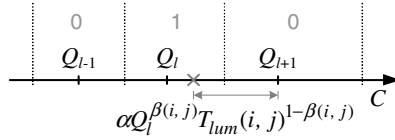


Fig. 4. Three consecutive quantization points

$$s(i, j, k) = \max \left\{ T_{lum}(i, j, k), |C(i, j, k)|^{\beta(i, j)} T_{lum}(i, j, k)^{1-\beta(i, j)} \right\}, \tag{5}$$

where $\beta(i, j)$ is a constant value between 0 and 1. Watson suggests $\beta(i, j)=0.7$ for all $(i, j) \neq (0,0)$ and $\beta(i, j)=0$ for $(i, j)=(0, 0)$.

In order to control the watermark imperceptibility, the coefficient distortion caused by watermark embedding should be proportional to the *slack* values. By analysis of equation (5), it can be concluded that, for $|C(i, j, k)| > T_{lum}(i, j)$, *slack* values increase with the absolute values of DCT coefficients. This suggests the use of non-uniform quantization functions with larger quantization steps for coefficients with larger absolute values.

Consider figure 4, which represents three consecutive quantization points. Suppose that the value of $C(i, j, k)$ is the value marked with a cross, and that the watermark bit to be embedded is ‘0’. According to the proposed embedding scheme, the DCT coefficient will be modified to match the value of Q_{l+1} , thus the error introduced will be $Q_{l+1} - C(i, j, k)$. Since the maximum distortion caused by the watermark for any coefficient in the interval $[Q_l; Q_{l+1}]$ is $Q_{l+1} - Q_l$ (which occurs if $C(i, j, k)=Q_l$), we should have:

$$Q_{l+1} - Q_l = \alpha \cdot s(i, j, k), \tag{6}$$

where α regulates the embedding strength proportionally to the *slack* values, thus controlling watermark imperceptibility.

Let’s consider that $C(i, j, k) = Q_l > T_{lum}(i, j)$. Under this condition, (5) and (6) lead to:

$$Q_{l+1} - Q_l = \alpha Q_l^{\beta(i, j)} T_{lum}(i, j)^{1-\beta(i, j)}. \tag{7}$$

By relating Q_l with Q_{l+1} , the quantization functions can be defined recursively. The initial quantization value was chosen to be the value of $\alpha T_{lum}(i, j)/2$, which ensures model consistency for $|C(i, j, k)| < T_{lum}(i, j)$. It comes, then:

$$Q_{l+1} = \begin{cases} \alpha \frac{T_{lum}(i, j)}{2} & \text{if } l = 0 \\ Q_l + \alpha Q_l^{\beta(i, j)} T_{lum}(i, j)^{1-\beta(i, j)} & \text{otherwise} \end{cases}. \tag{8}$$

Negative quantization values, for $l = -L, \dots, -1$, are computed by symmetry (e.g.: $Q_{-1}=-Q_0, Q_{-2}=-Q_1$, etc.). By analysis of (8), it can be seen that the quantization functions depend on the coefficient position within the block. Considering (4), it can also be concluded that quantization functions depend on the DC coefficients values. At the extraction, these values may be different from the original (due to distortion), thus this dependency must be eliminated. This issue has been solved by neglecting luminance masking, setting $T_{lum}(i, j) = T(i, j)$.

3 PSNR Estimation

The PSNR of a distorted image – I_d – can be defined as:

$$PSNR(I_{ref}, I_d) = 10 \log_{10} \frac{M^2}{MSE(I_{ref}, I_d)}, \tag{9}$$

where I_{ref} is a reference image (usually the original one), $MSE(...)$ represents the mean square error and M is the maximum pixel value. Using the proposed watermarking scheme, the PSNR is estimated by measuring the distortion of the watermark signal, i.e.:

$$PSNR_w = 10 \log_{10} \frac{M_w^2}{\frac{1}{N_w} \sum_{i,j,k} d(i, j, k)^2}, \tag{10}$$

where N_w is the number of watermarked points, M_w is a normalization constant, and $d(i, j, k)$ is the distance between the received DCT coefficient and the nearest quantization value correspondent to the bit value embedded in the coefficient (which is known by the receiver). Formally, $d(i, j, k)$ can be defined as:

$$d(i, j, k) = \begin{cases} |C_d(i, j, k) - Q_n(i, j)|, & \text{if } w_d(i, j, k) = w(i, j, k) \\ \min \left\{ \begin{array}{l} C_d(i, j, k) - Q_{n-1}(i, j), \\ Q_{n+1}(i, j) - C_d(i, j, k) \end{array} \right\}, & \text{otherwise} \end{cases}, \tag{11}$$

where Q_n now represents the quantization value nearest to C_d . Since the DCT is a unitary transform, the value of PSNR can be indifferently measured in the pixel domain or in the transform domain. Assuming that the watermark distortion is equal to the image distortion, M_w was set to same value of M .

The estimated PSNR should be close to the true PSNR if the image is subject to a distortion that is lower than the quantization step. However, as compression rate increases, some coefficients become distorted in such a way that the extracted watermark bit may be the correct one, but with a quantization level different from the one assigned during embedding. In these situations, *false positives* occur.

Figure 5 shows an example of such an event. Suppose that a watermark bit with value ‘1’ was embedded by quantizing the DCT coefficient value to Q_n (represented with a cross). The image was then subject to distortion, which caused the value of the DCT coefficient to change to the value represented with a circle. In this situation, the correct bit is extracted, but instead of $|Q_n - C|$, an underestimated distance $|Q_{n-2} - C|$ is computed, resulting in overestimated PSNR values. Similarly, it is also possible the occurrence of *false negative* situations, i.e., watermark bits with extraction errors, whose corresponding distances are also underestimated.

These situations may also involve quantization points separated by a larger number of quantization levels. Let D represent the difference, measured in number of quantization levels, between quantization values assigned during embedding and the ones retrieved during extraction. False positives occur when D is even and different from 0, while false negatives occur when D is odd and different from 1.

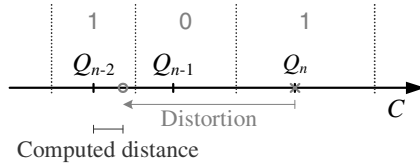


Fig. 5. A false positive

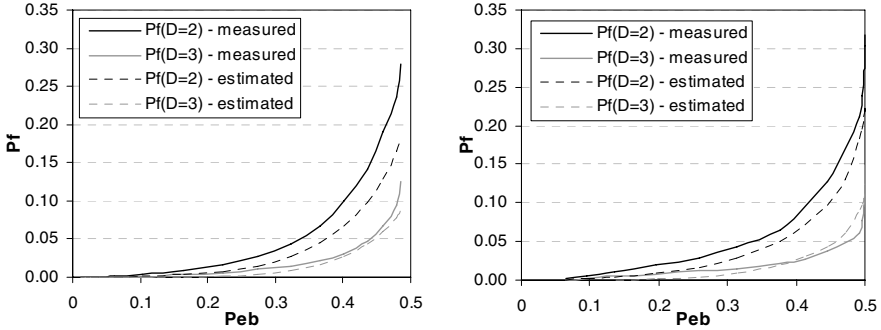


Fig. 6. False positive/negative rates for two test frames

In order to improve the accuracy of PSNR estimation, distances are weighted at the extraction, by accounting for *false negative/positive* rates, P_f . Since these rates are not known *a priori*, statistics were obtained for several frames from several video sequences (fig. 7), by computing P_f as a function of the watermark extraction bit error rate – P_{eb} – and the value of D . As an example, results for two test frames are depicted in figure 6, showing the evolution of a false positive rate (for $D=2$) and a false negative rate (for $D=3$). It can be observed that P_f increases with P_{eb} and that the maximal amplitude of P_f decreases with increasing D . It has also been verified that exponential decay of these rates is less accentuated if the variance (and mean of absolute values) of DCT coefficients are larger, thus P_f also depends on image statistics.

By looking into the evolution of the P_f curves, and accounting for the previous conclusions, one possible function type to approximate the rate – $P_f(D)$ – of false positives (or negatives) for different values of D is:

$$P_f(D) = e^{-f_1(P_{eb}, D, \sigma)} f_2(P_{eb}, D), \tag{12}$$

where σ represents standard deviation for the DCT coefficients located in the watermarking area. The function f_1 regulates the slope of the exponential decay while f_2 regulates its maximal amplitude and the value of P_f for $P_{eb}=0$. Through curve fitting, it was found that the following functions conduct to estimations of $P_f(D)$ close to the measured ones:

$$f_1(P_{eb}, D, \sigma) = \left[\frac{(P_{eb} - 0.5)}{\sigma} \right]^{\lambda c} ; \quad f_2 = \rho P_{eb}^D. \tag{13}$$

Table 1. Values for ρ and λ used in the experiments

| Frame class | ρ | λ |
|-------------|--------|-----------|
| I | 1.1 | 2.0 |
| P | 1.0 | 1.6 |
| B | 0.8 | 1.4 |

where ρ and λ are constants that have been adjusted according to frame class (I, P or B) under analysis. The value of parameter c has been found assuming that the shape of the P_f curves is related with the distribution of the DCT coefficients in the watermarking area, which has been experimentally verified. Admitting that those coefficients follow a generalized Gaussian pdf., the distribution’s shape parameter c can be estimated by [7]:

$$c = F^{-1}\left(\frac{\sigma^2}{E^2[C(i, j, k)]}\right), \tag{14}$$

with

$$F(x) = \frac{\Gamma\left(\frac{3}{x}\right)\Gamma\left(\frac{1}{x}\right)}{\Gamma\left(\frac{2}{x}\right)^2}, \tag{15}$$

where $\Gamma(\cdot)$ is the *gamma* function. $F(x)$ is also referred as generalized Gaussian ratio function. An approximate solution for (14) can be easily found by using a lookup table with values of x and $F(x)$.

Figure 6 also shows plots for the estimated $P_f(D)$ using (12) and (13) for $D=2$ and $D=3$, using the parameters ρ and λ specified in table 1.

The estimated values for false positive/negative rates are used together with the reference watermark (which is known by the receiver) to compute a weighted distortion – $d'(i, j, k)$ – for the watermark:

$$d'(i, j, k) = \begin{cases} P_0 d(i, j, k) + \sum_{t=1}^{D_{MAX}/2} P_f(2t) \cdot d_f(2t), & \text{if } w_d(i, j, k) = w(i, j, k) \\ P_1 d(i, j, k) + \sum_{t=1}^{D_{MAX}/2-1} P_f(2t+1) \cdot d_f(2t+1), & \text{otherwise} \end{cases} \tag{16}$$

The summation at the upper term accounts for the probability of false positives (even values of D), while the summation at the bottom term accounts for false negatives (odd values of D). P_0 and P_1 are normalized values that equal $1 - \sum P_f(D)$ for even D and odd D , respectively. D_{MAX} is the maximum value for the quantization level difference to be accounted, and

$$d_f(D) = \min\left\{ |C_d(i, j, k) - Q_{n+D}|, |C_d(i, j, k) - Q_{n-D}| \right\} \tag{17}$$



Fig. 7. Video sequences used in the experiments. From top left to bottom right: *Stephan*, *Table-Tennis*, *Foreman*, *Paris*, *Akiko*, *News*.

Table 2. Video sequences used in the experiments

| Sequence | Short description |
|--------------|--|
| Stephan | Fast camera and content motion |
| Table-Tennis | Slow camera and slow moving objects |
| Foreman | Still camera over fast human motion; fast pan at the end |
| Paris | Still camera on human subjects; fast local motion |
| Akiko | Still camera on almost still human subject; |
| News | Still camera on human subjects; motion in the background |

represents the distance between the received DCT coefficient and the nearest quantization point at D levels distance.

4 Results

The effectiveness of the proposed algorithm has been evaluated using the sequences displayed in figure 7 in the presence of H.264 / AVC video coding standard. Table 2 synthesizes the main features of the used sequences.

The PSNR of the encoded sequences has been computed frame-by-frame (using the uncompressed watermarked frames as reference frames), and compared with the PSNR estimated from the extracted watermarks, using the proposed non-uniform quantization scheme with (DW) and without (NUQ) distance weighting, as described in sections 2 and 3. In order to increase watermark imperceptibility, only one half of the coefficients located in the watermarking area (fig. 2) were used. The embedding strength α was set to 1.0 for all the experiments.

Sequences have been encoded at different rates (256kb/s@25Hz, 128kb/s@15Hz and 64kb/s@10Hz). It has been used a GOP-12 frame structure with a prediction step

Table 3. Mean error for PSNR estimation for sequences encoded at 256kbit/s @ 25Hz

| Sequence | Global | | I-Frames | | P-Frames | | B-Frames | |
|------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | DW | NUQ | DW | NUQ | DW | NUQ | DW | NUQ |
| Stephan | 0.66 | 2.44 | 0.41 | 1.72 | 0.56 | 2.13 | 0.73 | 2.66 |
| Table | 0.70 | 1.89 | 0.25 | 1.06 | 0.57 | 1.98 | 0.81 | 1.97 |
| Akiko | 0.85 | 0.73 | 1.32 | 1.05 | 0.91 | 0.80 | 0.76 | 0.65 |
| News | 0.92 | 0.56 | 1.41 | 0.85 | 0.91 | 0.54 | 0.85 | 0.54 |
| Foreman | 1.04 | 0.35 | 0.54 | 0.26 | 1.07 | 0.31 | 1.10 | 0.37 |
| Paris | 1.00 | 1.90 | 0.13 | 1.11 | 0.81 | 1.88 | 1.20 | 2.01 |
| Mean error [dB] | 0.86 | 1.31 | 0.67 | 1.01 | 0.80 | 1.27 | 0.91 | 1.37 |

Table 4. Mean error for PSNR estimation for sequences encoded at 128kbit/s @ 15Hz

| Sequence | Global | | I-Frames | | P-Frames | | B-Frames | |
|------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | DW | NUQ | DW | NUQ | DW | NUQ | DW | NUQ |
| Stephan | 1.01 | 3.83 | 0.49 | 2.85 | 0.74 | 3.44 | 1.18 | 4.11 |
| Table | 0.95 | 2.70 | 0.44 | 2.47 | 0.77 | 2.58 | 1.08 | 2.78 |
| Akiko | 1.01 | 0.84 | 1.41 | 1.03 | 1.11 | 0.95 | 0.91 | 0.78 |
| News | 0.90 | 0.43 | 1.38 | 0.48 | 1.01 | 0.44 | 0.79 | 0.42 |
| Foreman | 0.78 | 1.20 | 0.44 | 1.05 | 0.80 | 1.08 | 0.82 | 1.26 |
| Paris | 1.36 | 2.65 | 0.23 | 1.81 | 0.89 | 2.41 | 1.69 | 2.86 |
| Mean error [dB] | 1.00 | 1.94 | 0.73 | 1.61 | 0.89 | 1.82 | 1.08 | 2.03 |

Table 5. Mean error for PSNR estimation for sequences encoded at 64kbit/s @ 10Hz

| Sequence ¹ | Global | | I-Frames | | P-Frames | | B-Frames | |
|------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | DW | NUQ | DW | NUQ | DW | NUQ | DW | NUQ |
| Table | 1.25 | 3.31 | 0.49 | 2.66 | 0.96 | 3.20 | 1.47 | 3.44 |
| Akiko | 1.09 | 0.86 | 1.40 | 1.11 | 1.20 | 0.92 | 1.01 | 0.80 |
| News | 0.59 | 1.10 | 1.22 | 0.60 | 0.75 | 1.04 | 0.45 | 1.19 |
| Foreman | 1.08 | 3.39 | 0.61 | 2.34 | 0.97 | 3.22 | 1.19 | 3.60 |
| Mean error [dB] | 1.00 | 2.17 | 0.93 | 1.68 | 0.97 | 2.10 | 1.03 | 2.26 |

of 3 frames, which means that each 12-frame group consists of one I-frame, three P-frames and 8 B-frames, in the order IBPBBP...

Tables 3 to 5 depict the mean PSNR estimation error that has result from the experiments, discriminated for each frame class, for both DW and NUQ strategies. As can be observed from the tables, the proposed distance weighting strategy generally

¹ The results for the sequences Stephan and Paris have not been considered, since the quality of the resulting encoded sequences (at 64kbit/s @ 10hz) is too low for any practical use.

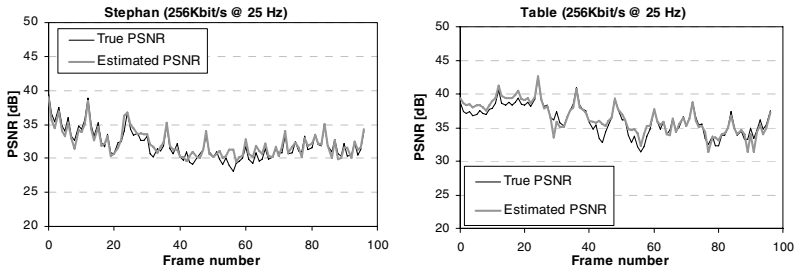


Fig. 8. Results for 256Kbit/s QCIF 25Hz Left: *Stephan*; Right: *Table-tennis*

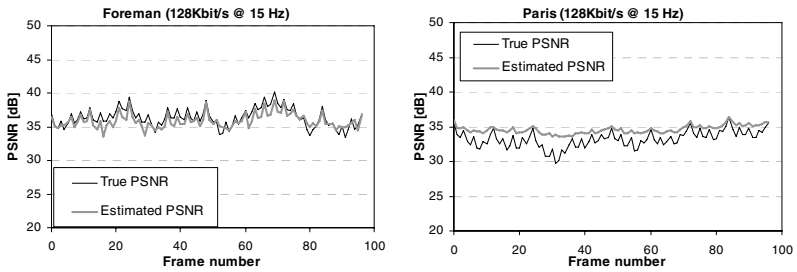


Fig. 9. Results for 128Kbit/s QCIF 15Hz Left: *Foreman*; Right: *Paris*

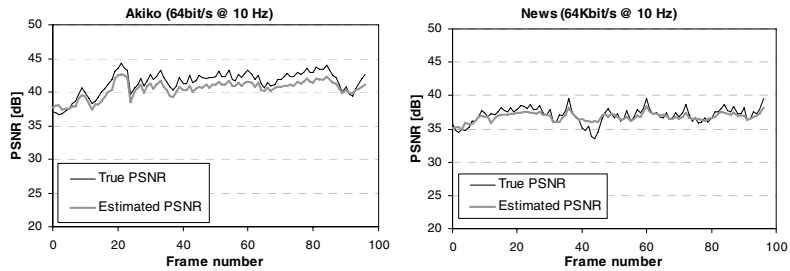


Fig. 10. Results for 64Kbit/s QCIF 10Hz: Left: *Akiko*; Right: *News*

leads to better PSNR estimations, especially for the test sequences where motion is present. As the encoding rate decreases, it becomes more evident that the performance of the distance weighting strategy is better than using non-uniform quantization alone.

Figures 8 to 10 illustrate the temporal evolution of both the true and the estimated PSNR during the first 97 frames of the video sequences. The frame-by-frame PSNR estimation closely follows the true PSNR in the generality of the experiments, a result that has not been shown in any of the related references [1,2,5,6,8,9]. For sequence *Paris* (fig. 9) PSNR estimation is generally overestimated, except for the I-frames (every 12th frame). For sequence *Akiko* (fig. 10), the estimated PSNR is 1 to 2 dB underestimated, yet it still follows the true PSNR. The reason for this higher estimation error is probably related with lack of accuracy in the estimation of P_f ; this can be due

to fast local motion in the sequence *Paris*, and to the absence of motion in the sequence *Akiko* (remember that the expression for P_f has been obtained empirically considering all the test sequences).

Using the proposed algorithm, different strategies for PSNR estimation could also be considered. For instance, PSNR estimation could be performed over an entire group of pictures (GOP), or by considering I-frames alone (since results show that the estimation accuracy is better for this class of frames).

5 Conclusions

In this paper, a watermark-based algorithm that blindly estimates the PSNR for each frame of a video sequence was proposed. One of the main paper contributions, compared with related work in the area, is the derivation of a non-uniform and frequency adapted quantization scheme for the DCT domain, which considers some of the human visual perception characteristics. The other key idea is the use of a weighting function which generally leads to more accurate distortion measurements in situations where the distortion is higher than the quantization step used during embedding. In the majority of the experiments, distance weighting as proven to be effective.

Since the PSNR is a rough objective metric, a topic that deserves further investigation is to analyze new ways of using the proposed scheme to obtain other objective metrics that correlate better with the human perception of quality.

References

1. S. Bossi, F. Mapelli, R. Lancini, "Objective video quality evaluation by using semi-fragile watermarking", *Proc. of Picture Coding Symposium 2004*, S. Francisco, USA, December 2004.
2. P. Campisi, M. Carli, G. Giunta, A. Neri, "Blind quality assessment system for multimedia communications using tracing watermarking", *IEEE Transactions on Signal Processing*, Vol. 51, N. 4, April 2003.
3. B. Chen, G. W. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding", *IEEE Transactions on Information Theory*, Vol. 47, N. 4, May 2001.
4. I. Cox, J. Kilian, T. Shamon, "Secure spread spectrum watermarking for multimedia", *IEEE Transactions on Image Processing*, Vol. 6, N. 12, December 1997.
5. M. C. Q. Farias, M. Carli, A. Neri, S. K. Mitra, "Video quality assessment based on data hiding driven by optical flow information", *Proc. of SPIE Image Quality and System Performance*, San Jose, January 2004.
6. M. Holliman, M. M. Yeung, "Watermarking for automatic quality monitoring", *Proc. of SPIE Security and Watermarking of Multimedia Contents IV*, S. Jose, EUA, January 2002.
7. K. Sharifi, A. Leon-Garcia, "Estimation of shape parameter for generalized gaussian distributions in subband decompositions of video", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 5, N. 1, February 1995.
8. S. Saviotti, F. Mapelli, R. Lancini, "Video quality analysis using a watermarking technique", *Proc. of WIAMIS2004*, Lisbon, Portugal, April 2004.

9. S. Wang, J. Zhao, W. J. Tam, F. Speranza, "Image quality measurement by using watermarking based on discrete wavelet transform", *22nd Biennial Symposium on Communications*, Kingston, Ontario, Canada, May-June 2004.
10. Z. Wang, H. R. Sheikh, A. C. Bovik, "Objective video quality", in *The handbook of video databases, design and applications*, pp. 1041-1078, CRC Press, 2003.
11. A. B. Watson, "DCT quantization matrices optimized for individual images", *Human Vision, Visual Processing, and Digital Display IV*, SPIE, 1993.

Pattern Recognition Using Neighborhood Coding

I.R. Tsang¹ and I.J. Tsang²

¹ Center for Informatics, Federal University of Pernambuco (UFPE),
P.O. Box 7851, 50.732-970 Recife PE, Brazil
`tir@cin.ufpe.br`

² Alcatel Bell N.V., Research & Innovation,
Francis Wellesplein 1, 2018 Antwerpen, Belgium
`ing-jyh.tsang@alcatel.be`

Abstract. We propose a new coding algorithm for binary images based on neighborhood relations. The shape is transformed into a set of representative vectors (position invariant) by coding each pixel according to the number of neighbors in the four directions (north, east, south, west). These neighborhood vectors are transformed into a set of codes satisfying the boundary condition imposed by the size of the image in which the shape is imbedded. A label is attached to the codes to indicate a sequential order of the pixels. The combined code and label characterize an exact shape. Thus following the label ordering and performing simple comparison of the codes an exact shape match is obtained. It is interesting to note that each shape will represent a polyomino. Neighborhood image operators are developed by applying mathematical and logical operations on the code vectors. A code reduction scheme for the purpose of information reduction and generalization of the shape image is proposed. Using the digits 1 and 0 of the NIST handwritten segmented characters set, we show a preliminary application for pattern recognition.

1 Introduction

The problem of shape recognition is fundamental in pattern recognition. Diverse methods for the characterization and classification of shapes can be found in the literature. In general, algorithms follows two criteria: whether they examine only the shape boundary or the whole area, and whether they describe the image in scalar measurements or through structural descriptions [1]. Recently, different methods for shape similarity and recognition have been studied. Using morphological function [2] a given shape is decomposed into primitive parts. The properties and relationships among these primitives are then used to describe the different objects. Another technique is the potential-based approach [3], which identifies the best match from a selected group of shape templates by measuring the repulsive force and torque when the template and sample are put to interact through a potential field. A weighted graph that represents the structure and the quantitative elements of a contour is used for estimation of shape similarity

in [4]. There are many other methods for shape matching and recognition using a variety of different techniques like for instance Fourier descriptors, template matching, stochastic models or moment invariant.

Many of the low-level image analysis operations can be performed using neighborhood operator [4]. Since these operators act locally we have looked for methods that better describe the global structure in the image using neighborhood relations. In this paper, we present a novel method of coding binary shapes. Each pixel is coded according to the number of neighbors in the 4 directions (north, east, south, west). In this way, information about the patterns structure is maintained under translation invariance, while the shape is examined under the whole area instead of just the borders. It is possible then to do different image transformations according to each pixel code, so that global structural of the original image is maintained. A pixel ordering label is used to reinforce the structural information of the shape. Differently from all the other shape description techniques mentioned, we construct a description of the exact shape rigorously to a point where the difference of a single pixel is detectable. Classification of similar shape patterns can be achieved more precisely with the exact description of the shape. A preliminary application of this code scheme and a simple technique for shape generalization and recognition are presented.

The structure of this paper is as follows: Section 2 describes the shape structure, the pixel to neighborhood vector representation and the invariance of this representation. Section 3 presents an exact shape matching algorithm based on the neighborhood coding. Section 4 the neighborhood image operator and applications. Section 5 describes in details the neighborhood coding scheme, a code reduction technique, experiments and results. Section 6 summary and conclusions.

2 Shape Structure

A pattern can be composed of a single connected structure or of many different structures, as for example letters “a” and “i”. The pattern is decomposed into its components using the 4-connected neighborhood and each pixel is coded according to the numbers of neighbors in the four directions. If a pixel is in the inner part of the image the number of neighbors in one of the directions would be the amount of pixel until the border of the image. The structural information of the image is maintained on this set of codes in the same way that pieces of a jigsaw puzzle put together form a distinct pattern. These codes are translation invariant, and rotation can be accomplished in the code level, so no further operation on the image is necessary.

2.1 Component Description

Using a connected component algorithm, we divide a binary image in different clusters or components. Each pixel of the image is transformed into a vector containing the number of neighbors in the four directions $V = (n, e, s, w)$, in

which each letter respectively represents the direction north, east, south and west, see Fig. 1 A set of these neighborhood vectors describes a component and all these sets describe the image, see Fig. 2.

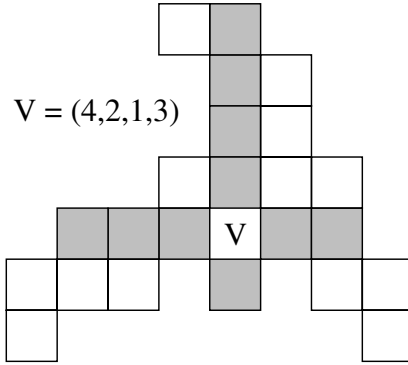


Fig. 1. Neighborhood vector $V = (n, e, s, w)$

We used the Hoshen-Kopelman (HK) algorithm [6] in the implementation of the proposed code scheme. Even though it is sequential, the algorithm is very efficient for connected component analysis, since it just needs one pass over the image to discriminate the different components. It is straightforward to implement the necessary changes to obtain the neighborhood vectors for each pixel of the image. These implementations do not affect greatly the general performance of the HK algorithm.

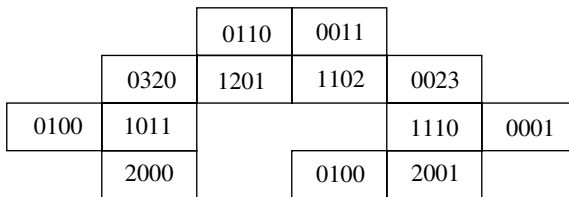


Fig. 2. Neighborhood code

2.2 Invariance

Three types of invariance are addressed here, translation, rotation and reflection invariance. Translation invariance comes as a natural consequence of this representation scheme, which is an excellent property when shape analysis and recognition is of concern. However, rotation and reflection invariance must be

implemented. On one hand, this means additional computation effort. On the other hand, certain applications do require a non-rotational invariance property, like the distinction between characters 6 and 9. Here, only 90°, 180° and 270° rotations are taken into consideration. The reflection invariance addressed are in respect to the horizontal and vertical axes.

Table 1. Changes of coordinates due to clockwise rotations and reflections. The first row shows the original coordinate.

| Rotation | north | east | south | west |
|-----------------|-------|-------|-------|-------|
| 90° | east | south | west | north |
| 180° | south | west | north | east |
| 270° | west | north | east | south |
| <i>Hor.Ref.</i> | south | east | north | west |
| <i>Ver.Ref.</i> | north | west | south | east |

A 90°, 180° or 270° rotation of the image means a shift of coordinates in the neighborhood vectors. If we fix at a specific pixel and rotate the image, it is quite intuitive that a rotation will mean a change of coordinate on the neighborhood vector. Similarly, a horizontal and vertical reflection operation respectively represents a swap between the codes north and south and between codes east and west. As a result, no further manipulation on the original image is necessary for these operations. It is just necessary to change the components of the neighborhood vectors. Table 1 shows the changes of coordinates due to clockwise rotations and to reflection operations.

3 Exact Shape Matching

In most of the shape analysis techniques an image is represented either by scalar measurements or by structural description. Basically, the information of similarity among different shapes is extracted with certain information loss. As a result, a complete reconstruction of a shape to its last pixel is not possible. Although some information reduction is necessary for most of the pattern recognition encoding scheme, we concentrate our effort on an exact description of the connected components of a binary image. Our objective is to retain the maximum amount of information on the description of the image. In this way, the structural information of the shape will only be lost in a controlled and precise manner, according to a specific shape similarity purpose. It is interesting to note that with such exact structural description each component will represent a polyomino [7] or lattice animal.

3.1 Shape and Volume Identification

A label indicating an order sequence of the pixel is attached to the vector, which complements the structural information of the shape. This ordering must follow

a general rule. If a sequential algorithm is used for the connected components analysis, a natural way of implementing it will be following the order in which the image is scanned, i.e. top to down and left to right sequence, see Fig. 3. Consequently, the components of an image are described by a set of neighborhood vectors which follow a specific ordering.

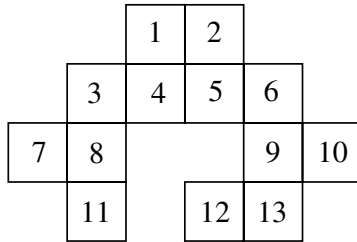


Fig. 3. Ordering label

To show that two shapes will not have the same set of neighborhood vectors in the same ordering label, let's try to perform a structural change in a shape without altering the order sequence and contents of the neighborhood vectors. Here, two shapes will be comparable only if they have the same amount of pixels. A change in the position of any pixel \mathcal{A} will infer a change in its ordering label and/or in its neighborhood vector content, moreover it also causes a change in at least another pixel \mathcal{B} neighbor to pixel \mathcal{A} , since any pixel must be connected. It will also cause a change in at least another neighborhood vector and/or in its ordering label.

In this way, the exact shape match is done by comparison of the neighborhood vectors in the proper ordering sequence. In section 5 we show how to transform the neighborhood vector into a single code. So that, in practice the shape match is done by simply comparing two series of numbers.

It is possible to extend such code scheme from shape to volume identification. In this case, a three dimensional image will be coded according to 6 directions (north, east, front, south, west, back) where the front and back directions account for the extra dimension. The ordering label is still necessary and a set of neighborhood vectors following the ordering label will characterize an exact volume.

3.2 Rotation and Reflection

A rotation and reflection operation on the shape represent a shift or swap of codes in the neighborhood vectors, as already mentioned before. The change on the ordering label will depend on the scanning sequence of the connected components analysis. So, for a correct reordering of the pixel label due to rotation or reflection, a second scan on the original image is necessary. If all the rotations or reflections are of importance this can be done right after the connected

components analysis. This will create an ordering vector associated with each rotation or reflection. Unfortunately, in a sequential algorithm this cannot be done without computational cost.

4 Neighborhood Operators

Image operators are a set of transformations that can simplify the image data, preserving their essential shape characteristics and can eliminate irrelevancies. Many applications in image processing require a preprocessing to normalize or to reduce the amount of data in the image, so that a pattern recognition system is feasible. The neighborhood coding vectors presented yields a practical procedure to implement these image operators. Some transformation can be equivalent to a thinning or skeleton algorithm. In contrast with the mathematical morphology transformation, in which two sets of vectors are combined by using vector operations on the set elements, a image operator is obtained by transforming the image pixels according to the codes in the 4 neighborhood directions.

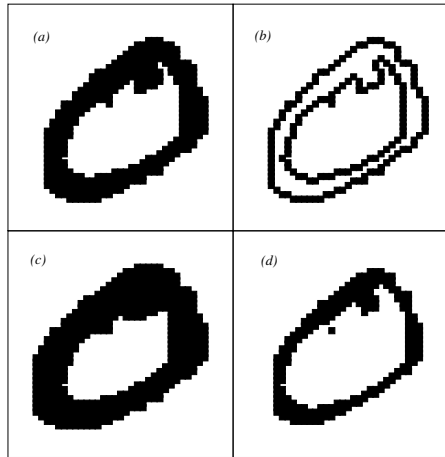


Fig. 4. Image 0 (a) Original, (b) Border Operator, (c) Growing Operator, (d) Thinning Operator

The transformation can be obtained using mathematical and logical operations on the neighborhood codes. By choosing certain properties on the vector codes of the image we can eliminate or add pixels that will fulfill desired characteristics. This could be equivalent to an erosion and dilation transformation in mathematical morphology. As an example, we applied three different operators into the handwritten image of the number 0 and the letter A respectively shown in Fig. 4 and Fig. 5. The first image Fig. 4(a) and 5(a) are the original ones. In the vector code $V = (n, e, s, w)$ the following criteria were applied to obtain

the other images. In Fig. 4 (b) and 5(b) we maintained all the vectors that have at least one of the (n, e, s, w) code equal to zero and we removed all the other vectors from the image. So, just the border pixels are maintained. In Fig. 4 (c) and 5(c) two pixels were added to the neighbor point in which any of the codes in the vector (n, e, s, w) is equal to zero. That is if $n = 0$ then add in the direction north 2 pixels neighbor to the pixel representing this vector. Thus generating a growing operator. Fig. 4 (d) and 5(d) we eliminate the pixels which code vector has an element (n, e, s, w) equal to zero and has $n + s \neq 0$ and $e + w \neq 0$. So, we reduce the shape of the images, but still maintain its characteristics, yielding an operation similar to thinning. Other types of operations are possible and can be designed for specific applications. This transformations show the robustness of the neighborhood coding scheme, since it can be applied not only as a feature extraction for the pattern recognition, as shown in the next section, but also as a preprocessing operator.

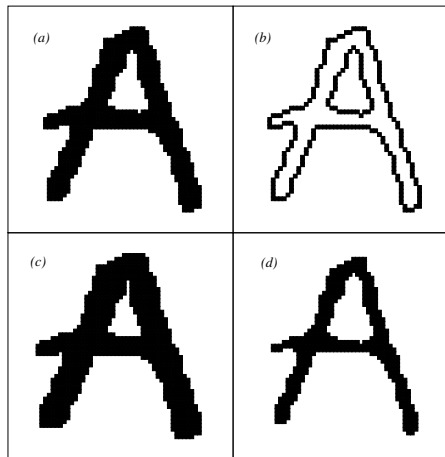


Fig. 5. Image 0 (a) Original, (b) Border Operator, (c) Growing Operator, (d) Thinning Operator

5 Neighborhood Coding

For the recognition of different shapes we transform the neighborhood vectors $V = (n, e, s, w)$ into a single code C . This transformation depends on the size of the image and in practice can yield a large amount of codes. So, a code reduction technique with the minimum amount of information loss is necessary. A controllable scheme for information reduction and learning are very close related. The success of any recognition system is dependent on the amount of information gathered and processed by the learning algorithm.

5.1 Coding According Neighborhood

The transformation function for the vector $V = (n, e, s, w)$ into a code $C = \mathcal{F}(n, e, s, w)$ depends on the size \mathcal{L} of the image. We will assume a square image, the equations for a rectangular image follow straightforward. Since the greatest value that $n, e, s,$ or w can assume is $\mathcal{L} - 1$, the transformation function has the following boundary conditions

$$n + s < \mathcal{L} \tag{1}$$

$$e + w < \mathcal{L} \tag{2}$$

Not taking into account the boundary conditions for the moment. The variables α and β generated by the related pairs (n, s) and (e, w) respectively is given by

$$\alpha = n\mathcal{L} + s \tag{3}$$

$$\beta = e\mathcal{L} + w \tag{4}$$

Considering the case where $\mathcal{L} = 2$, see Table 2. One can verify that various vectors do not obey properties (1) and (2). As a result, a significant amount of code can be reduced. By induction we have that the numbers of vectors that does not satisfy the boundary conditions follow an arithmetic series. So, when the boundary conditions are obeyed equations (3) and (4) become

$$\alpha = n\mathcal{L} + s - \frac{n(n-1)}{2} \tag{5}$$

$$\beta = e\mathcal{L} + w - \frac{e(e-1)}{2} \tag{6}$$

After coding for α and β , a unique representation for the vector V follows

$$C = \alpha\mathcal{K} + \beta \tag{7}$$

To define \mathcal{K} in this equation, we set β to its maximum value and $\alpha = 0$. β is maximum when $e = \mathcal{L} - 1$ and $w = 0$, so we have

$$\beta_{max} = \frac{\mathcal{L}^2}{2} + \frac{\mathcal{L}}{2} - 1 \tag{8}$$

leading to

$$\mathcal{K} = \beta_{max} + 1 = \frac{\mathcal{L}^2 + \mathcal{L}}{2} \tag{9}$$

substituting \mathcal{K} in equation (7) we finally arrive to the transformation function for the vector V to the code C

$$C = \alpha \frac{\mathcal{L}^2 + \mathcal{L}}{2} + \beta \tag{10}$$

To calculate the total number of codes generated in equation (10) set α and β to the maximum, yielding the inequality

$$0 \leq C \leq \frac{(\mathcal{L}^2 + \mathcal{L})^2}{4} - 1 \tag{11}$$

since n, e, s and w are positive integer. The total number of codes is then given by

$$\xi = \frac{(\mathcal{L}^2 + \mathcal{L})^2}{4} \tag{12}$$

According to equation (12), if a square image has the size $\mathcal{L} = 128$, the amount of possible codes are $\xi = 68161536$. For an image with size $\mathcal{L} = 1024$ the number of possible codes then is enormous. To deal with this problem a code reduction technique is necessary.

Table 2. Example of all possible codes for $\mathcal{L} = 2$. The empty space in C are the codes that do not satisfy the boundary condition.

| $(nesw)$ | α | β | C | $(nesw)$ | α | β | C | $(nesw)$ | α | β | C | $(nesw)$ | α | β | C |
|----------|----------|---------|-----|----------|----------|---------|-----|----------|----------|---------|-----|----------|----------|---------|-----|
| 0000 | 0 | 0 | 0 | 0010 | 1 | 0 | 3 | 1000 | 2 | 0 | 6 | 1010 | 3 | 0 | |
| 0001 | 0 | 1 | 1 | 0011 | 1 | 1 | 4 | 1001 | 2 | 1 | 7 | 1011 | 3 | 1 | |
| 0100 | 0 | 2 | 2 | 0110 | 1 | 2 | 5 | 1100 | 2 | 2 | 8 | 1110 | 3 | 2 | |
| 0101 | 0 | 3 | | 0111 | 1 | 3 | | 1101 | 2 | 3 | | 1111 | 3 | 3 | |

5.2 Code Reduction

The objective of reducing the amount of codes is two-fold. Firstly, we reduce the number of codes to be able to tackle the problem in a practical sense. As the amount of codes increases, more computer resources are needed both in memory and processing power. And secondly, information reduction is necessary for generalization of the shape patterns.

To reduce the amount of possible codes we re-scale the components of the vector $V = (n, e, s, w)$ into $V' = (n', e', s', w')$. This can be done in different ways. Here we will approach a linear and logarithmic re-scaling. Suppose that our image size is \mathcal{L} and we want to re-scale so that $\mathcal{L} \rightarrow \mathcal{L}'$, in this way the maximum value that n', e', s' or w' can assume is $\mathcal{L}' - 1$. Choosing an \mathcal{L}' in which will yield a reasonable amount of codes and a re-scaling function, which preserves the interesting structural information of the image, is problem dependable. In addition, we must not forget to fulfill the new boundary conditions:

$$n' + s' < \mathcal{L}' \tag{13}$$

$$e' + w' < \mathcal{L}' \tag{14}$$

A linear re-scaling is straightforward. An interesting case for the linear re-scaling will be when an *a priori* knowledge of the objects imbedded in the image is available. As an example, the objects will not have a pixel vector component greater than \mathcal{M} . In this case a multiplication by $(\mathcal{L}' - 1)/\mathcal{M}$ and a division by a normalization factor, so that the boundary conditions are satisfied, will do the re-scaling. For the logarithmic re-scaling case, let $v = n + s$ and $h = e + w$. Taking into consideration (13) and (14), we have that:

$$n' = \frac{n \log_b v}{v}, s' = \frac{s \log_b v}{v}, e' = \frac{e \log_b h}{h}, w' = \frac{w \log_b h}{h} \tag{15}$$

where $b = \sqrt[\mathcal{L}']{\mathcal{L}}$. If an *a priori* knowledge of the objects is available one can combine the re-scaling functions.

5.3 Experiment and Results

We used the NIST handwritten segmented characters database [8] to perform preliminary experiment using this technique. The database consists of a square binary images with $\mathcal{L} = 128$. As a learning set we used 561 images of 0 and 597 images of 1. Using a logarithmic re-scaling and setting $\mathcal{L}' = 8$, so that the total amount of codes is reduced to 1296, we obtained the probability distribution function (pdf) of these codes shown in Fig. 6 and Fig. 7

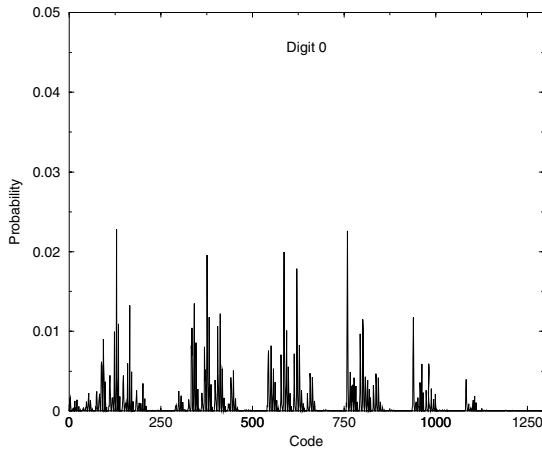


Fig. 6. Pdf of the codes in image 0

For the recognition task we used a simple chi-square analysis [9]. The recognition was done on a test set also taken from the same NIST database. Moreover the learning set and the test set consist of different images. The recognition scores are shown in Table 3. These results are promising, however further research on systems with more objects and different shape patterns are under evaluation.

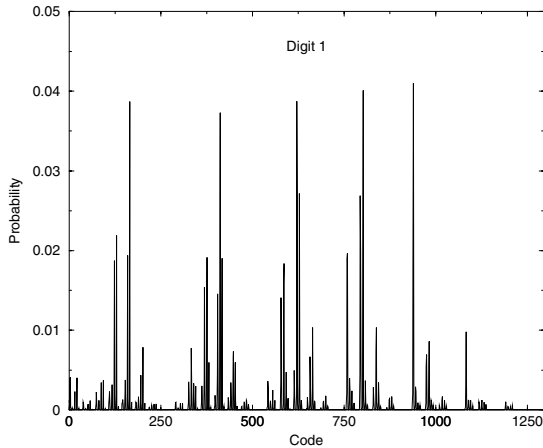


Fig. 7. Pdf of the codes in image 1

6 Summary and Conclusions

We presented a novel method to code and characterize binary shapes. It is based on the pixel neighborhood relation, thus it operates as a low-level image analysis. This approach tries to maintain the maximum structure information of the image, so that information loss for generalization of different shape images can be made more accurate and in a problem dependent way.

Table 3. Recognition scores of experiment

| Image | Learning Set | Test Set | Correct | % |
|-------|--------------|----------|---------|-------|
| 0 | 561 | 554 | 537 | 96.93 |
| 1 | 597 | 615 | 588 | 95.61 |

Using the neighborhood coding scheme we described an exact shape and volume matching algorithm. We also showed that it is possible to use neighborhood operator applying these ideas. A proper mathematical function to transform the neighborhood vector to codes was introduced together with a code reduction scheme. Applications of this technique for different image process problems as well as the use of different pattern recognition techniques such as Bayesian learning, decision tree or neural network using the neighborhood codes are possible.

References

1. T. Pavlidis, Survey a Review of Algorithms for Shape Analysis, Computer Graphics and Image Processing 7 (9) (1978) pp. 243-258.
2. D. Sinha and C. R. Giardina, Discrete Black and White Object Recognition via Morphological Functions, IEEE Transactions on PAMI 12 (3) (1990) pp. 275-293.

3. J. H. Chuang, A Potential-Based Approach for Shape Matching and Recognition, *Pattern Recognition* 29 (3) (1996) pp. 463-470.
4. K. Y. Kupeev and H. J. Wolfson, A New Method of Estimating Shape Similarity, *Pattern Recognition Letter* 17 (1996) pp. 873-887.
5. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Addison-Wesley, 1992.
6. J. Hoshen and R. Kopelman, Percolation and Cluster Distribution. I. Cluster Multiple Labeling Technique and Critical Concentration Algorithm, *Phys. Rev. B* 14 (8) (1976) pp. 3438-3445.
7. S. W. Golomb, *Polyominoes: Puzzles, Patterns, Problems, and Packings*, Princeton University Press, 1994.
8. M. D. Garris and R. A. Wilkinson, Handwritten Segmented Characters Database, *NIST special database 3*, NIST Technical Report and CDROM, February 1992.
9. W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical recipes in C*, Cambridge University Press, 1992.

Naming of Image Regions for User-Friendly Image Retrieval

Andrea Kutics¹ and Akihiko Nakagawa²

¹ Tokyo University of Technology
1404-1, Katakura, Hachioji-shi, Tokyo, 192-0982 Japan

² University of Electro-Communications
1-5-1, Chofugaoka, Chofu-shi, Tokyo, 182-8585 Japan
andi@media.teu.ac.jp, ranaka@ppp.bekkoame.ne.jp

Abstract. This paper presents a novel method for facilitating user-friendly image retrieval by attaching names to image regions. We first detect only the most prominent regions in images when such entities exist, using our own nonlinear image segmentation technique. Besides their visual features, the layout and relations between selected regions are also emphasized. Next, we apply an adaptive and multi-modal classification and naming of image regions using subsequent clustering methods to the features of the regions and related words as well as relevancy information. For both the naming and the testing, we have added a set of illustrations acting as abstract prototypes of the regions to randomly selected natural images. Experiments on 20,000 natural images show the efficacy of using this multilayer region naming model as well as of extensively interacting with users, enabling them to present their queries by a combination of region names, sketches and example images or regions.

1 Introduction

At present, the number of images put on various internet sites exceeds 40 trillion. There are also huge numbers of images in home photo and video collections, not to mention other image sets gathered for specific purposes. These large and unstructured image collections have given rise to the need for intelligent image retrieval tools to index or categorize these data. Several sophisticated systems including Netra, Blobworld, SIMPLicity etc., have been developed. A detailed analysis of these systems and of progress in the early years in this field can be found in [1]. Furthermore, several new methods have emerged and earlier ones have been updated, and the focus has moved to capturing the user's retrieval semantics in a given search context rather than finding images similar in color, texture or shape features. Thus, the so-called semantic gap has become a central problem in this research area.

A number of multi-modal methods for combining textual and visual features have been developed to overcome this problem. These methods include relevance feedback-based methods [2], various approaches drawn from the field of information retrieval and document processing such as methods using latent semantic indexing [3], statistical learning techniques based on mixture models [4, 5], HMM-based methods [6], and several neural-network-based approaches [7]. In most of these methods,

low-level visual features like histogram-based color- and/or wavelet-based texture descriptions, etc. are extracted and then unified into composite feature vectors representing the whole image or evenly divided image blocks. These features are then utilized to directly connect words to images by using pre-annotated training data and various learning approaches. One of the most promising approaches, described in [4, 5], uses image segmentation to capture visual semantics, and matches the detected regions with words by using a joint clustering approach on both feature spaces based on unsupervised learning using EM.

There also exist new promising image annotation techniques that emphasize various machine learning methods [8, 9, 10]. However, these vary greatly in their vocabularies or are set up to specific image sets, and are also sometimes restricted to very simple visual features.

This paper proposes a novel method that facilitates both region- or sketch-based visual and also textual queries in an interactive and iterative manner in order to bridge the semantic gap. First, prominent image regions are determined on the basis of a self-developed nonlinear segmentation technique including the extraction of structural and visual features of the regions. Next, a classification of both regions and words is accomplished by using both psychophysical studies on low-level visual features and related words. Thirdly, simple clustering techniques are applied to determine region names with strong visual coherence using a sketch collection acting as abstract object prototypes. Finally, a tree of abstract concepts is assigned to the regions. Relevance feedback is utilized not only to refine retrieval results but also to create and adapt possible semantic relations among region parts, and also names or keyword categories. Results of experiments on a large domain of natural images show that higher retrieval precision can be achieved by applying keyword, region, or sketch-based and combined queries as well as higher-level user interaction. Our purpose here is thus not the direct automatic annotation of images or their regions, but provision of retrieval results for a given user in a given search context that approximate well enough their understanding of the image content.

2 Detection of Prominent Regions

With our method, an image is represented via its prominent regions, as well as by its visual features including its internal structure and layout. We also determine the size, the position, and also the spatial relations between these regions. This is one of the

Table 1. Prominent region detection


















| squirrel | garden | dish | woman | building |
|---|---|---|---|---|
|  |  |  |  |  |

Table 2. Comparing our prominent region detection results to other methods'

| Original image | | | |
|---|---|---|---|
|  |  |  |  |
| Our method | | | |
|  |  |  |  |
| Blobworld | | JSEG | |
|  |  |  |  |

most important tasks in carrying out in this method as segmentation failures are the main reasons for retrieval errors. However, it would be very difficult, if not impossible, to carry out adequate image segmentation on images of a natural image domain. Consequently, a quasi-adequate or so-called weak [1] segmentation is utilized, that is, only the most prominent regions of an image are focused on. For this purpose, we proposed our nonlinear inhomogeneous diffusion model defined by both color and texture properties, as using single features of either color or texture would not provide enough information to semantically preserve the boundaries of the main regions. This model is expressed by the following partial differential equation and initial value problem:

$$\begin{cases} \frac{\partial}{\partial t} I(x, y, t) = \text{div} \cdot (d(x, y, t) \text{grad}\{I(x, y, t)\}) \\ I(x, y, 0) = I_0(x, y) \end{cases} \quad (1)$$

where *div* and *grad* express the divergence and $(DG_\sigma * I)$ indicates the Gaussian-smoothed gradient $(DG_\sigma * I)$. Function $d(x, y, t)$ represents the diffusivity function or in other words expresses the inhomogeneity. This natural image segmentation model was defined in one of our previous works [11] and it has the advantage of using multiple image properties, or in other words considering both color and textured regions by applying an adaptive conductance parameter defined over either the color and/or the texture gradient accordingly and depending on the texturedness and its gradient. This latter can be given by:

$$C_{ij} = \max_{k \in \{T\}} \sum_{l,m} \left| \frac{T_{i,j}^{(k)}}{T_{\max}^{(k)}} - \frac{T_{i+l,j+m}^{(k)}}{T_{\max}^{(k)}} \right| \tag{2}$$

$$\{l,m \mid -1 \leq l \leq 1, -1 \leq m \leq 1, (l,m) \in \mathbf{Z}, |l| + |m| > 0\}.$$

where T_{ij} indicates a texture feature component (k) at a given image location (C_{ij}).

The model is capable of preserving the boundaries of even highly textured regions by adaptively adjusting the gradient threshold. Preserving region boundaries is very difficult and also very desirable for obtaining adequate prominent regions. In this work, we apply a further refinement on this model, such as using a combination of local edge histograms and Gabor features (three scales and four orientations) to tune the calculation of the diffusivity coefficient with a more adequately defined texture gradient to obtain accurate segment boundaries and thus more precisely detect regions. The reader is referred to [11] for a detailed explanation of the above inhomogeneous diffusion model and that of the image segmentation process. Table 1 illustrates some segmentation examples, while Table 2 compares our results to those obtained from two widely used segmentation methods: (a) Blobworld and (b) JSEG. A new online demonstration of the prominent region detection will soon be available on the URL: <http://www.teu.ac.jp/akm>.

3 Classification and Naming of Image Regions

3.1 Structure and Features of Regions

In further processing steps, we select segments as image regions that are most relevant to the retrieval: at most 4-5 regions per image including the background. These are determined on the basis of their size, position, and structural properties, or of the saliency of their features. Next, we create a visual description of each region by calculating MPEG-7 compliant features for color and texture, such as representative colors over the HSV space, scalable colors (SCD) and color layout, and also homogeneous texture features (Gabor features). Moment invariants and contour-based descriptors including contour shape and also region-based descriptors such as CSD and RSD are calculated to obtain a shape description. We also calculate the position of the regions inside the image via determining their center co-ordinates, lengths of their main axes and their projections. The tree-like structure of the main regions considering the entire image as root and the adjacency graph of the prominent regions are also determined in order to be able to handle composite regions when they are defined by the user. An example image and its region structure are illustrated in Fig. 1.

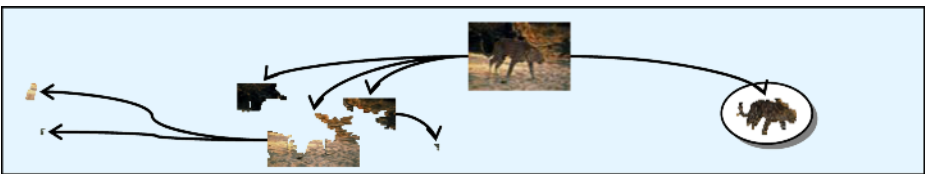


Fig. 1. Structure of prominent image regions

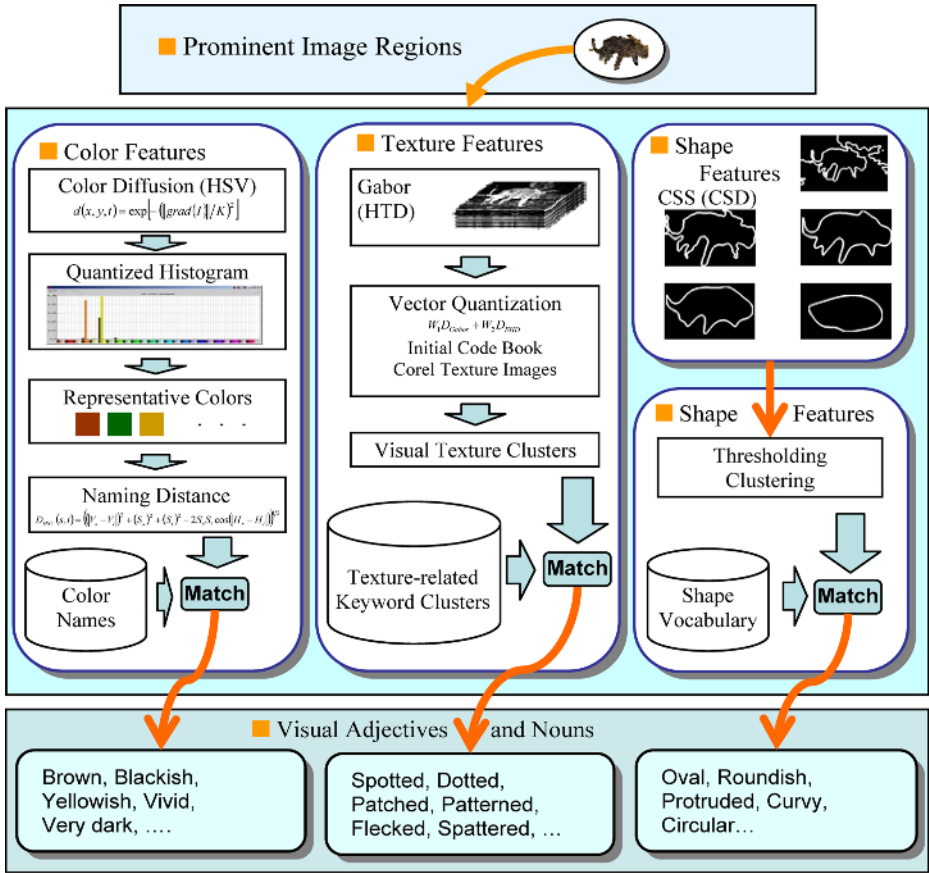


Fig. 2. Regions versus Adjectives

3.2 Regions Versus Adjectives

In order to properly classify and attach names to regions, we first try to express visual features of image regions to related words like names expressing color, texture, shape and/or layout by transforming the primitive visual features of the regions to the textual domain. We utilize feature lexicons here that are built up using human perception-based clustering on the above mentioned low-level visual features, and on related adjectives and nouns expressing these features textually. That is, we carry out a clustering of the regions using their color, texture, shape and layout features, and then map the resulting region clusters to psychophysically predetermined word clusters. This cluster mapping scheme is based on studies carried out by cognitive psychologists [12] and it is derived from experimental results, which suggests that the clustering of low-level region-based visual features and also of visual-related adjectives carried out by humans is nearly objective, such that there is no significant difference observable between the group of individuals in these experiments.

Color features are matched to words in three steps: (1) intra-region smoothing using an exponential diffusivity function is applied on the HSV channels, (2) color quantization is carried out and representative colors are defined for the regions over a quantized HSV space, and (3) these colors are then mapped to color names using the metric:

$$D_{HSV}(s, i) = \left((V_s - V_i)^2 + (S_s)^2 + (S_i)^2 - 2S_s S_i \cos(H_s - H_i) \right)^{1/2} \tag{3}$$

to calculate the distance from color name prototypes [13].

Texture feature mappings are handled by first obtaining a hierarchy of texture feature clusters using self-organizing map and/or learning vector quantization and repeated vector quantizations on the texture features of the regions by selecting prototypes from the Corel “Texture” categories utilized as training data. Next, these texture clusters are mapped to texture-related keyword clusters (11 main clusters) proposed as the texture lexicon in [14] and to several sub-clusters to obtain texture-related adjectives.

Shape clusters are determined by using thresholding and agglomerative clustering on contour- and region-based features and heuristically-determined shape-related words such as “ellipsoid”, “curvy”, “protruded”, etc. Calculation of this cluster matching process is illustrated in Fig. 2.

At the end of these calculations, we attach a number of words, mainly adjectives and a few nouns, to each image region. It is emphasized that here we do not match visual and textual clusters uniquely (in a one-to-one manner), except for texture features, as regions can possess multiple representative colors and shape properties. In order to obtain joint region-word clusters, a soft vector representation of visually-related words is utilized. This can be defined by vectors $(\vec{v}_c, \vec{v}_t, \vec{v}_s)$. An element of a vector, $v_{rj} \in [0,1]$, defines the probability of which word (j) expresses image region (r). These visual word probabilities can be determined by calculating the weighted normalized distance $(\alpha_c D_{ic}, \beta_t D_{it}, \gamma_s D_{is})$ from the given visual cluster center. (The distances for texture and shape can be written by $D_{it} = D_m(i, j) = \sum_k \left(|f_k^{(i)} - f_k^{(j)}| / \delta_k \right)$ and $D_{is} = D_{css}(i, j) = \sum_k (\max(CSS^{(i)}_k) - \max(CSS^{(j)}_k))$ respectively.) Here, weights $(\alpha_c, \beta_t, \gamma_s)$ are used to express the relations between the corresponding visual and textual clusters.

3.3 Region Clusters Versus Region Names

In the third step, we try to determine names for the real image regions as appropriately and as objectively as possible. For this purpose, we utilize visually-related words that we assigned to the regions in the previous step, as well as both layout and structural properties of regions within the image. Here we use the position of each region inside the image, its size, structure and adjacency relations to the other prominent regions or to the background.

First we apply a so-called preprocessing step, in which we separate objectable images from non-objectable ones with the meaning of whether or not the given image contains prominent regions. This decision can be easily made by examining the size, the position and the structure of the detected prominent regions as written above.

Next we apply a simple k -NN clustering to obtain region clusters that are similar in their visual properties. Our purpose with this clustering is to possibly classify regions into top object categories, such as organism: person, animal and botanic, or object (artifact): indoor objects or outdoor objects. This is a very simplified set of the MUSCLE [15] vocabulary recommended for image annotation. We use here only this simply classified and nearly disjointed set as we focus on finding the most commonly used natural image objects, where using whole sets would be quite difficult and time-consuming to implement. As a training set, we use 200 pre-annotated images randomly selected from 10 categories of the Corel Collection and also 100 annotated illustration images. We used 5 or 6 neighbors for the k parameter as this showed the best results (75-88% clustering ratio) in our experiments carried out for all categories.

Next, we apply further vector quantizations to the regions to obtain more specific region clusters that have similar visual features. Our purpose here is to possibly cluster regions and names together, such as several organisms and artifacts with significant shape, texture, color or layout features, like four-legged animals (e.g. wolves), long or circular objects like tools or flowers, or objects with the same color or texture, like the sky, sea, rocks, etc. We also wanted to obtain region clusters with special combinations of some or all of these features, like dress, dish, jaguar, etc. We carry out this calculation on the basis of the statistical properties of the visual features or visually-related words of the regions, and adaptively select the most significant ones for the given clustering task. (For example, shape features are obviously the most important ones for clustering four-legged animals.) Here we emphasize again that our interest in this work is in how well this four-step region naming can help to find objects for a given user, and we do not try automatic annotation of regions.

In order to determine specific region clusters, we have to estimate the conditional probability that a region name ($name_i$) belongs to a given region cluster ($cluster_j$). These probabilities are obtained by calculating the frequencies of region names or their direct synonyms and their lexical abstractions (top categories or super-ordinates), over the predetermined region clusters obtained by vector quantization over the soft vectors of visual related words. Here we utilize the well-known WordNet [16] lexicon, which is sometimes quite ambiguous and we must rely on polysemy counts and restrict the number of senses used. We also use a lexicon of words and pictures, as WordNet does not contain enough interpretations. Here we apply weighting ($w(p,r)$) by the probability of which a region with visual features (f_r) belongs to a top object category (cat_p). This calculation can be expressed by the following equations:

$$P(name_i | cluster_j) = \frac{P(cluster_j | name_i)P(name_i)}{\sum_{l=1}^I P(cluster_j | name_l)P(name_l)} * \arg \max_p P(cat_p | f_r) = \frac{\text{count}(name_i) \in cluster_j}{\sum_{l=1}^I (\text{count}(name_l) \in cluster_j)} * w(p,r) \quad (4)$$

The illustration or sketch-based images are involved in the naming for two main reasons: (1) The annotations of these images are almost always restricted to words or names of regions with direct visual semantics. For example, for an illustration of a skirt, the appearance of more abstract words like fashion, model, etc. are very rare.

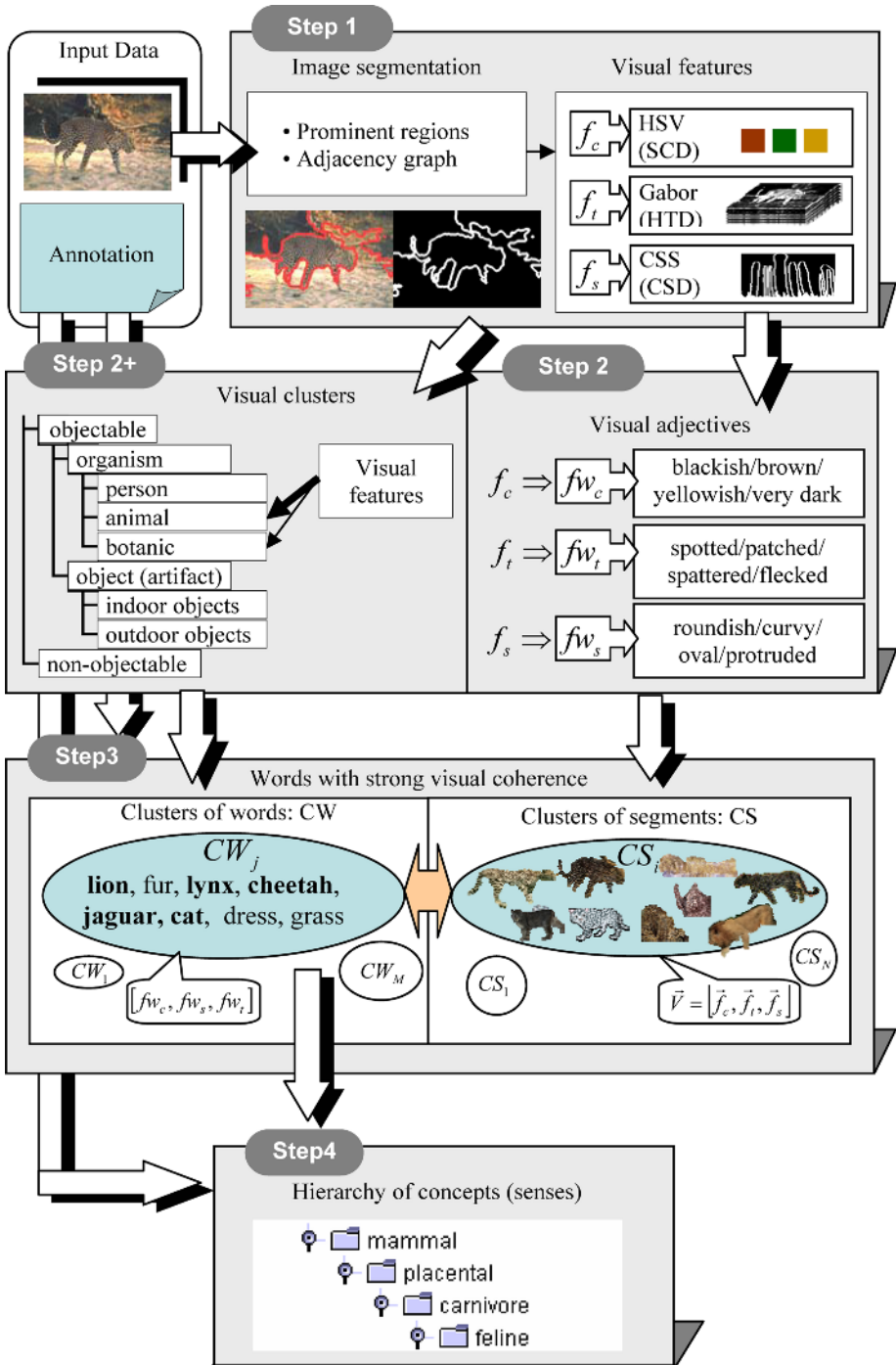


Fig. 3. Overview of the region naming process

This perfectly fits our purpose here, as in this stage of the clustering process we would like to assign names with strong visual coherence. (2) These images usually depict a given region, with its shape, simple color and sometimes simple texture properties, and these can act as abstract prototypes of real-life objects. Finally, we assign a maximum of 5 names or words with strong visual coherence to regions determined on the basis of their probabilities ranked over the clusters.

3.4 Concept Trees for Browsing

In order to support keyword-based browsing, we also determine a hierarchy of abstract concepts. This is accomplished by assigning global sense hierarchies by using WordNet and our simple illustration dictionary. Here we select a concept tree on the basis of word frequency, co-occurrence and polysemy counts based on the words defined in the previous steps. It is also possible to determine concept or word hierarchies by applying a combination of asymmetric and symmetric clustering of words and regions on the second layer as proposed in [4, 5]. In this way, regions and words with higher probabilities are put on the higher levels. However, here we argue that global semantic concept hierarchies are more suitable for browsing in natural data sets, even if these are sometimes unbalanced. It is also possible to apply Expectation-Maximization using mixtures of Gaussians over the features and various distributions of the textual properties, as well as SVM-based learning algorithms, to obtain estimated categories or vocabularies. An overview of the region classification and naming process is illustrated in Fig. 3.

4 Region and Image Retrieval

The method can handle various types of query presentations and it is very useful for keyword or category-type browsing, as well as searches specified by sketch or example images. The user can present a query by using keywords, example images or regions. Except for a very specific query image of a user (e.g. an adult hippopotamus in full profile view; finding it by keywords can take hours on the Internet) he can start his search by keyword or category-based browsing. In this latter case, a concept hierarchy and corresponding images are retrieved, enabling the user to interact with the system in several ways, such as freely browse in the concept tree, find example images and regions or sketch them and start a single or a composite search, etc.

For queries ($qitem_{1..K}$) of both keywords and region or image examples, images ($reg_{i..I}$) are retrieved by calculating query-item probabilities over each cluster ($cluster_j$) weighted by the probability of the cluster with respect to the given region (reg_i):

$$P(qitem_{1..K} | reg_i) = \sum_j \left(P(cluster_j | reg_i) * \prod_{k=1}^K P(qitem_k | cluster_j) \right). \quad (5)$$

The user can also refine the retrieval results by specifying relevant and/or irrelevant images among the retrieved images. Via this relevance feedback, not only are the

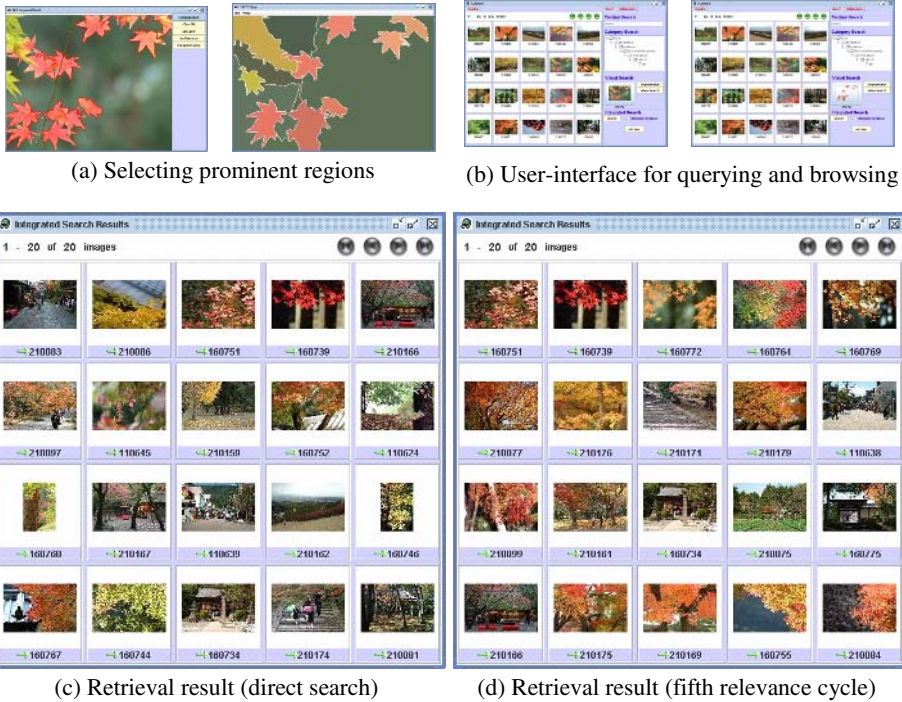


Fig. 4. Retrieval result of a combined search using “autumn leaf” regions and keywords

probabilities of both visual and conceptual words updated, but also user-specific query words are saved in a “user dictionary” to enable user subjectivity to be handled more effectively. Retrieval result examples are illustrated in Fig. 4. The selected regions of “autumn leaves” are shown in Fig. 4(a). Fig. 4(b) illustrates the query and user interface as well as the determined concept tree. Results obtained by the proposed method for “autumn leaf” keywords and regions are shown in Fig.4(c) and (d).

5 Retrieval Results and Discussion

Retrieval experiments were conducted on more than twenty thousand natural images obtained from various randomly selected categories of the Corel Gallery collection and also selected from home photo and video collections to evaluate the method for non-professional images. We did not use any specific keywords or annotations for training purposes, but only the most simple words, almost always the group titles (1 title/100 images) assigned by Corel, like “Churches”, “Tools”, etc. Home photos were set up with very few usable annotations. For the illustration or sketch database, we have one word/image usable for training. We randomly chose 100 test images from 10 additional categories to produce a query set, regardless of the existence of

Table 3. Retrieval precision for composite (5 cycles) and visual-based searches

| Search [%] | animal | garden | dish | woman | building | Ave. |
|------------|--------|--------|------|-------|----------|------|
| composite | 92.5 | 88.3 | 91.3 | 89.2 | 90.0 | 90.3 |
| visual | 48.4 | 64.9 | 62.1 | 51.1 | 56.7 | 56.6 |

well-defined relevant regions. The 5 persons participating in the experiments represented different genders, ages, occupations and cultural backgrounds. Each of them assigned various test query words to each test image according to his/her own background. The retrieval precision obtained for some test image categories is shown in Table 3.

The highest retrieval precision was obtained for composite queries specified by both keyword(s)/categories and regions/images and/or sketches. About 65% of the most relevant regions or pictures were retrieved for each test person in the set of 20 result images in the first retrieval cycle and about 90% of these regions or pictures were retrieved after five consecutive relevance feedback cycles. Obviously, especially good results could be obtained by iterative searches when the user first selected a category and continued the search inside that category. However, even for these searches, we can obtain lower (less than 70%) retrieval precision, because of category assignment errors. We also carried out traditional visual feature-based searches to measure the performance improvement of the method, using the same test images or image regions, and the retrieval precision dropped to an average of 58%. Retrieval failures occur for three main reasons: (1) segmentation errors, which have the most severe effect on shape features, and mostly occur with images containing areas of inhomogeneous texture or a large number of small regions that were aggregated by error (However, sketch images can be a great help, especially with their more precise shape features.), (2) region clustering and naming errors, when the method fails to select the relevant words with visual coherence or assigns mismatching ones, and (3) errors of failing to select proper abstract concept hierarchies and senses in WordNet. These errors can be avoided or at least significantly decreased in number by applying more user interaction, i.e., by incorporating more browsing and also relevance feedback into the retrieval, or applying a more precise statistical model for naming and concept matching, which is an ongoing research topic in our laboratory.

6 Conclusions

In this work, we proposed a new approach to classifying image regions and assigning names to them by trying to capture more semantics in image content for efficient image retrieval. In order to accomplish this, we first applied our self-developed nonlinear multi-scale segmentation algorithm for detecting quasi-appropriate prominent image regions, which themselves carry important visual semantics for the user. In our method, we tried the naming of image regions in several consecutive steps. First we assigned visual adjectives and nouns to region clusters on a psychophysical basis. Next, as a preprocessing step, we classified the regions into five top semantic categories using the k -NN method. Then we assigned semi-abstract names using a soft vector representation defined in the previous step to create joint clusters of regions

and names of direct visual coherence by applying a vector quantization using selective visual features. Finally, we also determined a hierarchy of higher-level concepts. The advantage of this method is that it supports various browsing and searching schemes by enabling the user to present textual, sketch and/or visually-based queries and several different combinations of them. A sketching tool can be an invaluable help for queries, especially those that are difficult to specify only by words or by image examples. By comparing the results of experiments over a large, natural image domain to reported results of traditional visual feature-based CBIR methods or other image-word matching approaches, it can be shown that the proposed method achieves higher retrieval accuracy. This is accomplished by using multiphase region-word clustering and naming and thus semantic inference in several consecutive steps, as well as incorporating user interaction rather than matching images and words directly. By taking greater account of spatial region-relations on a conceptual and semantic basis and applying high-level reasoning, the method can be further developed for image mining and generic image recognition purposes.

References

1. Smeulders, A., et al.: Content-based image retrieval at the end of the early years. *IEEE Trans. PAMI*, Vol.22 No. 12 (2000) 1349-1380
2. Zhou, X. S. and Huang, T. S.: Unifying Keywords and Visual Contents in Image Retrieval. *IEEE Multimedia*, Vol. 9 No. 2 (2002) 23-33
3. Zhao, R. and Grosky, W. I.: Negotiating the semantic gap: from feature maps to semantic landscapes. *Pattern Recognition*, Vol. 35 (2002) 593-600
4. Barnard, K. et al.: Matching Words and Pictures. *Journal of Machine Learning Research*, Vol. 3 (2003) 1107-1135
5. Hofmann, T., Learning and Representing Topic. A Hierarchical Mixture Model for Word Occurrences in Document Databases. *Proc. of CONALD*, Pittsburgh (1998)
6. Wang, J. Z., and Li, J.: Learning-based linguistic indexing of pictures with 2-D MHMMs. *Proc. ACM Multimedia* (2002) 436-445
7. Lim, J-H., Tian, Q., Mulhem, P.: Home Photo Content Modeling for Personalized Event-Based Retrieval. *IEEE Multimedia*, Vol. 9, No. 2 (2003) 28-37
8. Carbonetto, P., de Freitas, N., Barnard, K.: A Statistical Model for General Contextual Object Recognition. *Proc. of Eighth European Conf. on Computer Vision* (2004) 350-362
9. Gabrilovich, E., Markovitch, S.: Feature Generation for Text Categorization Using World Knowledge. *Proc. of The 19th International Joint Conf. for Artificial Intelligence* (2005)
10. Metzler, D., and Manmatha, R.: An inference network approach to image retrieval. *Proc. of the International Conference on Image and Video Retrieval* (2004) 42-50
11. Kutics, A., Nakagawa, A.: Detecting Prominent Objects for Image Retrieval. *IEEE International Conference on ICIP Volume 3*, (2005) 445-448
12. Hendee, W. R., Wells, P. N. T.: *The Perception of Visual Information*. Springer (1997)
13. Mojsilovic, A.: A method for color naming and description of color composition in images. *Proc. of the ICIP* (2002)
14. Bhusnan, N., Rao, A. R., Lohse, G. L.: The texture lexicon: Understanding the categorization of visual texture terms and their relationship to texture images. *Cognitive Science*, Vol. 21(2) (1997) 219-246
15. Hanbury, A.: *MUSCLE*, Guide to annotation, Version 2.0, Tech. Univ. of Vienna (2005)
16. Fellbaum, C. (ed.): *WordNet: An Electronic Lexical Database*. MIT Press (1998)

Shape Retrieval Using Shape Contexts and Cyclic Dynamic Time Warping

Vicente Palazón and Andrés Marzal*

Dept. Llenguatges i Sistemes Informàtics. Universitat Jaume I de Castelló. Spain
{palazon, amarzal}@lsi.uji.es

Abstract. Belongie *et al.* used sets of shape contexts for contours at certain landmarks and showed that they offer good results in shape matching. We propose a better selection of landmark points based on a low-pass filtering of the shape contour and consider that shape context at landmarks along the contour form a cyclic sequence. These cyclic sequences can be properly compared by means of Cyclic Dynamic Time Warping, as shown in the paper. Experiments on the MPEG7 CE-Shape-1 and the SQUID Demo databases show that our proposal outperforms the WARP system and Belongie's approach for shape retrieval.

1 Introduction

Content-based image retrieval is being increasingly demanded in many applications: digital libraries, broadcast media selection, multimedia editing, etc. [15]. The MPEG-7 standard, for instance, is a multimedia content description specification that defines visual feature descriptors in order to allow images retrieval based on their own visual content rather than text [3]. The shape of 2D objects (written characters, trademarks, pre-segmented object contours, 2D/3D object boundaries, etc.) usually provides a more powerful semantical clue for similarity matching than color or texture: humans can recognize characteristic objects from their shape boundary. In order to be effective in classification and retrieval tasks, shape descriptions, combined with (dis)similarity measures, must be robust to noise and invariant to transformations such as translation, scaling, and rotation.

When an arbitrary starting point is chosen, the shape contour can be described as a (counter-clockwise) sequence of Freeman chaincodes, 2D points, curvature and/or distance to centroid values, etc. Another interesting shape descriptor was introduced by Belongie *et al.* [2] and showed very discriminative results: the shape is described as a set of shape contexts at specific landmarks. These descriptions can be compared by means of bipartite matching.

Recently, Bartolini *et al.* [1] have proposed a new Discrete Fourier Transform based approach to compactly represent contours with sequences of points which are invariant to translation, scale, rotation and election of the starting point.

* This work has been supported by the Spanish *Ministerio de Ciencia y Tecnología* and FEDER under grant TIC2002-02684 and the Conselleria d'Empresa, Universitat i Ciència under grant GV06/302.

In the WARP system, these sequences can be compared by means of Dynamic Time Warping (DTW) [13]. This system presents two drawbacks: the technique employed to reconstruct the shape produces contours with an ambiguity modulo a rotation of π radians [5] (which also affects the starting-point selection); and perceptually similar shapes may have significantly different reconstructed contours (in orientation and starting point selection). These problems affect the performance of DTW-based comparisons.

In [2], the ordering information that a contour supplies is not taken into account by Belongie *et al.* When order is considered, DTW can be used to compare shapes. Sequences of shape contexts are invariant to translation, scale and rotation, but not to starting-point election. The WARP system can not help us at this election, but the reconstructed shape seems appropriate for the election of landmark points in order to obtain shape contexts sequences. To obtain starting-point invariance, we propose to consider the problem under the framework of cyclic sequences and to use Cyclic Dynamic Time Warping (CDTW) to compare the outer contours of shapes.

The paper is organized as follows: In Sect. 2 and Sect. 3, shape contexts and DTW are respectively reviewed. In Sect. 4 we critically study the WARP system. A CDTW procedure that provides invariance to the starting point when comparing sequences is presented in Sect. 5. In Sect. 6 the proposed method is summarized. In Sect. 7, experimental results on image retrievals tasks for the SQUID Demo and MPEG-7 CE-Shape-1 databases compare different methods. Finally, some conclusions are presented in Sect. 8.

2 Comparison of Sets of Shape Contexts by Means of Bipartite Matching

In [2], Belongie *et al.* introduced sets of shape contexts and obtained good results in shape classification and retrieval. In their approach, the shape is described as a spatial distribution of points from the contour. Given a set $A = \{a_0, a_1, \dots, a_{m-1}\}$ of landmark points from the contour of a shape, the shape context at point a_i is defined as a histogram h_i of the relative coordinates of the remaining $m - 1$ points: $h_i(s) = \#\{a_j : j \neq i, a_j - a_i \in \text{bin}(s)\}$, where bins are uniform divisions of the log-polar space centered at point a_i (see Fig. 1)). In this description of the contour as a set of histograms, there is an intrinsic translation invariance. It is also possible to obtain scale invariance normalizing all radial distances by the mean distance between all pairs of points in the shape, and rotation invariance using a relative frame based on treating the tangent vector at each point as the positive x -axis.

In order to compare two shapes A and B , Belongie *et al.* propose a measure based on a weighted combination of three scores: (i) the appearance difference (it uses colour information in the gray-scale image patches surrounding points); (ii) the bending energy [4]; and (iii) the shape contexts distance.

We are specially interested in the shape contexts distance. Therefore, let us study more in depth this score. Let A and B be two sets describing contours

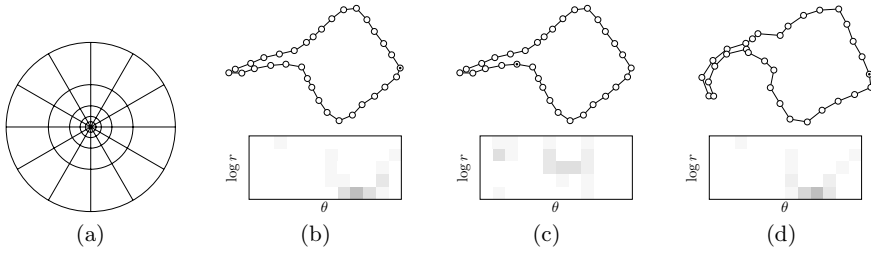


Fig. 1. Shape context computation. (a) Diagram of log-polar histogram bins using in computing the shape contexts. Five bins for $\log r$ and 12 bins for θ . (b) It shows the landmark points from a shape (using Fourier Descriptors) and the corresponding histogram of the point marked by a black dot. (c) The same shape but using a different point. (d) Another shape similar to (b). There is a visual similarity of the histograms (b) and (d), since the situation of the points is alike.

of shapes with the same number of 2D points. Let $\gamma(a_i, b_j)$ be a measure of dissimilarity between a_i and b_j . The γ function is used to determine a matching of two sets. Then, a matching between A and B is a permutation $\pi : \{0, \dots, m - 1\} \rightarrow \{0, \dots, m - 1\}$ where a_i is matched to b_j if $\pi(i) = j$. The permutation π should minimize the total matching cost defined as $\sum_{i=0}^{m-1} \gamma(a_i, b_{\pi(i)})$. The value of $\gamma(a_i, b_j)$ is the cost of matching a_i to b_j , and it is measured using the χ^2 statistic,

$$\gamma(a_i, b_j) = \frac{1}{2} \sum_{s=0}^{S-1} \frac{(h_{a_i}(s) - h_{b_j}(s))^2}{h_{a_i}(s) + h_{b_j}(s)},$$

where h_{a_i} and h_{b_j} are the shape context histograms at a_i and b_j , and S is the number of histogram bins.

Since histograms computation for all points is a computationally expensive procedure, it is only performed at some landmarks on the shape.

Belongie *et al.*, in [2], deal with the permutation π as an instance of the weighted bipartite matching problem, which can be solved in $O(M^3)$ time using the Hungarian method [12], where M is the number of landmark points. Bipartite matching is a general solution for the optimal matching of two *sets* of points, since it minimizes the total matching cost. However, contours are ordered *sequences* of points and it is natural to match them by means of DTW.

3 Dynamic Time Warping

An *alignment* between two sequences $A = a_0a_1 \dots a_{m-1}$ and $B = b_0b_1 \dots b_{n-1}$ is a sequence of pairs $(i_0, j_0), (i_1, j_1), \dots, (i_{k-1}, j_{k-1})$ such that (a) $0 \leq i_\ell < m$ and $0 \leq j_\ell < n$; (b) $0 \leq i_{\ell+1} - i_\ell \leq 1$ and $0 \leq j_{\ell+1} - j_\ell \leq 1$; and (c) $(i_\ell, j_\ell) \neq (i_{\ell+1}, j_{\ell+1})$. The pair (i_ℓ, j_ℓ) is said to *align* a_{i_ℓ} with b_{j_ℓ} . Each pair is weighted by means of a local dissimilarity function $\gamma : \Sigma \times \Sigma \rightarrow \mathbb{R}^{\geq 0}$ and the

weight of an alignment is $\sum_{0 \leq \ell < k} \gamma(a_{i_\ell}, b_{j_\ell})$. An alignment between A and B is optimal if its weight is minimum among that of all possible alignments. The DTW dissimilarity measure between A and B will be denoted with $DTW(A, B)$ and is defined as the weight of the optimal alignment between both sequences. The DTW dissimilarity can be computed as $DTW(A, B) = d(m - 1, n - 1)$, where d is this recurrence:

$$d(i, j) = \begin{cases} \gamma(a_0, b_0), & \text{if } i = j = 0; \\ d(i - 1, j) + \gamma(a_i, b_0), & \text{if } i > 0 \text{ and } j = 0; \\ d(i, j - 1) + \gamma(a_0, b_j), & \text{if } i = 0 \text{ and } j > 0; \\ \min \left\{ \begin{array}{l} d(i - 1, j - 1), \\ d(i - 1, j), \\ d(i, j - 1) \end{array} \right\} + \gamma(a_i, b_j), & \text{if } i > 0 \text{ and } j > 0. \end{cases} \quad (1)$$

This equation formulates the $DTW(A, B)$ computation problem as a shortest path problem in the so-called *warping graph*, an array of nodes (i, j) , where $0 \leq i < m$ and $0 \leq j < n$, connected by horizontal, vertical and diagonal arcs (see Fig. 2). All arcs incident on the same node (i, j) are weighted with $\gamma(a_i, b_j)$. Each path from $(0, 0)$ to $(m - 1, n - 1)$ defines an alignment between A and B containing a pair (i, j) for each traversed node (i, j) . The weight of a path (and of its corresponding alignment) is the sum of the weights of its arcs. Since the warping graph is acyclic, the optimal path can be computed by Dynamic Programming in $O(mn)$ time.

Sequences of shape contexts descriptions for contours provide invariance to translation, scale and rotation. However, there is still a fundamental problem when trying to compare sequences of shape contexts with DTW: these sequences are cyclic sequences, i.e., they have no defined starting point. Invariance to translation, rotation, scale and starting point in order to be able to apply DTW comparison has been tried to obtain with a different approach in the so-called WARP system.

4 Comparison of Canonical Sequences of Points by Means of DTW: The WARP System

DTW does not lead to good dissimilarity measures when the starting point of a shape is undefined and, thus, arbitrarily chosen. The WARP images retrieval system [1] is based on the DTW-based comparison of compact, normalized signatures of shapes (using as a local dissimilarity function, γ , the Euclidean distance between points). These signatures are obtained by applying the Inverse Discrete Fourier Transform (IDFT) to the shape's Fourier Descriptors (FDs) after a normalization procedure.

Let $A = a_0 a_1 \dots a_{m-1}$ be a sequence of complex numbers where each element denotes a point in the complex plane (thus defining a 2D point). The Discrete Fourier Transform (DFT) of A is an ordered set of complex values $DFT(A) = \hat{A} = (\hat{a}_{-m/2}, \dots, \hat{a}_{-1}, \hat{a}_0, \hat{a}_1, \dots, \hat{a}_{m/2-1})$ where $\hat{a}_i = \sum_{0 \leq k < m} a_k e^{-j2\pi ki/m}$ and

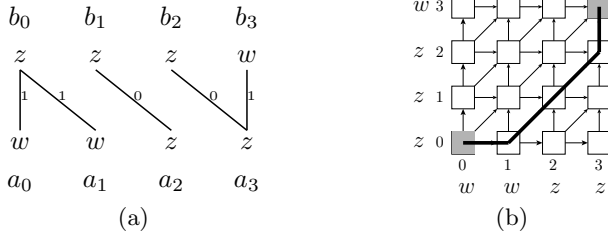


Fig. 2. (a) An optimal alignment between the sequences $A = wwzz$ and $B = zzzw$. The lines represent aligned pairs of values and are labeled with the local dissimilarity between the aligned pairs ($\gamma(w, z) = 1$). (b) Warping graph underlying the solution procedure of the recursive equation (1). The weight of all arcs arriving at node (i, j) is $\gamma(a_i, b_j)$. Each possible alignment corresponds to a path departing from the lower-left node and arriving to the upper-right node. The thick line is the optimal alignment associated to (a).

$j = \sqrt{-1}$. These coefficients are the FDs and model the contour of a shape as a composition of ellipses revolving at different frequencies [5]. The main ellipse is centered at the contour centroid, \hat{a}_0 , and translation of the contour only affects this descriptor. Scaling by a factor α scales the FDs by α . Rotating the shape by an angle θ yields a phase shift of θ in the FDs. A cyclic shift of k positions in A produces a linear phase shift of $2\pi ki/m$ in \hat{a}_i . Thus, the shape can be reconstructed to a canonical form invariant to: (i) translation, setting the \hat{a}_0 descriptor to 0; (ii) scale, dividing all the descriptors by r_1 ($\hat{a}_i = r_i e^{j\theta_i}$); (iii) rotation, subtracting $(\theta_{-1} - \theta_1)/2$ (the orientation of the basic ellipse) to each θ_i ; and (iv) starting point, adding $i(\theta_{-1} - \theta_1)/2$ to each θ_i .

Taking only $M \ll m$ low frequency components before computing the IDFT provides dimensionality and noise reduction. The WARP system only uses the $M = 32$ lower frequency FDs before computing the IDFT. The DTW computation in the WARP system is $O(M^2)$, for $M \ll m, n$.

In Section 2, the election of the starting point was the only invariance related problem that was still unsolved, since shape contexts provides invariance to translation, scale and rotation. The basic idea of the WARP system is that, after normalization, all shapes have a canonical version with a “standard” starting point and thus, are amenable to be compared by means of the DTW dissimilarity measure. But, on one hand, it should be noted that subtracting $(\theta_{-1} + \theta_1)/2$ to the orientation of all FDs only provides rotation invariance modulo π radians [5]. On the other hand, invariance is only achieved for different starting points of the same shape. Different shapes (even similar ones) may differ substantially in their normalized starting point. Fig. 3 shows three perceptually very similar figures (in fact, the second and third ones have been obtained from the first one by slightly compressing the horizontal axis) whose normalized version are significantly different in terms of orientation and starting point. This problem appears frequently in shapes whose basic ellipse is almost a circle. Invariance to starting point election should be provided by a different method.

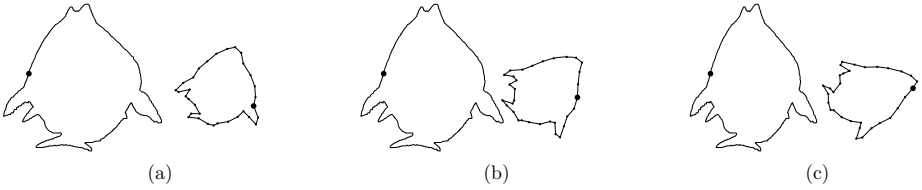


Fig. 3. (a) Original shape and its normalized version. (b) The same shape compressed in the X axis and its normalized version, which has a different rotation and starting point. (c) A bit more compressed shape and its normalized version, which is also different.

5 Cyclic Dynamic Time Warping

It is useful to consider the problem under the framework of cyclic alignments, i.e., alignments between cyclic sequences.

A cyclic sequence can be viewed as the set of sequences obtained by cyclically shifting a representative sequence (i.e., by choosing different starting points). Let $A = a_0a_1 \dots a_{m-1}$ be a sequence from an alphabet Σ and let Σ^* be the closure under concatenation of Σ . A cyclic shift σ of A is a mapping $\sigma : \Sigma^* \rightarrow \Sigma^*$ defined as $\sigma(a_0a_1 \dots a_{m-1}) = a_1a_2 \dots a_{m-1}a_0$. Let σ^k denote the composition of k cyclic shifts and let σ^0 denote the identity. Two sequences A and A' are cyclically equivalent if $A = \sigma^k(A')$ for some k . A cyclic sequence is an equivalence class $[A] = \{\sigma^k(A) : 0 \leq k < m\}$.

Let $[A] = [a_0a_1 \dots a_{m-1}]$ and $[B] = [b_0b_1 \dots b_{n-1}]$ be two cyclic sequences. A *cyclic alignment* between $[A]$ and $[B]$ is a sequence of pairs $(i_0, j_0), (i_1, j_1), \dots, (i_{k-1}, j_{k-1})$ such that, for $0 \leq \ell < k$, (a) $0 \leq i_\ell < m$ and $0 \leq j_\ell < n$; (b) $0 \leq i_{(\ell+1) \bmod m} - i_\ell \leq 1$ and $0 \leq j_{(\ell+1) \bmod n} - j_\ell \leq 1$; and (c) $(i_\ell, j_\ell) \neq (i_{(\ell+1) \bmod m}, j_{(\ell+1) \bmod n})$. The *weight* of a cyclic alignment $(i_0, j_0), (i_1, j_1), \dots, (i_{k-1}, j_{k-1})$ is defined as $\sum_{0 \leq \ell < k} \gamma(a_{i_\ell}, b_{j_\ell})$, where γ is the local dissimilarity measure. An optimal cyclic alignment is a cyclic alignment of minimum weight.

The Cyclic Dynamic Time Warping (CDTW) measure $CDTW([A], [B])$ is defined as the weight of the optimal cyclic alignment between A and B . First, we are going to show that the optimal cyclic alignment can be defined in terms of alignments between non-cyclic sequences, i.e., in terms of $DTW(\cdot, \cdot)$; then, we will present an efficient procedure to compute it¹.

Lemma 1. *If $m, n > 1$ and $(i_0, j_0), (i_1, j_1), \dots, (i_{k-1}, j_{k-1})$ is an optimal alignment between two sequences $a_0a_1 \dots a_{m-1}$ and $b_0b_1 \dots b_{n-1}$, there is at least one ℓ such that $i_\ell \neq i_{(\ell+1) \bmod m}$ and $j_\ell \neq j_{(\ell+1) \bmod n}$.*

Lemma 2. *The CDTW dissimilarity between $[A] = [a_0a_1 \dots a_{m-1}]$ and $[B] = [b_0b_1 \dots b_{n-1}]$, $CDTW([A], [B])$, can be computed as*

$$\min_{0 \leq k < m} \min_{0 \leq \ell < n} DTW(\sigma^k(A), \sigma^\ell(B)).$$

¹ The reader is addressed to [9] to obtain proofs for the following lemmas and theorem.

According to Lemma 2, the value of $CDTW([A], [B])$ can be trivially computed in $O(m^2n^2)$ time by solving mn recurrences like equation 1. Maes showed in [8] that the Cyclic Edit Distance (CED), a related dissimilarity measure, can be computed in $O(m^2n)$ time by performing cyclic shifts only on one of the sequences. This observation finally led to a $O(mn \lg m)$ time algorithm.

Is it possible to perform cyclic shifts on only one of the sequences when computing the CDTW? The answer is no: in general, $CDTW([A], [B])$ is neither $\min_{0 \leq k < m} DTW(\sigma^k(A), B)$ nor $\min_{0 \leq k < n} DTW(A, \sigma^k(B))$, as the following counter-example shows: let z and w be two elements from Σ such that $\gamma(z, w) = 1$; the value of $CDTW([z wz], [w z w])$ is 0, since $DTW(z z w, z w w) = 0$, but $DTW(z w z, w z w) = 2$ and $DTW(w z z, w z w) = DTW(z z w, w z w) = DTW(z w z, z w w) = DTW(z w z, w w z) = 1$. Therefore, an equivalent of Maes' algorithm for the CED computation cannot be directly applied to CDTW dissimilarity computation, contrarily to what is supposed in [14].

Theorem 1. *The CDTW dissimilarity between cyclic sequences $[A]$ and $[B]$ can be computed as*

$$CDTW([A], [B]) = \min_{0 \leq k < m} (\min(DTW(\sigma^k(A), B), DTW(\sigma^k(A)a_k, B))).$$

The value of $DTW(\sigma^k(A), B)$ and $DTW(\sigma^k(A)a_k, B)$, for each k , can be obtained by computing shortest paths in an *extended warping graph* similar to the extended edit graph defined by Maes [8] (see Fig. 4 (a)). Since the non-crossing property of edit paths also holds for alignment paths (see Fig. 4 (b)), the Divide-and-Conquer approach proposed by Maes can be applied to CDTW. The Divide-and-Conquer procedure is depicted in Fig. 5. It should be taken into account that, unlike in Maes' algorithm, the optimal path starting at $(k, 0)$ can finish either at node $(k + m - 1, n - 1)$ or $(k + m, n - 1)$.

6 The Proposed Method: CDTW of Shape Contexts Cyclic Sequences

We propose to describe shapes as cyclic sequences of shape contexts at certain landmarks and to compare them by means of CDTW.

Belongie *et al.*, in [2], select landmark points ensuring that these points have a certain minimum distance along the contour. It can be seen in Fig. 6 that the result by uniform spacing is rather poor and noisy: in some cases the shape can be totally deformed. The landmarks resulting from reconstructing the contour from low frequency FDs seem more suitable and detailed: noise is filtered and the original shape is better preserved. Although we do not need to take advantage of the canonical forms that the WARP system provides, it seems appropriate to use FDs to compute landmark points.

Finally, to summarize, in order to compare two shapes, the proposed method has the following stages: (i) the election of landmark points using FDs (in $O(m \log m + n \log n)$ time, where m and n are the number of points of the

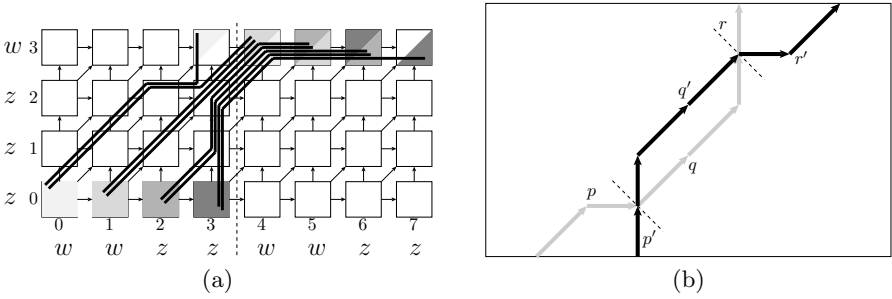


Fig. 4. (a) Extended warping graph for $A = wuzzz$ and $B = zzzzw$. $\gamma(z, w) = 1$. Arcs ending at node (i, j) are weighted $\gamma(a_i, b_j)$. The optimal alignment for $[A]$ and $[B]$ is the minimum weight path starting from any colored node in the lower row and ending at a node containing the same color in the upper row (all path candidates are shown with thick lines). (b) Optimal crossing paths can be avoided: if the weight of the subpath q is greater than the weight of the subpath q' , the black path can be improved by traversing q' instead of q .

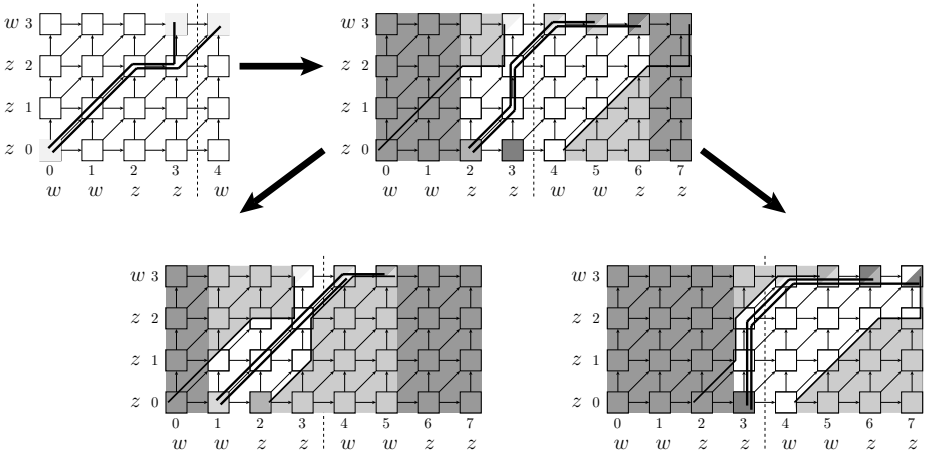


Fig. 5. Divide-and-Conquer procedure to compute the CDTW dissimilarity between the sequences of Fig. 4. First, the optimal alignment (path) between A and B and between $\sigma^0(A)a_0$ and B is computed. The first optimal path is used as a left and right frontier in the extended graph: only the white region must be explored to compute the optimal alignment between $\sigma^2(A)$ and B and between $\sigma^2(A)a_2$ and B . This idea is applied recursively to the computation of the other optimal alignments, but using also the optimal alignment between $\sigma^2(A)$ and B as a new left or right frontier.

original contours) for both shapes, (ii) the computation of the cyclic sequence of shape contexts for these points (in $O(M^2)$ time, where $M \ll m, n$ is the number of landmark points) for both shapes, and (iii) to use CDTW to measure the dissimilarity between these cyclic sequences (in $O(M^2 \log M)$ time).

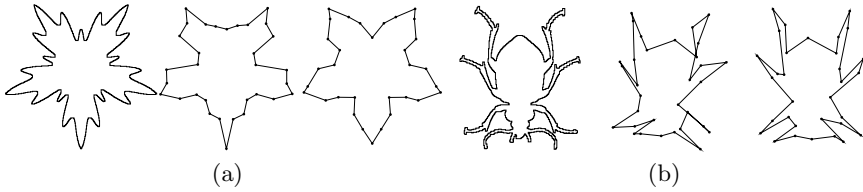


Fig. 6. Examples of the election of landmark points. For each example: the first shape is the original one, the second shape is the election of landmark points by sampling using uniform spacing (as in [2]) and the third one is the election of landmark points by FDs. These examples show how noise can deform the original shape using uniform spacing while the FDs reconstruction provides a better approximation of the shape.

7 Evaluation and Results

The proposed method was tested on the SQUID database [10] and the MPEG7 CE-Shape-1 database. For all the experiments we use shape contexts with 5 log-distance bins and 12 orientation bins (see Fig. 1 (a)).

7.1 SQUID Database

The SQUID Demo database consists of 1100 contours of marine species (see Fig. 7) and is used as a demonstration application of the Shape Queries Using Image Databases system see Fig.. The original database does not divide the contours into classes. Bartolini *et al.*, in [1], manually classified 252 images into 10 semantic categories: seahorses (5 images), seamoths (6), sharks (58), soles (52), tonguefishes (19), crustaceans (4), eels (26), u-eels (25), pipefishes (16), and rays (41). They conducted some precision (P) versus recall (R) experiments with 30 query images from the 10 semantic categories. For each query, images in the same category were considered relevant and all the others were considered irrelevant. Since the set of query images is unknown, we have run queries on the 252 labeled shapes. We use 64 landmark points.

Fig. 8 depicts the precision/recall graph for 5 retrieval procedures (precision: percentage of relevant shapes among the retrieved shapes; and recall: percentage of relevant shapes retrieved w.r.t. the relevant shapes in the database):

- CDTW-SC-FD: CDTW comparing shape contexts and sampling landmark points using low frequency FDs.
- CDTW-SC-US: comparison of shape contexts by means of the CDTW. Sampling landmark points using uniform spacing.
- WARP-SC: the WARP system using shape contexts.
- WARP: the standard WARP system [1].
- DTW-SC-US: DTW comparing sequences of shape contexts with arbitrary starting point and sampling landmark points using uniform spacing.

This graph shows how a better election of points affects the performance of the used method. It can also be seen that the method proposed in this paper (CDTW-SC-FD) significantly outperforms the other ones.

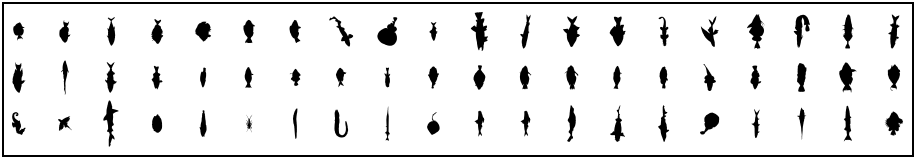


Fig. 7. Some images in SQUID database

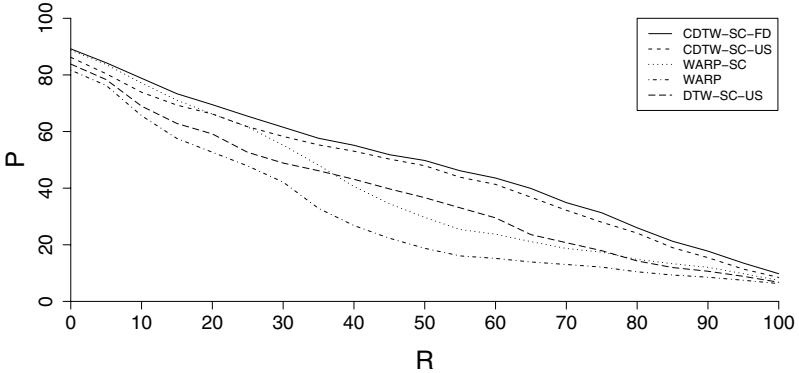


Fig. 8. Results on the SQUID Demo database. Precision P (number of relevant shapes among the retrieved ones, in %) as a function of recall R (% of relevant shapes recovered w.r.t. all relevant shapes in the database).

7.2 MPEG7 Database

The MPEG7 CE-Shape-1 database consists of 1400 shapes (see Fig. 9 (a)) divided in 70 categories, each category with 20 images [7]. Performance is measured by the so-called “bulleye test”. This is a frequently used test in shape retrieval and

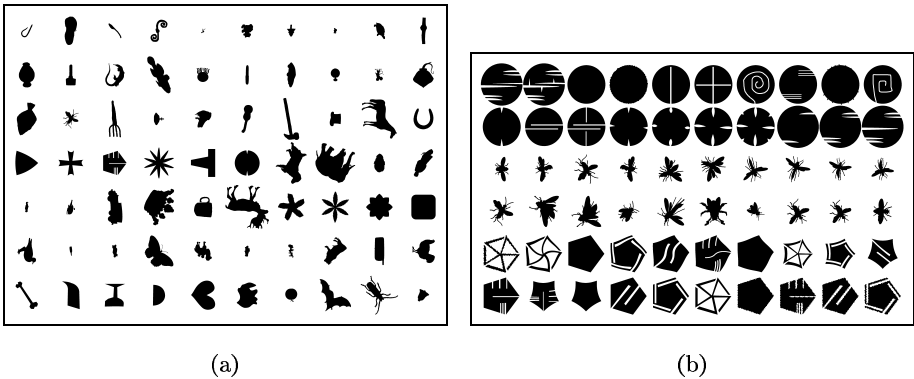


Fig. 9. (a) Some images in MPEG7 CE-Shape-1 database. (b) The three types with the lowest performance in the “bulleye test”.

| Method | Performance |
|----------------------------------|---------------|
| WARP [1] | 29.25% |
| CSS [11] | 37.72% |
| Visual Parts [7] | 38.22% |
| Shape Contexts [2] | 38.25% |
| Curve Edit Distance [14] | 39.05% |
| Distance Sets [6] | 39.18% |
| CDTW-SC-FD (32 landmark points) | 36.41% |
| CDTW-SC-FD (64 landmark points) | 39.28% |
| CDTW-SC-FD (100 landmark points) | 40.23% |

Fig. 10. Best published retrieval rates for MPEG7 Database and our approach performance (FDs, Shape Contexts and CDTW) with different number of landmark points

permits us to compare our algorithm against many of the best performing shape retrieval methods. It consists in comparing each shape to every other shapes in the database; the retrieval rate for the query object is measured by counting the number of objects from the same category which are found in the first 40 most similar shapes. The maximum number of objects from the same category is 20 (the maximum precision attainable in that experiment is 50%). The currently published best performing approaches are shown in Fig. 10. The best retrieval rate for our approach is 40.23% for 100 landmark points. Even our second best retrieval rate (with only 64 points) outperforms the best known results.

8 Discussion

We have presented a new approach to shape matching. Where contours are described by cyclic sequences of shape contexts and landmark points to compute these shape contexts are obtained from low-frequency FDs. The sequences are compared by means of CDTW in $O(M^2 \log M)$ time, where $M \ll m, n$. This approach outperforms the obtained results in previous published methods as shown in the experiments.

Acknowledgments

The authors wish to thank S. Abbasi, F. Mokhtarian, and J. Kittler for making the SQUID database publicly available and to I. Bartolini, P. Ciaccia, and M. Patella for providing their labeled version of the SQUID database.

References

1. I. Bartolini, P. Ciaccia, and M. Patella. WARP: Accurate Retrieval of Shapes Using Phase of Fourier Descriptors and Time Warping Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(1):142–147, 2005.

2. Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.
3. M. Bober. Mpeg-7 visual shape descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):716–719, 2001.
4. Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-11(6):567–585, June 1989.
5. A. Folkers and H. Samet. Content-based Image Retrieval Using Fourier Descriptors on a Logo Database. In *Proc of the 16th Int Conf on Pattern Recognition*, pages 521–524, 2002.
6. Cosmin Grigorescu and Nicolai Petkov. Distance sets for shape filters and shape recognition. *IEEE Transactions on Image Processing*, 12(10):1274–1286, 2003.
7. J. Latecki, R. Lakämper, and U. Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 424–429, 2000.
8. M. Maes. On a Cyclic String-to-String Correction Problem. *Information Processing Letters*, 35:73–78, 1990.
9. Andrés Marzal and Vicente Palazón. Dynamic time warping of cyclic strings for shape matching. In *ICAPR (2)*, volume 3687 of *Lecture Notes in Computer Science*, pages 644–652. Springer, 2005.
10. F. Mokhtarian, J. Kittler, and S. Abbasi. Shape queries using image databases. <http://www.ee.surrey.ac.uk/Research/VSSP/imagedb/demo.html>.
11. Farzin Mokhtarian, Sadegh Abbasi, and Josef Kittler. Robust and efficient shape indexing through curvature scale space. In *BMVC*, 1996.
12. Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, 1982.
13. D. Sankoff and J. Kruskal, editors. *Time warps, string edits, and macromolecules: the theory and practice of sequence comparison*. Addison-Wesley, Reading, MA, 1983.
14. T. B. Sebastian, P. N. Klein, and B. B. Kimia. On aligning curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):116–125, 2003.
15. T. Sikora. The mpeg-7 visual standard for content description – an overview. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):696–702, 2001.

Topological Active Nets for Object-Based Image Retrieval

D. García-Pérez¹, S. Berretti², A. Mosquera¹, and A. Del Bimbo²

¹ Departamento de Electrónica y Computación,
Universidade de Santiago de Compostela, Spain
davidgp@usc.es, mosquera@dec.usc.es

² Dipartimento di Sistemi e Informatica,
University of Firenze, Italy
{berretti, delbimbo}@dsi.unifi.it

Abstract. Extraction of relevant image objects and their matching for retrieval applications is proposed in this paper. Objects are represented by using a two dimensional deformable structure, referred to as *active net*, capable to adapt to relevant image regions according to chromatic and edge information. In particular, an extension of the active nets has been defined which permits the nets to break themselves, thus increasing their capability to adapt to objects with complex topological structure (e.g., objects with holes). The resulting representation allows a joint description of color, shape and structural information of extracted objects. A similarity measure between active nets is also defined and validated in a set of retrieval experiments on the ETH-80 objects database.

1 Introduction

Effective access to modern archives of digital images requires that conventional searching techniques based on textual keywords be extended by content-based queries addressing visual features of searched data. To this end, a number of models have been experimented which permit to represent and compare images in terms of quantitative indexes of visual features. In particular, different techniques have been identified and experimented to represent content of single images according to low-level features, such as color, texture, shape and structure, intermediate-level features of saliency and spatial relationships, or high-level traits modeling the semantics of image content [1,2]. In so doing, extracted features may either refer to the overall image (e.g., a color histogram), or to any subset of pixels constituting a spatial entity with some visual cohesion in the user perception (e.g., an object).

Among these approaches, image representations based on chromatic indexes have been largely used for general purpose image retrieval systems [3,4], as well as for object based search partially robust to changes in object shape and pose [5]. This mainly depends on the capability of color-based models in combining robustness of automatic construction with a relative perceptual significance of

the models. However, such approaches are not appropriate for precise retrieval, accounting for perceptual details in the image. More suited to this end are region based solutions. In fact, recently much research has focused on region based techniques that allow the user to specify a particular region of an image and search for images containing similar regions [6]. However, most existing region or object-based systems rely on color segmentation [7,8], causing these systems to fail when accurate segmentation is not possible. As opposed to color information, other retrieval schemes are entirely based on shape content. Most of the work on region-shape recognition relies on matching sets of local image features (e.g., edges, lines and corners), usually through statistical analysis which disregard relational information among extracted features. Most of these methods have been proved to be adequate only for simple, flat and man-made objects, but shape features alone are rarely adequate to discriminate objects for the purpose of object based retrieval. Only few approaches have tried to conjugate color and shape information to improve the significance of object representations. For example, in [9], color and shape invariants are combined into a histogram based representation which basically relies on the invariant properties of sets of points in the image. However, many of the current solutions suffer from the difficulty in extracting effective object representations capable to jointly capture color, shape and structural information of image objects.

In this paper, we propose a descriptor modeled through a graph which accounts for structural elements and color of regions (objects) of interest in an image. This graph directly corresponds to an elastic structure (*active net*) which through a deformation process is used to separate regions from the background. In particular, due to their deformable structure, active nets can adapt to the borders and internal part of a region encoding, at the same time, information on color, shape and spatial structure of the region. The use of an illumination invariant color space for the detection of region borders, makes the model partially robust also to changes in illumination conditions. The basic structure and the deformation process of active nets have been extended to make a net capable to break the edges connecting its nodes. This permits the net to better adapt to regions of complex shape or regions with complex topology (e.g., regions with holes). Once the net is adapted to a region it is transformed to a graph accounting for the region chromatic content of the nodes, and for their relative distance and spatial position. Finally, based on a similarity measure defined between graph representations, graph models are compared to support region-based retrieval.

The rest of the paper is organized in three Sections and a Conclusion. In Sect.2, the active net model is defined. In particular, the dynamic adaptation of active nets to relevant image objects based on color and edge image information is discussed. An extension of the active nets model, which allows the net to break itself, is also proposed in order to make the net able to adapt to objects with complex topological structures. In Sect.3, the model is cast to a graph representation in order to support effective and efficient comparison between two nets. Retrieval experiments on the ETH-80 object database are reported in Sect.4. Finally, conclusions and future research directions are outlined in Sect.5.

2 Modeling Image Content by Active Nets

An active net is a discrete implementation of an elastic sheet [10]. In parametric form, it can be defined as $u(r, s) = (x(r, s), y(r, s))$, where $(r, s) \in ([0, 1] \times [0, 1])$. The domains of parameters (r, s) are discretized to a regular grid of *nodes* defined by the internode spacing (k, l) . This parametrization defines a two dimensional net that acts in the two-dimensional space of an image. The net can deform under the control of the following energy function:

$$E(u) = \sum_{(r,s)} (E_{int}(u(r, s)) + E_{ext}(u(r, s))) \quad (1)$$

The internal energy of the net E_{int} , controls the shape and structure of the net and is defined as:

$$E_{int}(u(r, s)) = \alpha \left(|u_r(r, s)|^2 + |u_s(r, s)|^2 \right) + \beta \left(|u_{rr}(r, s)|^2 + 2|u_{rs}(r, s)|^2 + |u_{ss}(r, s)|^2 \right) \quad (2)$$

where the subscripts indicate the orthogonal partial derivatives, and α and β are coefficients controlling the first and second order smoothness of the net. In particular, the first derivatives make the net contract and the second derivatives enforce smoothness and rigidity of the net. In our specific setting, better results have been obtained for $\alpha = 3.0$ and $\beta = 0.01$ ¹.

The external energy of the net E_{ext} , accounts for external forces acting on the net, and is defined as:

$$E_{ext}(u(r, s)) = f[I(u(r, s))] \quad (3)$$

where f is a general function of the properties of the image $I(u)$. The objective is to find a function f that makes the nodes of the net to be attracted to significant zones of an image according to an energy minimization process (*fitting*). To this end a greedy algorithm is used that improves the convergence of the fitting process and makes it more robust to image noise [10].

2.1 Image Attractors for the Active Net

Active nets are used to represent relevant zones of an image due to their capability to adapt to contour and capture internal information of image regions.

¹ α and β have been selected by evaluating the fitting results of active nets on a set of image objects. For α , we found that values between 1 and 2 make the net very rigid. Values between 2 and 10 provide similar results, while values greater than 10 make the internal energy very high, so that the net collapses over itself. For β , values between 0.01 and 0.5 give the net enough elasticity to get a good deformation over an object. Values greater than 1, make the net more sensible to the presence of noise thus causing possible undesired deformations.

However, this requires that relevant image features be extracted and used as attractors for the nodes of the net. To this end, nodes are divided into *external* nodes (i.e., nodes belonging to the border of the net $u(r, s) : r, s = 0, 1$), and *internal* nodes (i.e., nodes that do not belong to the border of the net $u(r, s) : r, s \neq 0, 1$) as shown in Fig.1.

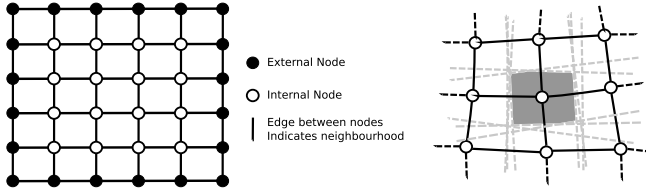


Fig. 1. Basic structure of an active net. The grey area is the influence zone of an internal node (the Voronoi region of its second level neighbor nodes).

In our approach, color information is used to drive the fitting process which controls the deformation of a net. This is obtained by creating two external energy images, one locating the borders of a relevant region, and the other describing its internal area. During the net fitting process, the external nodes are guided to the borders of a region by the internal nodes, while the internal nodes are attracted to the internal zones of a region.

The energy images attracting the internal nodes, are created by clustering the image color content using a modified version of the clustering algorithm proposed in [11]. This algorithm iteratively selects a pixel randomly from the image and assigns it to a winner cluster by calculating the Euclidian distance between the color of the pixel and the centroid of the cluster (both expressed in the $L^*a^*b^*$ color space). While in the original formulation the maximum number of clusters is fixed a priori, and the creation of new clusters depends on the number of elements in the existing clusters, here this number can adaptively change on the basis of the cluster color variance and the median color of the image. Fig.2(b) shows the two energy images identified for the image in Fig.2(a).

Positioning of external nodes is guided by energy images associated to the border of an object. These energy images are based on the color edge detector proposed in [5]. It uses a color space defined by the relation of the R, G, B color components existing between two neighbor pixels of an image. This results in a high robustness to illumination changes of image objects (Fig.2(c)). Unlike internal nodes, external nodes are repelled by this energy. The motivation for this, is that the external nodes have a low external energy where there is no border, and a high energy value near to or in the border of an object. In this way, external nodes can move in zones of low energy but cannot cross the border of a region in that this would increase their energy. The result is that external nodes anchor around the edges of the region, while internal nodes take position inside of the region. In so doing, borders of an object act like a barrier for the external nodes and anchor their position.

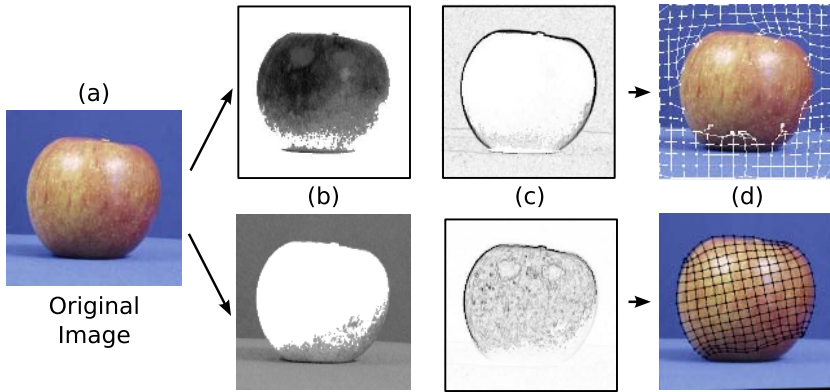


Fig. 2. The fitting process of an active net. The clustering algorithm finds two color clusters, one for the background (white areas upper image in (b)), and one for the apple (white areas lower image in (b)). White pixels are associated with the lowest energy, thus resulting the best attractors for the internal nodes of the net. In (c), the color ratios algorithm provides the color edges of these two regions, and the black pixels indicate energy barriers that the external nodes of an active net should not cross. An active net is fitted to each cluster (d). As final result, the image is represented by two nets modeling the background and the apple, respectively.

According to this modeling approach, the equation for the external energy is given by:

$$E_{ext}(v(r, s)) = \begin{cases} \gamma f(I_{cluster}(r, s)) & \text{Internal nodes} \\ \delta f(I_{border}(r, s)) & \text{External nodes} \end{cases} \quad (4)$$

where γ and δ are constants that ponderate the effect of each external energy component. $f(I_{cluster}(r, s))$ depends on the intensity of color cluster image, while $f(I_{border}(r, s))$ depends on the intensity of the color border image.

2.2 Topological Analysis

According to the active deformation process, a net can adaptively reshape to a region of interest. However, due to the physical properties of a net, the precision of the fitting process can be hampered by two main problems.

First, if an object has a complex shape (i.e., the legs of the horse in Fig.3(b)), the net can have difficulties to adapt its borders to the shape of the object. To make a perfect fit, the net needs to increase the distance between its nodes, but this increases its internal energy value (in that the internal energy makes the net contract over itself). To solve this problem, a rupture algorithm that breaks links (*edges*) between external nodes has been developed.

The rupture algorithm only acts over the external nodes and is activated once the net has finished its fitting process. At this point, the algorithm looks for *candidate* external nodes for which an edge can be broken. When an edge is

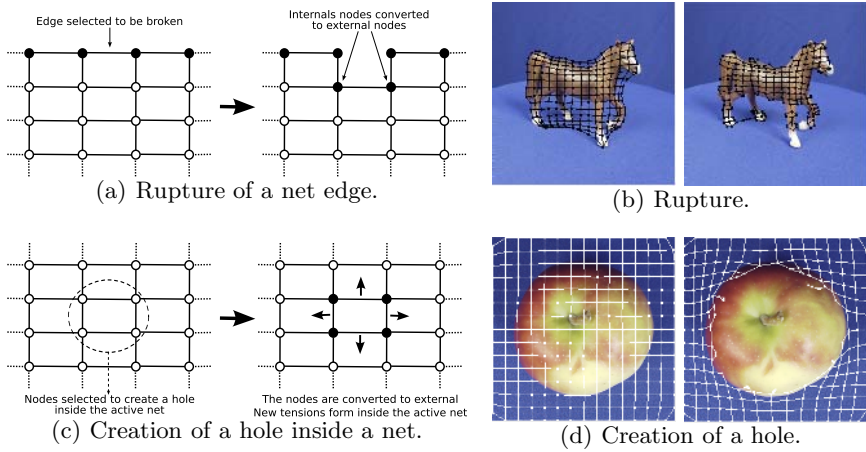


Fig. 3. Typical problems in the adjusting process of an active net. The leftmost image of Fig.3(b) shows the problem of adapting a net around the legs of a horse. If net rupture is allowed (rightmost image), the net can adapt itself to the contour of the legs and get a better result. The problem of creating a hole inside of a net is shown in Fig.3(d). In this case, the active net is attracted to the blue background of the image and, in the image on the left, a lot of nodes cannot move outside the apple because an equilibrium between its internal and external nodes is reached. If a hole is created, rightmost image, the net can avoid this problem getting a better fit to the image background.

broken, two internal nodes are converted to external nodes (Fig.3(a)) and they start to be under the influence of $\delta f(I_{border}(r, s))$ instead of $\gamma f(I_{cluster}(r, s))$. After an edge is broken, the net tries to reach a new stable position and the cycle restarts until no more edges are broken. A particular case occurs for the external nodes located at the corners of the net. In fact, a corner node is not allowed to break one of its edges thus avoiding the creation of a thread in the net structure.

The distance of an external node to the borders of a region of interest is considered to identify *candidate* external nodes. In particular, only external nodes that are far from the borders of the object are taken into consideration. The external node with the greatest distance from the object boundary is selected. If its neighbors nodes are also candidates to rupture, the two neighbors with the highest distance are selected and the edge between them is removed.

A second difficulty, can occur when an active net tries to fit to an object with holes inside (Fig.3(d)). In order to manage such objects, the fitting process is modified by searching for holes inside of objects. In particular, this is obtained by considering the external energy of the internal nodes ($\gamma f(I_{cluster}(r, s))$). In fact, internal nodes in the hole areas of a region, have a high energy value compared with the energy of well positioned nodes (energy value near or equal to 0). This selection is automatically done by constructing an external energy histogram for the internal nodes. A histogram thresholding is used to select a value T , that

distinguishes between nodes which are in a bad or in a good position (Fig.3(c)). If all the nodes have a similar energy, they are all considered in good position. Otherwise, the algorithm looks for the node with the highest energy value and checks if this node has four neighbors inside the hole too. If this condition is not fulfilled, the algorithm searches for the next node inside the hole with the highest external energy value. Once four candidates internal nodes are found, these are converted to external nodes. Combining the internal energy, which makes the active net to contract over itself, and the rupture algorithm, all the nodes can escape the hole area of the object (Fig.3(d)).

2.3 Fitting Process of an Active Net

In summary, the fitting process follows the following steps. First, the energy images for a given picture are computed. For each color selected by the clustering algorithm, an active net is created. Then, each net searches for the relevant zones of an image (Fig.4(b)). The main orientation of each zone is evaluated by computing its main geometric moment. In the second step, the region is rotated according to the orientation angle and a second net is created and initialized on the region minimum embedding rectangle (Fig.4(c)). This new net adapts itself over this particular area of the image (Fig.4(d)). In this way, the final configuration of the net is independent from the initial position and orientation of the object in the image. At the end of the second training process, the rupture and hole detection algorithms run modifying the topology of the net if necessary (Fig.4(d)).

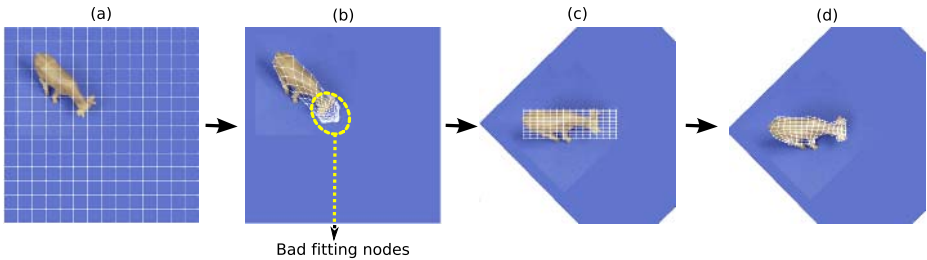


Fig. 4. The fitting process of an active net. (a) In the first step, the active net starts with its nodes equally distributed over all the image. (b) Since the region of interest is in one corner of the image, some nodes reach it before other nodes of the net. As a consequence, some nodes take a good position over the object, while other nodes remain out of it. (c) To solve the previous problem, a second net is initialized on the minimum embedding rectangle of the region selected by the previous net. At the same time, the image is rotated according to the principal moments of the object. In this way, the net is more robust to rotations of the object. (d) In the second fitting process, all the nodes find the object at the same time, allowing a better distribution of the nodes according to the external energies.

3 Similarity Model for Topological Active Nets

According to the modeling approach of Sect.2, an image is represented by a set of active nets capturing the relevant objects in the image. In the perspective to compare objects for retrieval purposes, a distance measure between two active nets has been defined. To this end, an active net is regarded as an attributed relational graph so that graph properties can be usefully exploited to enhance the net representation. In this way, the comparison between two nets is reduced to the problem of matching their corresponding graphs.

Given an active net $u(r, s)$, it is cast to a graph G by mapping n nodes of the net to vertices of the graph, and links between nodes of the net to edges of the graph:

$$\begin{aligned}
 G &\stackrel{def}{=} \langle V, E, \Phi, \Psi \rangle, \\
 V &= \text{set of vertices} \\
 E &\subseteq V \times V = \text{set of edges} \\
 \Phi &: V \mapsto L_V, \text{ vertices labeling function} \\
 \Psi &: E \mapsto L_E, \text{ edge labeling function}
 \end{aligned}
 \tag{5}$$

where L_V and L_E are the sets of vertices and edge labels, respectively.

In our framework, active nets adapt to image regions according to the overall energy function of Eq.(1) so that nodes are constrained to some relevant point of the image (those providing the stable minimal configuration for the energy of the net). In so doing, the average color (in the $L^*a^*b^*$ color space) of the Voronoi regions surrounding the nodes in the image is used as the vertices labeling function Φ of the graph (an example is the grey region shown in Fig.1). Vertex attributes are compared by exploiting the metric properties of the $L^*a^*b^*$ color space. In particular, given two vertices v_1 and v_2 , their distance is evaluated as: $\mathcal{D}_v(v_1, v_2) = \sqrt{(L_{v_1}^* - L_{v_2}^*)^2 + (a_{v_1}^* - a_{v_2}^*)^2 + (b_{v_1}^* - b_{v_2}^*)^2}$.

In order to account for the deformation of the net with respect to its initial configuration, the normalized distance existing between two nodes n_1 and n_2 is used as edge labeling function Ψ of the edge e_{v_1, v_2} connecting the graph vertices v_1 and v_2 to which n_1 and n_2 are mapped, respectively, in the graph representation. The distance between edges e_j and e_k is defined as: $\mathcal{D}_e(e_j, e_k) = |l_{e_j} - l_{e_k}|$ being l_{e_j} and l_{e_k} the labels associated to e_j and e_k , and measuring their length.

Before to convert an active net to a graph, all the edges broken during the fitting process are reconstructed to recover the original net topology (Fig.5). In this way, edges of the net are removed during the net adaptation process to increase its capability to deform according to the shape of complex objects; then, the complete topological structure of the net is recovered by inserting the missing edges so as to include this information in the graph representation.

The comparison of the graph models of a *query net* Q and an archive *description net* D involves the association of the vertices in the query with a subset of the vertices in the description. Using an additive composition, and indicating

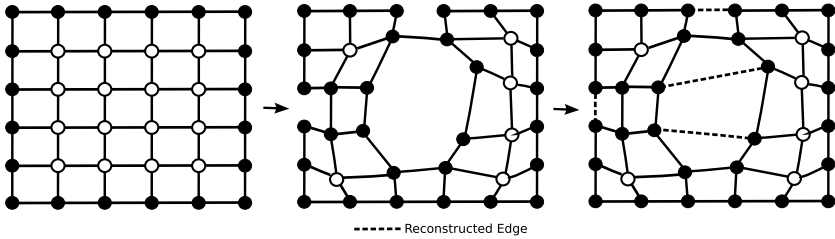


Fig. 5. After an active net is stable, the rupture and hole detection algorithm enter in action deleting edges. This allows a better fit of the active net to regions of complex shape. When this process is completed, the edges are re-inserted into the net so as to fully account for the deformation of the active net.

with Γ an injective function which associates vertices v_k in the query graph with a subset of the vertices in the description graph, this is expressed as follows:

$$\mu^\Gamma(Q, D) \stackrel{def}{=} \lambda \sum_{k=1}^{N_q} \mathcal{D}_v(v_k, \Gamma(v_k)) + (1 - \lambda) \sum_{k=1}^{N_q} \sum_{h \in C(k)} \mathcal{D}_e([v_k, v_h], [\Gamma(v_k), \Gamma(v_h)]) \tag{6}$$

where N_q is the number of vertices in the query graph Q , $C(k)$ is the set of graph vertices directly connected to the vertex k (i.e., $v_h \in C(k)$ if an edge $e_{h,k} = e_{k,h}$ connecting the vertices v_h and v_k exists in the set of graph edges E), and $\lambda \in [0, 1]$ balances the mutual relevance of edge and vertex distance (e.g., for $\lambda = 1$, the distance only accounts for the chromatic distance).

In general, given Q and D , a combinatorial number of different interpretations Γ are possible each scoring a different value of distance. The distance is thus defined as the minimum under any possible interpretation: $\mu(Q, D) = \min_\Gamma \mu^\Gamma(Q, D)$. In so doing, computation of the distance between two nets becomes an *optimal error-correcting (sub)graph isomorphism problem*, which is a NP-complete problem with exponential time solution algorithms [12].

However, due to the particular structure of active nets, it is possible to find the optimal match between their graph representations in polynomial time. In fact, we assume that nets with the same number of nodes are used to describe every object in the image database (i.e., $N_q = N_d = n \times m$, being n and m the number of rows and columns of a net, respectively). This is motivated by the fact that nets with the same number of nodes represent image objects at the same spatial resolution. A second basic assumption is that during comparison, only homologous graph vertices can match (i.e., vertices having the same position in the grid $u(r, s)$). This corresponds to assume that the injective function Γ of Eq.(6) maps any vertex v_k in the graph Q to the homologous vertex d_k in D (i.e., $\Gamma(v_k) = d_k$). In so doing, indicating with T_v and T_e the time spent in computing

the vertex and the edge distance, respectively, the complexity in matching two graphs is upper bounded by $O(mn \cdot T_v + 4mn \cdot T_e)$.

4 Experimental Results

The proposed approach for object representation and retrieval has been experimented by deriving measures of *precision* and *recall* on the *ETH-80* objects database [13]. This includes biological and artificial (human-made) objects organized in eight basic-level categories from the following superordinate areas: “fruits & vegetables” (apples, pears, tomatoes); “animals” (cows, dogs, horses); “human-made, small” (cups); “human-made, big” (cars). For each category, 10 objects that span large in-class variations are provided. Each object is represented by 41 images taken from viewpoints spaced equally over the upper viewing hemisphere. The viewing positions are obtained by subdividing the faces of an octahedron to the third recursion level. This results in a total of 3280 images. For each object, a reference segmentation mask is provided in the *ETH-80* database, for comparison and evaluation purposes. Fig.6 shows some segmentation results obtained by using the color clustering algorithm of Sect.2.1. In particular, we found that the algorithm gives perfect segmentation results in the 66% of database images.

In Fig.7(a) average precision-recall (P-R) curves are reported for active nets of different size (i.e., $n \times m$ nodes). Precision is defined for each query as the number of correctly retrieved images relative to the number of images retrieved from the database. Recall is the number of correctly retrieved images relative to the overall number of relevant images in the database to a given query. The ideal result is to get precision 1 to every value of recall.

In these experiments, a set of five queries for each of the 10 objects comprised in the eight categories are used. Results show that nets of size 5×5 or 10×10 are not able to capture enough details of the objects in comparison with net of size 15×15 . In our experiments this size resulted as a reasonable tradeoff between the effectiveness of retrieval and the efficiency of the match in that further increasing the size of the nets has not provided significant improvements in the P-R curves.

Fig.7(b) compares the average P-R curves for active nets of size 15×15 , and for three different object categories (namely, horses, cups and tomatoes). Five different queries are used for each category. The better results which are attained

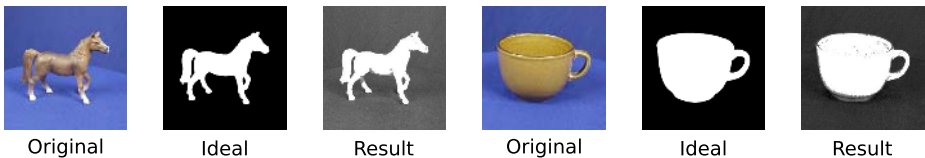
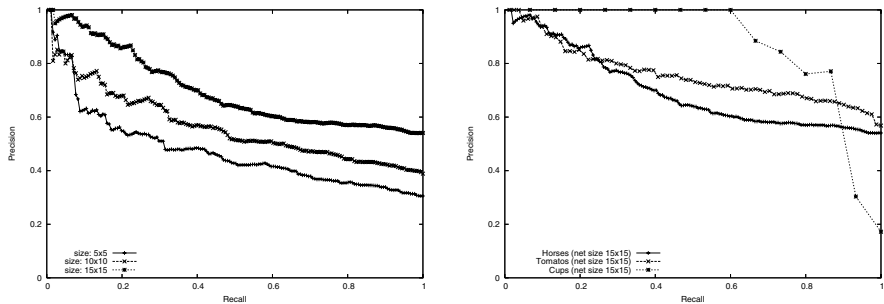


Fig. 6. Some results of the color clustering algorithm. These are compared against the ideal segmentation masks provided in the *ETH-80* object database.



(a) Average P-R curves for different net size. (b) P-R curves for three object categories.

Fig. 7. Precision-Recall results for the ETH-80 image database

by the “horses” category are mainly motivated by the stronger shape characterization that can be captured by active nets for these objects thus improving the discriminative information with respect to other object classes.

In all these experiments, relevant images to a query object have been determined through a user based evaluation where users have been asked to select from the database the images containing the most similar objects to the query according to their visual perception.

5 Conclusions

In this paper, we have proposed an original approach for the extraction and representation of significant perceptual regions (objects) of an image for object based retrieval applications. This is obtained by first extracting relevant regions and their borders on the basis of image chromatic content, then by using active nets to identify and represent image regions. In particular, we extended the basic structure and adaptation process of active nets in order to allow the rupture of edges of the net and the creation of holes inside it. In this way, we found that a better fit of the net to the shape of complex objects can be attained. Casting an active net to a graph representation allows for the embedding of additional information in the model and for an efficient matching algorithm. Experimental results validated the proposed approach on a test database.

Future work will address the inclusion of texture information in the active net model and investigate the combination of the model with indexing mechanisms capable to speed up the retrieval process in very large databases.

References

1. Del Bimbo, A.: Visual Information Retrieval. Academic Press, San Francisco (1999)
2. Smeulders, A., Worring, M., Santini, S., Gupta, A., Jain, R.: Content based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(12) (2000) 1349–1380

3. Flickner, M., Niblack, W., Sawhney, H., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D., Steele, D., Yanker, P.: Query by image and video content: the qbic system. *IEEE Computer* **28**(9) (1995) 23–32
4. Wang, J., Li, J., Wiederhold, G.: Simplicity: semantics-sensitive integrated matching for picture libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(9) (2001) 947–963
5. Gevers, T., Smeulders, A.: Color based object recognition. *Pattern Recognition* **32** (1999) 453–464
6. Hoiem, D., Sukthankar, R., Schneiderman, H., Huston, L.: Object-based image retrieval using the statistical structure of images. In: *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*. Volume 2., Washington, DC, USA (2004) 490–497
7. Xu, Y., Duygulu, P., Saber, E., Tekalp, A., Yarman-Vural, F.: Object-based image retrieval based on multi-level segmentation. In: *IEEE Conf. on Acoustics, Speech, and Signal Processing*. (2000)
8. Carson, C., Belongie, S., Greenspan, H., Malik, J.: Blobworld: Image segmentation using expectation maximization and its application to image querying. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **24**(8) (2002) 1026–1038
9. Gevers, T., Smeulders, A.: Pictoseek: Combining color and shape invariant features for image retrieval. *IEEE Transactions on Image Processing* **9**(1) (2000) 102–119
10. Ansia, F., Penedo, M., Mari, C., Lez, J., Mosquera, A.: Automatic 3d shape reconstruction of bones using active net based segmentation. In: *Proc. Int. Conf. on Pattern Recognition*, Barcelona, Spain (2000) 486–489
11. Uchiyama, T., Arbib, M.: Color image segmentation using competitive learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **10**(12) (1994) 1197–1206
12. Berretti, S., Del Bimbo, A., Vicario, E.: Efficient matching and indexing of graph models in content based retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(10) (2001) 1089–1105
13. Leibe, B., Schiele, B.: Analyzing appearance and contour based methods for object categorization. In: *Proc. Int. Conf. on Computer Vision and Pattern Recognition*, Madison, USA (2003)

Iterative 3-D Pose Correction and Content-Based Image Retrieval for Dorsal Fin Recognition^{*}

John Stewman, Kelly Debure, Scott Hale, and Adam Russell

Eckerd College, St. Petersburg FL 33712, USA

Abstract. Contour or boundary descriptors may be used in content-based image retrieval to effectively identify appropriate images when image content consists primarily of a single object of interest. The registration of object contours for the purposes of comparison is complicated when the objects of interest are characterized by open contours and when reliable feature points for contour alignment are absent. We present an application that employs an iterative approach to the alignment of open contours for the purposes of image retrieval and demonstrate its success in identifying individual bottlenose dolphins from the profiles of their dorsal fins.

1 Introduction

Marine mammalogists study numerous characteristics of their target populations, estimate population size and longevity, identify range limitations and make observations on behaviors and association patterns. Identification of individual animals is essential to these studies. For decades the identification method of choice has been photo-identification. This technique is far less invasive than traditional mark and recapture methods which used tagging or freeze branding, and a wealth of additional information is captured in the identification photographs acquired during field studies.

Bottlenose dolphins are very active, social, sometimes aggressive animals. Their dorsal fins acquire damage over time, particularly in the form of nicks, notches, scratches, and other markings. As Figure 1 shows, these features are sufficient to identify individual adult dolphins. This is similar to the use of a biometric identifier, such as a fingerprint, to recognize a person, as an alternative to token-based identification requiring ID cards or PIN numbers.

This paper focuses on the application of image processing and analysis techniques to semi-automate the process of photo-identification. The methods described should apply equally to other species of marine mammals that have dorsal fins of significant extent. The application described here has been specifically developed and tested using images of the dorsal fins of bottlenose dolphins (*Tursiops truncatus*).

^{*} The authors would like to thank the National Science Foundation for funding of this research under grant number DBI-0445126 and the Eckerd College Dolphin Project for use of their field photographs.



Fig. 1. Dolphin Dorsal Fin with Distinctive Natural Markings

2 Background

Researchers photograph dolphins in their natural surroundings and compare new photographs from each sighting against a catalog of reference images of known dolphins in order to verify the identity of the recently sighted individual. The catalogs, in either photo, slide or digital form, may be organized into categories of distinct fin shapes and each of the shape categories may be further divided into subcategories denoting regions where the predominant fin damage is present. The manual photo-identification process, although effective, is extremely time consuming and visually stressful, particularly when large collections of known dolphins are involved. In some studies, the tentative identification, or the finding of a truly new dolphin not in the catalog, must be confirmed by a second researcher, replicating the same image search. The sheer quantity of image data and the somewhat constrained properties of this class of images makes the automation of the photo-identification process an appealing and very feasible subject for content-based image retrieval. The complications associated with such a project include the difficulty of creating a reliable and intuitive method for database query and the inherent computational intensity associated with the registration and comparison of image data.

Methods for the computer assisted identification of individual marine mammals have been proposed for a variety of different species including gray seals[5], sperm whales[15], [7], dolphins[4], [10], [1], humpback whales[12], [14] and right whales[2], [6]. Identifying dolphins by the comparison of dorsal fins is similar to whale identification based upon fluke characteristics such as highly visible pigmentation patterns. While more subtle pigmentation and injury patterns such as rake marks may be used as secondary identifiers, it is the overall shape and the specific details of the damage found along the fin profile that have become the primary identifiers used within the cetacean research community to identify individual bottlenose dolphins. Although methods have been developed which use the profile of the trailing edge of whale flukes for individual identification, characterizing the outline of a dorsal fin has some unique challenges. Dolphin dorsal fin outlines have no easily determined beginning and ending points. The user must arbitrarily decide where the dorsal fin and the dolphin's back meet. Often the dorsal fin is partially obscured by waves or water if the dolphin is diving or surfacing. This uncertainty leads to significant variation in the portions

of the fin outline that are used to determine matches and creates complications in scaling and alignment of outlines.

The Digital Analysis and Recognition of Whale Images on a Network (DARWIN) system is in many aspects, a straightforward application of image processing, computer vision, and content-based image retrieval. The novelty of the DARWIN system, in its current form, is in the use of well known techniques to provide an almost completely automatic process, from image load to display of a rank ordered list of possible dolphin identities, while also providing numerous user check points and correction tools, allowing the user to enhance or even override the system's automatic processes. This produces significant time savings, and still leaves the human user in the position of final decision maker.

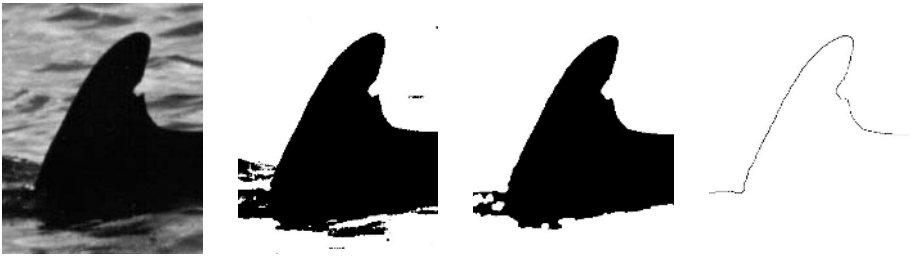


Fig. 2. Left: original image; Center left: image which results from unsupervised selection of threshold value; Center right: smoothing of the fin region and removal of small regions with morphological processing; Right: the resulting outline

3 Sketch-Based Image Query

A researcher wishing to identify an unknown dolphin must open a digital image of a dolphin dorsal fin and orient the image such that the leading edge of the dorsal fin is at the left of the image. The software automatically generates an outline of the dorsal fin. It uses an unsupervised minimum error threshold selection [9] to select an intensity threshold that will effectively segment the image into dorsal fin and background regions. It then uses morphological processing to simplify the region of interest and generate its outline, as shown in Fig. 2. If the software is unable to identify the dorsal fin region in the image, the user has an opportunity to roughly trace a general outline of the leading and trailing edges of the dorsal fin with the cursor. The trace of the outline, automatically generated or hand-sketched, initializes the positions of a series of evenly spaced points along the edge of the fin and provides a constraint space for their automated repositioning. DARWIN uses active contours [8] to move the points from their initial locations to the actual edge of the fin. The movement is controlled by several energy measures. Large magnitude values in the image gradient attract the points, while internal energy measures that maintain even point spacing and curve smoothness restrict point movement. Figure 3 illustrates the movement



Fig. 3. Left: Initial location of outline; Right: repositioned outline using active contours

of the chain of points. Once the points have been appropriately repositioned, the user has the option to manually relocate individual points along the edge of the fin. This opportunity for user interaction is necessary since photographic artifacts such as glare spots are interpreted by the program as nicks and notches and poor contrast or intensity variation on the surface of the dorsal fin may necessitate manual positioning of the outline. In addition, parameters of the active contour which encourage the convergence toward a smooth outline, may cause the automatic repositioning to miss nicks or notches with extreme angles.

Several processes, including active contours, feature point location, and mapping of unknown to reference outline, have configurable constants that are set to values based on an assumption of approximate overall fin size. The actual values of these constants and the standard fin size have been determined empirically, such that the best results are obtained in all processing of the fin outlines. All fin outlines are scaled to an approximate height of 600 units. The height measure is computed and scaling is applied before the locations of feature points are known. Therefore, the two points used to compute the height give only an approximation of the vertical distance between the dolphin's back and the tip of the fin. As Figure 4 shows, Q is the midpoint of a line segment connecting the first outline point and the last outline point, and R is the outline point with the index $\frac{n}{2}$, where n is the total number of points along the outline. Q is a reasonable estimate of the midpoint along the dolphin's back where the fin is attached, and R is a point in the vicinity of the tip of the fin. A scale factor

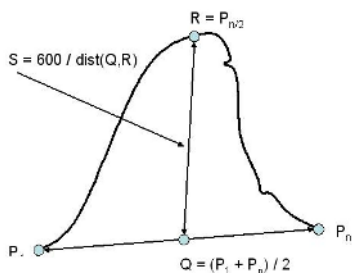


Fig. 4. Scale factor computation for the standardization of outline size

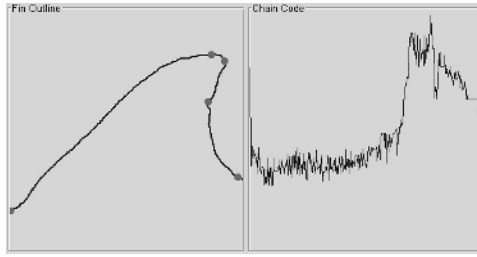


Fig. 5. Left: Dorsal fin outline with feature points; Right: Chain code of outline

$s = \frac{600.0}{\text{dist}(Q,R)}$ is multiplied times the image coordinates of each outline point in order to standardize the outline size. After the outline size is standardized, the outline (interpreted as a polyline) is traversed and a new sequence of outline points is computed with an approximate, uniform point to point separation of 3.0 units. This typically produces an outline containing 400 to 600 points. The reason the point spacing is approximate, is that the even spacing computations are done while the outline is still in an integer coordinate form, so all point coordinates are rounded to the nearest integer value.

4 Feature Point Identification

The DARWIN system represents each fin with an outline contour (a sequence of evenly spaced points approximating the its 2-dimensional placement and shape), a chain code (a one dimensional representation of the orientation of successive outline edge segments), and a set of salient features (Tip, Notch, ...). These feature points are used to establish initial alignment of two fin outlines during matching, and they are identified by an unsupervised process using the chain codes extracted from the contour. A single process converts the initial outline contour to even point spacing and creates the chain code.

Dorsal fin outlines contain only a few consistently identifiable features suitable for alignment. As illustrated in Figure 5, the DARWIN system automatically locates 1) the start of the leading edge, 2) the end of the leading edge (where the leading edge angle begins to bend toward the fin tip), 3) the tip of the dorsal fin, 4) the most prominent notch on the trailing edge, and 5) the end of the trailing edge.

Figure 5 shows the general shape of the plot of a fin's absolute chain code as that of a step edge and the location of the tip corresponds to the point of transition from low to high. DARWIN uses a quadratic spline wavelet [11] which produces large coefficients for step edge type features in the transformed signal. In order to identify the tip of the dorsal fin, a wavelet decomposition of the chain code comprising the outline is computed. At the coarsest level of analysis, the tip is identified by locating the maximum coefficient value. This initial position is tracked back through each of the finer levels of the decomposition, to more

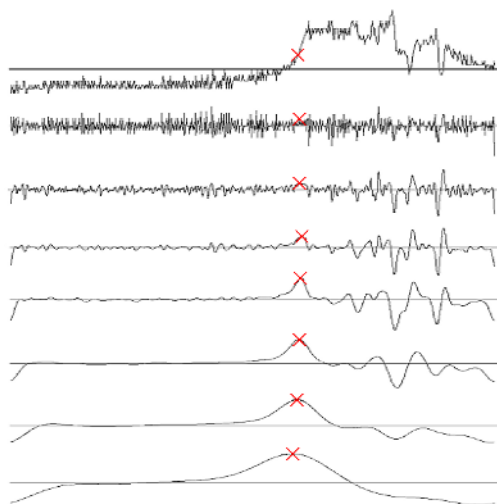


Fig. 6. Finding the dorsal fin tip from the wavelet transform of the outline. A plot of the original chain code is shown on top, with increasingly coarse details below. The position of the tip is found on the coarsest (bottom) level, and tracked back through the finer levels to obtain a more accurate location.

precisely identify the position of the tip in the chain code [11], as illustrated in Fig. 6.

Notches along the trailing edge of a dolphin dorsal fin are often the primary features used in the manual photo-identification process. Prominent notches in a fin outline appear as valleys in the plot of a fin's absolute chain code and a wavelet decomposition will produce local minima for features of this type. Since it is the most prominent notch along the trailing edge of the dorsal fin that is desired, only that portion of the outline following the fin tip is analyzed. Local minima with the largest magnitudes are identified at an intermediate transform level as candidate notches and are tracked to coarser levels. The minima that decreases most slowly in magnitude across the coarser levels is selected as the most prominent notch. As with the fin tip, the precise location of the notch is determined by back-tracking through increasingly finer levels of the decomposed signal.

To automatically identify the starting point of the dorsal fin outline at the base of the leading edge, the absolute angles between successive points along the outline are examined. A threshold angle is selected to maximize between class variance of the angles which comprise the proper leading edge of the fin and the angles which are associated with the body of the dolphin. This approach is based on Otsu's method for unsupervised threshold selection in image data [13]. Line segments at the leftmost end of the outline are discarded if their angles diverge significantly from the predominant orientation of the leading edge, accurately identifying the point at which the dorsal fin meets the dolphin's back. If this junction is occluded or missing from the initial outline, the first point of the

edge is stored as the start of the dorsal fin, indicating a truncated trace. The ending point of the leading edge is similarly identified as the point at which the angles of the line segments begin to significantly diverge from the predominant orientation of the leading edge.

5 Mapping Unknown Fin Outline to Reference Fin Outline

When photographing dolphins in the wild, there is little control over the orientation (3-D pose) of the fin that appears in an image. Several constraints are imposed, and must be assured by the user. The fin must be upright, and the dolphin must be swimming to the left. Tools are provided in the software to allow the user to flip the image as needed to assure this. Since the software is only concerned with the outline or profile of the fin, it does not matter that an image is inverted and processed as if an image of the right side of a dolphin's dorsal fin were an image of its left. These modest constraints simplify the process used for feature point location, ensuring that the trailing edge of one dolphin is compared to the trailing edge of the other. In order to correct for the differences in 3-D pose between two fin outlines (F_i and F_j), one is mapped to the other using a three point affine transformation. The three points used are selected from the previously described identifiable fin features and are guaranteed to be non-colinear. The 2-D coordinates of the six feature points, taken from the scaled, evenly-spaced fin outlines define two triangles (T_i and T_j) in the image coordinate space. A transformation matrix is found that maps T_i to T_j . This defines a skew transformation in the 2-D image plane, but it also corresponds to many 3-D affine transformations that map T_i to a 3-D pose that projects to T_j . Given several reasonable assumptions, this three point transformation gives a close enough approximation in the image plane, to a complete 3-D pose correction between the two fin outlines. These assumptions are :

- Actual 3-D pose is unimportant, only the corrected projection in the image plane is needed.
- The three feature points are reliably detectible
- Foreshortening effects are negligible
- The more similar two fin outlines are, the more accurate will be the pose correction/estimation

Once the affine transformation matrix is found, it is used to map all points on the unknown fin outline F_i to the orientation of the known fin outline F_j .

The degree to which two fin outlines, or their open polyline approximations, match is measured by a mean squared error computed using “corresponding points” along some fixed extent of the two outlines. First, the error measure is used in an iterative process to refine the alignments of the two outlines such that a better fit is obtained. The better fit is defined simply as the fit that produces the smaller mean squared error. Second, the ranking order of final match results is established by using the mean squared error for the best alignment of each

pair of reference and unknown outlines. The listing is in order of increasing mean squared error. The mean squared error measure has been satisfactory for purposes of alignment, but has been less than optimal for determination of the final rankings. Future enhancements to the software will employ additional measures of fit that allow more local, small scale analysis of particularly salient areas of the outlines. The use of more localized error measures would more closely emulate the discriminations made intuitively by biologists in their comparisons of two dorsal fins.

The original three point mapping was done once per fin pair, using the beginning of the leading edge, the tip and the position of the most significant notch as the alignment points. This produced reasonable results as long as the entire leading edge of the fin was visible in both images and was fully part of the outline, as long as the dorsal fin tip was well defined (not true if severely nicked or missing), and as long as the most significant notch was well defined and positioned at a significant distance from the tip. Full fin outlines from two images of the same dolphin, extending between correspondingly placed beginning and ending points and having a clear most significant notch, always produced excellent quality mappings. However, misplacement of the beginning of the leading edge by even relatively small amounts (10% of leading edge length) produced significant misalignments. Also, a significant notch close to the tip produced a very skinny triangle, such that only a few pixels of misplacement caused significant misalignment. Finally, it was found that the multiscale approach to notch detection was sensitive to the overall length and curvature of the trailing edge. Trailing edge traces that include part of the back could bias notch detection creating a preference for notches closer to the end of the trailing edge. The undesirable effects of highly variable user input indicated a need to move away from single applications of a pose correction mapping based upon these initial feature points.

The first significant improvement to the alignment of fin outlines and rankings of matches was through a "best of 13" approach. In this approach, the original mapping based on the feature points is applied, and the mean squared error is computed between the mapped unknown outline and the known outline. Then twelve alternative mappings are performed and error measures computed. Six of the alternatives test for shortened traces of the unknown fin outline and represent shortening of the leading edge in increments of 5%. Similarly, the other six alternatives test for shortened traces of the known fin outline. The mapping that produces the smallest mean squared error is selected to correct for orientation differences between the two fin outlines. In practice, this approach produced significantly better mapping results. Overall performance is measured by how close the correct "known" dolphin is to the top of the ranked listing of the database when queried with an "unknown." The median position of the correct identity using the "best of 13" approach was 8/200 or 4% down from the top of the list. The mean location of the correct identity was 22/200 or 11% down from the top. However, sensitivity to misplacement of the feature points at the notch and at the end of the trailing edge still produced numerous intuitively poor

mappings. The desire has always been to produce correct rankings of candidate identities, and to produce intuitively reasonable orientation corrections, even when the fin being matched is not that of the same dolphin.

Several iterative optimization approaches have been implemented in the software and tested. These all share a basic Newton-Raphson approach. Three point correspondences are used to map the unknown fin outline to the known fin outline in all cases. Shortening of leading and trailing edges, and movement of the tip, have all been done by testing feature point placement corrections of 1% of the index range of the fin leading edges and then making jumps of up to 16% based on the best correction indicated during each iteration. The maximum feature movement amount begins at 16% and decreases by half under various conditions, until either none of the 1% tests indicate improvement or the jump interval decreases to 0%. The major difference between all of the variations developed and tested is in the choice of which points to move.

Our current optimized approach allows the feature points at the beginning of the leading edge and the end of the trailing edge to move, shortening the outline. This repositioning is performed on both the unknown and reference outlines, and continues as such until the jump interval decreases to 4%. At this point, the feature point for the tip of the dorsal fin of the unknown dolphin may be shifted as well, to further reduce mean squared error between the outlines.

Since the jump intervals begin at 16% of the index range of the fin leading edges, and they monotonically decrease when a direction of improvement is found but the jump in that direction is too far to improve (decrease) the error, the optimization process is guaranteed to terminate. It is possible that the process will settle into a local minimum of the error function. This has been observed in practice. However, the use of the auto-trace feature to find the initial placement of the fin outlines appears to minimize occurrences of non-optimal minima being found.

6 Selection of Corresponding Points for Error Calculation

When comparing open curves (C_1 and C_2), or even their polyline approximations, it is difficult to calculate the location of the point on C_2 that best corresponds to a given point on C_1 . In general, the best corresponding point is the point on C_2 that is closest to the current point on C_1 , but located in the same direction of travel along C_2 as is the given point on C_1 from the previously selected pair of corresponding points. Essentially, this means that the entire sequence of points traversed along C_1 moving in the same direction has a single best set of similarly ordered corresponding points on C_2 . For the purpose of calculating a mean squared error between two mapped outlines, it is not essential that one solves exactly for the best corresponding points along the pair of curves. When alignments are near exact, error based on a reasonable point-to-point correspondence should be zero or nearly so. When alignments are not close, the outline curves are most likely from fins of different dolphins. In these cases it is acceptable to have point correspondences that are not exactly ordered in sequence on

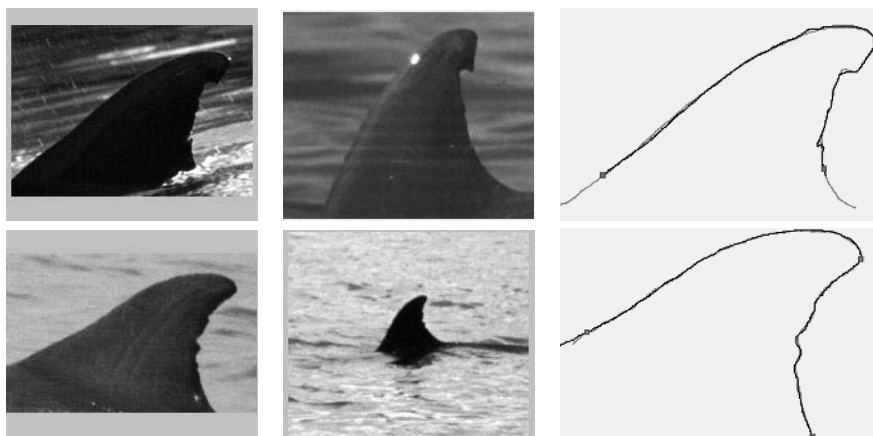


Fig. 7. For each set of images above, Left: the reference image from the database, Center: the unknown fin from a sighting, and Right: the registered outlines, unknown mapped to database

both curves. Such situations can be expected to produce larger errors in many cases, but this is quite acceptable since the outlines of different dolphins should not be given a minimal mean squared error match in any case. If the error calculation is slightly less than actual, this is acceptable as well, since the overall error magnitude of such grossly misaligned fin outlines will produce a large error and be ranked low in the final listing regardless. The point here is that an acceptable selection method should produce a sequence of corresponding points on the two curves that is “good enough” to produce a mean squared error suitable for use in fin outline registration and fin match ranking. The currently used method does just that.

It is important to note that an outline curve which is evenly spaced and similarly scaled to another outline before mapping (orientation correction) is no longer evenly spaced after mapping. Thus, the outlines cannot simply be traversed by using similar curve point indices to select individual points for comparison. Mapping is based on a 2-D skew transform that only solves for the approximate 2-D projection of a 3-D affine transformation of the mapped curve. Therefore, the sequence of line segments connecting similarly indexed points on the two curves, even after trimming extra points on curve ends, has the look of the ribs of an accordion and the calculated error, even when curves are well aligned, can be significantly higher than desired.

The current method of selecting corresponding point pairs uses an arc length based approach. The total arc length of each curve is computed between its repositioned endpoints. The ratio of the two arc lengths is used to determine the distance to traverse on the unknown curve for each unit step on the known curve. The known curve is then traversed using the indexed curve points. At

each point $P(i)$ at arc length $L(i)$ from the beginning of the curve, a point $Q(i)$ is found on the mapped unknown curve at $L(i)$ *ratio distance from the beginning of the unknown curve. The midpoint of the segment connecting $P(i)$ and $Q(i)$ is a point $M(i)$ on a medial axis of the region in the image plane bounded by the two outline curves. This sequence of medial axis points, $M(i)$ for $i = 0..m$, is then traversed. At each medial axis point, a perpendicular to the medial axis is intersected with the two fin outline curves. The two points of intersection are the i^{th} pair of “corresponding” curve points used in the error computation. The squared length of each segment between the m corresponding point pairs forms the basis of the mean squared error. In practice this approach yields closely corresponding point pairs. They are not the optimal pairs in all cases, but are “close enough” to produce meaningful error measures. Alignments based on these pairs are tight and intuitive as shown in Fig. 7.

7 Testing and Results

Preliminary testing of the software utilized a database of dorsal fin images for 200 individuals. Fifty additional images of individuals known to be contained in the database were used as “unknown” fins to test the query mechanism. Both the database images and the fifty previously identified dorsal fin images represented a range of quality and resolution and depicted fins with damage of varying location and degree. As shown in Figure 8, queries performed with forty of the fifty “unknown” fins produced rankings of the corresponding correct fin in the top 10% of the database. Of these, twenty three were ranked in the first position. On average, the optimized alignment method placed the correct fin in the top 9% of the ranked listing of database fins. The median ranking was in the top 1%. The outlines of the “unknown” fins were well aligned with the outlines of their counterparts in the database, thus the registration method

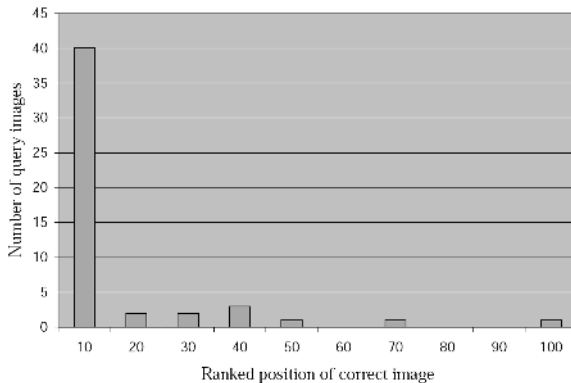


Fig. 8. Rankings of the correct fin in the database as a percentage of the total number of database entries

produced accurate alignments, even for images photographed from significantly different camera angles, as shown in Fig. 7. Also, in most cases, the outlines of the “unknown” fins were well aligned with outlines of dorsal fins of images of different individuals. Intuitive alignment is important even for the non-matched cases to give biologists the opportunity to visually compare pairs of dorsal fins removing the distortions that arise from difference in distance and camera angle. Of the test set of 50 “unknown” fins, only one of the corresponding correct fins from the database was ranked worse than 100 (random selection).

These results suggest that the registration methods presented herein show significant promise in perspective correction of dolphin dorsal fin images. The iterative nature of the alignment algorithm makes it less sensitive to the original positioning of the outline or subsequent identification of feature points. In any case, the process has the ability to improve the registration of outlines considerably. The improved registration makes fin outline comparisons easier and aids in the subsequent retrieval of appropriate images.

References

- [1] B. N. Araabi, N. Kehtarnavaz, G. Hillman, and B. Wursig, “A string matching computer-assisted system for dolphin photo-identification,” *Annals of Biomedical Engineering* **28**, pp. 1269–1279, 2000.
- [2] S. R. Burnell and D. Shanahan, “A note on a prototype system for simple computer-assisted matching of individually identified southern right whales, *Eubalaena australis*,” *Journal of Cetacean Research and Management Supplement* **2**, pp. 297–300, 2001.
- [3] J. Canny, “A computational approach to edge detection,” *IEEE Trans. on Pattern Analysis and Machine Intelligence* **8**, pp. 679–698, 1986.
- [4] R. H. Defran, G. M. Shultz, and D. W. Weller, “Technique for the photographic identification and cataloging of dorsal fins of the bottlenose dolphin (*tursiops truncatus*),” *Report to the International Whaling Commission Special Issue* **12**, pp. 53–55, 1990.
- [5] A. R. Hiby and P. Lovell, “Computer-aided matching of natural markings: A prototype system for grey seals,” *Report to the International Whaling Commission Special Issue* **12**, pp. 57–61, 1990.
- [6] A. R. Hiby and P. Lovell, “A note on an automated system for matching of callosity patterns on aerial photographs of southern right whales,” *Journal of Cetacean Research and Management Supplement* **2**, pp. 291–295, 2001.
- [7] R. Huele and H. U. de Haes, “Identification of individual sperm whale flukes by wavelet transform of the trailing edge of the flukes,” *Marine Mammal Science* **14**, pp. 143–145, January 1998.
- [8] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models,” *International Journal of Computer Vision*, pp. 321–331, 1988.
- [9] J. Kittler and J. Illingworth, “Minimum Error Thresholding,” *Pattern Recognition*, Vol. 19, No. 1, pp. 41–47, 1986.
- [10] A. Kreho, N. Kehtarnavaz, B. N. Araabi, G. Hillman, B. Wursig and D. Weller, “Assisting manual dolphin identification by computer extraction of dorsal ratio,” *Annals of Biomedical Engineering* **27**, pp. 830–838, 1999.

- [11] S. Mallat and S. Zhong, "Characterization of Signals from Multiscale Edges," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 7, pp. 710-732, July 1992.
- [12] S. A. Mizroch, J. A. Beard, and M. Lynde, "Computer-assisted photo-identification of humpback whales," *Report to the International Whaling Commission Special Issue 12*, pp. 63-70, 1990.
- [13] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-9, No. 1, pp.62-66, 1979.
- [14] R. K. Uyeyama and L. M. Herman, "Automated fluke-contour matching of humpback whales using a wavelet-based algorithm," 14th Biennial Conference on the Biology of Marine Mammals (*abstract*), Vancouver, p. 221, 2001.
- [15] H. Whitehead, "Computer-assisted individual identification of sperm whale flukes," *Report to the International Whaling Commission Special Issue 12*, pp. 71-77, 1990.

Feature Selection for Retrieval Purposes

Marco Reisert and Hans Burkhardt

University of Freiburg, Computer Science Department,
79110 Freiburg i.Br., Germany
{reisert, burkhardt}@informatik.uni-freiburg.de

Abstract. The quality of a retrieval system relies to major part on the quality of the used features. The features have to be small and compact, but also discriminative. Feature selection is one way to achieve both goals. We present a new feature selection method with the focus on retrieval purposes. The new method is based on the well known Relief algorithm. The new algorithm is shown to be superior to state-of-the-art methods both on toy problems and real-life 3D-Shape and image retrieval tasks. The algorithm is based on the intuition that distances to false detection has to be enlarged and distances to non-detected positives has to be shortened.

1 Introduction

Feature Extraction and Feature Selection are important tasks in many fields of computer vision. Given a retrieval or classification task, the goal is to find a transform from a possible high dimensional feature space to a low dimensional space, which is better suited for retrieval or learning purposes. While in Feature Extraction the transform may be nearly arbitrarily chosen, in Feature Selection one restricts to simply selecting a subset of features. In literature two types of Feature Selection algorithms are distinguished, the so-called wrapper and filter methods [1]. Wrapper methods estimate the usefulness of a subset of features by a given predictor or learning machine. Filter or Variable Ranking methods compute relevance scores for each single feature and choose the most relevant ones according to those scores. A popular example is Pearson's correlation coefficient expressing the correlation of a certain feature with the corresponding class labels. The main drawback of such simple filter methods is that they are not able to detect inter-feature-dependencies, one important example is the XOR-problem. Neither the first nor the second dimension alone helps to determine from which class an example is stemming, only both dimensions together contain enough information about the class membership. But there are methods, which may be categorized as variable ranking methods and are also able to reveal such feature dependencies, e.g. Relief [3] is one of those.

In this paper we present some fundamental modifications of the basic Relief algorithm, which lead to nice performance improvements and show that also

filter methods are able to cope with non-linear and multi-modal¹ environments. The paper is organized as follows: in Section 2 we uncover some drawbacks of the original Relief algorithm and propose the modifications. In Section 3 we examine various experiments on toy and real-world data, including image and shape retrieval examples. And finally in Section 4 we give a conclusion and outlook.

2 The Method

In this section we first give a introduction to Relief and then propose the modifications. Finally we make some short complexity considerations.

2.1 Relief

Our proposed idea has similarity with a popular feature ranking algorithm called Relief proposed in 1992 by Kira and Rendell [3]. The main idea of Relief is to compute a ranking score for every feature indicating how well this feature separates neighboring samples. The algorithm seeks for every training sample its nearest neighbor from the same class (nearest hit) and from the opposite class (nearest miss). The Relief score for a single feature is then the difference (or ratio) between the distance to the nearest miss and the distance to the nearest hit, projected on the specific feature. Several variants have been proposed. We give here a generalized and improved version of Relief which makes use of the k -nearest hits/misses and is based on the ratio:

Start with zero scores $\mathbf{h} = (0, \dots, 0)$ and $\mathbf{m} = (0, \dots, 0)^T$.

For every training sample $\mathbf{x} \in X$ do

Find k -nearest hits $H(\mathbf{x})$ and k -nearest misses $M(\mathbf{x})$ of \mathbf{x} .

Update $\mathbf{h} \mapsto \mathbf{h} + \mathbf{d}(\mathbf{y}, \mathbf{x})$ for all $\mathbf{y} \in H(\mathbf{x})$.

Update $\mathbf{m} \mapsto \mathbf{m} + \mathbf{d}(\mathbf{y}, \mathbf{x})$ for all $\mathbf{y} \in M(\mathbf{x})$.

Compute the ratios $w_i = \frac{m_i}{h_i}$.

Here $\mathbf{d}(\cdot, \cdot)$ denotes the feature-wise distances, i.e. the i -th component of $\mathbf{d}(\cdot, \cdot)$ is the distance between x_i and y_i with an appropriate metric. The w_i determines the relevance of feature i . If we want to select j features, we take the j features with the highest w_i -scores. The algorithm has received a high popularity in the past, in particular due to its easy implementation, its good performance and the ability to also work in multi-modal environments, where simple ranking criterias like Pearson's correlation coefficients or Fisher's criterion fail. But for retrieval purposes Relief shows some drawbacks. In the following section we want to reveal under what circumstances Relief has its problems and propose modifications, which circumvent those problems.

¹ In the following we call a single feature unimodal if its within-class-distribution is nearly Gaussian and form a cluster. Otherwise, if the distribution consists of two or more clusters the feature is called multi-modal.

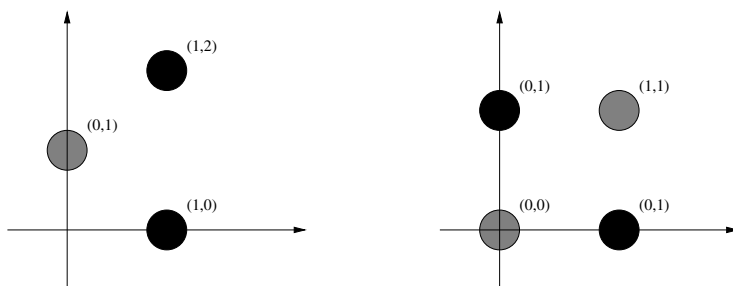


Fig. 1. a) Two class problem, where both dimensions are able to separate the classes, but the x-axis is better suited for retrieval purposes. b) The XOR problem. Only both dimensions together separate the classes.

2.2 The Method

To show this let us consider the two classes in figure 1a). Class A has two centers located at (1, 2) and (1, 0) and class B is located at (0, 1) in the Euclidean plane. Obviously, the first dimension (or feature) easily separates the two classes. And, if we restrict to the first feature, a query for (1, 2) actually gives a result from cluster (1, 2) or cluster (1, 0) of class A, this is the desired result. Though the second feature is also able to separate A and B (of course not linearly), a query for (1, 2) retrieves members of B much earlier than objects around the second cluster (1, 0) of class A. Such a behavior is clearly not desired for retrieval purposes. Although both features keep information about the class membership, one would favor the first dimension. But which dimension does Relief predict as more relevant? Since Relief only considers nearest hits, the algorithm is not able to establish the connection between the two clusters of class A, because cluster (1, 0) is farther to cluster (1, 2) than to the center of class B. In this case Relief selects the second dimension as more relevant. Hence we have to modify the algorithm such that also objects far away (in terms of the original distance measure) belonging to the query's class are taken into account. Instead of taking the nearest hits we should take the "farthest" hits, meaning those results belonging to the same class as the query but not found by the search, i.e. the false negative answers. We define $N(\mathbf{x})$ by the set of all objects, which are in the same class as \mathbf{x} but are not found under the first c objects in the results list for query \mathbf{x} , where c denotes the number of class members in the queryclass. Similar we can define the set of all false positives $P(\mathbf{x})$ by all those objects, which do not belong to the same class as \mathbf{x} but are found under the first c search results. We modify the original algorithm by using the sets $M(\mathbf{x})$ by $P(\mathbf{x})$ instead of $H(\mathbf{x})$ by $N(\mathbf{x})$:

Start with zero scores $\mathbf{n} = (0, \dots, 0)$ and $\mathbf{p} = (0, \dots, 0)^T$.

For every training sample $\mathbf{x} \in X$ do

Find false negatives $N(\mathbf{x})$ and false positives $P(\mathbf{x})$ for query \mathbf{x} .

Update $\mathbf{n} \mapsto \mathbf{n} + \mathbf{d}(\mathbf{y}, \mathbf{x}) / \|\mathbf{d}(\mathbf{y}, \mathbf{x})\|$ for all $\mathbf{y} \in N(\mathbf{x})$.

Update $\mathbf{p} \mapsto \mathbf{p} + \mathbf{d}(\mathbf{y}, \mathbf{x}) / \|\mathbf{d}(\mathbf{y}, \mathbf{x})\|$ for all $\mathbf{y} \in P(\mathbf{x})$.

Compute the ratios $w_i = \frac{p_i}{\alpha + n_i}$.

In contrast to the original Relief algorithm we introduced a normalizing step in order to not overweight the distances to the false negatives $N(\mathbf{x})$, whose distances are naturally larger than the distances to the false positives. We also introduce a balancing parameter α to adapt the influence of the false negatives \mathbf{N} .

Considering the XOR problem from figure 1b, we have a closer look at the interplay of the counterparts \mathbf{n} and \mathbf{p} . Suppose a feature vector with plenty of dimensions where the information containing features are feature one and two and the other features contain uncorrelated noise. One can imagine that both score vectors are of the same form $\mathbf{p} \approx (1, 1, \dots)$ and $\mathbf{n} \approx (1, 1, \dots)$, where the additional dimension contain mainly small noisy values. If we pay more attention to the false positive vector \mathbf{p} , i.e. we suppose large α , the relevance scores w_1 and w_2 are lifted and one can determine the first two features as the relevant ones. In the opposite case for small α we have a problem. Since \mathbf{n} stands somehow for the non-relevance of the particular feature, the first two weight scores are damped and the algorithm has difficulties to determine the first two dimensions as the most relevant. We have to make a tradeoff between two demands. The false negatives scores \mathbf{n} help us to merge different clusters like in the case in figure 1a. In other words, it detects the unimodal distributed features like Fisher's or Pearson's ranking criterias. But it also hinders the algorithm to find the relevant features in the XOR problem.

3 Experiments

In this section we present various experiments on toy and real-world data. For comparison we use three other ranking methods. The most simple method is Pearson's correlation coefficient², which is a very simple idea. We also used Fisher's coefficient, but due to the very similar behavior of both we restricted to Pearson's. As a second method we choose feature selection by Maximum Marginal Diversity (MMD) introduced by Vasconcelos [9]. The MMD idea is closely related to the infomax principle, which maximizes the mutual information between features and class labels. It was shown by Vasconcelos that for a special class of classification problems both principles are identical and in [10] he showed that it seems that vision related problems are actually a instance of those. Since the proposed feature selection method is based on Relief , of course, we also compare our experimental results to that. We use the version of Relief proposed above, where we adapted k , the number of neighbors, to the training set size. As Relief is not straightforward adaptable to multi-class problems, we used a version proposed by Kononenko. In [4] he compared two versions and showed that the so called Relief-F algorithm is superior. Instead of finding the k -nearest misses from the opposite classes, the algorithm finds the k -nearest misses for each different class and averages their contribution.

² Pearson's correlation coefficient for a feature x is given by $PCC = \frac{\sum (x_i - \bar{x})(c_i - \bar{c})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (c_i - \bar{c})^2}}$, where the index i ranges over the whole training set and c_i are the class labels.

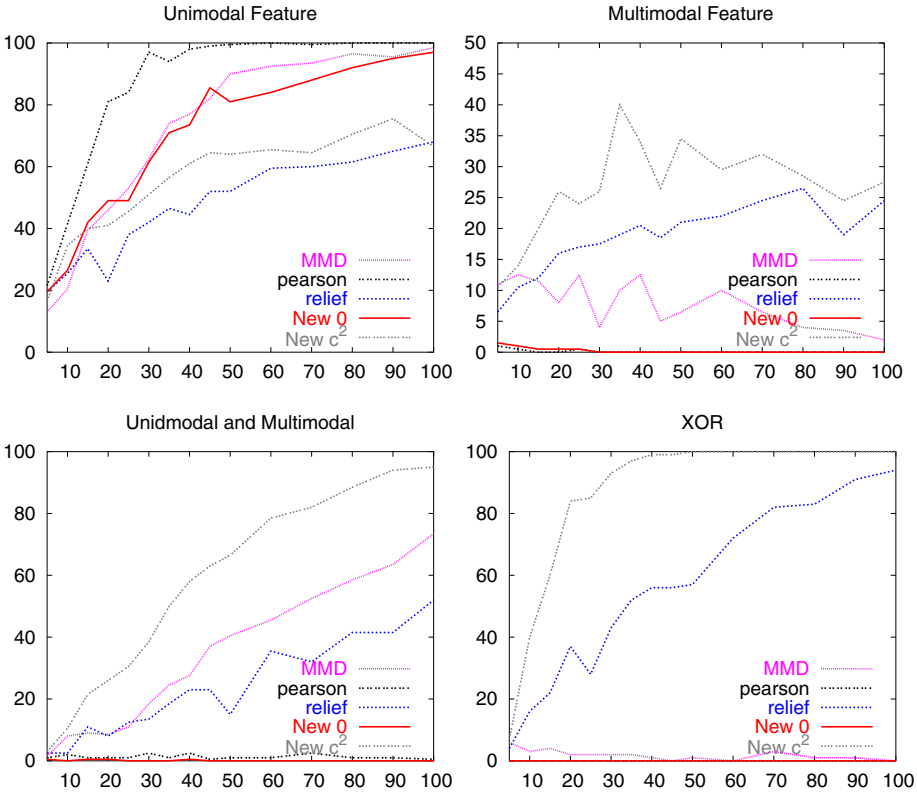


Fig. 2. Evaluation Plots for 2-class problem from figure 1a and the XOR problem from figure 1b. On the x-axis the number of training samples per class is given. The y-axis gives the percentage of correct detected features.

3.1 Toy Data

For the following three experiments we always use the Euclidean distance as the distance metric. First we consider the problem from figure 1a. Both classes occur with same probability. The feature vectors have length 20, where the first 2 features contain the information and the other 18 features are Gaussian noise with $\sigma = 1.0$. The clusters are also Gaussian distributed with the same standard deviation. We did three runs. For increasing number of training samples, we measured the percentage how often the algorithms detect the unimodal feature (x-axis in figure 1a) as most important or the multimodal feature (y-axis in figure 1a). Finally we measure how often both together are reported as the two most relevant features. To estimate the percentage each experiment is performed two hundred times. For the Relief algorithm and our new algorithm we choose $\alpha = 0$. But we also examined our algorithm with $\alpha = c^2$, where c is the number of training samples per class. It is obvious that the scaling behavior of α with respect to the number of training samples is quadratic. As already mentioned

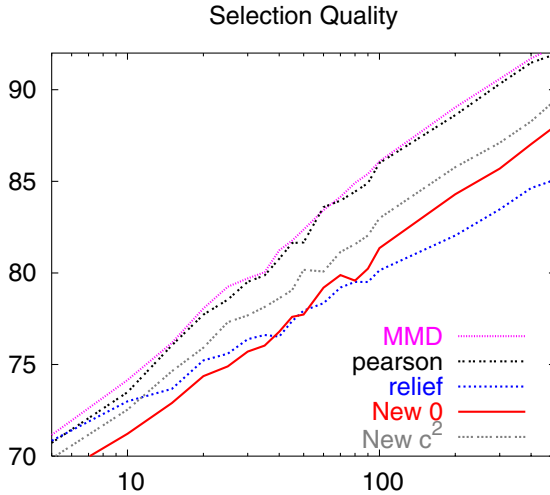


Fig. 3. Evaluation Plots for the problem introduced by Trunk. On the logarithmic x-axis the number of training samples per class is given. The y-axis gives the quality of the feature subset.

large α -values lead to a down weighting of false negatives. The results are given in figure 2. Pearson’s correlation coefficient easily detects the unimodal feature and outperforms all other methods in this case. MMD performs very similar to the new method with $\alpha = 0$. Like expected for $\alpha = c^2$ our method is not able to detect the unimodal feature with the same reliability as for $\alpha = 0$. But for $\alpha = 0$ it slightly outperforms the Relief algorithm, which approves our modifications. For the multimodal feature the situation is inverted. Pearson’s coefficient and our method for $\alpha = 0$ are totally unable to find the feature. Also MMD does not perform very well. Our method for $\alpha = c^2$ beats Relief and all other methods. In the last graph we see that its overall performance is also the best. Our method actually finds both features most. Only MMD can compete a little bit, further below Relief is still not too bad, but our method for $\alpha = 0$ and Pearson’s totally fail.

As a second experiment we consider the XOR problem. The training data is created like above by a Gaussian distributed process with means $(0,1)$, $(1,0)$ for class A and $(1,1)$, $(0,0)$ for class B and with standard deviation $\sigma = 1$. Again we measure the percentage that both features are detected as most relevant. In figure 2 the results are presented. Only Relief and our method can cope with this difficult case, but our method is much more reliable in detecting the relevant features. Only 50 training samples per class are enough to get 100% detection ratio.

As a last experiment we adopt a evaluation method introduced by Trunk [8] which was also used by Jain and Zongger [2] and Vasconcelos [9]. To evaluate the reliability of feature selection methods a 20 dimensional feature vector is

created, every feature is Gaussian distributed with $\sigma = 1$ and means $\mu_i = \frac{1}{\sqrt{i}}$ for class A and $\mu_i = -\frac{1}{\sqrt{i}}$ for class B, where the i stands for the i -th feature. The optimal j -subset is obviously the set of the first j features. The quality of the feature subset is computed by taking the number of commonalities in the optimal feature subset with the subset generated by the feature selection algorithm. This number is divided by the number of dimensions and averaged over the subset size j from 1 to 19. In figure 3 the results are given. Pearson's coefficient and MMD outperform the three other methods, both methods have also the highest slope, followed by our methods, which have both nearly the same slope. Relief seems to have the lowest slope, but for very small training set sizes it can compete with the others.

3.2 Image and Shape Retrieval

We consider three databases, two image databases and one 3D-Shape database. All three are divided in a test and training set. For Relief and our method we optimize the parameter K , and the parameter α by computing both, the ranking criteria and the performance results, on the training set. The feature rankings obtained with optimal parameters are then used for evaluation on the test set. As a performance measure we mainly use the First-Tier (1T) measure introduced in [7], since this measure focuses on the demands a user has on a retrieval system. The 1T-rate measures the percentage of objects in the query's class that appear within the top C matches, where C is the number of members in the query's class minus one. These statistics is similar to the "Bulls Eye Percentage Score".

ETH 80 database. The database was introduced by Leibe and Schiele in [5]. It consists of eight categories of high resolution color images. Each object is represented by 41 images from viewpoints spaced equally over the upper viewing hemisphere. We decided to consider the four most difficult categories, namely dogs, horses, cows and cars. The training set is of size 660, the test set is of size 1010. Sometimes it is even difficult for a human to discriminate if the viewpoint is unfortunately chosen. Since we want to evaluate feature selection algorithms, a large set of features is computed for each image. Each feature set is a four dimensional histogram of the distance between two randomly chosen points in the image, the relative orientation of the gradients at those points (the normalized dot-product, i.e. the cosine), the relative orientation between the gradient and the vector connecting both points and the absolute difference of the intensity values at the chosen points. The random points are chosen by the probability proportional to the gradient magnitude. As we deal with color images we apply this procedure for every color channel (RGB) and also crosswise, meaning that one random point is chosen from e.g. the R-channel and one from the G-channel. This results in 6 histograms for every combination (RR,GG,BB,RG,RB,GB). The overall feature size is then $\#f = \#bins * 6$, where in the experiments we chose $\#bins = 8 * 4 * 4 * 4 = 512$, resulting in $\#f = 3072$ features. The resulting histograms are all invariant to rotation and translation of the image plane and illumination changes. Since rotation invariance is not really necessary, we

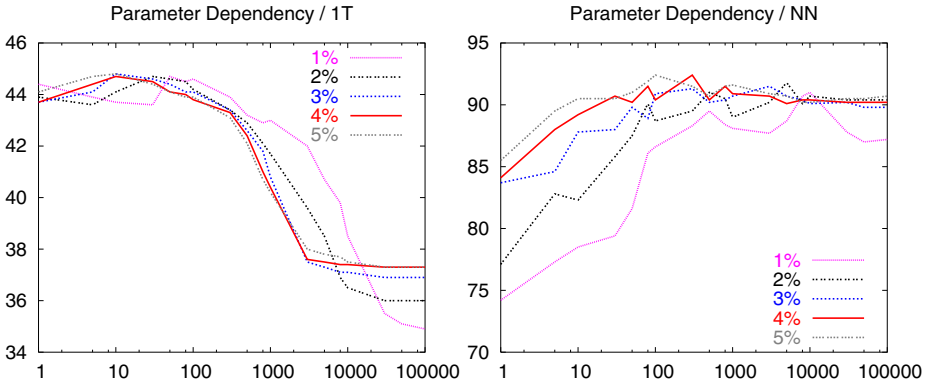


Fig. 4. Parameter Dependency on the ETH80 database with invariant features. On the x-axis the parameter α is drawn in logarithmic scale. On y-axis left the 1T-measure is given and on the right the Nearest-Neighbor Classification Accuracy (NN). The plot is evaluated for several fixed feature subset sizes given in percentage of the original size.

additionally computed two histograms, where the drawn samples are weighted by the cosine/sine of the absolute angle of the vector connecting the randomly chosen points, resulting in a feature size of 9216. We always compare the histograms using the L_1 -norm. The L_1 -norm is closely related to the Histogram-Intersection-Kernel which predestines it for our purposes. To gain an intuition how the parameter α of our method has to be chosen, we recorded the 1T-rate for different fixed feature sizes. Additionally we also consider the Nearest-Neighbor Classification Accuracy (NN), meaning the rate that an object of the query’s class appears as first. In figure 4 the corresponding plots are shown. The 1T- and NN-measure evolve in a opposite manner. While 1T has a diffuse maximum for small α and decrease with growing α , NN shows higher values for large α , i.e. for large α the precision of detecting first an object from the same class rises, but there are still lots of objects which cannot be found. We know that for large α the algorithm only considers negative samples nearby the query, positive samples far away are not considered which explains that in this case the algorithm is not able to merge different clusters belonging to the same class and hence shows a bad 1T-rate. We have to find a tradeoff between merging the intra-class clusters and obtaining a high NN-rate.

In the experiments we focus on the 1T measure and tune α to get a high 1T-rate, but still keep α as large as possible to obtain a high NN-rate as well. In other words we try to find the turning point of the 1T-rate for increasing α . All the pre-tuning is done manually for a fixed feature size of 5% from the original size.

In figure 5 the results for the invariant and variant features are shown. For the NN-rate our new algorithm is able to beat all the other three methods. For the 1T-measure our method obtains the highest scores for very small feature sizes (the first plot point is at 0.5%) and also the highest overall 1T-rates. But for

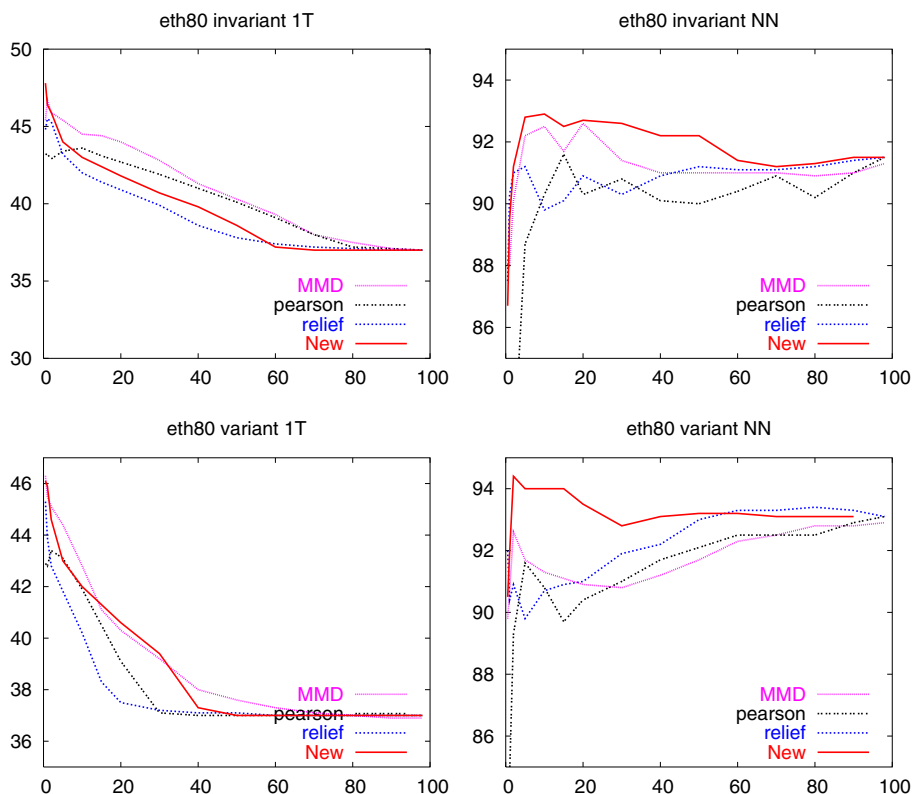


Fig. 5. Results for the ETH80 database. On the x-axis the feature size is given (in percentage of the original size). On the y-axis the 1T-rate and NN-rate respectively.

growing feature sizes the other methods, in particular MMD outperforms our algorithm with respect to the 1T-rate. It is astonishing that very small feature sizes (15 invariant features, 45 invariant features) lead to the highest 1T-rates. It seems that there is only a very small set of features which show a low intra-class variability. One can also notice that for both measures it is always worse to use all features instead of a cleverly selected subset.

Caltech database. As a second problem we consider a 2-class problem. Caltech (<http://www.robots.ox.ac.uk/vgg/data3.html>) provides several image databases. In our experiments we tried to discriminate between a set of airplanes and background (training set 687, testing set 987), and secondly between a set of motorbikes and background (training set 750, testing set 1005). We used the same variant features as above. One may argue that it is a little bit naive to use such global features for a dataset with a complex background, but since we are interested in examination of feature selection algorithms the use of such 'dirty' features is justified. Since the provided background image set is only in gray-value format, we only used the

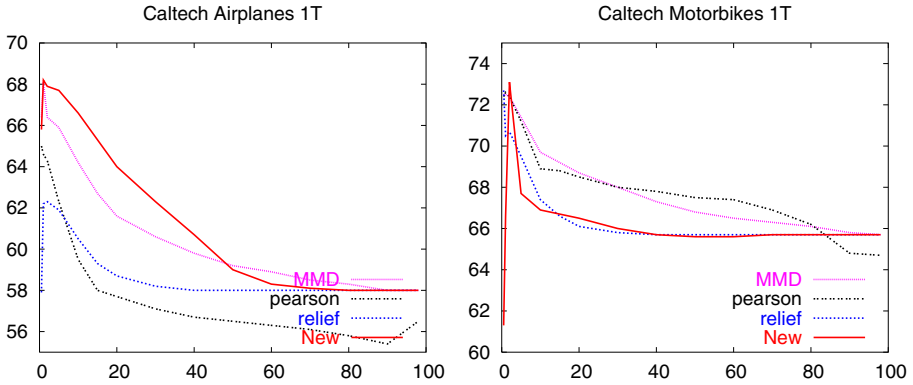


Fig. 6. Results for the Caltech database. On the x-axis the feature size is given (in percentage of the original size), on the y-axis the 1T-rate.

luminance information and neglected color. In figure 6 the results are presented. This time we only present the results for the 1T-measure. For the airplanes our method again outperforms the others. But for the motorbikes our method shows a little strange behavior. For the very small sizes the performance is very bad, for 2% it shows a very high peak and then again falls down to very low performance. Relief also performs very bad on this dataset.

Princeton Shape Benchmark (PSB). Finally we deal with the Princeton Shape Benchmark provided in [7]. The PSB consists of 1814 triangulated 3D-surface models. The database is divided into a test and training set, both of size 907. The authors also provide a hierarchical classification. We work with the three finest granularities, level 0: about 90 classes, level 1: about 40 classes, level 2: 7 classes. The used features are proposed in [6]. This time three dimensional histograms are used: the joint distribution of the distance between two equidistributed random points on the surface of the object, the relative orientation of the surface normals at those points and the relative orientation of one normal with the vector connecting both points. To keep more information we compute for each bin the angular distribution of the connecting vector falling into this bin. Since invariance to 3D-rotation is necessary a Spherical Harmonic Transform of the angular distribution is computed and the norms of the coefficient vectors are stored, which are known to be invariant to rotation. The resulting feature (SD) is of size 768. But it is also possible to preserve much more information. Instead of taking the norm for each SH-coefficient vector alone, crosswise dot-products of coefficient vectors from different bins are also invariant to rotation. Applying this procedure the feature size grows quadratically in the number of bins of the underlying histogram. The examined features (SDF) are in this case of size 6240. Again the same optimization procedure was applied like in the image retrieval case. The results for the 1T-rate are given in figure 7. Again our proposed

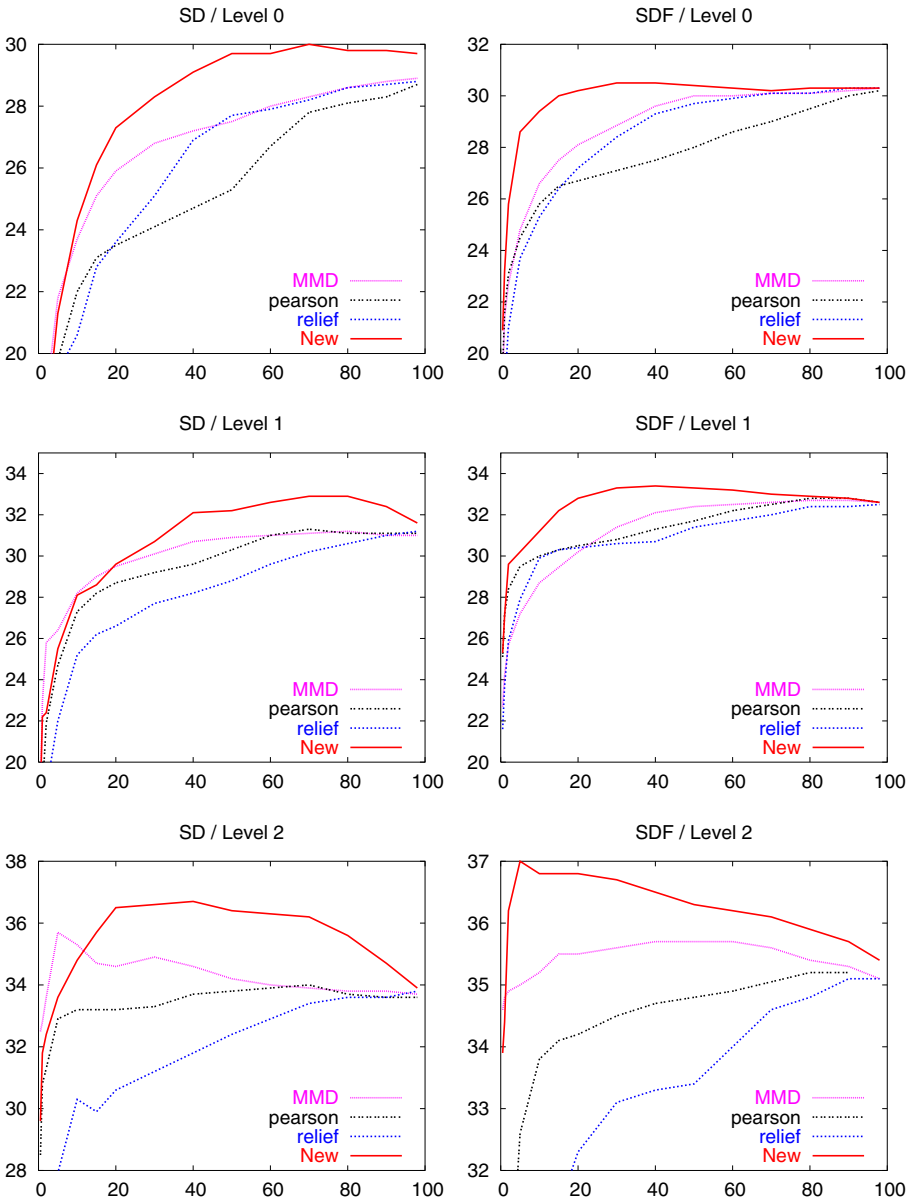


Fig. 7. Results for the Princeton Shape Benchmark. On the x-axis the feature size is given (in percentage of the original size), on the y-axis the 1T-rate.

method shows very nice results and seems to be superior in nearly all situations. Only for very small feature sizes the MMD method sometimes outperforms our algorithm, but in all cases our method achieves the highest overall 1T-rate.

4 Conclusion and Future Work

We have introduced a new feature selection algorithm, which is based on the well known Relief algorithm and is very well suited for retrieval purposes. We well motivated our modification and demonstrated that on both toy and real-world data our algorithm shows a good performance in comparison to other methods. It works quite well on a great diversity of databases.

A nice property of our algorithm is a parameter which helps us to find a tradeoff between the accuracy finding nearest neighbors from the query's class and the amount of false negatives within the first few result objects. But this parameter is also the main drawback of our algorithm, an automated estimation of this parameter with respect to the user's demands would be very helpful.

References

1. I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.
2. A. Jain and D. Zongker. Feature selection: Evaluation, application and small sample performance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(3):153–158, 1997.
3. K. Kira and L. Rendell. A practical approach to feature selection. In *Proceedings of the International Conference on Machine Learning. (Aberdeen 1992) Editors: D.Sleeman and P.Edwards, Morgan Kaufmann*, pages 249–256.
4. I. Kononenko. Estimating attributes: Analysis and extensions of relief. In *Proceedings of ECML-94, Catania, Sicily, April 1994 (F.Bergadano, L.de Readt (eds.)), Springer Verlag*, pages 171–182.
5. B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *Proceedings of CVPR 2003, Madison, Wisconsin*, 2003.
6. M. Reisert and H.Burkhardt. Second order 3d shape features: An exhaustive study. *accepted for publication in Computers and Graphics, Special Issue on Shape Reasoning and Understanding (publication date: April 2006)*, 30(2), 2006.
7. P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. In *Shape Modeling International, Genova, Italy*, 2004.
8. G. Trunk. A problem of dimensionality. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1(3):306–307, 1979.
9. N. Vasconcelos. Feature selection by maximum marginal diversity. In *Proceedings of NIPS 2002*, pages 1351–1358.
10. N. Vasconcelos. Feature selection by maximum marginal diversity: optimality and implications for visual recognition. In *Proceedings of CVPR 2003*, volume 1, pages 762–772.

DCT-Domain Image Retrieval Via Block-Edge-Patterns

K.J. Qiu¹, J. Jiang^{1,2}, G. Xiao¹, and S.Y. Irianto²

¹ Faculty of Informatics & Computing, Southwest China University, Chongqing, China
j.jiang1@bradford.ac.uk, g.xiao@swcu.edu.cn

² Department of EIMC, University of Bradford, UK

Abstract. A new algorithm for compressed image retrieval is proposed in this paper based on DCT block edge patterns. This algorithm directly extract three edge patterns from compressed image data to construct an edge pattern histogram as an indexing key to retrieve images based on their content features. Three feature-based indexing keys are described, which include: (i) the first two features are represented by 3-D and 4-D histograms respectively; and (ii) the third feature is constructed by following the spirit of run-length coding, which is performed on consecutive horizontal and vertical edges. To test and evaluate the proposed algorithms, we carried out two-stage experiments. The results show that our proposed methods are robust to color changes and varied noise. In comparison with existing representative techniques, the proposed algorithms achieves superior performances in terms of retrieval precision and processing speed.

1 Introduction

At present, many images and videos distributed on the Internet or stored inside a database are represented in compressed formats, due to the limitation of space and bandwidth. Joint Picture Expert Group (JPEG) is one of the most typical compressed standards widely used among industrial communities. As a result, methods used for indexing or retrieving images directly in compressed domain have become a recent focus of research and developments. To detect or extract the features from those compressed images, traditional approaches need to decode the images first to provide pixel domain for content feature extraction [1]. To avoid such an overhead operation and computing cost, many research efforts are directed to feature extraction in compressed domain [2-9], including texture, color, and shape etc. Representative work in such areas can be summarized as follows.

Shneier and Abdel.Mottaleb [2] calculated average value of DCT coefficients computed over a window to generate keys of JPEG images for retrieval. S. F. Chang [3] computed the statistical measures of DCT coefficients to form texture feature. Chong-Wah Ngo, Ting-chuen Pong [4] described an image-indexing algorithm via reorganization of DCT coefficients in Mandala domain [5], and representation of color, shape and texture features in compressed domain. G.C. Feng and J. Jiang [6] extracted mean and standard deviation of statistical features directly from DCT coefficients to retrieve

JPEG compressed images. Sim, Kim [7] reported fast texture description and retrieval of DCT-based compressed images. Lay, Ling [8] proposed a method using energy histograms of the low frequency DCT coefficients for retrieving images. Jianghong Liu, Haiming Gu [9] developed a method, which can retrieve images from different kinds of domains by using the wavelet coefficients. Daidi Zhong, Irek Defee [10] described a method based on histograms of quantized DCT blocks. These histograms can be optimized in order to achieve best retrieval performance by optimizing the selection of quantization factor and the number of DCT blocks under normalization of luminance.

Edge is a strong feature for characterizing an image, which can be used to construct an important clue to understand the content of images. While many methods as highlighted above extract features in DCT domain, most techniques reported in the literature extract edge features in pixel domain, including those widely reported for image segmentations, and thus existing efforts on edge-based image retrieval are limited in pixel domain. For example, Jun Weihai, Lei Guo [11] proposed an image retrieval method based on salient edges, in which Canny operator is applied to detect edge points followed by a Water-Filling technique to extract edge curves and shape features for salient edge selection. Minakshi Banerjee and Malay K. Kundu [12] presented a technique for extracting edge map of an image and formed a global feature (like fuzzy compactness) to process image content.

Recent trend on image processing is characterized by detecting edges and classifying their patterns at a block level either in pixel domain or compressed domain [18]. Kim and Lee [13] found out that the edge location is well captured by the polarities of the projection of DCT coefficients. Soltane et al. [14] suggested an adaptive edge operator selection scheme for image segmentation based on the mean, variance, and entropy of DCT coefficients. Shen and Sethi [15] further proposed a method to determine edge strength and orientation through pattern analysis of DCT coefficients. They examined the DCT coefficient patterns induced from an ideal edge model and show how the relative values or signs of different coefficients can be determined at block level. Lee et al. [16] has extracted and defined their scheme to detect scene change using direct feature extraction from MPEG compressed videos. Li et al [17] proposed an effective approach to edge classification from DCT domain. Chang et al [18] proposed a fast and systematic scheme to classify the edge orientation of each block in DCT domain. Five directional edge patterns (no edge, 0-directional edge, $\pi/4$ -directional edge, and $3\pi/4$ -directional edge) were proposed in their paper. Their algorithm is similar to the one proposed in [19].

Based on the analysis of the existing work as described above, we propose a new compressed image retrieval algorithm, in which three block-edge-patterns are extracted directly from DCT domain, which include no edge, horizontal edge, and vertical edge. We then combine the three block edge patterns to form three block edge pattern histograms (3_D histograms, 4_D histograms, Run-length histograms) to characterize the compressed image content and use them as indexing keys for image retrieval. The rest of the paper is organized into four further sections, where Section II describes how the three block edge patterns can be extracted from DCT coefficients based on a brief review of an existing work, Section III describes our proposed

algorithm to form indexing keys, including 3_D histogram, 4_D histogram, and a so-called run-length histogram. Finally, experimental results and their analysis are given in Section IV and conclusions are given in Section V.

2 Block Edge Pattern Extraction in DCT Domain

In reference [18], Chang et al proposed a technique to extract five edge patterns directly from DCT coefficients to characterize the distribution of edges inside an image. In practice, however, our observation and analysis reveal that image edge distribution can actually be characterized with three edge patterns, i.e. no edge, vertical edge and horizontal edge as shown in Figure-1. This is because that an arbitrary edge inside an image can always be approximated by many small vertical or horizontal edge elements. This is illustrated in Figure-2. As a matter of fact, if we zoom into any part of image to a certain extent, the arbitrary shaped image edge can be seen by artifacts consist of vertical and horizontal edges. As a result, such five edge patterns can be further reduced to three edge patterns, and thus further improvement can be achieved in terms of both processing speed and computing cost when the characterization of edge distribution inside an image is implemented by only three edge patterns.

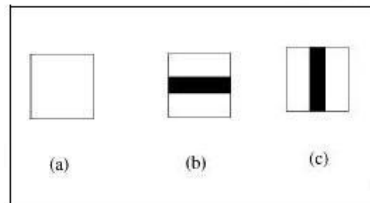


Fig. 1. Illustration of three edge patterns, where (a)=NE, (b)=HE, and (c)=VE

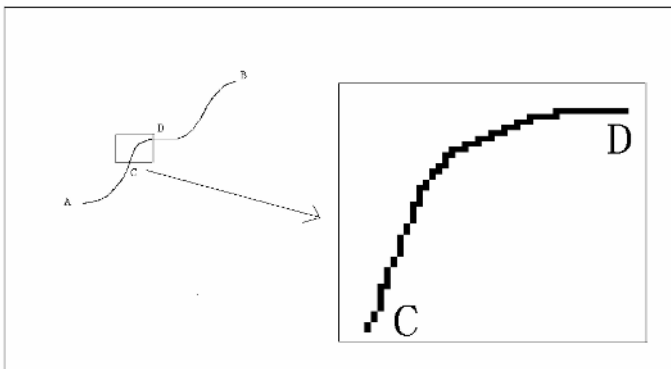


Fig. 2. Zoom-in of an arbitrary shaped edge

To support our analysis, we further compare the five edge pattern representation of two standard images, Lena and pepper, with their three edge pattern representations as shown in Figure-3 and Figure-4. From the comparisons, it can be seen that the two representations present little difference in terms of their edge distribution and content characterization. Therefore, the three edge patterns given in Figure-1 can be used to construct image content features, leading to effective image retrieval based on its content.

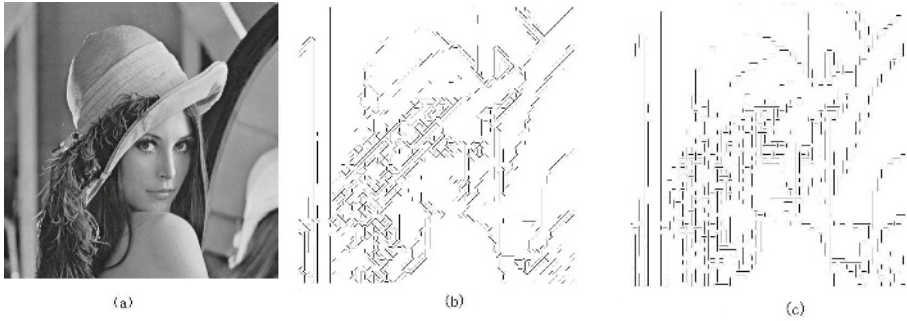


Fig. 3. Edge-pattern representation of image Lena, (b)=five pattern representation, and (c) = three pattern representation

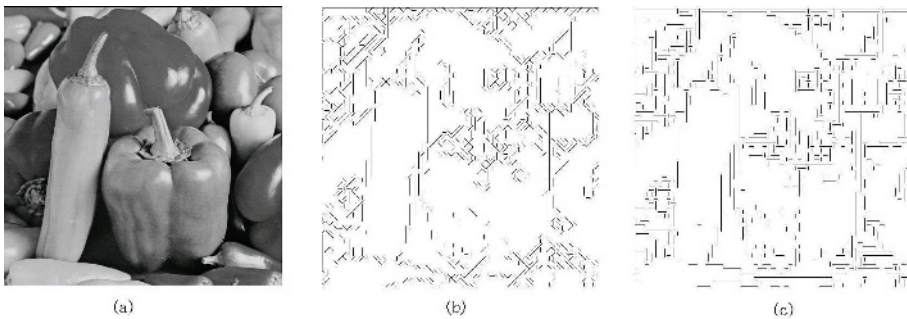


Fig. 4. Edge-pattern representation of image Pepper, (b)=five pattern representation, and (c) = three pattern representation

In 2005, Chang et al proposed a technique of extracting edge patterns from DCT coefficients in terms of 8×8 block of pixels [18]. This technique operates in compressed domain by just analyzing the value of the first four DCT coefficients and thus only limited decoding operations are required without involving the full decompression.

Given a block of 8×8 pixels, the technique divides the block into four regions, and the average pixel value for each region is represented as S_{ij} , $i, j \in [0, 3]$. In pixel domain, the edge pattern classification for the block can be conducted in terms of the four average pixel values by choosing the maximum measure out of all the measure values given in Table-1.

Table 1. Measure for Five Edge Patterns

| EdgeDirection (in radian) | Measure values |
|------------------------------|--|
| No edge (NE) | δ_{NE} (set by user) |
| 0 | $\delta_0 = \left \frac{S_{00} + S_{01}}{2} - \frac{S_{10} + S_{11}}{2} \right $ |
| $\pi/4$ | $\delta_{\pi/4} = \max \left\{ \left S_{00} - \frac{S_{01} + S_{10} + S_{11}}{3} \right , \left S_{11} - \frac{S_{00} + S_{01} + S_{10}}{3} \right \right\}$ |
| $\pi/2$ | $\delta_{\pi/2} = \left \frac{S_{00} + S_{10}}{2} - \frac{S_{01} + S_{11}}{2} \right $ |
| $3\pi/4$ | $\delta_{3\pi/4} = \max \left\{ \left S_{01} - \frac{S_{00} + S_{10} + S_{11}}{3} \right , \left S_{10} - \frac{S_{00} + S_{01} + S_{11}}{3} \right \right\}$ |

The contribution of [18] is to calculate all the above measure values in DCT domain, leading to edge block pattern classification via DCT coefficients. Since we only need the horizontal edge and vertical edge pattern as analyzed earlier for compressed image content characterization, the measure values for these two patterns can be summarized as follows according to the work reported in [18].

$$\delta_0 = 2|V| = |X(1,0)| \quad (1)$$

$$\delta_{\pi/2} = 2|H| = |X(0,1)| \quad (2)$$

Where $X(1,0)$ and $X(0,1)$ are the DCT coefficients decoded from JPEG compressed image data.

From the (1) and (2), it can be seen that the two edge pattern can be implemented very fast since only two DCT coefficients are required. As a result, the three edge pattern classification can be implemented as summarized in Figure-5, where λ is a threshold selected via empirical approaches. In our implementation, λ is selected to be 50.

3 Construction of Indexing Keys

Following extraction of block edge patterns, the remaining issue is how to construct an effective indexing key to enable content-based search via these edge patterns. Standard techniques are represented by histogram approaches, due to the fact that histograms are effective in representing random variable distributions such as colour, intensity and luminance etc.

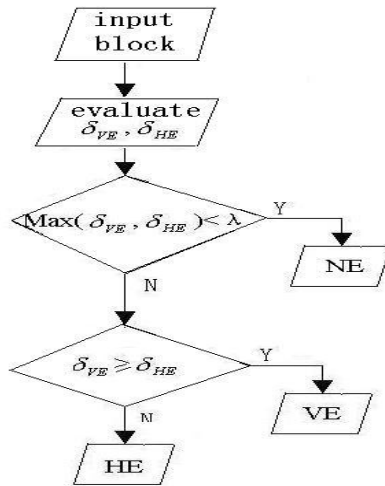


Fig. 5. Flow-chart for block edge pattern classifications (NE=no-edge, VE=vertical edge, HE=horizontal edge)

Given a block of 8x8 DCT coefficients, its edge pattern is firstly transformed into a set of three elements, $\psi = \{\#, -, |\}$, where # = no-edge, - = horizontal edge and | = vertical edge. When the histogram is constructed by calculating the number of occurrence for each edge pattern, the histogram will only have three elements, which is often referred to 1D edge-block-histogram (EBH). As such 1D edge-block-histogram has little information to reveal the image content, the occurrence of states can be implemented by considering combination of edge patterns, leading to construction of high dimensional histograms. As an example, if we examine the status of two edge patterns together, the total number of elements inside the histogram can be increased to 9 (a 2D EBH). A summary of the histogram dimension with respect to the number of it states is given in Table-II.

Table 2. Summary of histogram dimension with respect to its number of elements

| 1D EBH | 2D EBH | 3D EBH | 4D EBH | 5D EBH | 6D EBH |
|--------|--------|--------|--------|--------|--------|
| 3 | 9 | 27 | 81 | 243 | 729 |

From Table-2, it can be seen that 1D EBH and 2D EBH not only have small number of elements, but also contain little information for the location of those classified edge patterns inside the image, which are often useful to reflect the visual content. On the other hand, 5D EBH and 6D EBH incurs large number of states, which illustrate potential for high computing cost when millions of compressed images need to be indexed and searched via such keys. To this end, we focus on 3D EBH and 4D EBH in order to derive effective and efficient indexing keys.

To construct an edge block histogram based on combination of three or four edge block patterns, we define each state as shown in Figure-6.



Fig. 6. Definition of states for EBH, where (1) for 3D EBH and (2) for 4D EBH

Consequently, the edge block histogram (EBH) can be constructed as follows.

$$H_{rD} = \frac{\{C_1, C_2, \dots, C_N\}}{N} \quad (3)$$

Where $r \in [3, 4]$, C_i stands for the total number of occurrence of state- i , and N for the number of elements inside the histogram, which is 27 for 3D EBH and 81 for 4D EBH.

To fully reflect the positional information of the edge patterns inside images, we need higher dimensional histograms, yet their negative side is that the size of the histograms would be increased to a non-manageable level. To achieve an appropriate balance, we propose a variable dimensional histogram following the spirit of run-length coding in the area of lossless data compression. Specifically, we raft scan all block edge patterns and count the number of same patterns inside one single run. Every time a different edge block pattern is encountered, it is regarded a broken run and thus a new run will be started. The output code for each run can be represented as follows:

$$R_i = \{\alpha_i, \eta_i\} \quad (4)$$

Where $\alpha_i \in [\#, -, |]$, and η_i stands for the number of α_i inside this single run.

As a result, the histogram can be represented as follows:

$$H_{run} = \frac{\{C_1, C_2, \dots, C_M\}}{M} \quad (5)$$

Where C_i counts the number of the state- i occurred inside the image after the run-length coding is finished, and R_i is regarded the same state as R_j only when $\alpha_i = \alpha_j$ and $\eta_i = \eta_j$.

4 Experimental Results and Analysis

Extensive experiments are carried out to evaluate the proposed algorithm, where two important issues regarding the evaluation design are addressed. These include: (i) test data set is prepared by establishing a database of 10287 JPEG compressed images, which are classified into categories of car, people, flowers, outdoor-scenes, and animals; (ii) a benchmark is selected out of the existing technique reported in the published literature [6], which presents a fair comparability since both algorithms search and retrieve images via DCT coefficients. To calculate the similarity between images, the same distance measurement as described in [6] is adopted, which is given below.

$$d(x, q) = \sum_{j=1}^n |k_x(j) - k_q(j)| \quad (6)$$

Where n is the dimension of the key, $k_x(j)$ and $k_q(j)$ are the keys of image x and q respectively after normalization.

Two phases of experiments were carried out for benchmarking purposes. In the first phase, we choose 100 images from a database, 20 out of each category, as the query images to search the database. The retrieval performance was measured in terms of precision, which is defined below.

Let n_c and n_f be the number of correctly retrieved images and wrongly retrieved ones respectively among the first M retrievals. The precision for the query image q is defined as:

$$Precision_q = \frac{n_c}{n_c + n_f} = \frac{n_c}{M} \quad (7)$$

In our experiments, we designed M to be 12 in line with the benchmark [6]. All results of the first phase experiments are summarized in Table-3, from which it can be seen that the proposed RLEBH (run-length edge block histogram) achieves the best performance in terms of retrieval precision rate, and all the proposed block-edge-pattern histograms outperform the benchmark. Figure 7 illustrates a sample comparison for visual inspections.

Table 3. Experimental results summary for the first phase

| Average precision | Benchmark | 3DEBH | 4DEBH | RLEBH |
|-------------------|-----------|-------|-------|-------|
| Cars | 0.17 | 0.67 | 0.64 | 0.63 |
| Flowers | 0.17 | 0.25 | 0.27 | 0.28 |
| People | 0.74 | 0.74 | 0.76 | 0.83 |
| Outdoor-scene | 0.38 | 0.33 | 0.39 | 0.43 |
| Animals | 0.23 | 0.18 | 0.11 | 0.23 |
| Total-average | 0.34 | 0.40 | 0.44 | 0.49 |

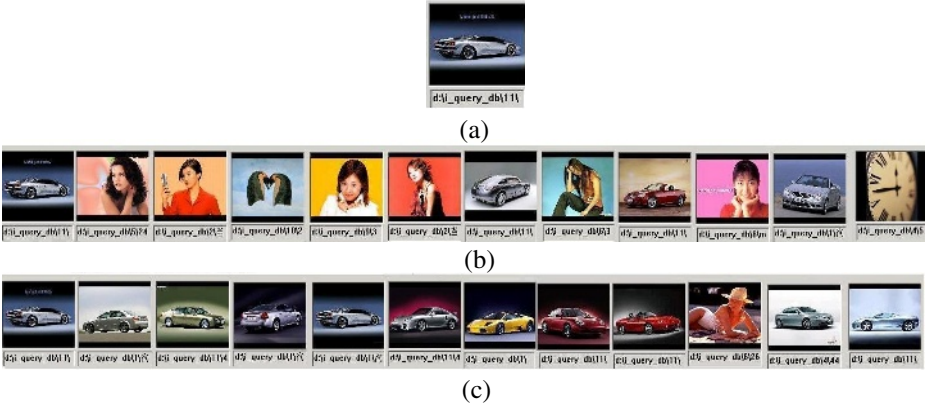


Fig. 7. Retrieval sample illustration, where (a) query image; (b) retrieved images by benchmark; (c) retrieved images by RLEBH

The second phase of experiments aims at evaluating the robustness of the proposed algorithm in retrieving images with orientation changes, such as rotation, zoom in or out, and noise affected etc. In this experiment, fifty images were chosen at random from the database. Specifically, each of the chosen images was processed as follows: clockwise rotation by 5 degrees, anti-clockwise rotation by 5 degrees, zoom out by 20%, zoom in by 20%, inverse color, horizontal reverse, vertical reverse, horizontal displacement by 10%, vertical displacement by 10%, adding Gaussian white noise (means equal to 0, variance equal to 0.04), adding salt & pepper noise (variance equal to 0.04) and adding speckle noise (variance equal to 0.04). Figure-8 illustrates an example of all the processes. Following that, all such processed images are added into the database. When one of them is taken as a query image, we would like to see how many of those processed images can be retrieved to test the robustness of indexing keys. To this end, we measure the retrieved results in terms of the recall and the average of the normalized modified retrieval rank over all queries (ANMRR) as described in [21].

Let n_c and n_m be the number of correct, and missed candidates, respectively, among the first M retrievals (M is 18 in our experiments). The recall for query image q is defined as:

$$Recall_q = \frac{n_c}{n_c + n_m} \quad (8)$$

To calculate ANMRR [22], we firstly produce an average rank for query image q as follows:

$$AVR(q) = \sum_{k=1}^{NG(q)} \frac{Rank(k)}{NG(q)}. \quad (9)$$

Where $NG(q)$ is the number of relevant images for query q .

Secondly, the average rank is modified into:

$$MRR(q) = AVR(q) - 0.5 - \frac{NG(q)}{2} \tag{10}$$

and thirdly, the normalized modified retrieval rank is derived by:

$$NMRR(q) = \frac{MRR(q)}{\max \left\{ 4 \times NG(q), \max_{q \in Q} (NG(q)) \right\} + 0.5 - 0.5 * NG(q)} \tag{11}$$

Where Q is the number of all queries.

Finally, the ANMRR is calculated for all queries as given below:

$$ANMRR = \frac{1}{Q} \sum_{q=1}^Q NMRR(q) \tag{12}$$

All the experimental results produced in the second phase are summarized in Table-4, which clearly indicates that the proposed algorithm is more robust than the benchmark in terms recall rates, and achieves superior performances measured by ANMRR values.

Table 4. Summary of second phase experimental results

| Methods | Benchmark | 3_D EBH | 4_D EBH | RL EBH |
|---|-----------|------------|------------|-----------|
| Average recall after clockwise rotate by 10° | 0.78 | 0.52 | 0.50 | 0.50 |
| Average recall after anti-clockwise rotate by 10° | 0.74 | 0.40 | 0.48 | 0.52 |
| Average recall after zoom out 20% | 0.96 | 0.18 | 0.12 | 0.72 |
| Average recall after zoom in 20% | 0.94 | 0.44 | 0.62 | 0.90 |
| Average recall after inverse color | 0.02 | 0.98 | 1.00 | 1.00 |
| Average recall after horizontal upset | 0.92 | 0.92 | 0.92 | 0.98 |
| Average recall after vertical upset | 0.92 | 0.90 | 0.88 | 0.94 |
| Average recall after horizontal displacement 10% | 0.82 | 0.44 | 0.44 | 0.50 |
| Average recall after vertical displacement 10% | 0.86 | 0.44 | 0.40 | 0.52 |
| Average recall after adding gaussian white noise | 0.02 | 0.64 | 0.66 | 0.90 |
| Average recall after adding salt& pepper noise | 0.02 | 0.18 | 0.22 | 0.42 |
| Average recall after adding speckle noise | 0.08 | 0.44 | 0.44 | 0.60 |
| Average recall | 0.59 | 0.54 | 0.56 | 0.71 |
| ANMRR | 0.1 | 0.12 | 0.14 | 0.08 |

5 Conclusions

In this paper, we proposed an effective algorithm for content-based image retrieval. In comparison with existing techniques, the proposed algorithm illustrates the following advantages and features: (i) robust to a range of orientation changes which are frequently encountered during transmission, processing and storages; (ii) directly operates in compressed domain, which are suitable for large compressed image databases; (iii) extremely low computing cost suitable for real-time implementation and fast image content management applications. As a matter of fact, the cost for classifying edge-blocks only requires partial decoding of two DCT coefficients $X(0,1)$ and $X(1,0)$ from (1) and (2). After that, all needed is to construct a histogram by counting the number of occurrences for each edge block pattern state. Finally, extensive experiments also support that the proposed algorithm outperforms the similar counterpart in terms of retrieval precision rates. Therefore, the proposed algorithm would have significant potential for practical applications either as a stand-alone tool or as a component working with other tools towards efficient and effective visual content management.

Finally, the authors wish to acknowledge the financial support under EU IST FP-6 Research Programme as funded for the integrated project: LIVE (Contract No. IST-4-027312).

References

- [1] M.K. Mandal, F. Idris and S. Panchanatha, A critical evaluation of image and video indexing techniques in the compressed domain, *Image and Vision Computing*, Vol.17, pp.513-529, 1999
- [2] M. Shneier, M. Abdel-Mottaleb, Exploiting the JPEG compression scheme for image retrieval, *IEEE Trans. Pattern Anal. Mach. Intell.* 18(8) ,pp.849-853, 1996.
- [3] Shih-Fu Chang, Compressed domain techniques for image/video indexing and manipulation, *IEEE International Conference on Image Processing*, pp.314-317, 1995.
- [4] Chong-Wah Ngo, Ting-chuen Pong, Exploiting image indexing techniques in DCT domain, *Pattern Recognition* 34 pp.1841-1851, 2001.
- [5] Y.S. Hsu, S. Prum, J.H. Kagel, H.C. Andrews, Pattern recognition experiments in the Mandala/Cosine domain, *IEEE Trans. Pattern Anal. Mach. Intell.* 5(5), pp.512-520, 1983.
- [6] Guocan Feng, Jianmin Jiang, JPEG compressed image retrieval via statistical features, *Pattern Recognition* 36, pp. 977-985, 2003.
- [7] Lay Jose A, Ling Guang, Image retrieval based on energy histograms of the low frequency DCT coefficients [A]. In: *Proc of International Conference on Acoustics, Speech and Signal Processing[C]*, Phoenix, Arizona, USA, 6: 3009-3012, 1999.
- [8] D.G. Sim, H.K. Kim, R.H. Park, Fast texture description and retrieval of DCT-based compressed image [J], *Electronic Letters*, 37(1): 18-19, 2001.
- [9] Jianghong Liu, Haiming Gu, Image retrieval in various domains, *Computers & Graphics* 27, pp.807-812, 2003.
- [10] Daidi Zhong, Irek Defee, DCT histogram optimization for image database retrieval, *Pattern Recognition Letters*, 2005.
- [11] Jun Wei Han, Lei Guo, A shape-based image retrieval method using salient edges, *Signal processing: Image communication* 18, pp.141-156, 2003.

- [12] Minakshi Banerjee, Malay K.Kundu, Edge based features for content based image retrieval, *Pattern Recognition* 36, pp.2649-2661, 2003.
- [13] D.S. Kim and S.U. Lee, Image vector quantizer based on a classification in the DCT domain, *IEEE Trans. Commun.*, vol. 39, no.4, pp.549-556, Apr.1991.
- [14] S. Soltane, N. Kerkeni, J.C. Angue, The use of two dimensional discrete cosine transform for an adaptive approach to image segmentation, *Proceedings of the SPIE Image and Video Processing IV*, pp.242-251, 1996.
- [15] B. Shen and I.K. Sethi, Direct feature extraction from compressed images, in *Proc. SPIE: Storage and Retrieval for Still Image and Video Databases IV*, vol.2670, pp.404-414, Mar.1996.
- [16] S.-W. Lee, Y.-M. Kim, and S.W. Choi, Fast scene change detection using direct feature extraction from MPEG compressed videos, *IEEE Trans. Multimedia*, vol.2, no.4, pp.240-254, Dec.2000.
- [17] H. Li, G. Liu, and Y. Li, An effective approach to edge classification from DCT domain, in *Proc. IEEE Int. Conf. Image Processing*, pp.940-943, Sep.2002.
- [18] Hyun Sung Chang, Kyeongok Kang, A compressed domain scheme for classifying block edge patterns, *IEEE Transactions on image processing*, vol.14, no.2, Feb. 2005.
- [19] C.S. Won, D.K. Park, and S.-J. Park, Efficient use of MPEG-7 edge histogram descriptor, *ETRI J.*, vol.24,no.1, pp.23-30, Feb.2002.
- [20] H.J. Bartsch *Handbook of mathematical formulas*, Academic press, 1974;
- [21] "MPEG Vancouver Meeting", ISO/IEC JTC1/SC29/WG11, Experimentation Model Ver.2.0, Doc. N2822, July 1999.
- [22] Ho Young Lee, Ho Keun Lee, and Yeong Ho Ha, Spatial color descriptor for image retrieval and video segmentation, *IEEE Transaction on Multimedia*, vol.5, No.3, Sep. 2003.

Visual Aspect: A Unified Content-Based Collaborative Filtering Model for Visual Document Recommendation

Sabri Boutemedjet and Djemel Ziou

DI, Faculté des Sciences
Université de Sherbrooke
Sherbrooke, QC, Canada J1K 2R1
{sabri.boutemedjet, djemel.ziou}@usherbrooke.ca

Abstract. This paper presents a generative graphical model (VC-Aspect) for filtering visual documents such as images. The proposed VC-Aspect extends the well-known Aspect model and combines both content based and collaborative filtering approaches in a unified framework. Instead of considering item indices in the model such as model-based collaborative filtering techniques, we use visual features in describing visual documents. This allows the model to predict ratings for new visual documents with the same set of parameters. Experimental results show the usefulness of such an approach in a real life application such as the content based image retrieval.

1 Introduction

The huge amount of information in current digital libraries motivates the use of recommender systems since only few items are known to a particular user. The main goal of a recommender system is to select only relevant information (Web sites, scientific papers, movies, etc.) for a specific user according to her preferences represented by user models. Indeed, the task of user modelling consists of using this small quantity of information about the user in order to elaborate models able of predicting her long term interests. Most user modelling tasks are made within the information filtering (IF) community in which the focus was mainly made on textual information. As an example, we may cite ResearchIndex recommenders which suggests scientific papers for specific users. Also, some movie recommenders suggest relevant movies according the user model built using his past history and also textual meta-data associated with movies (e.g. actors, genre, etc.). Visual document (e.g. images) filtering are far from being investigated by the IF research community even though the fact that images constitute 30% of information present in the World Wide Web. Visual document collections were studied mainly within information retrieval (IR) community and the goal was to develop techniques for indexing, searching and browsing image collections in order to respond to user ad hoc needs expressed as queries. However, user long term interests remain ignored in current image retrieval systems. We propose a new system in which user preferences related to visual documents

are modelled using a probabilistic framework. This allows another kind of interaction scenarios in which visual documents are suggested in a proactive manner.

This paper is organized as follows. In section 2 we present current research related to our work. In section 3 we elaborate our probabilistic model of user preferences related to the visual content and we present our method for learning this model. We detail our recommendation algorithm specific for visual document filtering in section 4. Experimental results will be presented in section 5. We conclude this paper by summary of the work and future directions.

2 Related Research Work

In literature, the task of user modelling [1] is addressed mainly within the information filtering (IF) [2] community, the technology used in recommender systems. For example, Amazon [3] makes personalized recommendations based on the books a user has bought and any explicit ratings (or votes) of books expressed on a five star scale. There are two classes of information filtering systems namely content based filtering (CBF) and collaborative filtering (CF). In CBF systems such as [4], user profiles are represented using the item's content and the relevance of items is computed using a similarity measure between the user profile and the content that describe the item. Indeed, CBF systems are useful in the case where user preferences may be expressed by content features that describe her interest. For example, scientific papers are indexed by keywords that may be used in user models. It has been noticed that CBF systems suffer from *overspecialization* in the sense that suggested items are always close to the content features of the user model. CF systems in other hand, alleviate the overspecialization problem by predicting the relevance of items for an active user on the basis of other like minded users. For example, memory-based CF systems identify user neighborhoods using correlation algorithms such as Pearson Correlation Coefficients (PCC) [5] or cosine similarity [6] measure while model-based CF systems construct first a probabilistic model of user preferences and then predict item's relevance using the constructed model. Examples include the Aspect model [7] and the Flexible Mixture Model [8] and many others [9] [10]. To the best of our knowledge the unique attempt of visual document filtering was the Viscors system [11] which build user neighborhoods using PCC measure in order to recommend cell-phone wallpapers (images) to the customers. Since Viscors employs a memory-based technique it inherits from its limitations mainly those related to the the poor prediction accuracy when user neighborhoods are small and also its inefficiency when used online with a large number of users and items. It is important to notice that in both CF techniques, users and items are represented by their indices and a user history is made up of a set of triplets (u, i, r) to denote a rating (or importance degree) r given by the user u to the item i . Indeed, it is impossible to recommend items on whose there is no associated ratings using these approaches. In our work, we consider that visual documents especially images constitute a special kind of information that need to filtered differently by taking into account their intrinsic characteristics. In fact,

the visual features of images such as color histograms or texture co-occurrence vectors may be modelled efficiently using probability distribution functions (pdfs). In other words, we propose Visual Content Aspect model (or simply VC-Aspect) a model-based hybrid filtering visual document recommender which combines both CBF and CF in a unified probabilistic framework. Our VC-Aspect model which we detail in the next section makes similar assumptions of the Aspect model and constructs user communities according to similar tastes on visual content instead of visual documents themselves (i.e. items indices are removed). This allows the prediction of ratings for newer visual documents that arrive into the collection even if there is no rating associated to them¹.

3 The Proposed Model

The domains we consider are a set of users $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$ and a set of visual documents $\mathcal{I} = \{i_1, i_2, \dots, i_{|\mathcal{I}|}\}$, $|\cdot|$ used to denote the cardinality of a set. Let us consider a utility function s in $\mathcal{U} \times \mathcal{I}$ where $s(u, i)$ quantifies the relevance of a visual document i for the user u . This function denotes a predicted relevance degree (i.e. rating) to a particular visual document for a certain user. We denote by $\mathcal{R} = \{r_1, \dots, r_{|\mathcal{R}|}\}$ the rating scale where r_1 (resp. $r_{|\mathcal{R}|}$) refers to the lowest (resp. highest) relevance degree. We propose to predict $s(u, i)$ on the basis of probabilities $p(u, i, r)$. More specifically, since we are interested by the prediction of a rating for a given user, we will compute $s(u, i)$ on the basis of probabilities $p(i, r|u)$. One may think to use the mean prediction rule² which computes a rating as $s(u, i) = \sum_r r p(r|i, u)$. One can also use the maximum prediction rule which returns a rating with the highest probability (i.e. $s(u, i) = \arg \max_r p(r|i, u)$). In this paper, we use the mean prediction rule for $s(u, i)$ and we focus hereafter on modelling $p(i, r|u)$. In the Aspect Model[7] (see Eq.(1)), a hidden variable z was introduced so that items and users are rendered conditionally independent given the state of z . This variable was interpreted as hidden preference patterns and also as user communities and their related items.

$$p(i, r|u) = \sum_z p(z|u)p(i, r|z) \quad (1)$$

Indeed, a user is considered as a distribution over hidden preference patterns which are in turn considered as a distribution over items and ratings. The Aspect model has proven to be efficient in CF tasks, however, it is inadequate for filtering visual documents. Let us consider the scenario that of some images arrive in the collection after the learning process. These images have similar images in the collection but with no rating associated to them by any of the users of the recommender system. The Aspect model as any other pure CF technique will be unable to suggest these images to the users. We propose our VC-Aspect

¹ We assume that the image collection is diversified enough so that we will not consider the problem of novelty detection.

² It is easy to show that $p(r|u, i) = p(i, r|u) / \sum_r p(i, r|u)$.

model to deal with such situations by identifying user communities on the basis of similar tastes on similar content. Let us consider the example (see Fig.1) of two users u_1 and u_2 who preferred different images of a sunset. It is convenient to assume that both users have similar tastes due to the similarity between the preferred content. Current CF however, will assume these images as different and hence will fail in assigning these users to the same community. In IF literature, the techniques that identify user communities on the basis of similar tastes on similar content, are referred to as hybrid filtering techniques.

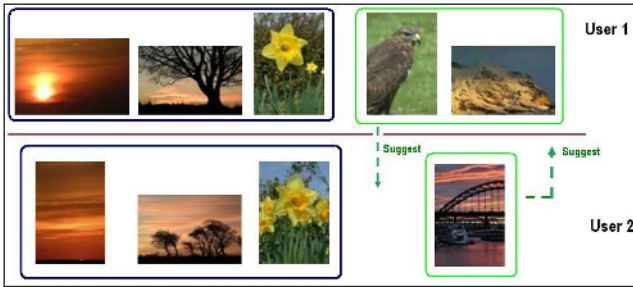


Fig. 1. collaborative filtering of visual content

In the VC-Aspect model, we consider a user as a distribution over hidden preferences patterns which are in turn considered as distributions over visual document classes. These classes are identified on the basis of the visual content (i.e. visual features) in the collection. We consider a visual document i as a vector of visual features v^i such as color histograms and texture co-occurrence vectors. The image collection can be written as a finite mixture model in Eq.(2). The variable c denotes the class of the feature vector.

$$p(v^i) = \sum_c p(c)p(v^i|c) \tag{2}$$

We assume a conditional independence between users u and visual content classes c given the hidden preference pattern z (i.e. $u \perp\!\!\!\perp c|z$)³ and another conditional independence between visual contents (i.e. visual features) and the preference pattern z given the class c of visual content (i.e. $v^i \perp\!\!\!\perp z|c$). This assumption is natural and means simply that we consider each visual document as generated by the class into which it belongs. To ensure a generative model, we assume also that the rating is generated only by the preference pattern z . We give the detailed formula of the model in Eq.(3). Experiments showed this last assumption has not great effect on prediction accuracy.

³ Phil Dawid's notation.

$$\begin{aligned}
 p(v^i, r|u) &= \sum_z p(z|u)p(r|z) \sum_c p(c|z)p(v^i|c) \\
 &= \sum_{z,c} p(z|u)p(r|z)p(c|z)p(v^i|c) \\
 &= \sum_{z,c} p(z, c|u)p(r|z)p(v^i|c)
 \end{aligned}
 \tag{3}$$

3.1 Data Model

In Eq.(3), we consider the model as mixture model in which $p(z|u)p(c|z) = p(z, c|u)$ are considered as mixture weights which we denote by β_{zc}^u . The probability $p(r|z)$ may be interpreted as the probability of a rating to be generated from a hidden preference pattern z . Since the variable r has $|\mathcal{R}|$ states, then we model $p(r|z)$ as a multinomial distribution with parameter vector $\phi^z = (\phi_{r_1}^z, \dots, \phi_{r_{|\mathcal{R}|}}^z)$ and we have thus to estimate $|\mathcal{R}| \times |\mathcal{Z}|$ values for $\phi = (\phi^{z_1}, \dots, \phi^{z_{|\mathcal{Z}|}})$. In modelling visual features such as the color histogram and the texture co-occurrence vector, we use a probability distribution function (pdf) for $p(v^i|c)$. It has been shown in a previous work [12] that the Dirichlet distribution (DD) is an efficient tool for modelling visual features than the Gaussian. In fact, the DD is defined in compact support and offers a higher flexibility due to the possibility to have different shapes. Generally, colors are encoded within the interval $[0, 255]$ which makes color histograms compactly supported and may be represented efficiently by a DD. A visual feature is represented by a vector $v^i = (v_1^i, \dots, v_n^i)$ of dimension n where n denotes for example the number of color bins in a color histogram. We say that v^i such as $v_l^i \in [0, A] \forall l \in [1, n]$, follows a DD in the class c with parameters $\alpha^c = (\alpha_1^c, \dots, \alpha_n^c, \alpha_l^c > 0$ when $p(v^i|c)$ has the form of the density given in Eq.(4). This density is defined in the simplex $\{(v_1^i, \dots, v_n^i), \sum_{l=1}^{n-1} v_l^i < A\}$ and we have $(v_n^i = A - \sum_{l=1}^{n-1} v_l^i)$. Notice that we have used a general form of the DD since we have put $\sum_{l=1}^n v_l^i = A$.

$$p(v^i|c) = \frac{\Gamma(|\alpha^c|)}{A^{|\alpha^c|-1} \prod_{l=1}^n \Gamma(\alpha_l^c)} \prod_{l=1}^n (v_l^i)^{\alpha_l^c-1}
 \tag{4}$$

We notice that the feature vector v^i is obtained using an auxiliary image processing software which extracts the low level visual features of the visual document i in only one vector.

3.2 Estimation of Model's Parameters

We will now show how to estimate $p(z, c|u) p(v^i|c), p(r|z)$ presented in Eq.(3). Their parameters are respectively β, α^c and ϕ . For convenience we put $\Theta = (\beta, \alpha^c, \phi)$. We use the Expectation-Maximization (EM) algorithm [13] in computing maximum likelihood estimates of the proposed model parameters since it contains hidden variables namely preference patterns z and visual feature

classes c . This algorithm alternates between two steps Expectation step (E-step) in which the model estimates the expectation of hidden variables given the observations and in the Maximization step (M-step) the parameters are estimated by considering complete data model. The algorithm stops when a convergence criterion is reached. We assume a sample of independent and identically distributed (iid) observations $\langle u, i, r \rangle$. To keep the derivatives clear, we write the log-likelihood L as a function of Θ only as given in the following:

$$L(\Theta) = \sum_{\langle i, u, r \rangle} \ln \left[\sum_{z, c} p(z, c|u) p(v^i|c) p(r|z) \right] \tag{5}$$

Since the states of the hidden variables c and z are not known, we introduce a so called *variational probability function* $Q(z, c|u, i, r, \Theta)$ over hidden variables z, c for every observation $\langle u, i, r \rangle$ to get a lower bound $\mathcal{F}(Q, \Theta)$ to be maximized. During the E-step, the model maximizes this lower bound $\mathcal{F}(Q, \Theta)$ with respect to Q and during the M-step, maximization is made with respect to Θ . We give the formula of EM steps in the following:

E-step:

$$Q^*(z, c|u, i, r, \hat{\Theta}) = \frac{\hat{\beta}_{zc}^u \hat{\phi}_r^z \hat{p}(v^i|c)}{\sum_{z, c} \hat{\beta}_{zc}^u \hat{\phi}_r^z \hat{p}(v^i|c)} \tag{6}$$

M-step:

$$\beta_{zc}^{u(t+1)} = \frac{\sum_{\langle u', i, r \rangle : u'=u} Q^{*(t)}(z, c|u, i, r, \hat{\Theta})}{\sum_{\langle u', i, r \rangle : u'=u} \sum_{z, c} Q^{*(t)}(z, c|u, i, r, \hat{\Theta})} \tag{7a}$$

$$\phi_r^{z(t+1)} = \frac{\sum_{\langle u, i, r' \rangle : r'=r} \sum_c Q^{*(t)}(z, c|u, i, r, \hat{\Theta})}{\sum_{\langle u, i, r \rangle} \sum_c Q^{*(t)}(z, c|u, i, r, \hat{\Theta})} \tag{7b}$$

We use the Fisher scoring method in estimating the parameters α^c of the Dirichlet distributions $p(v^i|c)$. Its principle is to update the parameters at the time $t+1$ using their value at the time t and based on the log-likelihood function and the Fisher information matrix \mathbf{I} . To ensure that Dirichlet parameters α_l^c (l^{th} parameter of c^{th} class) be always positive, we use transformation $\rho_l^c = e^{\alpha_l^c}$.

4 Recommendation

The goal of a visual content recommender is the suggestion of relevant unseen visual documents to an active user u_a according to her profile. We define the recommendation as the process of deciding which visual documents suggest and their ordering for an active user after the prediction of a rating using $s(u, i)$ defined in section 3. Most recommendation systems [7] [8] use the predicted rating as a criterion for suggestion and ranking. In other words, the content is suggested when its predicted rating is above a certain threshold and relevant items are sorted on the basis of their predicted rating. By this way, it is natural to see that two documents with the same predicted rating will have the same chance

to be suggested even if they are too similar. To avoid this problem and to improve the user's experience, we propose another ranking method for relevant visual documents. Its principle consists of building a list of ranked visual documents. We define a new score computed on the basis of the predicted rating and also on the basis of the diversity constraint. To do that we first build a list \mathcal{D}^u of relevant visual documents (i.e. $s(u, i) > T$ for $i \in \mathcal{D}^u$) ordered decreasingly based on $s(u, i)$. We denote by $\mathcal{D}^u[k]$ the visual document at the position k in the list \mathcal{D}^u . Then, we associate to each visual document $\mathcal{D}^u[k]$ a score $sc[k]$ which combines both similarity measure with the precedent visual documents in the list and the predicted rating (see Eq.8). \mathcal{D}^u is finally reordered decreasingly on the basis of this new score. We introduce another parameter γ in order to control the importance of the predicted rating in the ranking process.

$$sc[k] = \frac{s(u, \mathcal{D}^u[k])^\gamma}{(1 + \max_{k' < k} sim(\mathcal{D}^u[k], \mathcal{D}^u[k']))(1 - \gamma)} \quad (8)$$

According to the Eq.(8), the system assumes that the visual document with the highest rating is relevant. Then, the recommendation algorithm supports visual documents with highest score. This score, in the case where $\gamma = 0.5$, will be higher for documents that are relevant and at the same time not similar to those already selected. We notice that, when $\gamma = 1$, the presented algorithm will suggest visual documents according to their relevance only such as most recommender systems.

5 Experimental Results

In this section we present our method for evaluating the VC-Aspect model. We have made two kinds of evaluations. The first one focuses on measuring the rating's prediction accuracy while the second one is concerned by evaluating the usefulness of visual content filtering in a concrete application that is Content Based Image Retrieval (CBIR) for mobile environments.

5.1 The Data Set

We need a data set \mathcal{DS} of observations $\langle u, v^i, r \rangle$ for the learning and prediction phases. It should be noticed that there is no ground truth for filtering visual documents. To be able to conduct experiments, the data set needs to be generated synthetically. The generation process of observations involves an image collection, a set of users and the generation of associations (past histories of users) between users and visual documents of the collection. We have considered 30 simulated users and the collection of visual documents we have used in experiments contains 3227 images. represented by their color histogram in the RGB (*red, green, blue*) space. We have chosen images that are discriminated by the color feature so that the construction of image classes on the basis of the color feature will be meaningful. We have set two states (i.e. $\mathcal{R} = \{1, 2\}$) for the the rating r to denote the "*likes*" and "*dislikes*" preferences. Three user

classes are set according to the range of user indices while image classes are obtained by summarizing the image collection using a finite mixture model i.e. $p(v^i) = \sum_c p(c)p(v^i|c)$. The class of an image is identified using the Bayes' decision rule. Finally, we generate associations between user classes and image classes randomly where each user has at least 20 ratings.

5.2 Evaluating the Rating's Prediction Accuracy

This evaluation tries to evaluate the accuracy of the rating's prediction using cross-validation. We use a small part \mathcal{DS}^L from the data set for learning the model and the remaining part \mathcal{DS}^E for evaluation. Then, we measure the error between the predicted and observed ratings in the evaluation data set \mathcal{DS}^E . The method consists of computing the Mean Absolute Error (MAE) [14] between the predicted rating $s(u, i)$ and user's true rating say $r_t(u, i)$ obtained from the evaluation data set \mathcal{DS}^E . Notice that $r_t(u, i) \in \{1, 2\}$ while $s(u, i) \in [1, 2]$ since it returns an average predicted rating. We compare also our results with the Aspect model [7]. Results are shown in Figures 2. where we can see that the average absolute error for the VC-Aspect and the Aspect models are close each to other ($\simeq 0.095$). We notice that the small difference of error of both models is explained by some weak memberships of images to the classes c .

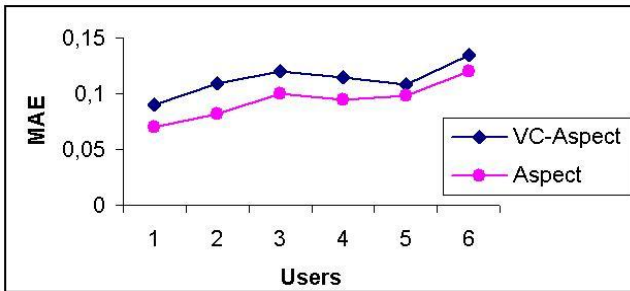


Fig. 2. MAE curves for VC-Aspect and Aspect models

In order to evaluate the rating's prediction accuracy for new images, we use the previous data set by excluding some images from the learning data set. We run the learning process in order to estimate the parameters of the model. Then, we predict the average rating for those images for some users. MAE results are shown in Fig.3. It is clear to see that VC-Aspect model, is able of generating ratings for newer images with an acceptable error ($\simeq 0.125$). We notice however, that we have not considered the problem of novelty detection since we have discarded from the learning data set some images from each class c and not a complete class of images at whole.

5.3 Evaluating the Recommendation Algorithm

In this experiment we want to measure the user’s satisfaction regarding the recommendation of visual documents. We validate the usefulness of visual document filtering in a concrete application namely CBIR. AtlasMobile is a mobile version of a previously developed [15] CBIR system designed to run in a mobile environment and we address the “cold-start” problem. In fact, current CBIR systems present an initial set (page zero) of randomly selected images to user from which he may mark one or more as positive or negative examples to construct a search query. In the case of mobile CBIR, the size of the page zero (i.e the number of displayed images) will be very small and hence the number of relevant images is reduced greatly. To alleviate this problem we propose to “recommend” images in the page zero instead of selecting them randomly. Figure 4 presents the page zero of two users with different preference patterns. We compare three versions of AtlasMobile according to the value of the parameter γ (see sect. 4): *VC-Aspect Recommend* ($\gamma = 0.5$), *VC-Aspect* ($\gamma = 1$) and *Random* ($\gamma = 0.5$). We use the precision-scope curve and the number of refinement iterations as performance metrics. A relevant visual document allows the user to find the sought images in the collection in few iterations when he uses this visual document in the query. The precision measure the portion of relevant representatives rather than relevant visual documents themselves. The scope denotes the number of “recommended” images. A human subject reports the number of relevant representatives and also the number of refinement iterations. A fail state for the number of iterations is generated when it exceeds 25.

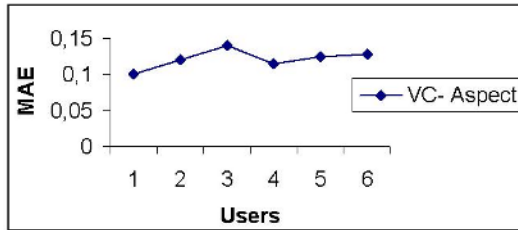


Fig. 3. MAE curve for some new images using VC-Aspect

We can see from Table 1 that the random selection of images in the page zero is widely outperformed by the user specific visual document recommendation. In some situations, the random selection fails to locate in the collection the relevant images in a reasonable number of refinement iterations. A “fail” state in Table 1 informs mainly about the fact that images used in the page zero are not relevant and very far from the ideal query a user should formulate. When the number of refinement iterations increases, this may be explained by a quality⁴

⁴ The term quality refers to the degree of closeness of image representatives to the sought images.

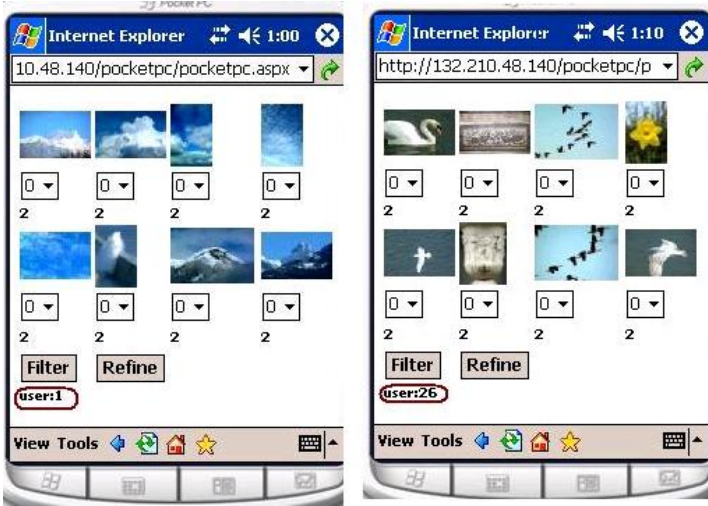


Fig. 4. AtlasMobile page zero for two users

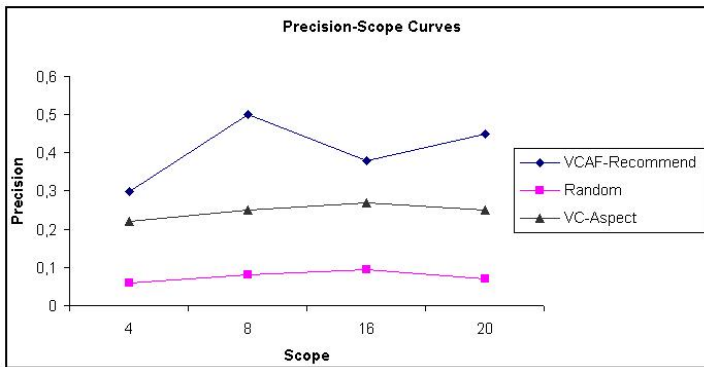


Fig. 5. Precision-scope curves

of image representatives that deteriorates. It should be noticed that *VC-Aspect-Recommend* selects relevant images with a quality better than those selected by the *VC-Aspect* system since the chance of obtaining relevant images increases with a relevant and diversified initial selection.

From these results, it is obvious to confirm the usefulness of modelling the users to improve their experience with the CBIR systems.

6 Conclusion

In this paper we have addressed a new problem of recommending visual documents using a hybrid filtering approach which combines content based and

Table 1. Number of refinement iterations using the three versions of AtlasMobile

| Iterations | |
|----------------------------|------|
| Experiment 1 | |
| Random | 20 |
| <i>VC-Aspect</i> | 8 |
| <i>VC-Aspect-Recommend</i> | 8 |
| Experiment 2 | |
| Random | Fail |
| <i>VC-Aspect</i> | 13 |
| <i>VC-Aspect-Recommend</i> | 9 |

collaborative filtering. The proposed VC-Aspect model allowed the prediction of accurate ratings as the pure collaborative filtering Aspect model. Under some assumptions, the VC-Aspect model predicted also accurate ratings for visual documents with no associated ratings. We have presented a new algorithm for top-N visual document ranking which takes into the diversity of recommended images into account. Experimental results showed the gain in efficiency of CBIR systems designed for mobile environments when initial example images are recommended according to the preferences of each user.

Acknowledgements

The completion of this research was made possible thanks to NSERC Canada for their support.

References

1. Kobsa, A.: Generic user modeling systems. *User Modeling and User-Adapted Interaction* **11**(1-2) (2001) 49–63
2. Hanani, U., Shapira, B., Shoval, P.: Information filtering: Overview of issues, research and systems. *User Modeling and User-Adapted Interaction* **11**(3) (2001) 203–259
3. Linden, G., Smith, B., York, J.: Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing* **7**(1) (2003) 76–80
4. Mooney, R., Roy, L.: Content-based book recommending using learning for text categorization. In: *Proc. Fifth ACM Conf. Digital Libraries*. (2000) 195–204
5. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: Grouplens: An open architecture for collaborative filtering of netnews. In: *In Proceeding of the ACM 1994 Conference on Computer Supported Cooperative Work*. (1994)
6. Breese, J.S., Heckerman, D., Kadie, C.M.: Empirical analysis of predictive algorithms for collaborative filtering. In: *In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, July 24–26, 1998, University of Wisconsin Business School, Madison, Wisconsin, USA. (1998) 43–52
7. Hofmann, T.: Probabilistic latent semantic indexing. In: *Proceedings of Twenty-second Annual International SIGIR Conference*. (1999)

8. Si, L., Jin, R.: Flexible mixture model for collaborative filtering. In: Machine Learning, Proceedings of Twentieth International Conference (ICML 2003), AAAI Press (2003) 704–711
9. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *The Journal of Machine Learning Research* **3** (2003) 993–1022
10. Marlin, B.: Collaborative filtering: A machine learning perspective. Master's thesis, University of Toronto (2004)
11. Kim, C.Y., Lee, J.K., Cho, Y.H., Kim, D.H.: Viscors: A visual-content recommender for the mobile web. *IEEE Intelligent Systems* **19**(6) (2004) 32–39
12. Bouguila, N., Ziou, D., Vaillancourt, J.: Unsupervised learning of a finite mixture model based on the dirichlet distribution and its applications. *IEEE Transactions on Image Processing* **13**(11) (2004) 1533–1543
13. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B* **39** (1977) 1–38
14. Herlocker, J.L.: Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems* **22**(1) (2004) 5–53
15. Kherfi, M., Ziou, D., Bernardi, A.: Combining positive and negative examples in relevance feedback for content-based image retrieval. *Journal of Visual Communication and Image Representation* **14**(4) (2003) 428–457

IntelliSearch: Intelligent Search for Images and Text on the Web

Epimenides Voutsakis¹, Euripides G.M. Petrakis¹, and Evangelos Milios²

¹ Dept. of Electronic and Computer Engineering
Technical University of Crete (TUC)
Chania, Crete, GR-73100, Greece

pimenas@softnet.tuc.gr, petrakis@intelligence.tuc.gr

² Faculty of Computer Science, Dalhousie University
Halifax, Nova Scotia
B3H 1W5, Canada
eem@cs.dal.ca

Abstract. *IntelliSearch* is a complete and fully automated information retrieval system for the Web. It supports fast and accurate responses to queries addressing text and images in Web pages by incorporating state-of-the-art text and Web link information indexing and retrieval methods in conjunction with efficient ranking of Web pages and images by importance (authority). Searching by semantic similarity for discovering information related to user's requests (but not explicitly specified in the queries) is a distinguishing feature of the system. *IntelliSearch* stores a crawl of the Web with more than 1,5 million Web pages with images and is accessible on the Internet¹. It offers an ideal test-bed for experimentation and training and serves as a framework for a realistic evaluation of many Web image retrieval methods.

1 Introduction

Searching for effective methods to retrieve information from the Web has been in the center of many research efforts during the last few years. The relevant technology evolved rapidly thanks to advances in Web systems technology [1] and information retrieval research [2]. Image retrieval on the Web, in particular, is a very important problem in itself [3]. The relevant technology has also evolved significantly propelled by advances in image database research [4].

Several approaches to the problem of content-based image retrieval on the Web have been proposed and some have been implemented on research prototypes (e.g., ImageRover [5], WebSEEK [6], Diogenis [7]) and commercial systems. The last category of systems, includes general purpose image search engines (e.g., Google Image Search ², Yahoo ³, Altavista ⁴ Ditto ⁵) as well as systems providing

¹ <http://www.intelligence.tuc.gr/intellisearch>

² <http://www.google.com/imghp>

³ <http://images.search.yahoo.com>

⁴ <http://www.altavista.com/image>

⁵ <http://www.ditto.com>

specific services to users such as detection of unauthorized use of images, Web and e-mail content filters (e.g., Cobion ⁶), image authentication, licensing and advertising (e.g., Corbis ⁷).

Image retrieval on the Web requires that content descriptions be extracted from Web pages and used to determine which Web pages contain images that satisfy the query selection criteria. The methods and systems referred to above differ in the type of content descriptions used and in the search methods applied. There are four main approaches to Web image search and retrieval.

Retrieval by text content: Typically images on the Web are described by text or attributes associated with images in `html` tags (e.g., filename, caption, alternate text etc.). These are automatically extracted from the Web pages and are used in retrievals. Google, Yahoo, and AltaVista are example systems of this category. The importance of the various text fields in retrieving images by text content depends also on their relative location with regard to the location of the images within the Web pages [8].

Retrieval by image annotations: The Web pages are indexed and retrieved by keywords or text descriptions which are manually assigned to images by human experts. This approach does not scale-up easily for the entire range of image types and the huge volumes of images on the Web. Its effectiveness for general purpose retrievals on the Web is questionable due to the specificity and subjectivity of image interpretations. This approach is typical to corporate systems specializing in providing visual content to diverse range of image consumers (e.g., authentication, licensing and advertising of logos, trademarks, artistic photographs etc.).

Retrieval by image content: The emphasis is on extracting meaningful image content from Web pages and in using this content in the retrieval process. Image analysis techniques are applied to extract a variety of image features such as histograms, color, texture measurements, shape properties. This approach has been adopted mainly by research prototypes (e.g., [5,6,9]).

Hybrid retrieval systems: Combining the above approaches such as systems using image analysis features in conjunction with text and attributes (e.g., [7,10,11]).

The problem of how to select authority Web pages and images on the topic of the query has not been addressed by any of the above methods, which focus mainly on image and text content. In Web text retrieval, link analysis methods such as HITS [12] and PageRank [13] have been applied to estimate the quality of Web pages and the topic relevance between the Web pages and the query. Incorporating page content within link analysis has also been proposed [14]. Extending these ideas to image retrieval on the Web is the natural next step. Building upon HITS, PicASHOW [15] shows how to handle pages that link to images and pages that contain images. WPicASHOW [11] shows how to handle image content in conjunction with link information.

⁶ <http://www.cobion.com>

⁷ <http://pro.corbis.com>

Queries on the Web are issued through the user interface by specifying keywords or free text. The system returns Web pages with similar keywords or text. The highest complexity of queries is encountered in the case of queries by example: The user specifies an example image along with a set of keywords (or annotation) expressing his or her information needs. Queries by example image require that that appropriate content representations be extracted from images in Web pages and matched with similar representations of the queries. However, image analysis approaches for extracting meaningful and reliable descriptions for all image types are not yet available. The adaptation of image descriptions to the different image types coexisting on the Web or to the search criteria or different interpretations of image content by different users is also very difficult. Typically, images are retrieved by addressing text associated with them (e.g., captions) in Web pages [8]. This is the state-of-the-art approach for achieving consistency of representation and high accuracy results.

IntelliSearch is motivated by these ideas. The link analysis and text retrieval methods referred to above are implemented and integrated into *IntelliSearch*. The resulting system provides an ideal test-bed for experimentation and training and also a framework for a realistic evaluation of state-of-the-art Web image retrieval methods. An analysis of the performance of all these methods is presented in [11,16]. The main points of this analysis are also discussed in this work. Furthermore, *IntelliSearch* supports fast and accurate responses to queries addressing Web pages or images by incorporating efficient indexing of text information extracted from Web pages. The system stores a crawl of the Web with 1,5 million Web pages with images.

2 Information Retrieval in *IntelliSearch*

IntelliSearch supports queries by free text and keywords (the most frequent type of image queries in Web image retrieval systems) addressing text or images in Web pages. Typically, images are described by text surrounding them in the Web pages (e.g., caption, title) [8]. The following image descriptors are derived from Web pages based on the analysis of `html` formatting instructions:

Image Filename: The URL entry (with leading directory names removed) in the `src` field of the `img` formatting instruction.

Alternate Text: The text entry of the `alt` field in the `img` formatting instruction. This text is displayed on the browser (in place of the image), if the image fails to load. This attribute is optional (i.e., is not always present).

Page Title: The title of the Web page in which the image is displayed. It is contained between the `TITLE` formatting instructions in the beginning of the document. It is optional.

Image Caption: A sentence that describes the image. It usually follows or precedes the image when it is displayed on the browser. Because it does not correspond to any `html` formatting instruction it is derived either as the text within the same table cell as the image (i.e., between `td` formatting

instructions) or within the same paragraph as the image (i.e., between p formatting instructions). If neither case applies, the caption is considered to be empty. In either case, the caption is limited to 30 words before or after the reference to the image file.

The similarity between a query Q and an image I is computed as a summation of similarities between the query and the above image descriptors:

$$S_{image}(Q, I) = S_{file_name}(Q, I) + S_{alternate_text}(Q, I) + S_{page_title}(Q, I) + S_{image_caption}(Q, I). \quad (1)$$

For queries addressing the text content of Web pages, the similarity between a query Q and a Web page W is computed as the sum of the similarities of the query with the text descriptions obtained from the the entire Web page and its title (if exists):

$$S_{text}(Q, W) = S_{page_title}(Q, W) + S_{page_text}(Q, W). \quad (2)$$

IntelliSearch implements the following two methods for computing similarity S .

2.1 Vector Space Model (VSM) [17]

Queries and texts are syntactically analyzed and reduced into term (noun) vectors. A term is usually defined as a stemmed non stop-word. Very infrequent or very frequent terms are eliminated. Each term in this vector is represented by its weight. The weight of a term is computed as a function of its frequency of occurrence in the document collection and can be defined in many different ways. The term frequency - inverse document frequency model [17] is used for computing the weight. Typically, the weight d_i of a term i in a document is computed as $d_i = tf_i \cdot idf_i$, where tf_i is the frequency of term i in the document and idf_i is the inverse frequency of i in the whole text collection. The formula is modified for queries to give more emphasis to query terms.

Traditionally, the similarity between two documents (e.g., a query Q and a document D) is computed according to the Vector Space Model (VSM) [17] as the cosine of the inner product between their vector representations

$$S(Q, D) = \frac{\sum_i q_i d_i}{\sqrt{\sum_i q_i^2} \sqrt{\sum_i d_i^2}}, \quad (3)$$

where q_i and d_i are the weights in the two vector representations. Given a query, all documents (Web pages or images in *IntelliSearch*) are ranked according to their similarity with the query.

2.2 Semantic Similarity Retrieval Model (SSRM) [16]

The lack of common terms in two documents does not necessarily mean that the documents are unrelated. Similarly, relevant text may not contain the same

terms. Semantically similar concepts may be expressed in different words in the documents and the queries, and direct comparison by word-based VSM is not effective. For example, VSM will not recognize synonyms or semantically similar terms (e.g., "car", "automobile").

SSRM works by discovering semantically similar terms using WordNet⁸ to estimate the similarity between different terms. The similarity between an expanded and re-weighted query q and a text d is computed as

$$S(Q, D) = \frac{\sum_i \sum_j q_i d_j \text{sim}(i, j)}{\sum_i \sum_j q_i d_j}, \quad (4)$$

where i and j are terms in the query and the query Q and document D respectively and $\text{sim}(i, j)$ denotes the semantic similarity between terms i and j [18]. Query terms are expanded with synonyms and semantically similar terms (i.e., hyponyms and hypernyms) while document terms d_j are computed as $tf \cdot idf$ terms (they are neither expanded nor re-weighted).

The method, although slow (due to its quadratic time complexity and extensive searches for terms over WordNet and computing their semantic similarity) has been demonstrated to outperform VSM for text and image queries [16].

2.3 HITS [12]

Co-citation analysis is proposed as a tool for assigning importance to pages or for estimating the similarity between a query and a Web page. A link from page a to page b may be regarded as a reference from the author of a to b . The number and quality of references to a page provide an estimate of the quality of the page and also a suggestion of relevance of its contents with the contents of the pages pointing to it.

HITS exploits co-citation information between pages to estimate the relevance between a query and a Web page and ranking of this page among other relevant pages. HITS computes authority and hub values by link analysis on the *query focused graph* \mathcal{F} (i.e., a set of pages formed by initial query results obtained by VSM expanded by backward and forward links). The page-to-page adjacency matrix W relates each page in \mathcal{F} with the pages it points to. The rows and the columns in W are indices to pages in \mathcal{F} . Then, $w_{ij} = 1$ if page i points to page j ; 0 otherwise. The Authority and Hub values of pages are computed as the principal eigenvectors of the page co-citation $W^T \cdot W$ and bibliographic matrices $W \cdot W^T$ respectively. The higher the authority value of an image the higher its likelihood of being relevant to the query.

2.4 PicASHOW [15]

Building upon HITS, PicASHOW shows how to handle pages that link to images and to pages that contain images. PicASHOW [15] demonstrates how to retrieve high quality Web images on the topic of a keyword-based query. It relies on the

⁸ <http://wordnet.princeton.edu>

idea that images co-contained or co-cited by Web pages are likely to be related to the same topic. Fig. 1 illustrates examples of co-contained and co-cited images. PicASHOW computes authority and hub values by link analysis on the *query focused graph* \mathcal{F} as in HITS. PicASHOW filters out from \mathcal{F} non-informative images such as banners, logo, trademarks and “stop images” (bars, buttons, mail-boxes etc.) from the query focused graph utilizing simple heuristics such as small file size.

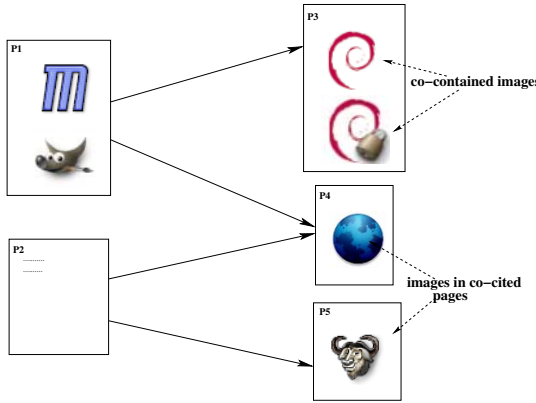












Fig. 1. The focused graph corresponding to query “Debian logo”












PicASHOW introduces the following adjacency matrices defined on the set of pages in the query focused graph:

- \mathcal{W} : The page to page adjacency matrix (as in HITS) relating each page in \mathcal{F} with the pages it points to. The rows and the columns in \mathcal{W} are indices to pages in \mathcal{F} . Then, $w_{ij} = 1$ if page i points to page j ; 0 otherwise.
- \mathcal{M} : The page to image adjacency matrix relating each page in \mathcal{F} with the images it contains. The rows and the columns in \mathcal{M} are indices to pages and images in \mathcal{F} respectively. Then, $m_{ij} = 1$ if page i points to (or contains) image j .
- $(\mathcal{W} + \mathcal{I})\mathcal{M}$: The page to image adjacency matrix (\mathcal{I} is the identity matrix) relating each page in \mathcal{F} both, with the images it contains and with the images contained in pages it points to.

Similarly to HITS, *PicASHOW* defined the so called image *co-citation* and *bibliographic* matrices $[(\mathcal{W} + \mathcal{I})\mathcal{M}]^T \cdot (\mathcal{M} + \mathcal{I})\mathcal{W}$ and $(\mathcal{W} + \mathcal{I})\mathcal{M} \cdot [(\mathcal{W} + \mathcal{I})\mathcal{M}]^T$ respectively. The ij -th entry of the image co-citation matrix is the number of pages that jointly point to images with indices i and j . The ij -th entry of the image bibliographic matrix is the number of images jointly referred to by pages i and j . Fig. 2 illustrates the adjacency and bibliographic matrices for the the focused graph of Fig. 1.

Image Authority and Hub values of images are computed as the principal eigenvectors of the image-co-citation $[(\mathcal{W} + \mathcal{I})\mathcal{M}]^T \cdot (\mathcal{W} + \mathcal{I})\mathcal{M}$ and

| |  |  |  |  |  |
|---|---|---|---|---|---|
|  | 0 | 0 | 1 | 1 | 0 |
|  | 0 | 0 | 0 | 1 | 1 |
|  | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 |

| |  |  |  |  |  |  |
|---|---|---|---|---|---|---|
|  | 0 | 0 | 1 | 1 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 0 |
|  | 1 | 1 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 1 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 1 |












| |  |  |  |  |  |  |
|---|---|---|---|---|---|---|
|  | 1 | 1 | 1 | 1 | 1 | 0 |
|  | 0 | 0 | 0 | 0 | 1 | 1 |
|  | 1 | 1 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 0 | 0 | 1 | 0 |
|  | 0 | 0 | 0 | 0 | 0 | 1 |

Fig. 2. \mathcal{W} , \mathcal{M} and $(\mathcal{W} + \mathcal{I})\mathcal{M}$ matrices computed by PicASHOW for the focused graph of Fig. 1

bibliographic matrices $(\mathcal{W} + \mathcal{I})\mathcal{M} \cdot [(\mathcal{W} + \mathcal{I})\mathcal{M}]^T$ respectively. The higher the authority value of an image the higher its likelihood of being relevant to the query.

PicASHOW can answer queries on a given topic but, similarly to HITS, it suffers from the following problems [14]:

Mutual reinforcement between hosts: Encountered when a single page on a host points to multiple pages on another host or the reverse (when multiple pages on a host point to a single page on another host).

Topic drift: Encountered when the query focused graph contains pages not relevant to the query (due to the expansion with forward and backward links). Then, the highest authority and hub pages tend not to be related to the topic of the query.

2.5 Weighted PicASHOW (WPicASHOW) [11]

PicASHOW does not show how to handle image content or image text context. This problem is addressed by WPicASHOW (or Weighted PicASHOW) [11], a weighted scheme for co-citation analysis is proposed. WPicASHOW relies on the combination of text and visual content and on its resemblance with the query for regulating the influence of links between pages. Co-citation analysis then takes this information into account. WPicASHOW has been shown to achieve better quality answers and higher accuracy results (in terms of precision and recall) than PicASHOW using co-citation information alone [11].

WPicASHOW handles topic drift and mutual reinforcement as follows:

Mutual reinforcement is handled by normalizing the weights of nodes pointing to k other nodes by $1/k$. Similarly, the weights of all l pages pointing to the same page are normalized by $1/l$. An additional improvement is to purge all intra-domain links except links from pages to their contained images.







Topic Drift is handled by regulating the influence of nodes by setting weights on links between pages. The links of the page-to-page relation \mathcal{W} are assigned a relevance value computed by VSM and Eq. 2 as the similarity between the term vector of the query and the term vector of the anchor text on the link between the two pages. The weights of the page-to-image relation matrix \mathcal{M} are computed by VSM and Eq. 1 (as the similarity between the query and the descriptive text of an image).

WPicASHOW starts by formulating the query focused graph as follows:

- An initial set R of images is retrieved. These are images contained or pointed-to by pages matching the query keywords according to Eq. 2.
- Stop images (banners, buttons, etc.) and images with logo-trademark probability less than 0.5 are ignored. At most T images are retained and this limits the size of the query focused graph ($T = 10,000$ in *IntelliSearch*).
- The set R is expanded to include pages pointing to images in R .
- The set R is further expanded to include pages and images that point to pages or images already in R . To limit the influence of very popular sites, for each page in R , at most t ($t = 100$ in this work) new pages are included.
- The last two steps are repeated until R contains T pages and images.

WPicASHOW then builds \mathcal{M} , \mathcal{W} and $(\mathcal{W} + \mathcal{I})\mathcal{M}$ matrices for information in R . Fig. 3 illustrates these matrices for the example set R of Fig. 1 with weights corresponding to query “Debian logo”. Notice that, in PicASHOW all non-zero values in \mathcal{M} and \mathcal{W} are 1 (non normalized weights).

| | | | | | |
|-------|-------|-------|-------|-------|-------|
| | P_1 | P_2 | P_3 | P_4 | P_5 |
| P_1 | 0 | 0 | .6 | .1 | 0 |
| P_2 | 0 | 0 | 0 | .1 | .1 |
| P_3 | 0 | 0 | 0 | 0 | 0 |
| P_4 | 0 | 0 | 0 | 0 | 0 |
| P_5 | 0 | 0 | 0 | 0 | 0 |

| | | | | | | |
|-------|---|---|---|---|---|---|
| |  |  |  |  |  |  |
| P_1 | 0 | 0 | .1 | .1 | 0 | 0 |
| P_2 | 0 | 0 | 0 | 0 | 0 | 0 |
| P_3 | .8 | .7 | 0 | 0 | 0 | 0 |
| P_4 | 0 | 0 | 0 | 0 | .2 | 0 |
| P_5 | 0 | 0 | 0 | 0 | 0 | .15 |







| | | | | | | |
|-------|---|---|---|---|---|---|
| |  |  |  |  |  |  |
| P_1 | .48 | .42 | .1 | .1 | .02 | 0 |
| P_2 | 0 | 0 | 0 | 0 | .02 | .015 |
| P_3 | .8 | .7 | 0 | 0 | 0 | 0 |
| P_4 | 0 | 0 | 0 | 0 | .2 | 0 |
| P_5 | 0 | 0 | 0 | 0 | 0 | .15 |

Fig. 3. \mathcal{W} , \mathcal{M} and $(\mathcal{W} + \mathcal{I})\mathcal{M}$ matrices computed by WPicASHOW for the focused graph of Fig. 1

| Image |  |  |  |  |  |  |
|------------------|---|---|---|---|---|---|
| Authority Values | .751 | .657 | .0418 | .0418 | .008 | 0 |

| Page | P_1 | P_2 | P_3 | P_4 | P_5 |
|------------|-------|-------|-------|-------|-------|
| Hub Values | .519 | .0001 | .854 | .001 | 0 |

Fig. 4. Image Authority (left) and Hub values (right) computed by WPicASHOW in response to query “Debian logo”

Fig. 4 illustrates authority and hub values computed by WPicASHOW in response to query “Debian logo”. The answers to the query are ranked by authority values. Notice the high authority scores of pages showing logo or trademark images of “Debian Linux”.

2.6 Weighted HITS [14]

Similarly to WPicASHOW, WHITS (weighted HITS) a weighting link analysis scheme for retrieval of Web pages is also implemented. HITS uses link information between pages (does not consider links to images or to pages containing images). Links are weighted by their text similarity (as computed by VSM).

3 IntelliSearch Architecture

A complete prototype Web image retrieval system is developed and is accessible on the Web ⁹. The system is implemented in Java. The architecture of *IntelliSearch* is illustrated in Fig. 5. It consists of several modules, the most important of them being the following:

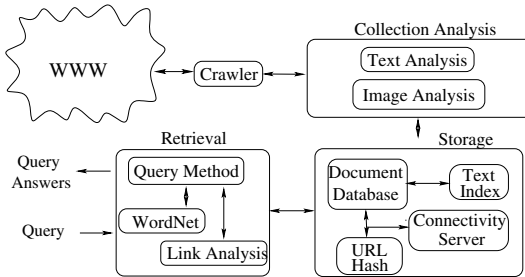


Fig. 5. *IntelliSearch* Architecture

Crawler module: Implemented based upon Larbin ¹⁰, the crawler assembled locally a collection of 1,5 million pages with images. The crawler started its recursive visit of the Web from a set of 14,000 pages which is assembled from the answers of Google image search to 20 queries on topics related to Linux and Linux products. The crawler worked recursively in breadth-first order and visited pages up to depth 5 links from each origin.

Collection analysis module: The content of crawled pages is analyzed. Text, images, link information (forward links) and information for pages that belong to the same site is extracted.

Storage module: Implements storage structures and indices providing fast access to Web pages and information extracted from Web pages (i.e., text, image descriptions and links). For each page, except from raw text and images, the following information is stored and indexed: Page URLs, image descriptive text (i.e., alternate text, caption, title, image file name), terms extracted from pages, term inter document frequencies (i.e., term frequencies in the whole collection), term intra document frequencies (i.e., term frequencies in image descriptive text parts), link structure information (i.e., backward and forward links). Image descriptions are also stored.

The Entity Relationship Diagram (ERD) of the database in Fig. 6 describes entities (i.e., Web pages) and relationships between entities. There are many-to-many (denoted as $N : M$) relationships between Web pages implied by the Web link structure (by forward and backward links), one-to-many (denoted as

⁹ <http://www.intelligence.tuc.gr/intellisearch>

¹⁰ <http://larbin.sourceforge.net>

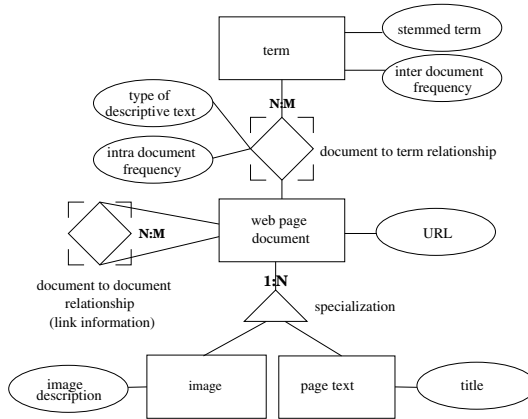


Fig. 6. The Entity Relational Diagram (ERd) of the database

1 : N) relationships between Web pages and their constituent text and images and N : M relationships between terms in image descriptive text parts and documents and. The ERD also illustrates properties of entities and relationships (i.e., page URLs for documents, titles for page text, image content descriptions for images, stemmed terms, inter and intra document frequencies for terms in image descriptive text parts.)

The database schema is implemented in BerkeleyDB¹¹ Java Edition. BerkeleyDB is an embedded database engine providing a simple Application Programming Interface (API) supporting efficient storage and retrieval of Java objects. The mapping of the ERD of Fig. 6 to database files (Java objects) was implemented using the Java Collections-style interface. Apache Lucene¹² is providing mechanisms (i.e., inverted files) for indexing text and link information. There are Hash tables for URLs and inverted files for terms and link information. Two inverted files implement the connectivity server [19] and provide fast access to linkage information between pages (backward and forward links) and two inverted files associate terms with their intra and inter document frequencies and allow for fast computation of term vectors.

Retrieval module: Queries are issued by keywords or free text. The user is prompted at the user interface to select mode of operation (retrieval of text pages or image retrieval). All methods in Sec. 2 are implemented.

4 Conclusions

*IntelliSearch*¹³ is a complete and fully automated system for retrieving text pages and images on the Web. It supports retrieval of important (authoritative)

¹¹ <http://www.sleepycat.com>

¹² <http://lucene.apache.org>

¹³ <http://www.intelligence.tuc.gr/intellisearch>

Web pages and images (by incorporating link analysis into its search and retrieval methods) as well as, searching by semantic similarity for discovering information related to the needs of the users (even if it is not explicitly specified in the queries). Retrievals are speeded-up by indexing text and link information specific to Web pages and images.

The results in [11,16] indicate that text searching methods like VSM and SSRM are far more effective than link analysis methods (text is a very effective descriptor of Web content itself). However, text similarity methods tend to assign higher ranking even to Web pages and images pointed to by very low quality pages such as pages created by individuals or small companies. Between the two, SSRM demonstrated promising performance improvements over VSM.

Link information alone (e.g., as in HITS and PicASHOW) is not an effective descriptor for Web pages and images. Link analysis methods tend to assign higher ranking to higher quality but not necessary relevant pages. High quality pages, on the other hand, may be irrelevant to the content of the query. Weighted link analysis methods (WHITS, WPicASHOW) attempted to compromise between text and link analysis methods.

IntelliSearch is currently being expanded to support queries by image content (e.g., queries by image example). This requires that image analysis methods be applied and appropriate image content representations extracted from images and used in retrievals. Future work includes also experimentation with larger data sets and more image types (i.e., video and graphics).

References

1. Arasu, A., Cho, J., Garcia-Molina, H., Paepke, A., Raghavan, S.: Searching the Web. *ACM Transactions on Internet Technology* **1**(1) (2001) 2–43
2. R. Baeza-Yates, E.: *Modern Information Retrieval*. Addison Wesley (1999)
3. Kherfi, M., Ziou, D., Bernardi, A.: Image Retrieval from the World Wide Web: Issues, Techniques, and Systems. *ACM Computing Surveys* **36**(1) (2004) 35–67
4. Smeulders, A.W., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-Based Image Retrieval at the End of the Early Years. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(11) (2000) 1349–1380
5. Taycher, L., Cascia, M., Sclaroff, S.: Image Digestion and Relevance Feedback in the ImageRover WWW Search Engine. In: *2nd Intern. Conf. on Visual Information Systems*, San Diego (1997) 85–94
6. Smith, J., Chang, S.F.: Visually Searching the Web for Content. *IEEE Multimedia* **4**(3) (1997) 12–20
7. Aslandongan, Y., Yu, C.: Evaluating Strategies and Systems for Content-Based Indexing of Person Images on the Web. In: *8th Intern. Conf. on Multimedia*, Marina del Rey, CA (2000) 313–321
8. Shen, H.T., Ooi, B.C., Tan, K.L.: Giving Meanings to WWW Images. In: *8th Intern. Conf. on Multimedia*, Marina del Rey, CA (2000) 39–47
9. Gevers, T., Smeulders, A.: The PicToSeek WWW Image Search Engine. In: *IEEE ICMS*. (1999)
10. Zhao, R., Grosky, W.: Narrowing the Semantic Gap Improved Text-Based Web Document Retrieval Using Visual Features. *IEEE Transactions on Multimedia* (2) (2002) 189–200

11. Voutsakis, E., Petrakis, E., Milios, E.: Weighted link analysis for logo and trademark image retrieval on the web. (In: IEEE/WIC/ACM International Conference on Web Intelligence (WI2005)) 581–585
12. Kleinberg, J.M.: Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM* **46**(5) (1999) 604–632
13. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Computer Systems Laboratory, Stanford Univ., CA (1998)
14. Bharat, K., Henzinger, M.R.: Improved Algorithms for Topic Distillation in a Hyperlinked Environment. In: Proc. of SIGIR-98, Melbourne (1998) 104–111
15. Lempel, R., Soffer, A.: PicASHOW: Pictorial Authority Search by Hyperlinks on the Web. *ACM Transactions on Information Systems* **20**(1) (2002) 1–24
16. Varelas, G., Voutsakis, E., Raftopoulou, P., Petrakis, E., Milios, E.: Semantic Similarity Methods in WordNet and their Application to Information Retrieval on the Web. In: 7th ACM International Workshop on Web Information and Data Management (WIDM 2005), Bremen, Germany (2005) 10–16
17. Salton, G.: *Automatic Text Processing: The Transformation Analysis and Retrieval of Information by Computer*. Addison-Wesley (1989)
18. Li, Y., Bandar, Z.A., McLean, D.: An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources. *IEEE Trans. on Knowledge and Data Engineering* **15**(4) (2003) 871–882
19. Bharat, K., Broder, A., Henzinger, M.R., Kumar, P., Venkatasubramanian, S.: The Connectivity server: Fast access to Linkage Information on the Web. In: Proceedings of the 7th International World Wide Web Conference (WWW-7), Brisbane, Australia (1998) 469–477

Automatic Shot-Change Detection Algorithm Based on Visual Rhythm Extraction

Kwang-Deok Seo¹, Seong Jun Park², Jin-Soo Kim³,
and Samuel Moon-Ho Song⁴

¹ Computer and Telecommunications Engineering Division, Yonsei Univ., Gangwon, Korea
kdseo@yonsei.ac.kr

² School of Electrical and Computer Engineering, Purdue Univ., West Lafayette, IN, USA
sjpark0420@purdue.edu

³ Division of Information Communication and Computer Engineering,
Hanbat Univ., Daejeon, Korea
jskim67@hanbat.ac.kr

⁴ School of Mechanical and Aerospace Engineering, Seoul National Univ., Seoul, Korea
smsong@snu.ac.kr

Abstract. To store and retrieve large-scale video data sets effectively, the process of shot-change detection is an essential step. In this paper, we propose an automatic shot-change detection algorithm based on Visual Rhythm Spectrum. The Visual Rhythm Spectrum contains distinctive patterns or visual features for many different types of video effects. For the improvement of detection speed, the proposed algorithm is executed by using the partial data of digital compressed video. The proposed detection algorithm can be universally applied to various kinds of shot-change categories such as *scene-cuts* and *wipes*. It is shown by simulations that the proposed detection algorithm outperforms other existing approaches.

1 Introduction

With the ever increasing digital video data and the advancement of digital video applications, *video indexing* is now becoming an integral part of multimedia services which include a wide variety of areas such as digital library, video editing, video summarization, and video database. Once the digital video is indexed, the storage and retrieval will be more manageable and efficient forms of presentation may be developed based on the index. Thus, the normal first step towards the development of a system to efficiently store and retrieve the digital video is the development of highly accurate automatic video parser.

The automatic video parsing algorithms usually consist of several parts or sub-algorithms. It must deal with abrupt, as well as gradual shot changes, which include wipes, fade-ins, fade-outs, etc. The algorithm must be able to detect sudden lighting changes (e.g., flashlights and lighting conditions), and camera related effects, such as zoom, pan, and track. This collection of shot-change scenarios must all be incorporated into the shot-change detection algorithm. The existing algorithms deal with one

or many of these scenarios with different parts of the algorithm dealing with different scenarios. The recent literature on video parsing may be divided into two broad groups. The first of which presents uncompressed domain processing and mainly deals with uncompressed video data and certain features of the video such as object[1], histograms[2], and edges[4]. The second group presents compressed domain processing and considers the DC image [3], [5], [6], [8], the number of forward versus backward motion vectors, and the macroblock data rate [7].

With such shot-change detection algorithms, abrupt changes called *scene-cuts* are detected fairly well. It is thus more challenging to detect gradual changes including *wipes* as these are often missed or falsely detected. In this paper, we focus on the detection of scene-cuts as well as wipes. The proposed algorithm begins by extracting *Visual Rhythm* to compose *Visual Rhythm Spectrum* (VRS). Visual Rhythm Spectrum (VRS) reflects different types of Visual Rhythms of various scene types in the time domain. The VRS contains distinctive patterns or visual features for many different types of video effects. The different video effects manifest themselves differently on the VRS. In particular, scene-cuts are presented as perpendicular line and wipes appear as curves that run from the top to the bottom of the VRS. Thus, using the VRS, it becomes possible to automatically detect scene-cuts and wipes simply by determining various lines and curves on the VRS.

2 Visual Rhythm Extraction

2.1 Visual Rhythm and Pixel Sampling

For the design of an efficient real-time shot-change detector, we resort using a portion of the original video. This partial video must retain most, if not all, video edit effects. We claim that our Visual Rhythm, defined below, satisfies this requirement. Let $f_v(x, y, t)$ be the pixel value at location (x, y) and time t of an arbitrary video V . Then, the video V may be represented as:

$$V = \{f_v(x, y, t)\}, \quad x, y, t \in \{0, 1, 2, \dots\}. \quad (1)$$

Let $f_T(x, y, t)$ be the representation of a reduced frame of a *spatially reduced video* V_T of the original video V . Each reduced frame, or thumbnails, is a (horizontally and vertically) reduced image of its corresponding frame in the video V by a factor r . Thus, the spatially reduced video or sequence of thumbnails may be expressed as:

$$V_T = \{f_T(x, y, t)\}, \quad x, y, t \in \{0, 1, 2, \dots\}. \quad (2)$$

The relationship between the video V and its spatially reduced video V_T can be represented using their pixel correspondences as follows:

$$f_T(x, y, t) = f_v(rx + k_x, ry + k_y, t), \quad x, y, t \in \{0, 1, 2, \dots\}, \quad 0 \leq k_x, k_y \leq r - 1. \quad (3)$$

where k_x and k_y are offsets in pixel units and r is a reduction factor. Using the spatially reduced video, we define the *Visual Rhythm* of the video V as follows:

$$VR = \{f_{VR}(z, t)\} = \{f_T(x(z), y(z), t)\}. \quad (4)$$

where $x(z)$ and $y(z)$ are one-dimensional functions of the independent variable z . Thus, the Visual Rhythm is a two dimensional image consisting of pixels sampled from a three-dimensional data (video). That is, the Visual Rhythm is constructed by sampling a certain group of pixels in each thumbnail and by temporally accumulating the samples along time. Thus, the Visual Rhythm is a two-dimensional abstraction of the entire three-dimensional video content pertaining to a particular scene type.

Depending on the mappings $x(z)$ and $y(z)$, we can obtain various types of Visual Rhythms such as horizontal, vertical, and diagonal Visual Rhythms. According to extensive scrutiny and simulations, we find that the diagonal sampling approach shows the best detection performance. For a constant α , diagonal pixel sampling is achieved by the application of $(x(z), y(z)) = (z, \alpha z)$.

Fig. 1 shows two types of shot-changes and Fig. 2 shows diagonal sampling strategy for the two shot-change types. The Visual Rhythm for wipe results from a repetitive diagonal sampling over many contiguous video frames during the wipe transition.

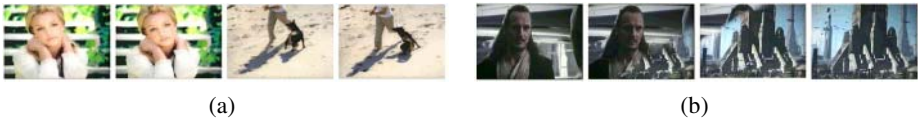


Fig. 1. Shot-change types: (a) scene-cut (b) wipe

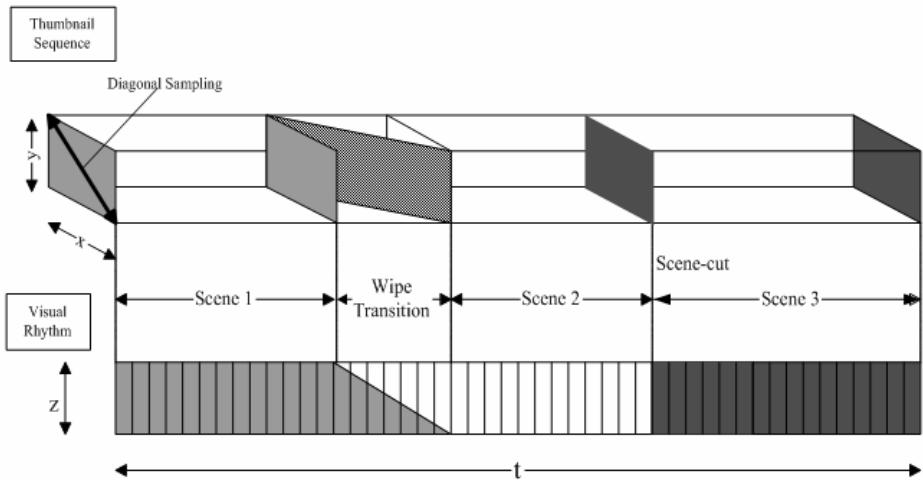


Fig. 2. Visual Rhythm composition from thumbnail sequence

2.2 Characteristics of VRS for Automatic Shot-Change Detection

The Visual Rhythm, an abstraction of a video, is itself a two-dimensional image. It is a depiction of the entire video, which retains visual features of shot changes. In a VRS, pixels along a vertical line are those pixels uniformly sampled along the

diagonal of a frame. The vertical lines on a VRS will have similar visual features if the lines are from the same shot. On the contrary, the lines will have different visual features if they belong to different shots. Thus, if there is a shot change, shot boundaries will become apparent on the VRS, as visual features of the vertical lines will change.

We will now discuss how different shot changes result in different visual features on VRS. Fig. 3 shows some typical Visual Rhythm patterns arising from different types of edit effects. Fig. 3(a) shows the expected pattern on the VRS for an abrupt change or a scene-cut corresponding to the case of Fig. 1(a). Note that a cut will appear as a *vertical line* on the VRS. If two adjacent vertical lines on a VRS belong to different shots, their visual difference leads to a vertical line right at the shot boundary as depicted in Fig. 3(a). Fig. 3(b) shows a left-to-right horizontal wipe. Note that this particular wipe appears as an *oblique line*. Top-to-bottom vertical wipe will also have the same pattern. Both right-to-left horizontal and bottom-to-top vertical wipes will also appear as an oblique line, but with an opposite slope. However, center-to-outskirts expanding wipe will appear as a curved line, as shown in Fig. 3(c). If an incoming shot is uniformly expanded in time, the wipe will appear as two oblique lines which have opposite slopes and meet at the center. Outskirts-to-center absorbing wipe will similarly appear as a curved line, except in the opposite direction. The important point is that all wipes will generate lines from top to bottom. And if this line is over a range of frames, instead of a vertical line over a single frame, then the shot change is probably due to a wipe. Obviously, a straight vertical line over a single frame implies an abrupt shot-change.

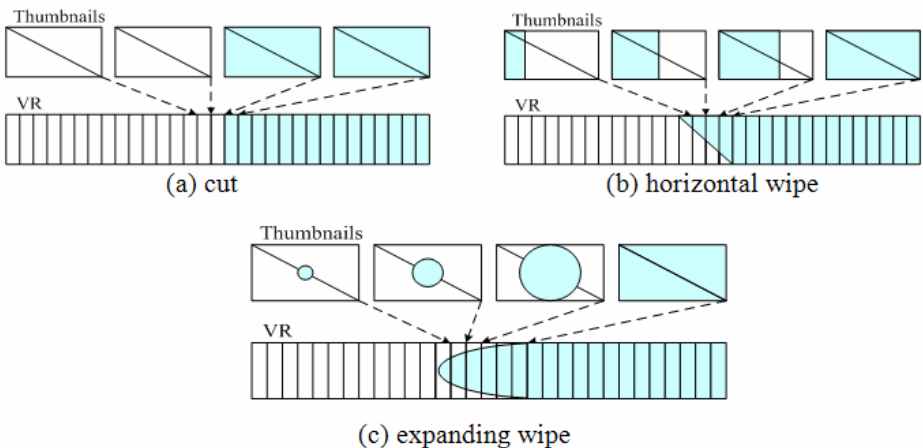


Fig. 3. Illustration of Edit Effects on Visual Rhythm

There are hundreds if not thousands of wipes that modern digital video editors can produce, and a user of the editor may still create an entirely different wipe. However, most wipes can be categorized into horizontal, vertical, expanding, absorbing, and diagonal types. Table 1 summarizes different manifestations of the five wipes on the VRS for diagonal sampling strategy.

Table 1. Visual Rhythms of wipe types for diagonal pixel sampling

| | Horizontal wipe | Vertical wipe | Absorbing wipe | Expanding wipe | Diagonal wipe |
|---------------|-----------------|---------------|----------------|----------------|---------------------|
| Visual Rhythm | Oblique line | Oblique line | Curved line | Curved line | Vertical line (cut) |

We will show that Visual Rhythm composed by projection diagonally in thumbnail sequence can reflect all the features of shot-change. Fig. 4 shows the thumbnail sequence generated from the parts of a 'France' video sequence including shot-change. We sampled thumbnail image in every 5 frame from the 250 thumbnail frames of the sequence, resulting in 50 thumbnail images in total. There are 5 scene-cuts in 75~80, 145~150, 170~175, 180~185 and 230~235 frames, and 1 wipe in 35~50 frames. Fig. 5 shows the corresponding VRS by projecting thumbnail sequence diagonally for the 250 frames. As shown in Fig. 5, there are 5 perpendicular lines corresponding to the scene-cuts and an oblique line corresponding to the wipe. As expected, abrupt changes appear as a straight vertical line on the VRS. And the wipe change results in a visually distinguishable line on the VRS. The line begins slanted which then appears to become vertical towards the top. According to these results, it is possible to design a new shot-change detection algorithm based on Visual Rhythm extraction to detect wipe-like effects (as well as cuts), because those effects manifest themselves as visually distinguishable lines (or curves) on the VRS.

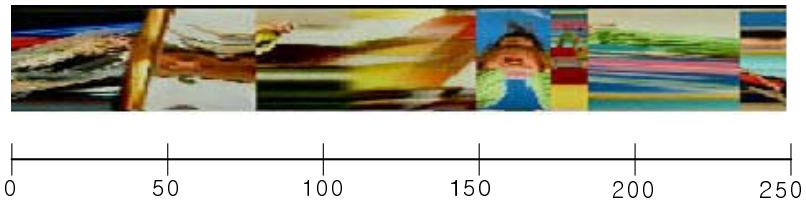
**Fig. 4.** 'France' video sequence including 5 scene-cuts and 1 wipe**Fig. 5.** VRS for 'France' video sequence

Fig. 6 shows another VRS extracted from a real video sequence including various wipe types and scene-cuts. As explained earlier, and indicated on the figure, edit effects such as scene-cuts and wipes can easily be identified from the VRS. Note that scene-cuts take only one frame time to be distinguished.

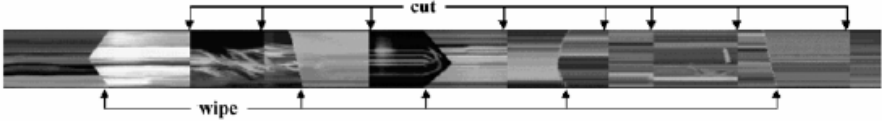


Fig. 6. Visual effects of cut and wipe on the VRS

2.3 Fast Generation of Visual Rhythms

The digital video is usually compressed due to its huge volume. But, because most compression schemes use the discrete cosine transform (DCT) for intra-frame encoding, we can generate the thumbnail sequence only by extracting the DC coefficient in the DCT domain without performing inverse DCT. Since the DC coefficient is actually the average pixel value of the block, the collection of these DC coefficients over the frame can serve as the thumbnail images. Noting this fact, Yeo and Liu [8] introduced the DC image which consists of the DC coefficients of the original frame, and a sequence of DC images called the DC sequence. By accepting the DC image as a thumbnail and the DC sequence as a spatially reduced video, the thumbnail and the original video are related as shown below:

$$f_T(x, y, t) = \frac{1}{8} \sum_{k_x=0}^7 \sum_{k_y=0}^7 f_v(8x+k_x, 8y+k_y, t), \quad x, y, t \in \{0, 1, 2, \dots\}. \tag{5}$$

As for the P- and B-frames of MPEG, algorithms for determining the DC images have already been developed [7], [9]. Therefore it is possible to generate Visual Rhythms fast for the DCT-based compression schemes, such as Motion JPEG, MPEG, H.261 and H.263 videos.

3 Proposed Shot-Change Detection Algorithm

3.1 Wipe Detection Algorithm

The proposed wipe detection basically searches for lines in VRS. The previous discussion indicates that during a wipe intensity change between the incoming and the outgoing shots give rise to abrupt intensity discontinuities on the VRS. The algorithm is designed to detect such discontinuities. Noting that the discontinuity will occur along time (horizontal detection) on the VRS, we would expect that the time derivative to be large during wipes. For the detection of such discontinuities, the following absolute value of the time derivative is computed:

$$d(z, t) = |f_{VR}(z, t) - f_{VR}(z, t - 1)|. \tag{6}$$

Note that for the detection of abrupt changes, $d(z, t)$ may be summed vertically (along z) first which may then be used to detect peaks. The detection of peaks in $d(z, t)$ is performed for one horizontal line at a time. For this purpose, the following statistics are computed:

$$\mu(z,t) = \frac{1}{N(B)} \sum_{k \in B} d(z,t+k), \quad (7)$$

$$\sigma^2(z,t) = \frac{1}{N(B)-1} \sum_{k \in B} (d(z,t+k) - \mu(z,t))^2, \quad (8)$$

where $\mu(z,t)$ and $\sigma^2(z,t)$ are sampled mean and variance of the time derivative image at (z,t) , respectively. The set B means the restrained interval, called sliding window. For instance, it may be $\{-16, -15, \dots, -2, -1, 1, 2, \dots, 15, 16\}$, in which case the sliding window size $N(B)=32$.

For the detection of peaks in $d(z,t)$, the following adaptive thresholding scheme is used:

$$b(z,t) = \begin{cases} 1, & \text{if } d(z,t) > \mu(z,t) + K\sigma(z,t) \\ 0, & \text{else.} \end{cases} \quad (9)$$

where K is a parameter that can be controlled by user. Note that a particular cell of the binary mask $b(z,t)$ will be set if the signal $d(z,t)$ is above its local mean by K standard deviations. We have experimentally observed that the choice of K is quite arbitrary and values from 3 to 8 works quite robustly. This binary mask is used for determining the lines and curves that divide the Visual Rhythm in half. This is done pure by checking the connectivity of the curve on the binary mask and can be implemented quite easily by a simple recursive calculation using the quadratic sliding window, as shown in Fig. 7. If we find a pixel with $b(z,t)=1$ at (z,t) point in the binary mask, we set a specific quadratic window whose top-left point is $(z-n, t-n)$ and bottom-right point is $(z+n, t+n)$, and accordingly with area of $(2n+1)^2$. By sliding this quadratic window for the detection of peaks, we find another point whose $b(z,t)$ value is 1. We repeat this procedure until we cannot find any more pixel values with $b(z,t)=1$. If this window moves from top to bottom in the VRS, we might declare the corresponding frame duration to be the *wipe*. Fig. 8 shows a trace of the quadratic sliding window for the detection of a wipe.

3.2 Scene-Cut Detection Algorithm

Scene-cut detection is similar to the Wipe, because it also detects discontinuous lines. Note that for the detection of abrupt shot-change, $d(z,t)$ in Eq. (6) should be summed vertically (along z). Thus the peaks in $d(z,t)$ line-up vertically. The abrupt changes are detected by first summing the derivative image vertically and then by the adaptive thresholding scheme, similar to the one described in Section 3.1 (except in one dimension), to determine the peaks in the summed data.

The required one dimensional statistics for detecting scene-cuts are as follows:

$$\mu(t) = \frac{1}{N(B)} \sum_{k \in B} d(k). \quad (10)$$

$$\sigma^2(t) = \frac{1}{N(B)-1} \sum_{k \in B} (d(k) - \mu(t))^2. \tag{11}$$

Finally, we decide scene-cut according to the following thresholding criterion:

If $d(t) > \mu(t) + K\sigma(t)$, then the t -th frame is a scene cut. (12)

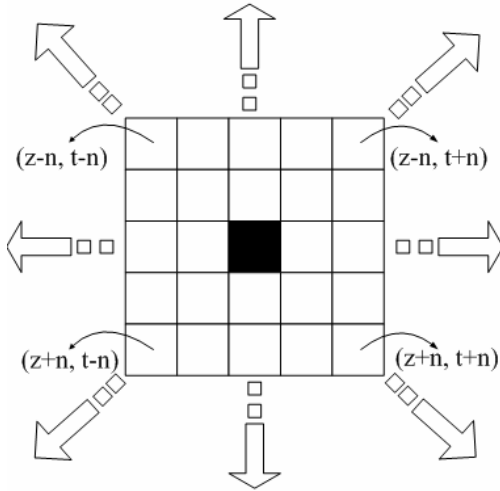


Fig. 7. Quadratic sliding window for the detection of peaks

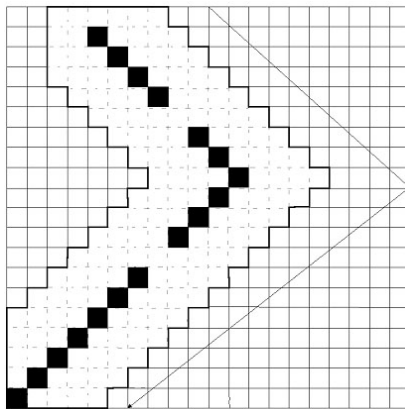


Fig. 8. A trace of the quadratic sliding window for the detection of a wipe

4 Experimental Results

To show the efficiency of the proposed algorithm, we compare with Yeo’s method [5] which is one of the well-known scene-cut detection algorithms. The method is similar

to the proposed one in that both algorithms make use of time derivative function and sliding window to get better detection of shot-change. According to [5], they form the difference sequence $D_t, t = 1, 2, \dots, N - 1$ as follows:

$$D_t = d(f_t, f_{t+1}) = |f(x, y, t) - f(x, y, t + 1)|. \quad (13)$$

where $f_t, t = 1, 2, \dots, N$ is a sequence of DC images, and (x, y) denotes pixel location of the DC image. They compare D_t with other difference sequences within a symmetric sliding window of size $2m-1$. They declare a shot-change from f_t to f_{t+1} , if D_t is the maximum within the sliding window, i.e., $D_t \geq D_j, j = t-m+1, \dots, t-1, t+1, \dots, t+m-1$.

The effectiveness of information retrieval systems is usually given by the well-known standard measures *recall* and *precision* [10]. The recall parameter defines the percentage of true detection (performed by the detection algorithm) with respect to the overall events (shot changes) in the video sequence. Similarly, the precision is the percentage of correct detection with respect to the overall declared event. The recall and precision are defined as

$$\begin{aligned} \text{Recall} &= \frac{N_c}{N_c + N_m}, \\ \text{Precision} &= \frac{N_c}{N_c + N_f}. \end{aligned} \quad (14)$$

where N_c, N_m and N_f indicate the number of correct detections, missed detections, and false detections, respectively.

The first experiment assesses the recall and precision of the detected shot-boundaries. Table 2 summarizes the contents of the test video sequences used in the experiments including total number of frames and scene-cuts.

Table 2. Video sequences types used in the experiment

| Sequences | Sequence type | Number of frames | Number of cuts |
|---------------|------------------|------------------|----------------|
| Dr. Dollittle | Movie | 56968 | 588 |
| CNN | News | 6350 | 41 |
| M.J. History | Music video | 16313 | 141 |
| French Kiss | Situation comedy | 14259 | 105 |

Fig. 9 shows the recall-precision plots for the proposed and Yeo's methods using Dr. Dollittle sequence. Each result is obtained by changing the required parameters and threshold values of the methods, and the results are verified manually by human observation. The result closest to the upper right-hand corner of the graph indicates the best performance. According to Fig. 9, the proposed method produces performances with higher recall and precision values. A high recall indicates the capability of detecting correct shot-changes, while a high precision indicates the capability of avoiding false detections.

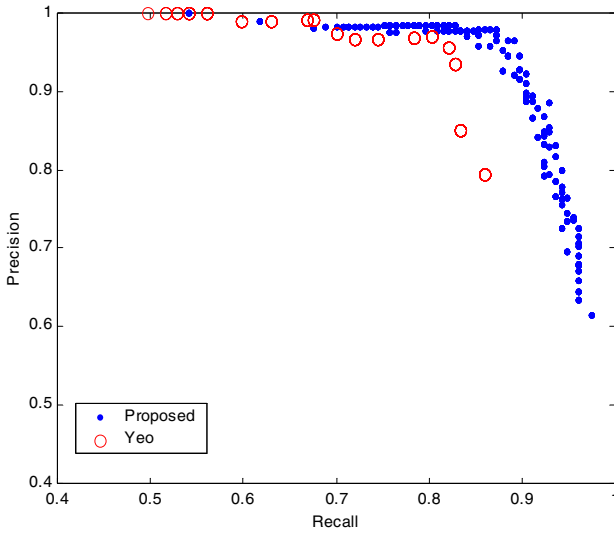


Fig. 9. Recall-precision curves for ‘Dr. Dollittle’ sequence

We show the recall-precision statistics for various test sequences in Table 3. The results of the table represent the best recall-precision combinations obtained by the proposed and Yeo’s methods.

Table 3. Comparison of recall-precision performances

| Sequences | Proposed method | | Yeo’s method | |
|---------------|-----------------|-----------|--------------|-----------|
| | recall | precision | recall | precision |
| Dr. Dollittle | 0.93 | 0.92 | 0.84 | 0.94 |
| CNN | 1.00 | 0.97 | 0.97 | 0.88 |
| M.J. History | 0.98 | 0.96 | 0.92 | 0.85 |
| French Kiss | 0.98 | 0.98 | 0.92 | 0.91 |

As shown in Table 3, the proposed approach outperforms Yeo’s in terms of recall and precision performance. For abrupt shot-change detection using the proposed method, the reported recall ranges between 0.93~1.0 and precision between 0.92~0.98.

The proposed method is demonstrated with a typical VRS containing wipes and cuts. Fig. 10 illustrates the typical scenario during the wipe and the scene-cut detection process. Fig. 10(a) shows the luminance of the Visual Rhythm. Note that there are seven scene-cuts (vertical lines) and five wipes. The absolute value of the time derivative of this Visual Rhythm is depicted in Fig. 10(b). Note that all shot-changes are clearly visible on the derivative image. Since it is easier to detect scene-cuts than wipes, we first detect and remove the abrupt changes. The detected abrupt changes are then used to nullify the derivative image. The resulting derivative image, without the abrupt changes (vertical lines), is depicted in Fig. 10(c). Successively, binary mask

$b(z,t)$ is formed using the sample mean and variance as described in Section 3. Fig. 10(d) shows the resulting binary mask, which is exactly the peaks of the image in Fig. 10(c). The algorithm then finds connected lines/curves that extend from the top to the bottom of the VRS. In this particular case, the recursive connectivity algorithm looked up to five pixels away from the current pixel, which corresponds to the quadratic window size of 10. Fig. 10(e) shows the connected lines that extend from the top to the bottom of the VRS. Fig. 10(f) shows the pixel count of the connected lines for each frame. The declaration of a wipe is based on this pixel count. Notice that all wipes are detected and that the duration of wipes is also determined using this approach. By following these procedures, we can finally obtain the exact wipe durations.

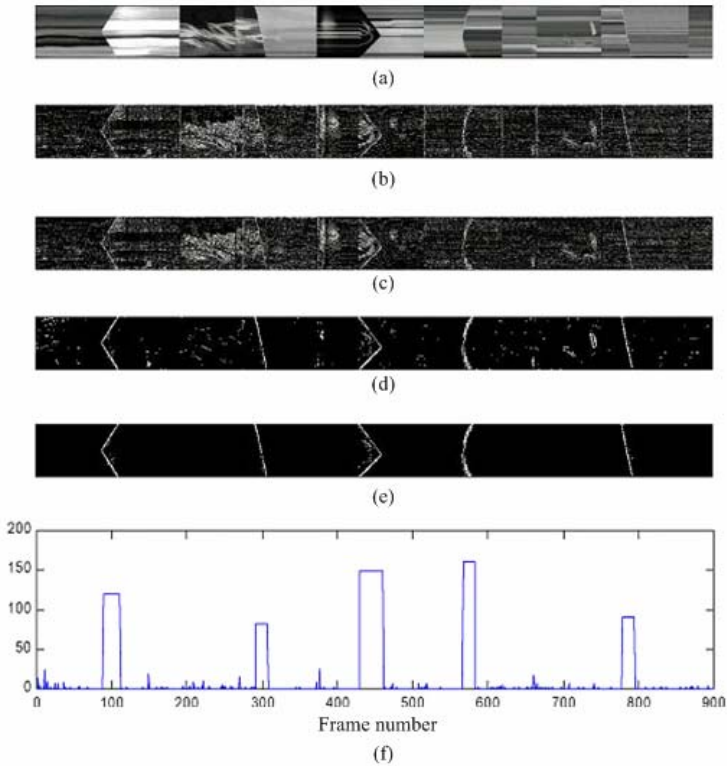


Fig. 10. Illustration of the wipe detector: (a) Visual Rhythm Spectrum, (b) absolute value of the time derivative image, (c) the derivative image without abrupt changes, (d) binary mask for wipe detection, (e) detected wipes, (f) connected pixel counts vs. frame number

5 Conclusion

In this paper, we proposed a new shot-change detection algorithm based on Visual Rhythm Spectrum (VRS) for database construction and systematic arrangement of multimedia data. The VRS contains distinctive patterns or visual features for various

types of shot changes. The different video effects manifest themselves differently on the VRS. In particular wipes appear as lines/curves that run from the top to the bottom of the VRS. Our proposed algorithm simply detects these curves. The proposed algorithm can be applicable not only to raw image but also to various DCT-based digital video standards including Motion JPEG, MPEG-1, MPEG-2, MPEG-4, and H.263.

The proposed algorithm can be extended to 3-D in a straightforward manner. Instead of processing the 2-D Visual Rhythm, which is a cut through a 3-D DC image sequence, the full 3-D DC image sequence may be processed. The wipes will then appear as surfaces of intensity discontinuity in the DC image sequence. However, the exact performance gain offered by 3-D processing is unclear at this point, especially considering the amount of additional processing required for the 3-D data sets.

Acknowledgement

This work was supported in part by Yonsei University Research Fund of 2005.

References

1. Chen S., Shyu M., Liao W., Zhang C.: Scene Change Detection by Audio and Video Clues. Proc. of IEEE International Conference on Multimedia and Expo, pp. 365-368, Aug. 2002.
2. Kim J., Suh S., Sull S.: Fast Scene Change Detection for Personal Video Recorder. IEEE Trans. on Consumer Electronics, Vol. 49, Issue 3, pp.683 – 688, Aug. 2003.
3. Lelescu D., Schonfeld D.: Statistical Sequential Analysis for Real-time Video Scene Change Detection on Compressed Multimedia Bitstream. IEEE Trans. on Multimedia, Vol. 5, Issue 1, pp. 106-117, March 2003.
4. Zabih R., Miller J., Mai K.: A Feature-based Algorithm for Detection and Classifying Scene Breaks. ACM Multimedia, pp. 189-200, 1995.
5. Yeo B., Liu B.: Rapid Scene Analysis on Compressed Video. IEEE Trans. on Circuits and Systems for Video Technology, Vol. 5, No. 6, pp. 533-544, Dec. 1995.
6. Meng J., Juan Y., Chang S.: Scene Change Detection in a MPEG Compressed Video Sequence. Proc. of SPIE Symposium on Digital Video Compression, pp. 14-25, 1995.
7. Calic J., Izquierdo E.: Efficient Key-frame Extraction and Video Analysis. Proc. of International Conference on Information Technology: Coding and Computing, pp. 28-33, April 2002.
8. Yeo B., Liu B.: On the Extraction of DC Sequence from MPEG Compressed Video. Proc. of International Conference on Image Processing, Vol. 2, pp. 260-263, Oct. 1995.
9. Song J., Yeo B.: Spatially Reduced Image Extraction from MPEG-2 Video: Fast Algorithms and Applications. Proc. of SPIE Storage and Retrieval for Image and Video Database VI, Vol. 3312, pp. 93-107, 1998.
10. Salton G.: Automatic Text Processing; the Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley Publishing Company, 1989.

Global Motion Estimation: Feature-Based, Featureless, or Both ?!

Rui F.C. Guerreiro¹ and Pedro M.Q. Aguiar²

¹ Philips Research, Eindhoven, Netherlands
rui.guerreiro@philips.com

² Institute for Systems and Robotics / Instituto Superior Técnico, Lisboa, Portugal
aguiar@isr.ist.utl.pt

Abstract. The approaches to global motion estimation have been naturally classified into one of two main classes: *feature-based methods* and *direct (or featureless) methods*. Feature-based methods compute a set of point correspondences between the images and, from these, estimate the parameters describing the global motion. Although the simplicity of the second step has made this approach rather appealing, the correspondence step is a quagmire and usually requires human supervision. In opposition, featureless methods attempt to estimate the global motion parameters directly from the image intensities, using complex nonlinear optimization algorithms. In this paper, we propose an iterative scheme that combines the *feature-based simplicity* with the *featureless robustness*. Our experiments illustrate the behavior of the proposed scheme and demonstrate its effectiveness by automatically building image mosaics.

1 Introduction

In this paper, we address the problem of estimating the global motion of the brightness pattern between a pair of images.

1.1 Motivation and State of the Art

Efficient methods to estimate global motion find applications in diverse fields. For example, in remote sensing and virtual reality, it is often necessary to build large images from partial views, *i.e.*, to register, or align, the input images. The key step for the success of these tasks is the estimation of the global motion between the images. In digital video, image alignment is also crucial, for stabilization and compression [1,2] and content-based representations [3].

Under common assumptions about the camera motion or the scene geometry, the motion of the image brightness pattern is described by a small set of parameters, see for example [4,5] for different parameterizations. In many situations of interest, the scene is well approximated by a plane, *e.g.*, when the relevant objects are far from the camera. In this case, the image motion is described by an 8-parameter homographic mapping. The homography also describes the image

motion when the scene is general, *i.e.*, unrestricted, but the motion of the camera is restricted to a (three-dimensional) rotation (an approximation is when the camera is fixed to a tripod), see for example [6,7]. In this paper, we address the estimation of the homographic mapping from a pair of uncalibrated input images.

Two very distinct approaches to homography estimation are found in the literature: *feature-based* methods estimate the homography by first matching feature points across the images; and *featureless, direct, or image-based*, methods estimate the homography parameters directly from the image intensity values. Other techniques include the use of integral projections [8] and Fourier transforms [9].

Feature-based methods, *e.g.*, [6,7,10], became popular due to the simplicity of the geometric problem of estimating the homography from the feature point correspondences. In fact, the homography parameters are linearly related to simple functions of the feature coordinates. Thus, given a set of point correspondences, the Least-Squares (LS) estimate of the homography parameters is obtained in closed-form by simply using a pseudo-inverse. The bottleneck of these methods is the feature correspondence step, which is particularly hard in several practical situations, *e.g.*, when processing low textured images [11,12].

Featureless methods, *e.g.*, [4,13,14], avoid pre-processing steps by attempting to estimate the homography directly from the images. Naturally, these methods lead to robust estimates. However, since the homography parameters are related to the image intensities in an highly nonlinear way, featureless methods use complex and time-consuming optimization algorithms, *e.g.*, Gauss-Newton, gradient descent. See [15,16] for a discussion on the feature-based/featureless dichotomy.

1.2 Proposed Approach

Our method splits the first image into four blocks (quadrants) and deals with each one as if it was a pointwise feature, *i.e.*, it determines the displacement of each quadrant by using standard correlation techniques. Then, it computes the homography described by these four displacements and registers the first image according to this homography. The registered image is naturally closer to the second image. The procedure is repeated (and the computed homographies are successively composed) until the displacement of each quadrant is zero, *i.e.*, until the registered image coincides with the second image.

Our approach combines the simplicity of the feature-based methods with the robustness of the featureless ones. It has the robustness of the image-based approaches because it matches the intensities in the entire image (rather than using only a subset of pointwise features). Our method is however much simpler than current featureless approaches because the nonlinear optimization is taken care of by using an iterative scheme where each iteration is as trivial as estimating the homography from feature point correspondences.

1.3 Paper Organization

In section 2, we introduce the notation needed to parameterize the image global motion in terms of an homographic mapping. Section 3 details the method we

propose in this paper to estimate the parameters of the homography describing the global motion between a pair of images. In section 4, we present experimental results that illustrate our approach and section 5 concludes the paper.

2 Global Motion Parameterization

An homography describing the global motion of the brightness pattern is characterized by an 8-parameter vector

$$\mathbf{h} = [a \ b \ c \ d \ e \ f \ g \ h]^T. \quad (1)$$

The homography parameter vector \mathbf{h} relates the coordinates (x_1, y_1) of a point in image \mathbf{I}_1 , with the coordinates (x_2, y_2) of the corresponding point in image \mathbf{I}_2 , through

$$\begin{cases} x_2(\mathbf{h}; x_1, y_1) = (ax_1 + by_1 + c) / (gx_1 + hy_1 + 1) \\ y_2(\mathbf{h}; x_1, y_1) = (dx_1 + ey_1 + f) / (gx_1 + hy_1 + 1). \end{cases} \quad (2)$$

For simplicity, in projective geometry, the vector \mathbf{h} is often re-arranged into a 3×3 homography matrix \mathbf{H} and the equalities in (2) are written in homogeneous coordinates $\mathbf{p} = [x, y, 1]^T$:

$$\begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} \propto \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \Leftrightarrow \mathbf{p}_2 \propto \mathbf{H}\mathbf{p}_1, \quad (3)$$

where \propto represents the projective space equality, *i.e.*, it denotes equal up to a scale factor [6,7].

Re-arranging (2,3), we see that the homography parameters in \mathbf{h} , or in \mathbf{H} , are linearly related to simple functions of the image coordinates. In fact, (2,3) can be written as

$$\Psi \mathbf{h} = \psi, \quad (4)$$

where

$$\psi = \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}, \quad (5)$$

and

$$\Psi = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_2x_1 & -x_2y_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y_2x_1 & -y_2y_1 \end{bmatrix}. \quad (6)$$

Given the correspondences of a set of N feature points, *i.e.*, given the set of pairs of coordinates $\{(x_1, y_1)^i, (x_2, y_2)^i, i = 1, \dots, N\}$, the LS estimate of the homography parameter vector \mathbf{h} is easily obtained by using the pseudo-inverse to invert the set of $2N$ linear equations like (4). Since vector \mathbf{h} is 8-dimensional (1), at least 4 feature points are required to unambiguously determine the homography between the images. In this case, which is the relevant one for the algorithm

we propose in this paper, the homography describing the global motion of the 4 feature points, is simply obtained as

$$\mathbf{h} = \begin{bmatrix} \Psi_1 \\ \Psi_2 \\ \Psi_3 \\ \Psi_4 \end{bmatrix}_{8 \times 8}^{-1} \begin{bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \end{bmatrix}_{8 \times 1}, \quad (7)$$

where Ψ_i and ψ_i denote matrices and vectors defined as in (4), now computed with the coordinates of each feature point i .

A reader familiar with feature-based methods may note that usually the number of feature points is much larger than the minimum 4 required and the estimate of the homography is usually computed from the singular value decomposition of a matrix that collects the observations, rather than from the generalization of (7). In opposition, we use the simple closed-form solution in (7) for 4 features, because, as it will become clear in the sequel, the robustness of our method comes from using the intensity values in the entire images, rather than from using an huge number of pointwise features.

3 Global Motion Estimation

Although the homography describing the global motion is easily estimated from point correspondences using (7), pointwise features are difficult to match from image to image in an automatic way. This inspired us to develop an algorithm that overcomes the difficulty without resorting to time-consuming nonlinear optimization, unlike current image-based, or featureless, methods. This section describes our algorithm.

3.1 Feature-Based, Featureless, or Both ?!

Rather than attempting to match hundreds of pointwise features, we use the minimum possible number, *i.e.*, four, but with a much larger dimension—each feature occupies one quadrant of the first image \mathbf{I}_1 . This way we take into account all the intensity levels of the entire images. Our method proceeds by matching each of these four feature blocks to the second image \mathbf{I}_2 , using standard correlation techniques. Naturally, this results in a very rough matching, unless the global motion between \mathbf{I}_1 and \mathbf{I}_2 is a pure translation. Then, using expression (7) with the coordinates of the centers of the feature blocks¹, we obtain a first estimate $\hat{\mathbf{h}}_1$ of the homography describing the global motion between images \mathbf{I}_1 and \mathbf{I}_2 .

Now, apply the homographic mapping characterized by the estimate $\hat{\mathbf{h}}_1$, to the first image \mathbf{I}_1 . If $\hat{\mathbf{h}}_1$ was an accurate estimate, this would align \mathbf{I}_1 with \mathbf{I}_2 .

¹ Due to the chosen location for the feature blocks, the matrix inversion in (7) is well conditioned. It is singular only in degenerate situations, *e.g.*, when the centers of three blocks become colinear or the centers of two blocks collapse into a point.

Since in general $\widehat{\mathbf{h}}_1$ will be a rough estimate, applying this homography to \mathbf{I}_1 will generate an image \mathbf{I}' that is just “closer” to being aligned with \mathbf{I}_2 . This is the fact exploited by our algorithm—we now use \mathbf{I}' as the first input image and proceed again as just described, now to estimate \mathbf{h}' , the homography describing the motion between \mathbf{I}' and \mathbf{I}_2 . The estimate $\widehat{\mathbf{h}}'$ will be more accurate than $\widehat{\mathbf{h}}_1$ because the corresponding input images are closer to being aligned, thus the block features (the image quadrants) will be better matched.

To obtain the second estimate $\widehat{\mathbf{h}}_2$ of the homography between the original image \mathbf{I}_1 and \mathbf{I}_2 , we just need to combine the first estimate $\widehat{\mathbf{h}}_1$ with the update $\widehat{\mathbf{h}}'$. From (3), we see that this operation is easily expressed using the homogeneous representation of the homographies in terms of the corresponding matrices $\widehat{\mathbf{H}}_2$, $\widehat{\mathbf{H}}_1$, and $\widehat{\mathbf{H}}'$:

$$\widehat{\mathbf{H}}_{i+1} \propto \widehat{\mathbf{H}}_i \widehat{\mathbf{H}}', \quad (8)$$

where, at this point, $i = 1$. The process is repeated until the displacements of the block features is zero (in practice, until they are below a small threshold). The update matrix $\widehat{\mathbf{H}}'$ converges to the identity $\mathbf{I}_{3 \times 3}$ and the sequence of estimates $\widehat{\mathbf{H}}_i$ converges to the homography describing the global motion between images \mathbf{I}_1 and \mathbf{I}_2 .

This paragraph summarizes our claims relative to the approach just described. Like the feature-based methods, our approach exploits the fact that the homography is easily estimated from a set of spacial correspondences but, unlike those, it avoids matching hundreds of pointwise features. Like the image-based methods, we match the intensity levels in the entire images, but, unlike those, we avoid complex and time-consuming optimization algorithms.

3.2 Multiresolution Processing

Naturally, attempting to match large regions, such as the image quadrants, with a simple translational motion model may lead to a very poor match when the global motion is far from a pure translation. As a consequence, in such situations, the algorithm described above may exhibit slow convergence or even get stuck at a point that does not correspond to the true solution. To speedup the convergence and better cope with global motions far from pure translations, we use a coarse-to-fine strategy.

We build a multiresolution pyramid [17] and start running the algorithm at its coarsest level. The estimate of the homography at each level is used to initialize the algorithm at the following (finer) level. The final estimate of the homography is then obtained at the finest level of the pyramid, *i.e.*, at the full resolution of the input images. The loss of detail at the coarser levels of the pyramid is what enables a fast convergence to the true solution, even with a model as simple as the pure translation for the motion of each quadrant.

3.3 Summary of the Algorithm

In short, our method computes the homography describing the global motion between images \mathbf{I}_1 and \mathbf{I}_2 through the following steps (image coordinates are normalized such that $x, y \in [0, 1]$):

- 1- Build multiresolution pyramids for \mathbf{I}_1 and \mathbf{I}_2 . Initialize the resolution to its coarsest level $l = 0$, the iteration number $i = 0$, and the estimate of the homography $\widehat{\mathbf{H}}_0 = \mathbf{I}_{3 \times 3}$.
- 2- Apply the current homography estimate $\widehat{\mathbf{H}}_i$ to image \mathbf{I}_1 at resolution l , obtaining \mathbf{I}' .
- 3- Split \mathbf{I}' into its four quadrants and compute their displacements that best match image \mathbf{I}_2 at resolution l , using standard correlation techniques.
- 4- Compute the vector \mathbf{h}' that describes the motion of the centers of the image quadrants, using (7).
- 5- Update the homography estimate $\widehat{\mathbf{H}}_{i+1}$ by composing the previous estimate $\widehat{\mathbf{H}}_i$ with $\widehat{\mathbf{H}}'$ through (8).
- 6- If the displacements of the quadrants are below a small threshold, increase the resolution level, $l = l + 1$ (if it was already the full resolution of the input images, then stop).
- 7- Increase the iteration number, $i = i + 1$, and go to 2-.

4 Experiments

In this section, we describe experiments that illustrate the efficiency of the proposed approach.

4.1 Chess Table Images

The first experiment illustrates the fast convergence of the algorithm by comparing the images before and after the first iteration. We applied an homographic mapping to an image of a chess table, obtaining an highly distorted version of it, see the two images on the top of Fig. 1. These images were the input to our algorithm. Superimposed with them, we represent the initial (very rough) correspondences of the 4 quadrants of the image. Using these 4 correspondences, our algorithm computes the corresponding homography and registers the first input image according to it. This leads to the bottom left image. Note how closer to the second input image is the bottom left one, when compared to the first input image (top left one). The displacements of the new blocks are now much smaller—see the rectangles superimposed to the bottom images of Fig. 1. Our experience has shown that they converge to zero in a very fast way, which, in turn, leads to a fast convergence of the estimated homography between the two images.

In order to demonstrate the performance of the proposed algorithm with real video, we built several mosaics from video sequences, in a fully automatic way. Our system estimates the homographies between successive video frames and composes them to align all the images. Then, the intensity of each pixel of the mosaic is computed by averaging the intensities of the frame pixels that overlap at the corresponding position. We now illustrate with two of these mosaics.

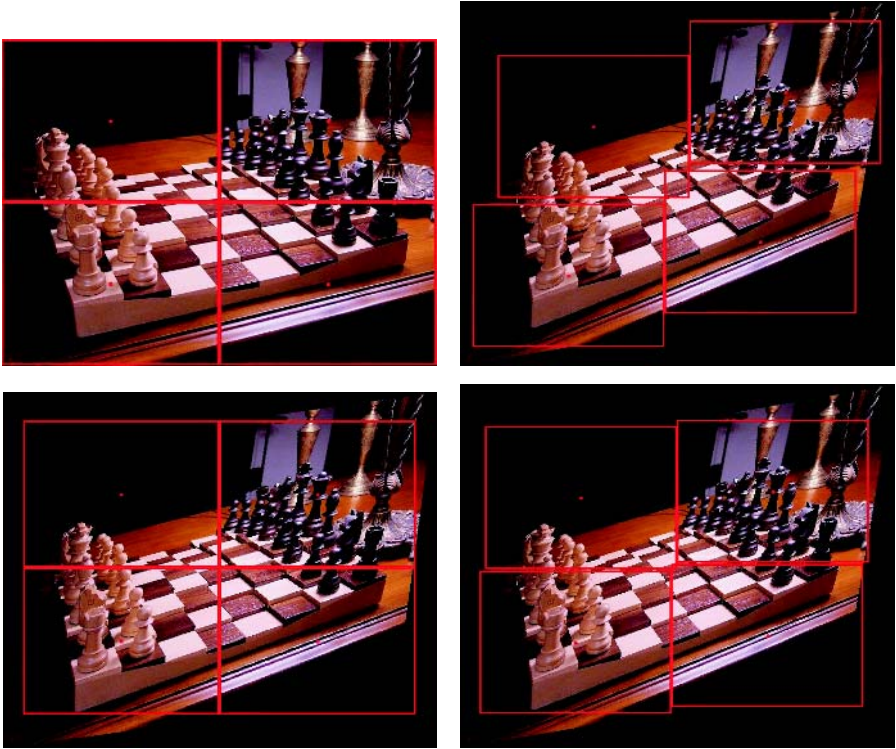


Fig. 1. Behavior of the proposed algorithm. Top line: images to register, with correspondences of the four quadrants superimposed. Bottom line: the same, after the first iteration. Notice how the left image approximates the right one.



Fig. 2. Towel video sequence: three sample frames

4.2 Beach Towel Video Sequence

We used several 640×480 images showing partial views of a beach towel, see the three sample images in Fig. 2. From our experience with images of this size, it suffices to use a multiresolution pyramid with four resolution levels. The number of iterations needed to estimate each homography was very small, typically less



Fig. 3. Mosaic recovered from the towel video sequence in Fig. 2



Fig. 4. Carpet video sequence: eight sample frames

than 5. The recovered mosaic, shown in Fig. 3, illustrates that the homographies estimated by our algorithm are accurate and appropriate for this kind of application.

4.3 Carpet Video Sequence

To demonstrate the performance of our method when dealing with long video sequences, we used a 512×512 video stream with 26 frames of a carpet. Sample images are shown in Fig. 4. Although these images, obtained with an ordinary

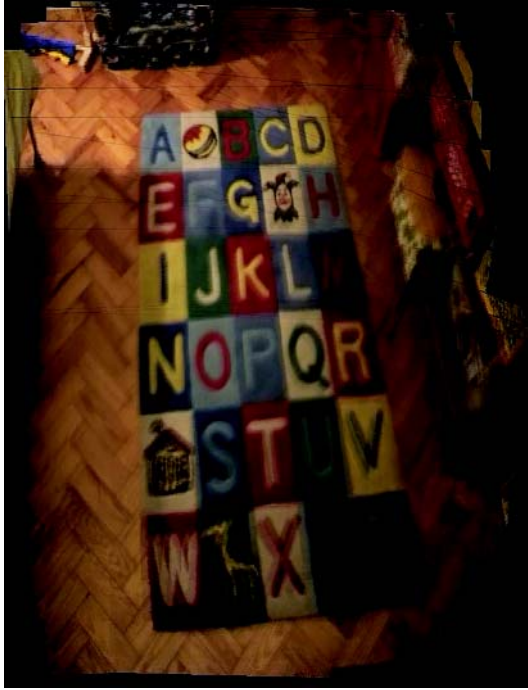


Fig. 5. Mosaic recovered from the carpet video sequence in Fig. 4

camera, are rather noisy due to video compression and fast camera movement, the recovered mosaic, shown in Fig. 5, confirms the good performance of our method. Note that feature-based methods usually fail to automatically align low contrast images, as the ones we use in this experiment, because it is very hard to compute the correspondences of pointwise features in this scenario.

5 Conclusion

We proposed a new method to estimate the homography describing the global motion between a pair of images. Our method combines the simplicity of the feature-based approaches with the robustness of the featureless ones. We illustrate the efficiency of the proposed algorithm when building, in a fully automatic way, image mosaics from uncalibrated video streams.

Acknowledgment

This work was partially supported by the (Portuguese) Foundation for Science and Technology, under grant # POSI/SRI/41561/2001.

References

1. Dufaux, F., Konrad, J.: Efficient, robust, and fast global motion estimation for video coding. *IEEE Trans. on Image Processing* **9**(3) (2000) 497–501
2. Petrovic, N., Jojic, N., Huang, T.: Hierarchical video clustering. In: *Proc. of IEEE Multimedia Signal Processing Workshop, Siena, Italy* (2004)
3. Aguiar, P., Jasinschi, R., Moura, J., Pluempitiwiriyawej, C.: Content-based image sequence representation. In Reed, T., ed.: *Digital Video Processing*. CRC Press (2004) 7–72 Chapter 2.
4. Mann, S., Piccard, R.: Video orbits of the projective group: a simple approach to featureless estimation of parameters. *IEEE Trans. on Image Processing* **6**(9) (1997) 1281–1295
5. Kim, D., Hong, K.: Fast global registration for image mosaicing. In: *Proc. of IEEE Int. Conf. Image Processing, Barcelona, Spain* (2003)
6. Faugeras, O.: *Three-Dimensional Computer Vision*. MIT Press, Cambridge, MA, USA (1993)
7. Hartley, R., Zisserman, A.: *Multiple View Geometry in Computer Vision*. Cambridge University Press (2000)
8. Lee, J., Ra, J.: Block motion estimation based on selective integral projections. In: *Proc. of IEEE Int. Conf. Image Processing, Rochester NY, USA* (2002)
9. Reddy, B., Chatterly, B.: An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE Trans. on Image Processing* **5**(8) (1996) 1266–1271
10. Perez, P., Garcia, N.: Robust and accurate registration of images with unknown relative orientation and exposure. In: *Proc. of IEEE Int. Conf. Image Processing, Genova, Italy* (2005)
11. Shi, J., Tomasi, C.: Good features to track. In: *IEEE Int. Conf. on Computer Vision and Pattern Recognition*. (1994)
12. Aguiar, P., Moura, J.: Image motion estimation – convergence and error analysis. In: *Proc. of IEEE Int. Conf. on Image Processing, Thessaloniki, Greece* (2001)
13. Altunbasak, Y., Merserau, R., Patti, A.: A fast parametric motion estimation algorithm with illumination and lens distortion correction. *IEEE Trans. on Image Processing* **12**(4) (2003)
14. Pires, B., Aguiar, P.: Featureless global alignment of multiple images. In: *Proc. of IEEE Int. Conf. Image Processing, Genova, Italy* (2005)
15. Irani, M., Anandan, P.: About direct methods. In: *Vision Algorithms: Theory and Practice*. Volume 1883 of Springer Lecture Notes in Computer Science. (1999)
16. Torr, P.: Feature based methods for structure and motion estimation. In: *Vision Algorithms: Theory and Practice*. Volume 1883 of Springer LNCS. (1999)
17. Rosenfeld, A., ed.: *Multiresolution Image Processing and Analysis*. Volume 12 of Springer Series in Information Sciences. Springer-Verlag (1984)

Background Updating with the Use of Intrinsic Curves^{*}

Joaquín Salas^{1,2}, Pedro Martínez¹, and Jordi González³

¹ CICATA Querétaro-IPN

² CVC

³ UPC-CSIC

Abstract. A primary tool to extract information about moving objects is background subtraction. In this technique, the difference between a model of what is static, or background, and the current image of the scene gives information about what is in the prime plane or foreground. This study focus on the pixelwise updating mechanism of the background model throughout the analysis of the images provided by a fixed camera. The concept of intrinsic curves, early introduced in the field of stereovision, is extrapolated to the problem of detecting the moving boundaries. We use a mixture of Gaussians to register information about the recent history of the pixel dynamics. Our method improves this model in two ways. Firstly, it reduces the chances of feeding the mixture of Gaussians with foreground pixels. Secondly, it takes into account not just the scalar pixel value but a richer description of the pixel's dynamics that carries information about the interpixel variation. Ample experimental results in a wide range of environments, including indoors, outdoors, for a different set of illumination conditions both natural and artificial are shown.

1 Introduction

Nowadays, there is an increasing demand for automatic monitoring systems based on image analysis fueled by factors such as the increasing existence of imaging devices, our own limitations to watch every available video source, and the development of important multimedia applications [18]. Motion is a basic capability of visual perception systems. Many important higher perceptual tasks can be built on top of it, including tracking, and recognition. A primary tool to extract information about moving objects is background subtraction. In this technique, the difference between a background model and the current image of the scene gives information about what is in the prime plane or foreground. Depending on the definition of background, its general solution will involve the distinction of foreground objects even when the background objects are moving [17], and indeed, even when the camera itself is moving [1]. However, there is a fairly large number of scenarios where the constraint of a static background imaged with static cameras is rather useful and interesting applications can be developed.

Piccardi [16] proposes a classification of background construction methods based on speed, memory requirements and accuracy. Functionally, the problem can be divided

^{*} This work has been partially supported by the SIP-IPN grant 20061072, and EC grant IST-027110 for the HERMES project and by the Spanish MEC under projects TIC2003-08865 and DPI-2004-5414. J. González also acknowledges the support of a Juan de la Cierva Postdoctoral fellowship from the Spanish MEC.

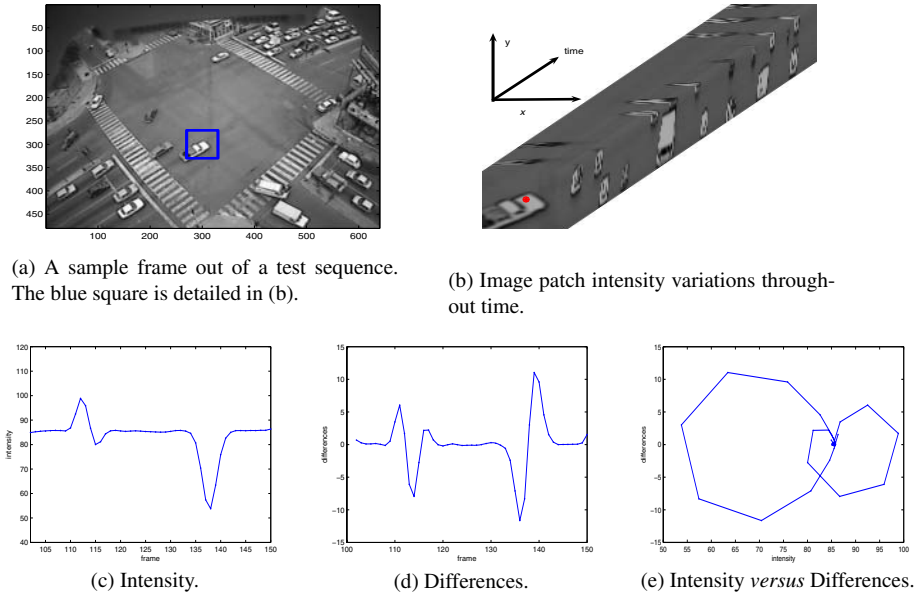


Fig. 1. A pixel is observed during 40 frames. The space-time representation of the image patch in (a) is shown in (b). The patch central pixel intensity variations(c) and differences(d), generate the intrinsic curve in (e).

into several interesting ones. Consider for instance, the initialization, or how to arrive to the initial background model. Since it may be seen as a classification problem some researchers have used optical flow to either formulate hypothesis[5], train Neural Networks[4], or use Support-Vector Machines [12]. Another problem is to reduce the time to figure out the background revealed by a moving foreground object. Kaup and Aach[10] exploit spatiotemporal correlation, and motion information. The surrounding known background area is analyzed and spatially extrapolated using an spectral domain extrapolation algorithm. These techniques are rather important for applications such as videoconferencing[13] where background coding for efficient compression and transmission is needed. The problem of updating the background model has many interesting facets. Since deciding a pixel classification based solely on the base of a single threshold may be too limiting, Kumar *et al.*[11] suggest the use of a hysteresis threshold technique. After a decision has been taken about what is background and what is foreground, there is still space for improving the results. Filling holes, dilating borders are among the usual strategies used. He *et al.*[7] present a background updating algorithm which combines the variation of a neighboring area with the difference between the current and previous values in order to predict the new values of the pixels on the background. Nowadays, it is widely accepted that color can be used to identify the shadows casted by objects[2]. Horprasert[8] designs a color model that separates the brightness from the chromaticity component. Their conclusion is similar to Han *et al.*[6] whom propose an algorithm to deal with gradual illumination changes. To cope with

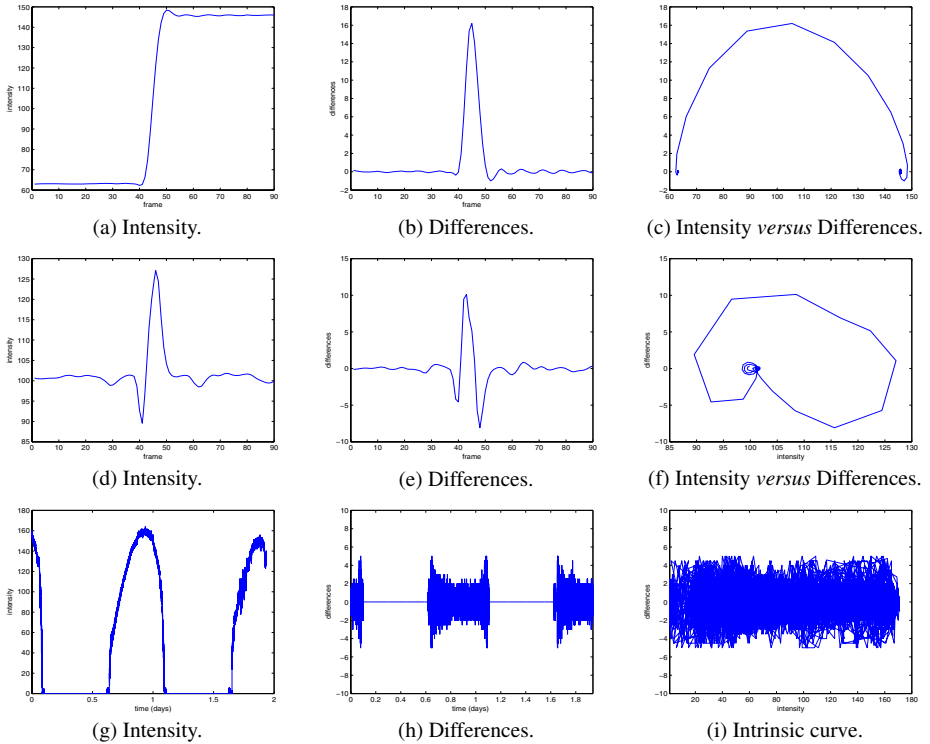


Fig. 2. The different classes of intrinsic curves that can be observed. In (a)-(c) the stability zone moves from one place to another. In (d)-(f) the curve comes back to the original stability zone after the occluding object pass by. In (g)-(i) the stability zone extends smoothly on the intensity axis. The first two sets of figures correspond to a few dozens of frames. The last set corresponds to two days of observations of the same pixel.

the problem of shadows, they propose a color model where chromaticity distortion is measured.

In this paper, we introduce a technique for background updating that uses intrinsic curves[21]. Intrinsic curves are N -values functions whose components are obtained from applying operators P_n to the intensity variation over time. This technique works at pixel level. It can be complemented with the knowledge generated at region level[25] or scene level[22]. The advantage of treating each pixel independently is that this gives a lot of flexibility at the expense of greater variability due to noise. For cases where the illumination conditions are controlled and there is the opportunity for observing the scene without foreground objects a simple median filter[24] could be used. However, this is rarely the case. Most common is the fact that the background image has to be adjusted as the time pass by. Huang *et al.*[9] address the problem of separating foreground from background in color images. They assume that background regions are relatively smooth but may have gradually varying colors or slightly textured. Voting may be another strategy to define the background pixel value from a set of samples.

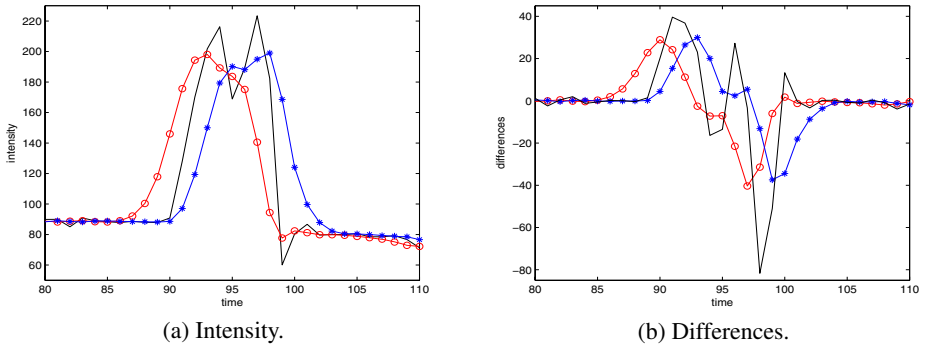


Fig. 3. Convolution with a Half-Gaussian that takes into account only the values before the current position (red-circled line) or the values after the current position (blue-starred line)

For a given pixel, Tai and Song [20] propose to accumulate the intensity value observations into a histogram. The most frequent value is used as an estimate of the background value. They apply the proposed method to a vision-based traffic monitoring system to segment moving vehicles from traffic image sequences. In a seminal paper, Stauffer and Grimson[19] proposed to model each pixel as a mixture of Gaussians. The gaussian distributions of the adaptive mixture model are evaluated to determine which are most likely to result from a background process. Each pixel is classified based on whether the Gaussian distribution which represent it most effectively is considered part of the background model. This model has been adapted widely, and studied extensively. For instance, Wayne and Schoonees[15] developed a tutorial paper to describe a practical implementation of the Stauffer-Grimson algorithm and provide values for all model parameters. In their document, they show what approximations to the theory were made and how to improve the standard algorithm by redefining those approximations.

In the rest of the paper, we introduce a technique to detect moving objects in a scene with static background imaged from a fixed camera. We focus on the background update stage, assuming that initialization, shadows, uncovered background, and region and frame level analysis can be deal with using some of the techniques mentioned before. In §2, we review the intrinsic curve paradigm and illustrate how it is useful to detect moving boundaries. Then, in §3, the background detection approach using mixture of Gaussians is visited. There, we show how both ideas can be blended. Next, in §4, we present experiments with several image sequences under an ample set of conditions. Finally, we summarize our results and outline future research.

2 Detecting Moving Boundaries

In this section, it is introduced an algorithm to distinguish, at pixel level, moving from static surfaces by analyzing a pixel dynamics in a particular intrinsic curve space.

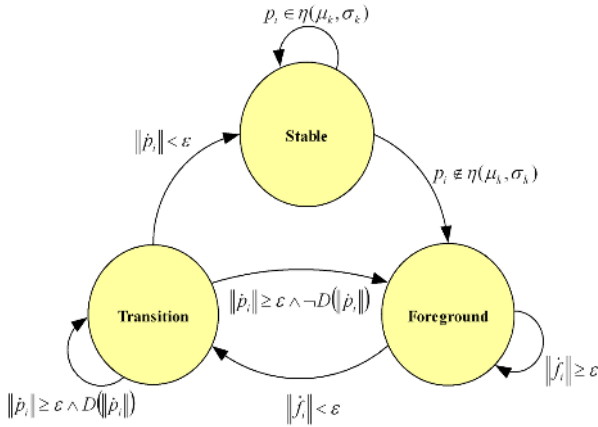


Fig. 4. State diagram for a pixel. The conditions to change state are commented within the text.

2.1 Intrinsic Curves

In the context of stereovision, Tomasi and Manduchi [21] proposed the description of scanlines in terms of a set of operators M_n . Thus, an intrinsic curve is a curve $\mathbf{m}(t) = \{m_n(t), t \in \mathbf{R}\}$ in \mathbf{R}^N parameterized by the real variable t such that \mathbf{m} is generated by applying N operators M_n to the original signal $m_0(t)$ to produce the new signals $m_n(t)$. Tomasi and Manduchi were specially interested in diffeomorphisms, *i. e.*, mappings between manifolds which are differentiable and have a differentiable inverse[23]. In particular, as Tomasi and Manduchi did in their paper, in this study is interesting to apply operators of the form

$$m_n(t) = [M_n m_0](t) = \frac{d^n}{dt^n} m_0(t). \tag{1}$$

Let us focus on a single pixel intensity dynamics throughout time. Suppose that it has a background intrinsic curve model $\alpha(t)$ while $\beta(t)$ is the intrinsic curve for the values being observed. Tomasi and Manduchi noted that in the intrinsic curve space, noise apart, both curves will follow the same trajectory. However, during an occlusion, they will experience a significant deviation of one respect to the other. When the occlusion ends, both curves will meet again in a place that depends on whether the occluding object became part of the background. This idea is illustrated in Fig. 1. In general, an intrinsic curve that belongs to the background is defined within what we call a *stability zone*. Here, stable corresponds to the notion that the observed curve values are drawn from a stable distribution[14]. For simplicity, we assume that the stable distribution follows a Gaussian model given by

$$h(\mathbf{x}) = \frac{1}{(2\pi)^{N/2} \|\Sigma\|^{1/2}} \exp \left[-\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right], \tag{2}$$

with $\mathbf{x} = \mathbf{m}(t)$, the observed intrinsic curve, and $\mu = [\bar{m}_0, \dots, \bar{m}_{N-1}]$, the intrinsic curve components mean value, and covariance matrix Σ . Background occlusions due

to foreground objects are rather difficult to describe in terms of a parametric distribution. Intrinsic curves help us to distinguish between both.

We have noted three kinds of intrinsic curves: *leaving*, *returning*, and *steady*. The *leaving* case occurs when the foreground objects integrates into the background (illustrated in Fig 2(a)-(c)). This is the typical case, for instance, of a car parking. Imagine that a intrinsic curve is in the stability zone. When the car occludes the background the curve departs from it. Then, when the car finally parks, the curve finds, in general, another stability zone and the parked car is integrated into the background model. In the *returning* case (illustrated in Fig 2(d)-(f)), the curve remains stable until an object occludes the background. At that moment, the curve leaves the stability zone. When the object passes, the curve comes back to its original stability zone. The previous two cases occur when the background is occluded. The *steady* case describes the long term behavior of a pixel that belongs to the background. This event occurs, for instance, when the illumination changes smoothly as the day passes by. Fig. 2(g)-(i) show two days of observation of the same pixel location. As the day passes by the illumination increases and decreases smoothly.

2.2 Detecting Motion

To detect moving boundaries using intrinsic curves, the prime problem is to distinguish the stability zone. The pixel will be classified as being part of the background until it leaves the stability zone, and during that time it will be part of the foreground class. It will return to the background class once it stabilizes again. In practice, image noise makes it hard to classify the state of a pixel. Let the pixel intensity $m_0(t)$ at time t be described by $m_0(t) = n(t) + \xi(t)$, where $n(t)$ is the true image intensity value and $\xi(t)$ is the image noise. Traditionally, one way to remove this noise has been by using low pass filters. However, applying filters across region boundaries may result in undesirable results because regions belong to different populations. To remove noise, a zero-mean Gaussian filter is applied but with a twist. Half-Gaussian are used to remove the noise, such that operators

$$g_p(t) = \begin{cases} \sqrt{\frac{2}{\pi\sigma^2}} \exp\left(-\frac{t^2}{2\sigma^2}\right) & -\infty < t \leq 0 \\ 0 & \text{otherwise} \end{cases}, \text{ and } g_f(t) = \begin{cases} \sqrt{\frac{2}{\pi\sigma^2}} \exp\left(-\frac{t^2}{2\sigma^2}\right) & 0 \leq t < \infty \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

are applied simultaneously. The original signal $m_0(t)$ is thus filtered out resulting in two estimates, $p_0(t)$ and $f_0(t)$, that take into account only frames before and after the current one, respectively, such that

$$p_0(t) = m_0(t) * g_p(t), \quad \text{and} \quad f_0(t) = m_0(t) * g_f(t). \tag{4}$$

Let $m(k)$ correspond to the observation of a pixel in position \mathbf{x} at time k . The sequence is smoothed using as many frames as required by the filter window width. Let us call k the present time. Half of the filter’s window include values corresponding to the previous frames and half of it values corresponding to the next frames. After smoothing, there will be signals $p_0(k)$ and $f_0(k)$ corresponding to estimates of the true pixel intensity. Intrinsic curve descriptors for the i -order difference can be computed using the estimates of the $(i - 1)$ -order. This way, there will be an intrinsic curve description that

considers the previous frames, $\mathbf{p}(k) = \{p_0(k), p_1(k), \dots, p_{N-1}(k)\}$, and another considering the next ones, $\mathbf{f}(k) = \{f_0(k), f_1(k), \dots, f_{N-1}(k)\}$. As it is illustrated in Fig. 3, \mathbf{p} and \mathbf{f} are useful to detect respectively the beginning and ending of an occlusion.

3 Adding Memory

Stauffer and Grimson[19] proposed one of the most popular methods to estimate the model of the intensity variations for an image sequence took from a fixed camera. Their method is based on computing on-line the parameters of a mixture of Gaussians (MOG) for each individual pixel. In this method, a given pixel intensity value is classified as either part of the background or the foreground depending on whether the value is likely to be interpreted by the respective statistical model. In practice, the method generates between 3 and 5 Gaussians. Intrinsic curves aims to improve this model in two ways. On the one hand, they do not feed the mixture of Gaussians with foreground pixels. On the other hand, they take into account not just the scalar pixel value but a richer description of the pixel's dynamics that carries information about the interpixel variation. In this section, we complement the intrinsic curve model with the addition of memory capabilities that could be used to describe what has been occurring in the past. First, the MOG model is described. Then, MOG and intrinsic curves are combined into a single paradigm.

3.1 Mixture of Gaussians

MOG aims to model the image of a dynamic scene as perceived from a fixed camera by a set of Gaussians. Given a set of n points in one dimension, $x_1, \dots, x_n \in \mathbf{R}$, and a family \mathcal{F} of probability density functions on \mathbf{R} , the problem is to find the probability density $f(x) \in \mathcal{F}$ that is most likely to have generated the given points. In this method, each member of the family \mathcal{F} has the same general Gaussian form. Each member is distinguished by different values of a set of parameters θ . That is

$$f(x; \theta) = \sum_{k=1}^K q_k g(x; \mu_k, \sigma_k), \quad \text{where} \quad g(x; \mu_k, \sigma_k) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp \left[-\frac{1}{2} \left(\frac{x - \mu_k}{\sigma_k} \right)^2 \right], \quad (5)$$

is a 1-dimensional Gaussian function and $\theta = (\theta_1, \dots, \theta_K) = ((q_1, \mu_1, \sigma_1), \dots, (q_K, \mu_K, \sigma_K))$, is a $3K$ -dimensional vector containing the mixing probabilities q_k as well as the means μ_k and standard deviations σ_k of the K Gaussian functions in the mixture. When a new observation x_t is available, it is compared again the parameters of the Gaussian models. If $\|x - \mu_k\| \leq \alpha \sigma_k$, then it is assumed that the observation is likely to be produced by a perturbation of the true value similar to the one expressed by the k -model. Typically, α is chosen to be 3, meaning that x is within 99.73% of the cases occurring under this model. If an observation occurs that can not be explained by the current set of Gaussians, a new model with high variance and centered around the observation is initialized. This model becomes part of the mixture of Gaussians. Otherwise, the new observations helps to learn the true value of the parameters. Most commonly the learning process followed is called Estimation-Maximization (EM). In the first place a lower bound function $b_i(\theta)$ is claimed to approximate f . Then the parameters that

maximize this function are found. Since not all the observations are available at once, an on-line version of the EM algorithm uses the following set of equations.

$$\mu_{t+1} = \rho\mu_t + (1 - \rho)x_t, \quad \text{and} \quad \sigma_{t+1}^2 = \rho\sigma_t^2 + (1 - \rho)(x_t - \mu_{t+1})^2, \quad (6)$$

where $\rho \in [0, 1]$ is the learning rate.

3.2 Estimating the Background Model

In our model, a pixel can be in anyone of three states (see Fig. 4): *stable*, *transition*, and *foreground*. To reduce the memory space and computing demands, let $\mathbf{p}(k) = (p_k, \dot{p}_k)$ and $\mathbf{f}(k) = (f_k, \dot{f}_k)$ be respectively the estimate for the intrinsic curves for a particular pixel at time k using the samples before and after the current frame. Initially, a pixel is classified in the *transition* state. The change of state is governed by the following rules:

- A *transition* pixel will stay like that if $\|\dot{p}_k\| > \epsilon$, and $\|\dot{p}_k\|$ is showing a decreasing tendency. Otherwise, if $\|\dot{p}_k\| > \epsilon$ but $\|\dot{p}_k\|$ does not show a decreasing tendency the pixel will change state to *foreground*. A *transition* pixel will change its state to *stable* if $\|\dot{p}_k\| < \epsilon$.
- An *stable* pixel will remain in that state provided \mathbf{p} is in the stability zone. Otherwise, the state will change to *foreground*.
- A *foreground* pixel will stay like that if \mathbf{f} is not in the stability zone. Otherwise, its state will change to *transition*.

The decreasing tendency function $D(x_i)$ is used to solve the problems caused by temporal aliasing. When an object has moving parts that occlude and accretes the background too fast, the noise filters receive samples from different distributions and the result is not accurate. A similar situation occurs when objects move too close to each other. The filter's window spans both objects mixing up their respective populations. However, in those cases \mathbf{f} returns for a moment to a stability zone and then bounces out in response to the following occluding object. If during this time \mathbf{p} keeps approaching the stability zone then it can be assumed that the pixel is a *stable* one.

A pixel classified as stable is not necessarily part of the background. That is, imagine that a large, homogeneous object is moving on top of a static background. The frontal part of the object will be detected as a moving region as well as the rear part. However, once a pixel enters the stable state, it will start building a model for the variation of the intensity of this pixel. What is being produced is a set of layers that model the pixel dynamics. So the criteria that we have used is the following one. If there is a pixel with more that one Gaussian description and one of the Gaussians has been used in the current frame, *i.e.*, the pixel is in the stable state, then that pixel is considered part of the foreground. This strategy has proved to be useful for instance when detecting leftover objects. Imagine for instance a parking lot. When a new car parks, its figure outstand because although it stabilizes, a different background was there before. A consequence of this is that when a parked car leaves, its silhouette will remain. So it really depends on what was the previous state of the scene.

Aging can be applied for pixels that have been in the scene for a long time, but whose Gaussian description has not been visited for some time, the application erases

the Gaussian that models it. This allows to eliminate the cases described before where parked objects leave the scene and left their ghost behind. Also, Gaussians with few samples not used for some time are eliminated from the Gaussian mixture because it is considered that they were the result of noise.

4 Experimental Results

The model just described was implemented on an Intel Pentium IV computer with 3GHz clock frequency. It was programmed in MATLAB and tested with several sequences that included indoors and outdoors scenarios, with natural and artificial light, imaged during nighttime and daytime. For the experiments, color images were converted into grayscale ones. The average processing speed was about 5,300 pixels per second. Due to lack of space, we illustrate the results with a few frames out of a poll of sequences.

PETS stands for Performance Evaluation of Tracking and Surveillance Systems[3] (first row in Fig. 5). The sequence has 2688 JPEG frames. The individual frame resolution is 768×576 color pixels. There are seven persons or groups of persons that at different times cross the scene. There are also three vehicles that park or do some kind of manoeuver, like moving back and forth. In general, when the objects move their shape is well delineated. In this sequence, the moving objects either pass by or arrive to the scene. There is one truck that enters the scene, parks, and afterwards leaves the scene. In such cases, the foreground objects are correctly detected and when they leave aging and small groups filtering can get ride of the traces left. However, see the *Laboratory* sequence, which has 182 color frames with resolution 640×480 (second row in Fig. 5). In this sequence a person walks around a computer laboratory. The person dresses a shirt with a black squares drawing and black pants. The sequence starts with the person standing at the door. As he leaves this position, his silhouette is left behind. Whether to leave or remove one of such figures depends on a decision that cannot be taken at pixel level and also on factors such as the initial state of the scene background and ultimate application of the surveillance system. Fig. 5(b.1) shows also the trajectory followed by the person. This trajectory was computed using the blob centroid in each frame of the sequence. Some problems related to shadows that appear in this sequence are more evident in the sequence *Soccer*. The *Soccer* sequence has 9,744 frames with resolution 320×240 (third row in Fig. 5). It shows a nocturne soccer game where 22 players, a referee, the ball and the public provide a quite dynamic scenario. The camera was placed about 80 meters from middle field. The illumination conditions, dominated by neon lamps, make the objects in the scene to cast multiple shadows. The shadows is the regular price that has to be paid for processing is grayscale instead of color. Still, in this scene moving objects are well detected even considering their size and the type of illumination. Another experiment shows the *Crossroads* sequence, which was took from a 28 meters high tower placed in one of the corners of a crossroads (Fig. 5). From that viewpoint is possible to cover the whole intersection. The sequence has 2000 frames, each one with resolution 320×240 . This sequence is very challenging because the vehicles become part of the background and then, after a while, they suddenly start moving again, at high speed. So the background has to recover quickly but there are other vehicles passing by. Also, long buses with homogeneous roofs are



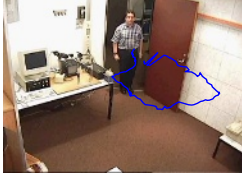
(a.1) Scene.



(a.2) Frame 950.



(a.3) Frame 2610.



(b.1) Scene.



(b.2) Frame 058.



(b.3) Frame 118.



(c.1) Scene.



(c.2) Frame 4000.



(c.3) Frame 8000.



(d.1) Scene.



(d.2) Frame 72.



(d.3) Frame 742.

Fig. 5. Experimental results

detected correctly. In general, depending on what is initially consider to be part of the background, our method can be used to detect left over objects, which could leave the scene at a later stage.

5 Conclusion

In this document, we have shown how an intrinsic curves could be used to detect accretion and occlusion regions in a scene containing moving objects. Using, this technique alone will produce fragmented segments for large objects with homogeneous texture. Hence, Gaussian mixture models provide an ideal vehicle to preserve information about the recent pixel intensity dynamics. Our algorithm shows good results for updating the model of the background under a variety of scenarios that include indoors and outdoors

environment in a number of different illumination conditions. It has been noted that depending on the initial background state and ultimate application, the algorithm is able to keep track of leftover objects. This is an important property that may be useful for security applications. As stated, the algorithm is based on the grayscale processing of individual pixels without regard of the behavior of the local neighborhood. Color information, and hierarchical analysis may be prove to be helpful to eliminate shadows, detect camouflage and sudden global illumination changes.

References

1. Shao-Yi Chien, Shyh-Yih Ma, and Liang-Gee Chen. Efficient Moving Object Segmentation Algorithm using Background Registration Technique. *Circuits and Systems for Video Technology, IEEE Transactions on*, 12(7):577 – 586, 2002.
2. R. Cucchiara, C. Grana, M. Piccardi, and A. Prati. Detecting Moving Objects, Ghosts, and Shadows in Video Streams. *IEEE Transactions on PAMI*, 25(10):1337 – 1342, 2003.
3. James Ferryman. Performance Evaluation of Tracking and Surveillance On-Line Evaluation Service. <http://visualsurveillance.org/PETS2001>, February 2006.
4. P. R. Giaccone, D. Tsaptsinos, and G. A. Jones. Foreground-Background Segmentation by Cellular Neural Networks. In *ICPR*, volume 2, pages 438 – 441, 2000.
5. D. Gutches, M. Trajkovics, E. Cohen-Solal, D. Lyons, and A. K. Jain. A Background Model Initialization Algorithm for Video Surveillance. In *ICCV*, volume 1, pages 733 – 740, 2001.
6. Hongzhe Han, Zhiliang Wang, Jiwei Liu, Zhengxi Li, Bin Li, and Zhongtao Han. Adaptive Background Modeling with Shadow Suppression. In *Intelligent Transportation Systems, Proceedings IEEE*, volume 1, pages 720 – 724, 2003.
7. Yinghua He, Hong Wang, and Bo Zhang. Updating in Illumination-Variant Scenes. In *Proceedings IEEE Intelligent Transportation Systems*, pages 515 – 519, 2003.
8. T. Horprasert, D. Harwood, and L. S. Davis. A Robust Background Subtraction and Shadow Detection. In *Proceedings of the Fourth Asian Conference on Computer Vision*, 2000.
9. Qian Huang, B. Dom, N. Megiddo, and W. Niblack. Segmenting and Representing Background in Color Images. In *ICPR*, volume 3, pages 13 – 17, 1996.
10. A. Kaup and T. Aach. Efficient Prediction of Uncovered Background in Interframe Coding using Spatial Extrapolation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 501–504, 1994.
11. P. Kumar, S. Ranganath, and W. Huang. Queue Based Fast Background Modelling and Fast Hysteresis Thresholding for Better Foreground Segmentation. In *Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on*, volume 2, pages 743 – 747, 2003.
12. Horng-Horng Lin, Tyng-Luh Liu, and Jen-Hui Chuang. A Probabilistic SVM Approach for Background Scene Initialization. In *ICIP*, volume 3, pages 893 – 896, 2002.
13. K. E. Matthews and N. M. Namazi. A Bayes Decision Test for Detecting Uncovered-Background and Moving Pixels in Image Sequences. *IEEE Transactions on IP*, 7(5):720 – 728, 1998.
14. John P. Nolan. Stable Distributions. <http://academic2.american.edu/~jpnolan/stable/chap1.pdf>, February 2006.
15. Johann A. Schoonees P. Wayne Power. Understanding Background Mixture Models for Foreground Segmentation. In *Proceedings Image and Vision Computing New Zealand*, page 267, 2002.
16. M. Piccardi. Background Subtraction Techniques: A Review. In *IEEE ICSMC*, volume 4, pages 3099 – 3104, 2004.

17. R. Pless, J. Larson, S. Siebers, and B. Westover. Evaluation of Local Models of Dynamic Backgrounds. In *CVPR*, pages 73–78, 2003.
18. R. J. Qian and M. I. Sezan. Video Background Replacement without a Blue Screen. In *ICIP*, pages 143–146, 1999.
19. C. Stauffer and W. E. L. Grimson. Adaptive Background Mixture Models for Real-Time Tracking. In *CVPR*, volume 2, pages 246 – 252, 1999.
20. Jen-Chao Tai and Kai-Tai Song. Background Segmentation and its Application to Traffic Monitoring using Modified Histogram. In *Networking, Sensing and Control, 2004 IEEE International Conference on*, volume 1, pages 13 – 18, 2004.
21. Carlo Tomasi and Roberto Manduchi. Stereo Matching as a Nearest-Neighbor Problem. *IEEE Transactions on PAMI*, 20(3):333–340, 1998.
22. K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and Practice of Background Maintenance. In *ICCV*, volume 1, pages 255 – 261, 1999.
23. Eric W. Weisstein. Diffeomorphism. From MathWorld—A Wolfram Web Resource. <http://mathworld.wolfram.com/Diffeomorphism.html>, September 2005.
24. Yufang Zhang, P. Shi, E. G. Jones, and Q. Zhu. Robust Background Image Generation and Vehicle 3D Detection and Tracking. In *IEEE Conference on Intelligent Transportation Systems*, pages 12 – 16, 2004.
25. Yue Zhou and Hai Tao. A Background Layer Model for Object Tracking through Occlusion. In *ICCV*, volume 2, pages 1079 – 1085, 2003.

Towards a New Paradigm for Motion Extraction

Luc Florack, Bart Janssen, Frans Kanters, and Remco Duits

Eindhoven University of Technology, Department of Biomedical Engineering,
Den Dolech 2, NL-5600 MB Eindhoven, The Netherlands
{L.M.J.Florack, B.J.Janssen, F.M.W.Kanters, R.Duits}@tue.nl
<http://www.bmi2.bmt.tue.nl/image-analysis/>

Abstract. A new multiscale paradigm is proposed for motion extraction. It exploits the fact that certain geometrically meaningful, isolated points in scale space provide unambiguous motion evidence, and the fact that such evidence narrows down the space of admissible motion fields. The paradigm combines the strengths of multiscale and variational frameworks.

Besides spatial velocity, two additional degrees of freedom are taken into account, viz. a temporal and an isotropic spatial scale component. The first one is conventionally set to unity (“temporal gauge”), but may in general account for creation or annihilation of structures over time. The second one pertains to changes in the image that can be attributed to a sharpening or blurring of structures over time.

This paper serves to introduce the new generalized motion paradigm de-emphasizing performance issues. We focus on the conceptual idea and provide recommendations for future directions.

Keywords: Motion extraction, scale space, reconstruction, regularization.

1 Introduction

In several recent papers algorithms have been proposed for reconstructing a static image given a finite set of linear constraints, or “features”. The reconstructed image can be said to be a representative of the metameric class of all images that satisfy these constraints. Since this class is in general infinite-dimensional, a so-called *prior* is needed in order to disambiguate a unique instance (unless the set of constraints is sufficiently rich, as is the case in the application by Kybic et al. [1]). Lillholm and Nielsen [2,3] have investigated the effect of several priors on the reconstruction, with the aim to maximize visual similarity with the original image from which the fiducial features had been extracted. In general this leads to a nonlinear system. Kanters et al., Duits et al., and Janssen et al. [4,5,6,7] show that one can potentially simplify the reconstruction scheme considerably without much loss of visual quality by suitable choice of prior, so as to obtain a *linear* system.

In any case, the possibility of approximate reconstruction from a sparse set of linear features evaluated at certain scale space anchor points has led to a paradigmatic shift. The idea is that, by virtue of the possibility of reconstruction,

a detour can be made via manipulation in the feature space of anchor points and their differential attributes, instead of working directly on image pixels. For instance, inspired by the pioneering work on SIFT features by Lowe et al. [8], further pursued by Schmid, Mikolajczyk et al. [9,10], Platel et al. [11,12] have shown that certain scale space anchor points endowed with suitable differential features can be used for robust image matching and retrieval invariant under similarities (scale-Euclidean transformations) and partial occlusion. *In this paper we wish to extend this paradigmatic shift to motion extraction and disambiguation.* We will de-emphasize performance issues (preliminary experiments are not yet competitive with state-of-the-art motion extraction), but concentrate on the basic principles underlying the new paradigm. In addition we will provide concrete recommendations that may lead to a significant improvement of performance.

The paper is organized as follows. In Appendix A we recapitulate the basic paradigm of image reconstruction in the well-established *static* case. To keep things as simple as possible we restrict ourselves to solutions that can be represented in analytically closed form. The appendix primarily serves to render this paper more or less self-contained, and to explain the concept of reconstruction in a simplified context. In Section 2 we generalize the theory so as to account for *dynamic* image sequences, with the aim to reconstruct a dense motion field from the dynamical properties of scale space anchor points. In Section 3 we summarize results.

2 Theory of Reconstruction—The Dynamic Case

In Section 2.1 we list notational conventions for later reference. In Section 2.2 we develop the theory in a rather general, yet fully operational form. In Section 2.3 concrete recommendations are given that will need to be taken into account in order to achieve state-of-the-art performance.

2.1 Definitions and Notations

Table 1 lists the most important symbols used in this paper for future reference.

Furthermore we define the following functional dependencies for motion and flux fields:

$$j \stackrel{\text{def}}{=} \gamma(f) v \quad (1)$$

$$J \stackrel{\text{def}}{=} \gamma(F) V \quad (2)$$

$$J_s \stackrel{\text{def}}{=} \gamma(F_s) V_s, \quad (3)$$

in which $\gamma : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$ is a smooth monotonic function, $\gamma' \geq 0$. In the sequel we shall only consider the cases $\gamma(\xi) = \xi^\alpha$ for $\alpha \geq 0$, and in particular the cases with $\alpha = 0$ and $\alpha = 1$. In the former case there is no distinction between motion and flux fields. In the latter case there is an explicit coupling between the greyvalue image and its motion field. In the usual temporal gauge ($v^t = 1$) the temporal component of all flux quantities then equals the scalar greyvalue image

Table 1. In all multiresolution expressions the symbol s denotes isotropic spatial scale. Time scale is considered a fixed constant. All vector field expressions account for spatial, temporal, and (spatial) scale components, yielding a total of $n+2$ vector components per base point. In the sequel, v is the “hidden” or “ground truth” motion field we are after, and w is a reconstruction, i.e. one preferred member of the equivalence class of high resolution motion fields that are consistent with v with respect to actual evidence, i.e. that share the same record of dynamical features at all anchor points.

| symbol | function prototype | significance |
|--------|---|---|
| f | $\mathbb{R}^{n+1} \rightarrow \mathbb{R}: (x, t) \mapsto f(x, t)$ | high resolution image sequence (“raw data”) |
| v | $\mathbb{R}^{n+1} \rightarrow \mathbb{R}^{n+2}: (x, t) \mapsto v(x, t)$ | high resolution motion field (“ground truth”) |
| j | $\mathbb{R}^{n+1} \rightarrow \mathbb{R}^{n+2}: (x, t) \mapsto j(x, t)$ | high resolution flux field |
| g | $\mathbb{R}^{n+1} \rightarrow \mathbb{R}: (x, t) \mapsto g(x, t)$ | high resolution reconstructed image sequence |
| w | $\mathbb{R}^{n+1} \rightarrow \mathbb{R}^{n+2}: (x, t) \mapsto w(x, t)$ | high resolution reconstructed motion field |
| k | $\mathbb{R}^{n+1} \rightarrow \mathbb{R}^{n+2}: (x, t) \mapsto k(x, t)$ | high resolution reconstructed flux field |
| F | $\mathbb{R}^{n+1} \times \mathbb{R}^+ \rightarrow \mathbb{R}: (s, x, t) \mapsto F(s, x, t)$ | multiresolution image sequence |
| V | $\mathbb{R}^{n+1} \times \mathbb{R}^+ \rightarrow \mathbb{R}^{n+2}: (s, x, t) \mapsto V(s, x, t)$ | multiresolution motion field |
| J | $\mathbb{R}^{n+1} \times \mathbb{R}^+ \rightarrow \mathbb{R}^{n+2}: (s, x, t) \mapsto J(s, x, t)$ | multiresolution flux field |
| F_s | $\mathbb{R}^{n+1} \rightarrow \mathbb{R}: (x, t) \mapsto F_s(x, t) = F(s, x, t)$ | fixed resolution image sequence, scale s |
| V_s | $\mathbb{R}^{n+1} \rightarrow \mathbb{R}^{n+2}: (x, t) \mapsto V_s(x, t) = V(s, x, t)$ | fixed resolution motion field, scale s |
| J_s | $\mathbb{R}^{n+1} \rightarrow \mathbb{R}^{n+2}: (x, t) \mapsto J_s(x, t) = J(s, x, t)$ | fixed resolution flux field, scale s |

($j^t = f, J^t = F, J_s^t = F_s$, et cetera), thus scalar image reconstruction reduces to a subproblem of flux reconstruction. Since at this stage it is an essentially unsolved question which definition is most appropriate, we adhere to a general definition using the function γ ; this raises no technical complications.

Moreover, we assume that the multiresolution fields are generated from the high resolution fields in the following ways¹:

$$F_s \stackrel{\text{def}}{=} e^{s\Delta} f \tag{4}$$

$$J_s \stackrel{\text{def}}{=} e^{s\Delta} j, \tag{5}$$

and, as a result, for nonnegative nonzero f ,

$$V_s \stackrel{\text{def}}{=} \frac{e^{s\Delta} j}{\gamma(e^{s\Delta} f)}. \tag{6}$$

In other words, F and J satisfy the heat equation, subject to initial conditions f and j , respectively. In the spatiotemporal domain they are obtained through convolution with a normalized Gaussian:

$$F_s = f * \phi_s \tag{7}$$

$$J_s = j * \phi_s, \tag{8}$$

¹ The linear operator $e^{s\Delta}$ is defined by virtue of its (multiplicative) Fourier representation, obtained by the formal identification $\Delta \sim -\|\omega\|^2$, in which $\omega \in \mathbb{R}^n$ denotes spatial frequency. In the spatial domain it corresponds to a convolution filter.

with

$$\phi_s(x, t) \stackrel{\text{def}}{=} \frac{1}{\sqrt{4\pi s}} \frac{1}{\sqrt{4\pi\epsilon}} \exp \left[-\frac{\|x\|^2}{4s} - \frac{t^2}{4\epsilon} \right], \tag{9}$$

with fixed “infinitesimal” time scale $\epsilon > 0$ (in practice in the order of frame separation). We henceforth ignore time scale and identify F_0 and J_0 with f and j , respectively.

Next, we define the *anchor point set* \mathcal{P}_t as a collection of identifiable isolated points in scale space at each moment t in time:

$$\mathcal{P}_t = \{p_i(t) = (s_i(t), x_i(t), t)\}_{i=1, \dots, N}. \tag{10}$$

The points in \mathcal{P}_t could be *scale space singular points* [13,14], *scale space critical points* [15,16,17,18], and/or other types of anchor points in scale space.

Finally, we introduce the convenient shorthand

$$\phi_i(x, t) \stackrel{\text{def}}{=} \phi_{s_i(t)}(x - x_i(t), t) \quad (i = 1, \dots, N), \tag{11}$$

recall Eq. (9).

2.2 Theory

Under mild transversality conditions² the anchor points $p_i(t) \in \mathcal{P}_t$ can be tracked across time frames. In fact, the geometric properties of the anchor point trajectories, such as scale space velocities, accelerations, et cetera, can be determined analytically in terms of partial spatiotemporal image derivatives evaluated at the respective points [14].

The movement of an anchor point is not confined to a spatial plane at a fixed level of resolution, because we have taken into account a scale component. In the t -parametrization above, which requires that the trajectories transversally intersect each time frame, the time component of an anchor point’s velocity equals unity by assumption (“temporal gauge”, cf. [19]), but this restriction can be relaxed by switching to an arbitrary parametrization instead, $p_i(\lambda) = (s_i(\lambda), x_i(\lambda), t_i(\lambda))$, say. In that case one can account for annihilations or creations of scale space anchor points over time, corresponding to a vanishing derivative $t'_i(\lambda) = 0$. By virtue of the incorporation of both scale as well as temporal component into the motion field the proposed framework for motion extraction is more general than conventional schemes, in which one accounts only for spatial motion while relying on the validity of the temporal gauge. Note also that the movement of anchor points does not require a constant brightness assumption [20], and that, at least for the anchor points themselves, there is no “aperture problem” to be resolved.

Let $F(\mathcal{P}_t) = \{F_i \equiv F(p_i(t) \in \mathcal{P}_t)\}_{i=1}^N$ and $V(\mathcal{P}_t) = \{V_i \equiv V(p_i(t) \in \mathcal{P}_t)\}_{i=1}^N$ denote the greyvalues and scale space velocities of the anchor points, respectively.

² The transversality conditions are mild in the sense that locally in spacetime they tend to be met almost everywhere. Globally, however, these conditions are rather restrictive. Cf. [19] for a discussion.

Then the flux vectors $J(\mathcal{P}_t) = \{J_i \equiv J(p_i(t) \in \mathcal{P}_t)\}_{i=1}^N$ are known as well, recall Eq. (2). Besides greyvalues we may endow anchor points with other measurable differential entities. Likewise, other dynamical properties than their velocities may be associated with anchor points, such as accelerations, and higher order geometric properties of their trajectories. Knowing all these intensive and geometric properties of anchor points the question arises of how to obtain a dense high resolution motion field consistent with all these constraints.

Consider the following Euler-Lagrange functional, analogous to Eq. (17) in Appendix A, using summation convention for scale spatiotemporal indices here and henceforth (μ in this case):

$$E_Q(k, \lambda) = Q(k) - \sum_{i=1}^N \lambda_\mu^i (\langle k^\mu | \phi_i \rangle - J_i^\mu), \tag{12}$$

in which Q is some disambiguating prior, and k denotes the approximate reconstruction of an unknown high resolution flux field j , the only certain knowledge of which we have is in terms of an N -tuple of vector-valued scale space measurements $J_i = \langle j | \phi_i \rangle \equiv \langle k | \phi_i \rangle$, $i = 1, \dots, N$. The inner product used in the constraint terms is the standard spatial inner product associated with a fixed time frame t . (Time is implicit in the notation.)

The Euler-Lagrange equations for the high resolution flux field k are given by:

$$\begin{cases} \frac{\partial E_Q(k, \lambda)}{\partial \lambda_\mu^i} = J_i^\mu - \langle k^\mu | \phi_i \rangle = 0 & (i = 1, \dots, N, \mu = 1, \dots, n+2) \\ \frac{\delta E_Q(k, \lambda)}{\delta k^\mu} = \frac{\delta Q(k)}{\delta k^\mu} - \sum_{i=1}^N \lambda_\mu^i \phi_i = 0 & \text{on } \mathbb{R}^n \quad (\mu = 1, \dots, n+2). \end{cases} \tag{13}$$

The first part reproduces the desired constraints (assuming we have extracted greyvalues and velocities of the scale space anchor points of interest and constructed the corresponding flux vectors J_i). If Q is a quadratic functional, say

$$Q(k) = \frac{1}{2} \langle k^\mu | \mathcal{Q}_{\mu\nu} | k^\nu \rangle, \tag{14}$$

in which all components $\mathcal{Q}_{\mu\nu} = \mathcal{Q}_{\nu\mu}$ are positive self-adjoint operators with bounded inverse, then Eq. (13) yields

$$k^\mu = \sum_{i=1}^N \sum_{j=1}^N \mathcal{R}^{\mu\nu} \phi_i \Psi_{\nu\rho}^{ij} J_j^\rho, \tag{15}$$

cf. Eq. (19) in Appendix A for the static case. Here we have defined the operators $\mathcal{R}^{\mu\nu} = \mathcal{R}^{\nu\mu}$ as follows:

$$\mathcal{R}^{\mu\rho} \mathcal{Q}_{\rho\nu} = \delta_\nu^\mu I, \tag{16}$$

in which I denotes the identity operator. The sources of motion evidence are given by $J_i^\mu = \langle k^\mu | \phi_i \rangle = \langle j^\mu | \phi_i \rangle$. They are, by definition, equal for all

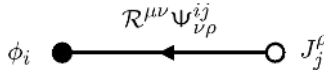


Fig. 1. Diagrammatic representation of Eq. (15): Coupling of source data J_j^ρ to sampling filter ϕ_i via propagator $\mathcal{R}^{\mu\nu}\Psi_{\nu\rho}^{ij}$

metamerical flux fields (members of the equivalence class consistent with data evidence). Cf. Figure³ 1.

We have established an operational scheme for motion extraction, condensely summarized by Eq. (15) or, equivalently, Figure 1. The sources J_i are known by definition. Recall that they capture motion and greyvalue evidence of isolated scale space anchor points, which can either be obtained empirically by tracking (seeking corresponding anchor points in successive time frames), or by theoretical derivation (exploiting the scale space paradigm and the defining equation for the specific anchor points). The latter typically requires the implicit function theorem, as anchor points are usually defined implicitly in terms of zero-crossings. The propagator $\mathcal{R}^{\mu\nu}\Psi_{\nu\rho}^{ij}$ expresses our choice of prior. It depends both on our choice of measurement filters ϕ_i as well as on our choice of regularization functional, Eq. (14). In some cases, such as exemplified in Appendix A for static reconstruction, it may be computed in analytically closed form, but in general cases one will need to resort to numerical integration and linear inversion schemes. Finally, the filters ϕ_i define our measurement paradigm, and span the projection space within which we seek for our high resolution reconstruction k . Instead of Gaussian filters one may adopt any set of linear filters instead.

2.3 Recommendations

Although we have established a fully operational scheme, choices will have to be made as described in the previous section, and some practical problems will have to be coped with. It is evident that such choices will affect performance in a critical way.

In their original article on static image reconstruction Lillholm and Nielsen [2,3] account for a prior that is given in terms of a standard L^2 inner product norm. As this gives quite unsatisfactory reconstruction results in the static case (unless one has a very rich set of anchor points), this is probably not a good choice for the dynamic case either. To stay within the linear framework Kanters et al., Duits et al., and Janssen et al. [4,5,6,7] have considered more general Hilbert spaces, notably Sobolev spaces of a certain type. These include $L^2(\mathbb{R}^n)$ as a limiting case of a vanishing coupling constant. Taking nonzero coupling constants has turned out to improve the visual quality of reconstruction considerably, at least in the static case. The example worked out in Appendix A is similar, but hinges on a Sobolev space of infinite order. It has some theoretical advantage

³ If $Q(k)$ contains higher order terms in k , with corresponding coupling constants, one may generalize the above scheme, and Figure 1, using a perturbative expansion, in terms of the coupling constants, akin to a Feynman diagrammatic expansion.

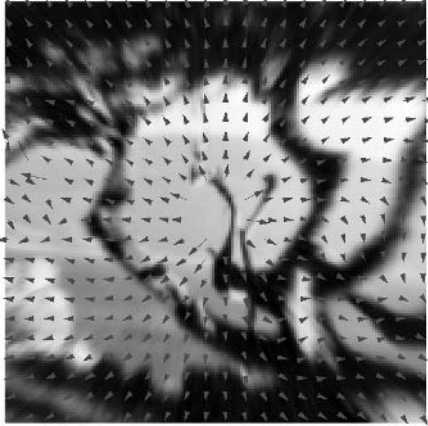


Fig. 2. Result of the *spatial component* of the reconstructed optic flow field, (w^x, w^y) , deduced from Eq. (15), or Fig. 1, on Diverging Tree Sequence [21], using the same prior and anchor points as in the static case, cf. Figure 4 in Appendix A. The time component has been gauged to unity following common practice, $w^t = 1$, whereas the scale component w^s (which has no counterpart in traditional optic flow models) has been ignored altogether in this experiment. The linear features that form the data evidence for reconstruction are the anchor point velocities V_i^μ , obtained analytically in terms of partial image derivatives evaluated at the anchor points, from which the local flux values are constructed as follows: $J_i^\mu = F_i V_i^\mu \equiv \langle j^\mu | \phi_i \rangle = \langle k^\mu | \phi_i \rangle$ (recall Eq. (2)). Note that no further differential image structure has been used to constrain the solution, as opposed to the static reconstruction shown in Figure 4. Despite the *extreme sparseness* of data evidence—roughly 1% of pixel data has actually been used—and despite the genericity of the prior—no knowledge about camera/scene relation has been accounted for—the result shows a qualitatively correct diverging motion field. (Especially near the image boundary the solution is hampered by a boundary problem.) This shows the feasibility of the new motion paradigm. Of course, quantitative results can only be expected if further constraints are imposed, possibly accompanied by an adaptation of anchor point set and model prior, as outlined in the recommendations.

that may be exploited in practice, such as closed form expressions for the Gram matrix, but it’s regularizing effect is qualitatively similar in the static case. In their application to generalized sampling Kybic et al. [1] considered a somewhat similar semi-norm instead, albeit in a different context.

Still, it remains doubtful whether any of the abovementioned regularizing priors will prove successful in the dynamic case. Preliminary studies indicate that reconstruction quality, measured in terms of the angular error introduced by Barron et al. [21], remains unsatisfactory compared to state-of-the-art on standard benchmark sequences, at least if the anchor points are scale space singularities, and endowed only with lowest order flux evidence (velocities and greyvalues). This poor performance is not really unexpected, as in this specific case the feature set is extremely sparse in practice (representing typically in the order of one percent of image information).

There are basically two ways to overcome the abovementioned shortcomings, viz. by adding evidence, or by improving one's prior. A straightforward way to add evidence is by enriching the given anchor point set with any further intensive and/or geometric evidence that can be extracted empirically, or computed analytically, such as differential image structure, accelerations, et cetera. Another way to do so is by extending the set of anchor points itself. Likely candidate anchor points are scale space critical points [15], scale space singularities [13,14], and similar points of higher order. Alternative anchor points, such as proposed by Lindeberg [22,23], Lowe [8], Schmid et al. [10], and Mikolajczyk et al. [9] are viable candidates as well. Both ways of adding evidence *will* improve performance, for the simple reason that motion is further constrained to be consistent with additional evidence. The extent of improvement remains, however, unclear, as these recommendations have not yet been carried out.

Even if data evidence has been exploited fully the option to change one's prior remains. Instead of using sophisticated but in a way "uncommitted" priors, such as those based on Sobolev inner product norms discussed above, it seems more promising to follow the direction proposed by Brox et al. [24], viz. to relate the prior to the constant brightness assumption. For although greyvalue attributes of anchor points will, in general, not be conserved, overall motion appears to be fairly consistent with greyvalue conservation. We refer to the literature for details.

Finally, one will need to account for boundary conditions, which have been ignored in our theoretical derivations.

3 Conclusion and Summary

We have proposed a new paradigm for generalized motion extraction, and provided an operational scheme for it. It combines the strengths of multiscale and variational frameworks. It exploits unambiguous motion cues obtained by tracking isolated anchor points in scale space, and a complementary model ("prior") for disambiguation of the high resolution motion field. It moreover generalizes the conventional concept of motion by accounting for two additional degrees of freedom besides spatial velocity, viz. a temporal and a scale component. The former is implicit in conventional schemes, where it is usually scaled to unity ("temporal gauge"). The second captures a hitherto unexplored dynamical degree of freedom, and can be related to a blurring or enhancement of structures over time. For instance, a gradual blurring of an otherwise stationary image over time is consistent with a generalized motion field in which the spatial part vanishes identically, but in which the scale component is nontrivial.

The significance of the additional degrees of freedom of the generalized motion field is subject to further study. Insight may lead to exploitation of these degrees of freedom beyond the usual scope of motion extraction techniques, e.g. in the case where annihilations, creations, sharpening, or blurring of structures over time occur.

There is no decisive conclusion regarding optimality of model priors; this, too, will require further investigation.

Acknowledgement

The Netherlands Organisation for Scientific Research (NWO) is gratefully acknowledged for financial support.

References

1. Kybic, J., Blu, T., Unser, M.: Generalized sampling: A variational approach—part I: Theory. *IEEE Transactions on Signal Processing* **50** (2002) 1965–1976
2. Lillholm, M., Nielsen, M., Griffin, L.D.: Feature-based image analysis. *International Journal of Computer Vision* **52** (2003) 73–95
3. Nielsen, M., Lillholm, M.: What do features tell about images? In Kerckhove, M., ed.: *Scale-Space and Morphology in Computer Vision: Proceedings of the Third International Conference, Scale-Space 2001, Vancouver, Canada*. Volume 2106 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin (2001) 39–50
4. Kanters, F., Florack, L., Platel, B., ter Haar Romeny, B.: Image reconstruction from multiscale critical points. In Griffin, L.D., Lillholm, M., eds.: *Scale-Space Methods in Computer Vision: Proceedings of the Fourth International Conference, Scale-Space 2003, Isle of Skye, UK*. Volume 2695 of *Lecture Notes in Computer Science*, Berlin, Springer-Verlag (2003) 464–478
5. Duits, R., Janssen, B., Kanters, F., Florack, L.: Linear image reconstruction from a sparse set of α -scale space features by means of inner products of Sobolev type. In Fogh Olsen, O., Florack, L., Kuijper, A., eds.: *Deep Structure Singularities and Computer Vision*. Volume 3753 of *Lecture Notes in Computer Science*, Springer-Verlag (2005) 96–111
6. Janssen, B., Kanters, F.M.W., Duits, R., Florack, L.M.J., ter Haar Romeny, B.M.: A linear image reconstruction framework based on Sobolev type inner products. In Kimmel, R., Sochen, N., Weickert, J., eds.: *Scale Space and PDE Methods in Computer Vision: Proceedings of the Fifth International Conference, Scale-Space 2005, Hofgeismar, Germany*. Volume 3459 of *Lecture Notes in Computer Science*, Berlin, Springer-Verlag (2005) 85–96
7. Janssen, B., Kanters, F.M.W., Duits, R., Florack, L.M.J., ter Haar Romeny, B.M.: A linear image reconstruction framework based on Sobolev type inner products (2006) To appear in the *International Journal of Computer Vision*.
8. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* **60** (2004) 91–110
9. Mikolajczyk, K., Schmid, C.: Scale and affine invariant interest point detectors. *International Journal of Computer Vision* **60** (2004) 63–86
10. Schmid, C., Mohr, R., Bauckhage, C.: Evaluation of interest point detectors. *International Journal of Computer Vision* **37** (2000) 151–172
11. Platel, B., Florack, L.M.J., Kanters, F.M.W., Balmachnova, E.G.: Using multiscale top points in image matching. In: *Proceedings of the 11th International Conference on Image Processing (Singapore, October 24–27, 2004)*, IEEE (2004) 389–392
12. Platel, B., Balmachnova, E.G., Florack, L.M.J., ter Haar Romeny, B.M.: Top-points as interest points for image matching. In Leonardis, A., Bischof, H., Prinz, A., eds.: *Proceedings of the Ninth European Conference on Computer Vision (Graz, Austria, May 2006)*. Volume 3951–3954 of *Lecture Notes in Computer Science*, Berlin, Heidelberg, Springer-Verlag (2006) 418–429

13. Damon, J.: Local Morse theory for solutions to the heat equation and Gaussian blurring. *Journal of Differential Equations* **115** (1995) 368–401
14. Florack, L., Kuijper, A.: The topological structure of scale-space images. *Journal of Mathematical Imaging and Vision* **12** (2000) 65–79
15. Koenderink, J.J.: A hitherto unnoticed singularity of scale-space. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11** (1989) 1222–1224
16. Kuijper, A., Florack, L.M.J., Viergever, M.A.: Scale space hierarchy. *Journal of Mathematical Imaging and Vision* **18** (2003) 169–189
17. Kuijper, A., Florack, L.M.J.: The hierarchical structure of images. *IEEE Transactions on Image Processing* **12** (2003) 1067–1079
18. Kuijper, A., Florack, L.M.J.: The relevance of non-generic events in scale space models. *International Journal of Computer Vision* **57** (2004) 67–84
19. Florack, L.M.J., Niessen, W.J., Nielsen, M.: The intrinsic structure of optic flow incorporating measurement duality. *International Journal of Computer Vision* **27** (1998) 263–286
20. Horn, B.K.P., Schunck, B.G.: Determining optical flow. *Artificial Intelligence* **17** (1981) 185–203
21. Barron, J.L., Fleet, D.J., Beauchemin, S.S.: Performance of optical flow techniques. *International Journal of Computer Vision* **12** (1994) 43–77
22. Lindeberg, T.: Detecting salient blob-like image structures and their scale with a scale-space primal sketch: A method for focus-of-attention. *International Journal of Computer Vision* **11** (1993) 283–318
23. Lindeberg, T.: Feature detection with automatic scale selection. *International Journal of Computer Vision* **30** (1998) 79–116
24. Brox, T., Bruhn, A., Papenber, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In Pajdla, T., Matas, J., eds.: *Proceedings of the Eighth European Conference on Computer Vision* (Prague, Czech Republic, May 2004). Volume 3021–3024 of *Lecture Notes in Computer Science.*, Berlin, Springer-Verlag (2004) 25–36

A Theory of Reconstruction—The Static Case

A.1 Definitions and Notations

- $i = 1, \dots, N$: label for linear constraints, features, and corresponding filters;
- $\mu = 1, \dots, n$: spatial index;
- (x_i, t_i) : scale space interest points;
- $\phi(x)$: fiducial filter template of unit scale centred at the origin;
- $\phi_i(x)$: i^{th} filter of scale $\sigma_i = \sqrt{2t_i}$ centred at x_i .
- F_i : feature value at scale space anchor point (x_i, t_i) , i.e. $F_i = \langle f | \phi_i \rangle$, v.i.;
- f : arbitrary high resolution image;
- g : approximation of f obtained by linear reconstruction;
- $\langle u | v \rangle$: standard inner product of u and v , i.e. $\langle u | v \rangle = \int u(x) v(x) dx$;
- $\langle u | A | v \rangle$: shorthand for $\langle u | Av \rangle = \langle Au | v \rangle = \langle \sqrt{Au} | \sqrt{Av} \rangle$ (A is positive self-adjoint);
- Φ : Gram matrix, i.e. positive symmetric $N \times N$ -matrix with components $\Phi_{ij} = \langle \phi_i | \phi_j \rangle$.
- Φ^ϵ : ϵ -regularized Gram matrix, i.e. positive symmetric $N \times N$ -matrix with components $\Phi_{ij}^\epsilon = \langle \phi_i | \exp(2\epsilon\Delta) | \phi_j \rangle$.
- Ψ, Ψ_ϵ : inverse matrices of Φ, Φ^ϵ , respectively $\Psi_\epsilon^\epsilon = \delta_j^i$ ($\epsilon \geq 0$).

A.2 Theory

Let f be a raw image, and $\{\phi_i\}_{i=1,\dots,N}$ a labelled set of linear filters. Each filter is characterized by some base point $x_i \in \mathbb{R}^n$ and some scale $\sigma_i = \sqrt{2t_i}$, in other words, it can be associated with a scale space interest point. We assume that all filter outputs $F_i = \langle f|\phi_i \rangle$ are known, but no further knowledge of f is available. The aim is to construct a unique representative g of f that satisfies the same constraints, i.e. $\langle g|\phi_i \rangle = \langle f|\phi_i \rangle = F_i$ for all labels $i = 1, \dots, N$. The image g will be referred to as the reconstructed image, or simply *reconstruction*. Recall that in general there will be many reconstructions satisfying any finite set of constraints, and that for this reason we need a prior for disambiguation.

In some sense we would like the reconstructed image g to be “as close as possible” to the source image f from which the samples F_i have been obtained. However, if, apart from these measurement samples, no further knowledge of f is available, this constraint becomes operationally void. Instead, it makes sense to insist on some kind of *regularity* in order to single out a unique reconstruction, since in practice images are spatially coherent objects. The role of the prior is precisely to achieve disambiguation through regularization.

Reconstruction can be achieved via a standard Euler-Lagrange minimization technique. We consider the following Euler-Lagrange functional (for the sake of simplicity we ignore boundary conditions due to finite domain restrictions):

$$E_\epsilon(g, \lambda) = \frac{1}{2} \langle g | \exp(-2\epsilon\Delta) | g \rangle - \sum_{i=1}^N \lambda^i (\langle g | \phi_i \rangle - F_i). \tag{17}$$

The linear prior is reflected in the data independent quadratic term. For each constraint we have introduced an auxiliary parameter $\lambda^i \in \mathbb{R}$. Variation with respect to both g as well as λ yields the following Euler-Lagrange equations:

$$\begin{cases} \frac{\partial E(g, \lambda)}{\partial \lambda^i} = F_i - \langle g | \phi_i \rangle = 0 & (i = 1, \dots, N) \\ \frac{\delta E(g, \lambda)}{\delta g} = \exp(-2\epsilon\Delta)g - \sum_{i=1}^N \lambda^i \phi_i = 0 & \text{on } \mathbb{R}^n. \end{cases} \tag{18}$$

Indeed, stationarity with respect to $\lambda \in \mathbb{R}^N$ implies the desired constraints to be satisfied. By substituting g from the second equation into the first equation one readily solves for λ , after which one obtains the following solution for g :

$$g = \sum_{i=1}^N \sum_{j=1}^N \exp(2\epsilon\Delta) \phi_i \Psi_\epsilon^{ij} F_j. \tag{19}$$

Cf. Figure 3 for a diagrammatic representation, and Figure 4 for an experimental verification.

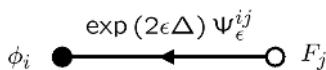


Fig. 3. Coupling of source data F_j to sampling filter ϕ_i via propagator $\exp(2\epsilon\Delta) \Phi_\epsilon^{ij}$

Note that the functional Eq. (17) may also be written as

$$E_\epsilon(g, \lambda) = \frac{1}{2} \langle \exp(-\epsilon\Delta)g | \exp(-\epsilon\Delta)g \rangle - \sum_{i=1}^N \lambda^i (\langle \exp(-\epsilon\Delta)g | \exp(\epsilon\Delta)\phi_i \rangle - F_i) ,$$

with

$$F_i = \langle \exp(-\epsilon\Delta)f | \exp(\epsilon\Delta)\phi_i \rangle .$$

The latter expression is intuitive: If, hypothetically speaking, you would ϵ -deblur the raw image, i.e. $f \rightarrow f_\epsilon \equiv \exp(-\epsilon\Delta)f$, you would encounter the same features F_i at anchor points that are shifted upward in scale space by an amount ϵ relative to the original interest points, in other words, that would have been obtained by using ϵ -blurred filters $\phi \rightarrow \phi_\epsilon \equiv \exp(\epsilon\Delta)\phi$ instead of the original ones. If we likewise consider the formal replacement $g \rightarrow g_\epsilon \equiv \exp(-\epsilon\Delta)g$ we observe that we could basically have started out from the well-established case of standard reconstruction, i.e. the non-regularized case obtained by setting $\epsilon = 0$, cf. [2,3,4,5,6,7], and subsequently apply the above formal replacement rules for f , g , ϕ_i , and Φ_{ij} , so as to obtain the ϵ -regularized reconstruction of interest. Since the Gram matrix Φ_{ij} is known in closed form for the case of standard reconstruction and Gaussian derivative filters ϕ_i of scale s_i , say, the ϵ -regularized reconstruction scheme is likewise in closed form, and boils down to scale replacements $s_i \rightarrow s_i + \epsilon$ in the unregularized Gram matrix, Φ^0 , and $s_i + 2\epsilon$ in the free expansion filters, ϕ_i , recall Eq. (19). (Note that Eq. (19) does not correspond to a mere ϵ -blurring of the standard reconstruction, for that would not be consistent with the constraints.)

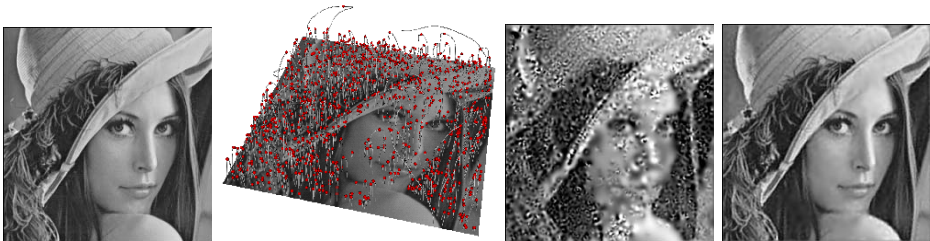


Fig. 4. Result of Eq. (19), or Fig. 3, on Lena image. From left to right: Original image f , scale space critical paths showing the singular points used as anchor points in the reconstruction, reconstruction g using standard L^2 -norm as disambiguating prior (i.e. the case $\epsilon = 0$), and reconstruction using a Sobolev-norm instead (i.e. $\mathcal{O}(\epsilon)$ -approximation $\exp(-2\epsilon\Delta) \approx I - 2\epsilon\Delta$ in Eq. (17)). The latter norm is not claimed to be optimal, but it is apparent that the reconstruction paradigm works in the static case provided care is taken with respect to the choice of prior. In this experiment the attributes of the anchor points that define the constraints are spatial image derivatives up to fourth order. Note that in a dynamic case this result will also be obtained as the time component $k^0 = g$ of Eq. (15) if we use the definition $j^\mu = f v^\mu$, respectively $k^\mu = g w^\mu$, and the usual temporal gauge $v^0 = w^0 = 1$, recall Eq. (1), given the same prior and set of constraints.

Fast Motion Estimation Using Spatio Temporal Filtering

V. Bruni, D. De Canditiis, and D. Vitulano

Istituto per le Applicazioni del Calcolo "M. Picone" - C.N.R.
Viale del Policlinico 137, 00161 Rome - Italy
{bruni, d.decanditiis, vitulano}@iac.rm.cnr.it

Abstract. In this paper a fast algorithm for motion estimation is presented. It models the temporal averaging of a group of frames as the spatial filtering of the reference one with a suitable Dirac comb function. This equality allows us to estimate a constant affine motion by comparing the phases of the FFT. Experimental results show that the proposed algorithm outperforms the available fast motion estimation techniques in terms of both quality and computational effort.

1 Introduction

Video coding is a topic which has receiving a growing interest since it is fundamental in various multimedia applications [1]. One of the most important steps during the encoding phase consists of an accurate estimate of the motion of the objects inside the scene under study. Even though different approaches such as pel recursive [2,3] and optical flow [4,5] have been proposed in literature, presently the most effective seems to be the Block Matching Algorithm (BMA) [6]. The main difference between this latter and the aforementioned approaches stems from the fact that BMA tries to estimate the motion of a block of pixels of the reference frame instead of one pixel at a time. The idea is to find the best block J of size $N \times N$ (belonging to the frame at time $t - 1$) that matches the block under study I of size $N \times N$ (belonging to the frame at time t), among all possible blocks inside a wider area of size $M \times M$ around I in the frame at time $t - 1$. As distance between blocks is usually selected MAD (Mean Absolute Difference) or the classical MSE (Mean Square Error). Once found the best block J , we are able to also know the displacement and then the motion of the pixels inside I . Although this strategy may be not accurate whenever the block I contains more than one dominant motion, it achieves good performances in terms of simplicity and bits budget. Its main drawback is that it requires a huge computational effort in the previously mentioned search. That is why various attempts oriented to reduce the search time have been proposed in literature — see for instance [7,8,9]. Another interesting trend is represented by the transform based approaches, in particular the FFT based one. They share the same philosophy of the Block Matching based approaches, since they treat blocks of the image instead of pixels. The underlining idea is to manage the phase of the Fourier

Transform that intrinsically contains the shift of a block — whenever there is motion [10,11,12,13,14]. Even in this case, the search is further reduced but not eliminated.

In this paper we present a novel approach based on FFT. It has the following advantages: *i)* an arbitrary number of frames can be simultaneously processed; *ii)* the motion estimation is solved without search; *iii)* a drastic reduction of the computational effort with respect to phase correlation methods is achieved, since the inverse FFT is avoided. The trick is very simple. Suppose that we use K frames for estimating the motion. Then, the average of K corresponding blocks in K subsequent frames is equivalent to the convolution of the reference block (i.e. in the reference frame) with a Dirac comb kernel — proof is in the Appendix. Hence, if on one hand we have the temporal average of the blocks, on the other we have the aforementioned convolution. In this latter the reference block is known as well as the Dirac comb function, apart from its sampling period that represents the motion d . Comparing the FFT phases in the first non zero frequency of both the mean of the K blocks and the convolution, we can determine the displacement d by a simple inversion formula. The method is also robust in presence of noise if data are preprocessed by a moving average operator. Moreover, it outperforms existing approaches in terms of speed and accurateness.

The paper is organized as follows. Next section presents the proposed model for both 1D and 2D cases, exploiting the result contained in Appendix. Section III shows some experimental results and makes some comparisons with other existing approaches. Finally, Section IV draws the conclusions.

2 The Proposed Model

For the sake of clarity, we first present a toy example: frames are replaced by 1D signals and each signal is shifted with respect to the reference one by an unknown motion d . The second part of the section will show the 2D generalization and the corresponding algorithm.

Let $\{f_k(x)\}_{0 \leq k \leq K-1}$ be a sequence composed of K successive signals $f_k(x)$ such that

$$f_k(x) = f_0(x - kd), \quad \forall k = 0, \dots, K-1, \quad 0 \leq x \leq 2M, \quad (1)$$

where d is the unknown quantity, $2M + 1$ is the signal length and $f_0(x)$ is the reference signal.

Let now $F(x)$ be the vector containing the temporal average of the K signals in the adopted sequence, that is

$$F(x) = \frac{1}{K} \sum_{k=0}^{K-1} f_k(x). \quad (2)$$

In the Appendix it is shown that $F(x)$ is equivalent to the spatial convolution of the reference signal $f_0(x)$ with the following Dirac comb function

$$g(x) = \frac{1}{K} \sum_{k=0}^{K-1} \delta(x - kd), \tag{3}$$

that is non zero whenever x is a multiple of the motion d . Therefore,

$$F(x) = (f_0 * g)(x). \tag{4}$$

By computing the Fourier transform of both members of the previous equation and exploiting the convolution property, it follows

$$\hat{F}(\omega) = \hat{f}_0(\omega)\hat{g}(\omega). \tag{5}$$

Both $\hat{F}(\omega)$ and $\hat{f}_0(\omega)$ are known, since they represent the data of our problem. The unknown object is $\hat{g}(\omega)$. Nonetheless, g is known except for the corresponding sampling period d , as arises from eq. (3).

In the following we will show that previous equation can be inverted with respect to the motion d . In particular, d will be computed by comparing the phases of both members of (5).

To invert (5) we have to explicitly write the dependence of g from the motion d . For properly computing the discrete convolution, the unitary circle is divided into $4M + 1$ frequency components. Hence,

$$\hat{F}(\omega) = \sum_{x=0}^{4M} F(x)e^{-i\frac{2\pi}{4M+1}\omega x}$$

$$\hat{f}_0(\omega) = \left(\sum_{x=0}^{2M} f_0(x)e^{-i\frac{2\pi}{4M+1}\omega x} \right)$$

and

$$\hat{g}(\omega) = \left(\sum_{x=0}^{(K-1)d} g(x)e^{-i\frac{2\pi}{4M+1}\omega x} \right) =$$

$$= |\hat{g}(\omega)|e^{-i\frac{2\pi}{4M+1}\frac{\omega}{2}d(K-1)} \tag{6}$$

(see Appendix for details).

Therefore, (5) can be rewritten as

$$\hat{F}(\omega) = \hat{f}_0(\omega)|\hat{g}(\omega)|e^{-i\frac{2\pi}{4M+1}\frac{\omega}{2}d(K-1)}.$$

This equality holds for both the absolute values and the phases of the complex numbers. Nonetheless, the absolute value of \hat{g} is a non linear function with

respect to d while its phase is linear. Hence, it is simpler to compare the phases of both members of the previous equation:

$$\text{Arg}(\hat{F}(\omega)) = \text{Arg}(\hat{f}_0(\omega)) + \text{Arg}\left(|\hat{g}(\omega)|e^{-i\frac{2\pi}{4M+1}\frac{\omega}{2}d(K-1)}\right),$$

that is

$$\text{Arg}(\hat{F}(\omega)) - \text{Arg}(\hat{f}_0(\omega)) = -\frac{(K-1)\pi}{4M+1}d\omega.$$

The equation can be inverted by fixing the frequency value ω . In particular, for $\omega = 1$ we have

$$\text{Arg}(\hat{F}(1)) - \text{Arg}(\hat{f}_0(1)) = -\frac{(K-1)\pi}{4M+1}d,$$

and then

$$d = \frac{(4M+1)}{(K-1)\pi} \left(\text{Arg}(\hat{f}_0(1)) - \text{Arg}(\hat{F}(1)) \right). \tag{7}$$

It is worth noticing that d is completely determined by the previous equation without any ambiguity on its sign. In fact, the model (1) includes the sign of the motion. Moreover, $\omega = 1$ is the more suitable choice among all the non zero frequencies whenever frames are corrupted by noise.

2.1 2D Motion Estimation

For the two dimensional case, we can adopt the same above mentioned strategy since the 2D motion can be split into its vertical (rows x axis) and horizontal (columns y axis) components. Let $\mathbf{d} = (d_x, d_y)$ be the motion vector, d_x and d_y respectively the vertical and horizontal motion and $\{f_k(x, y)\}_{0 \leq k \leq K-1}$ be a sequence composed of K frames of dimension $(2M+1) \times (2M+1)$ such that

$$f_k(x, y) = f_0(x - kd_x, y - kd_y), \quad k = 0, \dots, K-1, \quad 0 \leq x \leq 2M, \quad 0 \leq y \leq 2M.$$

Hence, the 2D *Dirac comb* is

$$g(x, y) = \frac{1}{K} \sum_{k=0}^{K-1} \delta(x - kd_x, y - kd_y).$$

It is worth noticing that the first diagonal block of the matrix g is

- a $K \times K$ identity matrix if $d_x = d_y = 1$;
- a $((K-1)d_x + 1) \times ((K-1)d_y + 1)$ diagonal matrix whose diagonal is a one dimensional Dirac comb having d_x as sampling period, if $d_x = d_y \neq 1$;
- a $((K-1)d_x + 1) \times ((K-1)d_y + 1)$ sparse matrix, if $d_x \neq d_y$.

Therefore, as for the 1D case, the temporal average of K subsequent frames is

$$F(x, y) = \sum_{h_1=0}^{(K-1)d_x} \sum_{h_2=0}^{(K-1)d_y} f_0(x - h_1, y - h_2)g(h_1, h_2) = (f_0 * g)(x, y),$$

that can be rewritten in the Fourier domain as follows

$$\hat{F}(\omega_x, \omega_y) = \hat{f}_0(\omega_x, \omega_y)\hat{g}(\omega_x, \omega_y), \tag{8}$$

where the frequency domain is divided into $(4M + 1)^2$ components.

Moreover,

$$\begin{aligned} \hat{g}(\omega_x, \omega_y) &= \sum_{x=0}^{(K-1)d_x} \sum_{y=0}^{(K-1)d_y} g(x, y)e^{-i\frac{2\pi}{4M+1}\omega_x x} e^{-i\frac{2\pi}{4M+1}\omega_y y} = \\ &= \sum_{k=0}^{(K-1)} e^{-i\frac{2\pi}{4M+1}\omega_x k d_x} e^{-i\frac{2\pi}{4M+1}\omega_y k d_y} = \sum_{k=0}^{(K-1)} e^{-i\frac{2\pi}{4M+1}k(\omega_x d_x + \omega_y d_y)} = \\ &= |\hat{g}(\omega_x, \omega_y)|e^{-i\frac{\pi}{4M+1}(d_x\omega_x + d_y\omega_y)(K-1)}. \end{aligned}$$

The absolute value of \hat{g} is again a non linear function with respect to d_x and d_y while its phase is linear.

Hence, we can compare the phases of both members of the equation (8), i.e.

$$Arg(\hat{F}(\omega_x, \omega_y)) = Arg(\hat{f}_0(\omega_x, \omega_y)) + Arg(\hat{g}(\omega_x, \omega_y))$$

and then

$$-\frac{\pi}{4M + 1}(d_x\omega_x + d_y\omega_y)(K - 1) = Arg(\hat{F}(\omega_x, \omega_y)) - Arg(\hat{f}_0(\omega_x, \omega_y)).$$

d_x and d_y can be finally computed choosing the couples of values $(\omega_x, \omega_y) = (0, 1)$ and $(\omega_x, \omega_y) = (1, 0)$ and then

$$Arg(\hat{F}(0, 1)) - Arg(\hat{f}_0(0, 1)) = -\frac{(K - 1)\pi}{4M + 1}d_y,$$

and

$$Arg(\hat{F}(1, 0)) - Arg(\hat{f}_0(1, 0)) = -\frac{(K - 1)\pi}{4M + 1}d_x.$$

By inverting both equations we have

$$d_x = -\frac{(Arg(\hat{F}(1, 0)) - Arg(\hat{f}_0(1, 0)))(4M + 1)}{(K - 1)\pi} \tag{9}$$

and

$$d_y = -\frac{(Arg(\hat{F}(0, 1)) - Arg(\hat{f}_0(0, 1)))(4M + 1)}{(K - 1)\pi}. \tag{10}$$

Also in this case, there is no ambiguity for the sign of both horizontal and vertical motions.

It is worth outlining that the equivalence between the temporal averaging and the convolution with a fixed kernel allow us to strengthen the correlation between two or more subsequent frames. This corresponds to the search of equally spaced high amplitude peaks in a classical phase correlation based approach.

2.2 The Algorithm

For a group of K frames, each of dimension $((2\overline{M} + 1) \times (2\overline{M} + 1))$, the motion of each frame with respect to the first one $f_0(x, y)$ (reference frame) is estimated as follows:

- Split the frames into non overlapping blocks $B_{j,k}$ of dimension $((2M + 1) \times (2M + 1))$. $B_{j,k}$ is the j^{th} block in the frame $f_k(x, y)$ and then $j = 1, \dots, \lfloor \frac{(2\overline{M}+1) \times (2\overline{M}+1)}{(2M+1) \times (2M+1)} \rfloor$.

For each j :

1. Compute the average of the K corresponding blocks as in (2): $F_j(x, y) = \frac{1}{K} \sum_{k=0}^{K-1} B_{j,k}(x, y)$.
2. Evaluate the phase of the DFT of F_j at the frequency values $\omega = (1, 0)$ and $\omega = (0, 1)$.
3. Compute the phase of the DFT of the reference block $B_{j,0}$ at the same frequency values.
4. Compute the horizontal d_y and vertical d_x motion using (10) and (9), where F is substituted for F_j and f_0 is substituted for $B_{j,0}$.
5. At each block $B_{j,k}$, $k = 1, \dots, K$, assign the motion vector whose components are (kd_x, kd_y) .

Table 1. Averages of PSNR results achieved by the proposed model on Mobile and Coastguard sequences using both 16×16 blocks and 8×8 blocks. The results are compared with the Two Bit Transform based algorithm in [15] and the full search block matching algorithm (FS-BMA) [6].

| Sequence | Method | PSNR (16×16) | PSNR (8×8) |
|------------|----------|-------------------------|-----------------------|
| Mobile | Proposed | 26.72 db | 28.52 db |
| | 2BT | 22.72 db | 22.99 db |
| | FS-BMA | 22.99 db | 23.88 db |
| Coastguard | Proposed | 33.85 db | 35.20 db |
| | 2BT | 29.93 db | 30.50 db |
| | FS-BMA | 30.47 db | 31.58 db |

3 Experimental Results

The proposed model has been tested on various test video sequences. In particular, in this paper we give the results achieved on Mobile and Coastguard gray scale sequences, whose first frames are shown in Figs. 1 and 2. The first one is composed of 140 frames of size 240×352 while the second one is composed of 300 frames of size 352×288 . Even if the model assumes one constant dominant motion for each successive block, results are very satisfying yielding better performances than the classical block matching based approaches [6,15], as shown in Table 1, and with a very low computational cost. Figs. 3 and 4 depict PSNR results achieved for each frame of Mobile and Coastguard sequences. The first sequence reveals a significant improvement of quality in its last part. It is due to the fact that there are less blocks with multiple motion, for example there are less blocks containing both the train and the calendar. With regard to the second sequence, firstly a new object enters in the scene, then from frame 65 to frame 74 there is a rough vertical change in the camera position along with a loss of quality of the images. On the contrary, the second half of the sequence presents an almost constant scene and then PSNR increases.

In the experiments presented in this section, we use blocks of dimension 16×16 and the motion is estimated using two frames ($K = 2$). Nonetheless, the model allows to estimate the motion of more than one frame simultaneously by increasing the number of frames K . We perform also some experiments using $K = 3$. The PSNR results do not significantly change. There is a decreasing of 0.7 db on average while the computational effort greatly decreases. Also wider or smaller blocks have been used. For 8×8 blocks there is an improvement of 1.5 db on average for the analyzed sequences. For wider blocks the difference in PSNR grows as the assumption of one constant global motion is not verified.

Further experiments have been also performed on the same sequences corrupted by additive gaussian noise $\mathcal{N}(0, \sigma^2)$. The algorithm is robust only under very moderate noise since the phase may suffer from differences between frames. We have found that for $\sigma \leq 7$ there is a loss of quality of about 0.8 db while for higher σ it is necessary to pre-process the data by a moving average filter to preserve the goodness of the results. It is worth highlighting that for noisy sequences PSNR results can be improved by estimating the motion estimation of each single frame using more than two subsequent frames. In fact, the temporal averaging allows a sort of de-noising of the analyzed frames.

The algorithm requires simple and fast operations and then it is suitable for compression purposes and real time applications. In fact, its complexity is $O(4(2M+1)^2+6)$ for $K = 2$. Steps 1-3 of the algorithm depend on the size of the selected blocks while the last one is shared by all the pixels composing the block. For a 16×16 block, 4 operations per pixel are required. Therefore, the proposed algorithm is faster than [15], which requires 7 operations per pixel, yielding better quality sequences, as shown in Table 1. Compared to phase correlation based models, the proposed approach avoids the inversion of the DFT and then the search of the best peak in the time domain.



Fig. 1. First frame of Mobile sequence



Fig. 2. First frame of Coastguard sequence

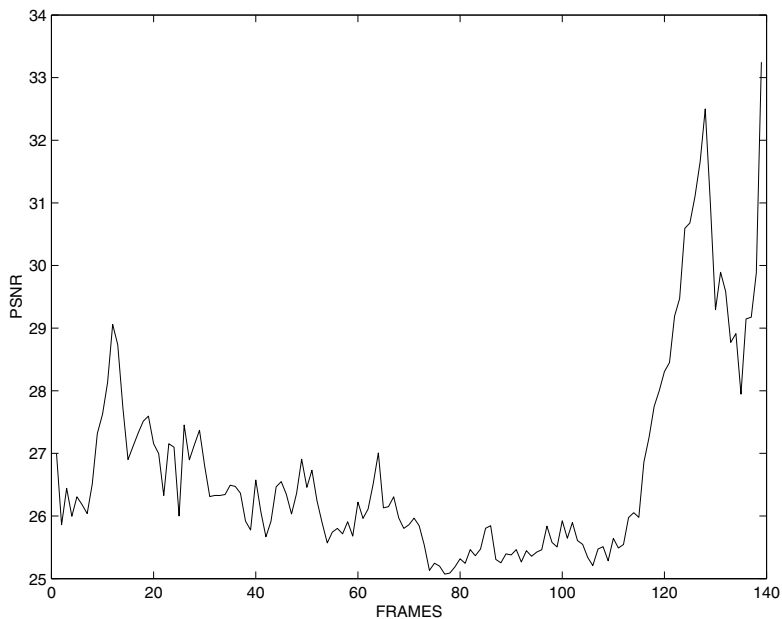


Fig. 3. PSNR versus number of frames of Mobile sequence processed by the proposed motion estimation algorithm using 16×16 blocks and 2 frames at a time

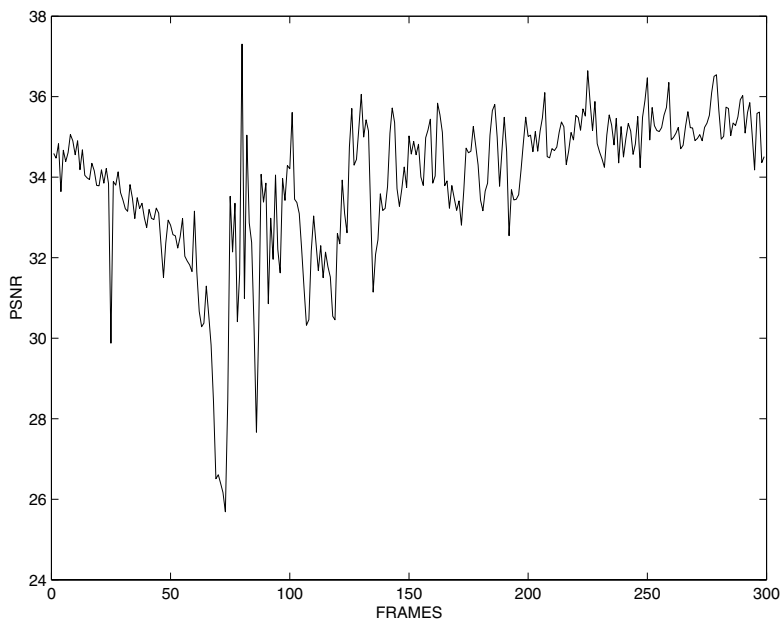


Fig. 4. PSNR versus number of frames of Coastguard sequence processed by the proposed motion estimation algorithm using 16×16 blocks and 2 frames at a time

4 Conclusions

In this paper we have presented a novel approach for motion estimation. It is based on the equivalence between the mean of corresponding blocks at different frames and the convolution of the reference block with Dirac comb kernel. This equivalence is then exploited in the Fourier domain and in particular looking at the phase for determining the motion. Experimental results show that the proposed model drastically reduces the computational effort and outperforms available approaches. Even though the topic of this paper was motion estimation for video coding, i.e. without noise, some experiments under moderate noise have been performed and discussed. Future research will consist of a deep investigation of the presented approach limits under severe noise conditions.

References

1. Y. Q. Shi, H. Sun, *Image and Video Compression for Multimedia Engineering*, CRC Press, New York, 2000.
2. A. N. Netravali, J. D. Robbins, *Motion-Compensated Television Coding: Part I*, *Bell Syst. Tech. J.*, 58 (3), 631-670, 1979.
3. J. D. Robbins, A. N. Netravali, *Recursive Motion Compensation: A Review*, in *Image Sequence Processing and Dynamic Scene Analysis*, T. S. Huang, Ed. Berlin, Germany: Springer-Verlag, pp. 73-103, 1983
4. G. Le Besnerais, F. Champagnat, *Dense Optical Flow by Iterative Local Window Registration*, *Proc. of IEEE International Conference on Image Processing*, pp. 137-140, 2005.
5. M. Ye, R.M. Haralick, L.G. Shapiro *Estimating Piecewise - Smooth Optical Flow with Global Matching and Graduated Optimization*, *IEEE PAMI*, Vol. 25, No. 12, pp. 1625-1630, December 2003.
6. J. M. Jou, P.-Y. Chen and J.-M. Sun, *The Gray Prediction Search Algorithm for Block Motion Estimation*, *IEEE Trans. Circuits Syst. Video Technol.*, vol 9, no. 6, pp. 843-848, Sept. 1999.
7. K. N. Nam, J.-S. Kim, R.-H. Park and Y. S. Shim, *A Fast Hierarchical Motion Vector Estimation Algorithm using Mean Pyramid*, *IEEE Trans. Circuits Syst. Video Technology*, vol. 5, no. 4, pp. 344-351, Aug. 1995
8. S. Zhu and K.-K. Ma, *A new Diamond Search Algorithm for Fast Block Matching Estimation*, *IEEE Trans. Circuits Syst. Video Technology*, vol. 9, no.2, pp. 287-290, Feb. 2000.
9. C. J. Kuo, C. H. Yeh, and S. F. Odeh, *Polynomial Search Algorithm for Motion Estimation*, *IEEE Trans. Circuits Syst. Video Technology*, vol. 10, no.5, pp. 813-818, Aug. 2000.
10. U.-V. Koc, K. J. R. Liu, *DCT-Based Motion Estimation*, *IEEE Trans on Image Processing*, vol. 7, no. 7, July 1998
11. C.D. Kughlin, D.C. Hines, *The phase correlation image alignment method*, *Proceedings of IEEE Int. Conf. Systems, Man. and Cybernetics*, pp. 163-165, September 1975.
12. M.Li, M. Biswas, S. Kumar, T. Nguyen, *DCT- based phase correlation motion estimation*, *Proceedings of IEEE Int. Conf. on Image Processing*, pp. 445-448, 2004.

13. M. Biswas, S. Kumar, T. Nguyen, *Efficient phase correlation motion estimation using approximate normalization*, Proceedings of IEEE Int. Conf. on Signals, Systems and Computers, Vol. 2, pp. 1727-1730, November 2004.
14. Y.M. Chou, H.M. Hang, *A New Motion Estimation Method using Frequency Components*, Journal of Visual Communication and Image Representation, Vol. 8, No. 1, pp. 83-96, March 1997.
15. A. Erturk, S. Erturk, *Two Bit Transform for Binary Block Estimations*, Transactions on Circuits and Systems for Video Technology, Vol. 15, No. 7, July 2005.

A Spatio-temporal Filtering

By putting eq. (1) into (2) we have

$$F(x) = \frac{1}{K} \sum_{h=0}^{K-1} f_0(x - hd). \tag{11}$$

Each element of $F(x)$ is then the sum of K elements of f_0 , suitably sampled.

Using the function defined in eq.(3), eq. (11) can be formally rewritten as the convolution of the function $f_0(x)$ with $g(x)$, i.e.

$$\begin{aligned} F(x) &= \frac{1}{K} \sum_{h=0}^{(K-1)d} f_0(x - h) \sum_{k=0}^{K-1} \delta(h - kd) = \\ &\sum_{h=0}^{(K-1)d} f_0(x - h)g(h) = (f_0 * g)(x). \end{aligned} \tag{12}$$

It turns out that a temporal averaging of corresponding signals is equivalent to a spatial convolution using Dirac comb, whose sampling period depends on the motion d .

Moreover, the Discrete Fourier Transform of $g(x)$ is

$$\hat{g}(\omega_s) = \sum_{x=0}^{4M} g(x)e^{-i\omega_s x},$$

where $\omega_s = \frac{2\pi}{4M+1}\omega$.

By using (3), the compact support of $g(x)$ and the shift property of the Fourier transform , we have

$$\begin{aligned} \hat{g}(\omega_s) &= \frac{1}{K} \sum_{x=0}^{(K-1)d} \left(\sum_{k=0}^{(K-1)} \delta(x - kd) \right) e^{-i\omega_s x} = \\ &\frac{1}{K} \sum_{k=0}^{(K-1)} \sum_{x=0}^{(K-1)d} \delta(x - kd)e^{-i\omega_s x} = \sum_{k=0}^{K-1} e^{-i\omega_s k d}. \end{aligned}$$

It is the partial sum of the geometric series and then it can be rewritten as

$$\begin{aligned}\hat{g}(\omega) &= \frac{\sin\left(\frac{\omega_s}{2} K d\right)}{\sin\left(\frac{\omega_s}{2} d\right)} e^{-i\frac{\omega_s}{2} d(K-1)} = \\ &= |\hat{g}(\omega_s)| e^{-i\frac{\omega_s}{2} d(K-1)}.\end{aligned}\tag{13}$$

Optic Flow from Multi-scale Dynamic Anchor Point Attributes

B.J. Janssen, L.M.J. Florack, R. Duits, and B.M. ter Haar Romeny

Eindhoven University of Technology,
Den Dolech 2, Postbus 513
5600 MB Eindhoven, The Netherlands

{B.J.Janssen, L.M.J.Florack, R.Duits, B.M.terHaarRomeny}@tue.nl

Abstract. Optic flow describes the apparent motion that is present in an image sequence. We show the feasibility of obtaining optic flow from dynamic properties of a sparse set of multi-scale anchor points. Singular points of a Gaussian scale space image are identified as feasible anchor point candidates and analytical expressions describing their dynamic properties are presented. Advantages of approaching the optic flow estimation problem using these anchor points are that (i) in these points the notorious aperture problem does not manifest itself, (ii) it combines the strengths of variational and multi-scale methods, (iii) optic flow definition becomes independent of image resolution, (iv) computations of the components of the optic flow field are decoupled and that (v) the feature set inducing the optic flow field is very sparse (typically $< \frac{1}{2}\%$ of the number of pixels in a frame). A dense optic flow vector field is obtained through projection into a Sobolev space defined by and consistent with the dynamic constraints in the anchor points. As opposed to classical optic flow estimation schemes the proposed method accounts for an explicit scale component of the vector field, which encodes some dynamic differential flow property.

1 Introduction

Typical optic flow estimation algorithms are based on the brightness constancy assumption initially proposed by Horn & Schunck [1]. This constraint is insufficient to determine optic flow unambiguously since constant brightness occurs on surfaces of codimension one (curves in 2D, surfaces in 3D, etc.). The intrinsic ambiguity has become known as the *aperture problem*. The ambiguity is typically resolved by adding extra constraints to the optic flow constraint equation, or similarly, through an appropriate regularizer in a variational formulation. Horn & Schunck used a quadratic regularizer. Ever since many alternative regularisation schemes have been proposed, essentially following the same rationale.

Apart from variational methods [2,3,4] that are similar to the method proposed by Horn & Schunck, correlation-based [5,6], frequency-based [7] and phase-based methods [8] have been proposed. In order to cover large displacements several coarse-to-fine strategies of these techniques have been devised [9,10]. Werkhoven et al., Florack et al. and Suinesiaputra et al. [11,12,13] developed biologically inspired optic flow estimation methods incorporating “optimal” local scale selection. The work by Florack et al.

shows that taking notion of scale may lead to superior performance compared to the optic flow estimation algorithms evaluated by Barron et al. [14]. Recent work by Brox et al. and Bruhn et al. [2,15] show impressive results of current state of the art variational optic flow estimation techniques.

In this work we combine the strength of the variational rationale with the added value of the multi-scale framework, by proposing a new paradigm for optic flow estimation. To this end we investigate flow of so called *anchor points* in Gaussian scale space. These anchor points could be any type of identifiable isolated points in Gaussian scale space. In these points the aperture problem is nonexistent and therefore the flow can be unambiguously measured. If these points carry “sufficiently rich” dynamic information one may hypothesise that any high resolution dense optic flow field that is consistent with the constraints posed by the anchor points and their dynamic features will be a potential representative of the “underlying” optic flow field one aims to extract. This has the additional advantage that optic flow definition becomes independent of image resolution, as opposed to “constant brightness” paradigms. Note that we completely abandon the constant brightness ansatz, as the anchor points typically have non-conserved intensity attributes. The proposed optic flow estimation method which is depicted in Figure 1 is of a *truly multi-scale* nature since each anchor point lives at a certain scale. This leads to a method in which automatic scale selection is manifest as opposed to the approach by Florack et al. [11] which require an extrinsic criterion for scale selection. We apply our method to a specific set of anchor points. One should however notice that any set of intrinsic points can be used. Section 2 discusses the subject of anchor points.

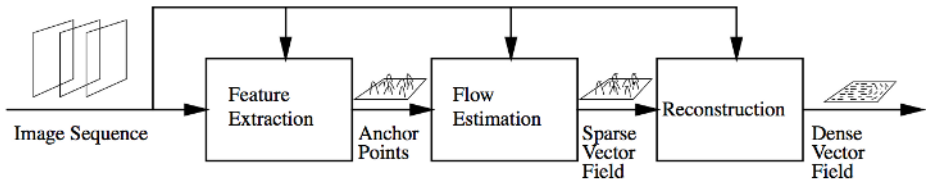


Fig. 1. An overview of the proposed optic flow estimation algorithm. First anchor points are selected. The second step is the flow estimation that is done on the sparse set of selected anchor points that are present in the scale space of a single frame. Finally this sparse multi-scale vector field is converted to a dense vector field at grid scale.

The sparse multi-scale vector field constructed by the motion of the anchor points has to be converted to a dense high resolution vector field at scale $s = s_0$, in which $s_0 > 0$ is related to grid scale. To this extent we apply a reconstruction method similar to the method proposed by Janssen et al. [16,17,18,19]. The case of stationary reconstruction is discussed in Section 3. In Section 4 the extension of the reconstruction algorithm to vector valued images and results of its application to flow reconstruction are discussed.

We stress that the emphasis in this article is not on performance (which, indeed, is not quite state-of-the-art), but on a paradigm shift for optic flow extraction with great potential. Many improvements to the ansatz proposed here are readily conceived of, such as the extension of intrinsic anchor pointset, utilization of higher order dynamic attributes

(e.g accelerations), modification of Sobolev inner product space, slick exploitation of time scale, etc. . We believe that a thorough investigation of these suggestions will significantly improve performance.

2 Anchor Points

A Gaussian scale space representation $u(\mathbf{x}; s)$ in n spatial dimensions is obtained by convolution of a raw image $f(\mathbf{x})$ with a normalized Gaussian:

$$\begin{aligned}
 u(\mathbf{x}; s) &= (f * \varphi_s)(\mathbf{x}) , \\
 \varphi_s(\mathbf{x}) &= \frac{1}{\sqrt{4\pi s}^n} e^{-\frac{\|\mathbf{x}\|^2}{4s}} .
 \end{aligned}
 \tag{1}$$

We aim to identify and determine the velocity of anchor points in the scale space of an image sequence. The scale space of an image sequence is a concatenation of the scale spaces of the images of the sequence. This means time scale is not taken as an explicit dynamic parameter. The type of anchor point that is discussed in this paper is the so called *singular point*. Apart from or in addition to these points any type of generically isolated point that is intrinsic in the scale space of an image could have been considered. An introduction to scale space theory can be found in a tutorial book by ter Haar Romeny [20] and monographs by Lindeberg [21] and Florack [22].

2.1 Singular Points

A *singular point* is a *non-Morse critical point* of a Gaussian scale space image. Scale s is taken as a control parameter. This type of point is also referred to in the literature as a *degenerate spatial critical point* or as a *toppoint* or *catastrophe*.

Definition 1 (Singular Point). A *singular point* $(\mathbf{x}; s) \in \mathbb{R}^{n+1}$ is defined by the following equations.

$$\begin{cases} \nabla u(\mathbf{x}; s) = 0 , \\ \det \nabla \nabla^T u(\mathbf{x}; s) = 0 . \end{cases}
 \tag{2}$$

Here ∇ denotes the spatial gradient operator.

The behavior near singular points is the subject of *catastrophe theory*. Damon studied the applicability of established *catastrophe theory* in a scale space context [23]. Florack and Kuijper have given an overview of the established theory in their paper about the topological structure of scale space images for the generic case of interest, and investigated geometrical aspects of generic singularities [24]. More on catastrophe theory in general can be found in a monograph by Gilmore [25].

Singular points can be found by following critical paths, which are curves of vanishing gradient. A generic singular point manifests itself either as a creation or annihilation event involving a pair of critical points.

2.2 Flow

When tracking anchor points over time we have one extra state variable. Spatial non-Morse critical points follow a path through scale spacetime in the scale space of an image sequence. Such a path can be described by a parameterised curve $p(t)$ as long as it is transversal to time frames. Along this curve, of which a visualisation can be found in Figure 2, the properties of the point as described in equation (2) – given that the t -transversality criterion holds¹ – do not change.

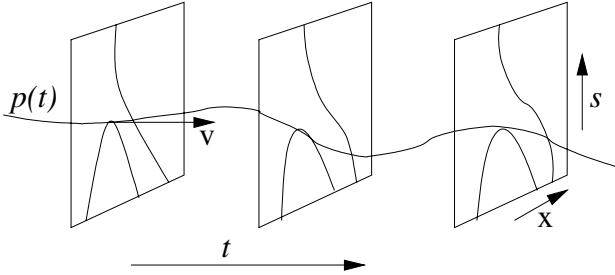


Fig. 2. The path $p(t)$ that a singular point travels through time t . In the frames the zero crossings of the image gradient are depicted as a function of scale s . These lines are called critical paths.

We are interested in finding a flow vector in each singular point that is present in the scale space of an image.

Definition 2 (Flow Vector). *The motion of a singular point is described by a flow vector $\mathbf{v} \in \mathbb{R}^{n+1}$,*

$$\mathbf{v} = \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{s} \end{bmatrix}. \tag{3}$$

Here $\dot{\mathbf{x}} \in \mathbb{R}^n$ denotes a spatial vector describing the flow’s spatial components and $\dot{s} \in \mathbb{R}$ describes the scale component. The (suppressed) time component² of this vector is implicit and taken as $\dot{t} = 1$. In other words it is assumed that time t is a valid parameter of the flow vector’s integral curve. A dot is shorthand for $\frac{\partial}{\partial t}$.

We ignore boundaries of the image sequence. By definition of scale space and the fact that we are studying natural image sequences it can be assumed that the image sequence of interest is at least piecewise continuously differentiable (even analytical with respect to spatial variables). By definition the properties of the tracked point along the curve $p(t)$ (see Figure 2),

$$\begin{bmatrix} \nabla u(t; \mathbf{x}, s) \\ \det \mathbf{H}(t; \mathbf{x}, s) \end{bmatrix} = \mathbf{0}, \tag{4}$$

¹ This transversality condition is implicitly required in the constant brightness based optic flow rationale.

² This assumption is called the “temporal gauge” and reflects the transversality assumption.

do not change. $\mathbf{H}(t; \mathbf{x}, s)$ is the $n \times n$ matrix with components $\frac{\partial^2 u(t; \mathbf{x}, s)}{\partial x^a \partial x^b}$ where a and b index the spatial dimensions. If the Jacobian determinant does not degenerate,

$$\det \begin{bmatrix} \mathbf{H}(t; \mathbf{x}, s) & \nabla \frac{\partial u(t; \mathbf{x}, s)}{\partial s} \\ \vdots & \vdots \\ \nabla^T \det \mathbf{H}(t; \mathbf{x}, s) \dots & \frac{\partial \det \mathbf{H}(t; \mathbf{x}, s)}{\partial s} \end{bmatrix} \neq 0, \tag{5}$$

the implicit function theorem can be applied to obtain the flow along the parameterised curve $p(t) = (\mathbf{x}(t), s(t))$ through time. Assuming that equation (5) holds, the movement of the anchor point in a sufficiently small neighborhood of the initial position is given by inversion of

$$\begin{bmatrix} \mathbf{H}(t; \mathbf{x}, s) & \nabla \frac{\partial u(t; \mathbf{x}, s)}{\partial s} \\ \vdots & \vdots \\ \nabla^T \det \mathbf{H}(t; \mathbf{x}, s) \dots & \frac{\partial \det \mathbf{H}(t; \mathbf{x}, s)}{\partial s} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{s} \end{bmatrix} = \begin{bmatrix} \nabla \frac{\partial u(t; \mathbf{x}, s)}{\partial t} \\ \frac{\partial \det \mathbf{H}(t; \mathbf{x}, s)}{\partial t} \end{bmatrix}. \tag{6}$$

This follows straight forward by setting

$$\frac{d}{dt} \begin{bmatrix} \nabla u(t; \mathbf{x}, s) \\ \det \mathbf{H}(t; \mathbf{x}, s) \end{bmatrix} = \mathbf{0}, \tag{7}$$

in which $\frac{d}{dt}$ denotes the total time derivative along the flow.

Problems arise when the point that is followed disappears, i.e. if the t -transversality condition fails: which may happen at isolated points in the image sequence (in-between image frames). At such an event the point in scale spacetime changes to a non-Morse critical point and the implicit function theorem does not hold anymore. This can be detected by checking if the Jacobian determinant degenerates. As such this can be used to evaluate the validity of the estimated flow vector.

2.3 Error Measure for Anchor Point Localisation and Tracking

Because of noise the resulting vector of equation (6) will not point to the exact position of the anchor point in the next frame. A refinement of the solution can be made by using a Taylor expansion around the estimated position of the anchor point as proposed by Florack and Kuijper [24]. Evaluation of

$$\begin{bmatrix} \mathbf{x}' \\ s' \end{bmatrix} = \begin{bmatrix} -\det \mathbf{H}(t; \mathbf{x}, s) \mathbf{H}^{-1}(t; \mathbf{x}, s) \frac{\partial \nabla u(\mathbf{x}; s)}{\partial s} \\ \det \mathbf{H}(t; \mathbf{x}, s) \end{bmatrix} \tag{8}$$

near a singular point results in a vector pointing to the actual position of the singular point. The prime in \mathbf{x}' and s' denotes differentiation with respect to a path parameter p

implicitly defined by $\frac{ds}{dp} = \det \mathbf{H}(t; \mathbf{x}, s)$. For a detailed derivation of equation (8) we refer to the original article [24]. Notice that the so-called ‘‘cofactor matrix’’

$$\tilde{\mathbf{H}}(t; \mathbf{x}, s) \stackrel{\text{def}}{=} \det \mathbf{H}(t; \mathbf{x}, s) \mathbf{H}^{-1}(t; \mathbf{x}, s) \tag{9}$$

is well defined even in the limit where $\det \mathbf{H}(t; \mathbf{x}, s) \rightarrow 0$. Equation (8) gives us an error correction mechanism that can be incorporated in tracking. Note that one can march this vector field in order to obtain a higher accuracy.

3 Stationary Reconstruction

In order to obtain a dense flow field from singular points endowed with their dynamic attributes a so called dense flux field is generated that is consistent with these features. In this section we will study the essence of this so-called ‘‘reconstruction’’ algorithm in the simplified context of stationary scalar images. Optic flow will be discussed in the next section.

Stationary reconstruction from differential attributes of singular points can be connected to generalized sampling theory as proposed by Unser and Aldroubi [16]. They find a consistent reconstruction of a signal from its integer shifted filter responses, i.e. a reconstruction that is indistinguishable from its original when observed through the filters the features were extracted with. The consistency requirement is adopted by Lillholm, Nielsen and Griffin [17,18] in pursue of stationary reconstruction from scale space interest points. They describe a variational framework that finds a consistent reconstruction that minimizes a so called prior.

The reconstruction problem boils down to the selection of an instance of the metameric class consisting of $g \in \mathbb{L}_2(\mathbb{R}^2)$ such that

$$(\psi_i, g)_{\mathbb{L}_2} = c_i, \quad (i = 1 \dots N) \tag{10}$$

with ψ_i denoting the distinct localized filters that generate the i^{th} filter response $c_i = (\psi_i, f)_{\mathbb{L}_2}$.

In case the prior is formed by an inner product the reconstruction scheme aims to establish a reconstruction g that satisfies equation (10) and minimizes

$$E(g) = \frac{1}{2}(g, g)_{\mathcal{H}}. \tag{11}$$

Here $(\cdot, \cdot)_{\mathcal{H}}$ denotes an inner product on a Hilbert space \mathcal{H} (to be specified later). Since g satisfies equation (10) we may as well write

$$g = \operatorname{argmin} E(g) = \frac{1}{2}(g, g)_{\mathcal{H}} - \lambda^i ((\kappa_i, g)_{\mathcal{H}} - c_i). \tag{12}$$

Summation convention applies to upper and lower feature indices $i = 1 \dots N$. The first term in the Euler-Lagrange formulation, cf. equation (12), is referred to as the *prior*. The remainder consists of a linear combination of *constraints*, recall equation (10),

with Lagrange multipliers λ^i . κ_i are the filters in \mathcal{H} that correspond to the filters ψ_i in \mathbb{L}_2 , i.e. by definition we have,

$$(\kappa_i, f)_{\mathcal{H}} = (\psi_i, f)_{\mathbb{L}_2}. \tag{13}$$

The solution to equation (12) can be found by orthogonal projection in \mathcal{H} of the original image f on the linear space V spanned by the filters κ_i , i.e.

$$g = \mathcal{P}_V f = (\kappa^i, f)_{\mathcal{H}} \kappa_i. \tag{14}$$

Here we have defined $\kappa^i \stackrel{\text{def}}{=} G^{ij} \kappa_j$ with Gramm matrix

$$G_{ij} = (\kappa_i, \kappa_j)_{\mathcal{H}} \tag{15}$$

and $G^{ik} G_{kj} = \delta_j^i$.

We adopt the prior that is formed by the first order Sobolev inner product

$$(f, g)_{\mathcal{H}} = (f, g)_{\mathbb{L}_2} + (\gamma \nabla f, \gamma \nabla g)_{\mathbb{L}_2}. \tag{16}$$

with $\gamma \in \mathbb{R}^+$ a scalar that controls the smoothness of the reconstructed function. Because of this choice the explicit form of the filters κ_i is as follows,

$$\begin{aligned} \kappa_i &= (I - \gamma^2 \Delta)^{-1} \psi_i \\ &= \mathcal{F}^{-1} \left(\omega \mapsto \frac{1}{1 + \gamma^2 \|\omega\|^2} \mathcal{F}(\psi_i)(\omega) \right). \end{aligned} \tag{17}$$

For two dimensions ($n = 2$) the convolution filter that represents the linear operator $(I - \gamma^2 \Delta)^{-1}$ equals

$$\phi_{\gamma}(x, y) = \frac{1}{2\pi\gamma^2} K_0 \left[\frac{\sqrt{x^2 + y^2}}{\gamma} \right] \tag{18}$$

with K_0 representing the zeroth order modified Bessel function of the second kind [26].

4 Flow Reconstruction

The reconstruction algorithm that is described in the previous section will be used to obtain a dense flux field that explicitly contains the desired flow information. The velocity of the anchor points (recall equation (6)) is encoded in a multi-scale flux field \mathbf{J} .

4.1 Proposed Vector Field Retrieval Method

Definition 3 (Dense Flux Field). *The dense flux field $\mathbf{j} \in \mathbb{L}_2^m(\mathbb{R}^n)$ at scale $s = s_0$, in which $s_0 > 0$ is related to the grid scale, is defined by*

$$\mathbf{j} = f \mathbf{v} \tag{19}$$

with $f \in \mathbb{L}_2(\mathbb{R}^n)$ the image intensity and $\mathbf{v} \in \mathbb{L}_2^m(\mathbb{R}^n)$ the dense optic flow field introduced in Definition 2.

In order to find a dense flux field from flux measurements in scale space the minimization with respect to $\mathbf{k} \in \mathbb{L}_2^m(\mathbb{R}^n)$ of the following Euler-Lagrange formalism is proposed, in which P denotes the collection of anchor points,

$$E[\mathbf{k}] = \sum_{a=1}^m \left[\frac{1}{2}(k_a, k_a)\mathcal{H} + \sum_{p \in P} c_p^a [(k_a, \phi_p)\mathcal{H} - J_p^a] \right] \stackrel{\text{def}}{=} \sum_{a=1}^m E_a[k_a]. \quad (20)$$

With $\mathbf{k} = (k_1, \dots, k_m)$ denoting the estimated dense flux field (gray value times scale space velocity) at scale $s = s_0$. The filter responses for each point $p \in P$ are

$$\mathbf{J}_p = \int_{\mathbb{R}^n} \mathbf{j} \phi_p dV. \quad (21)$$

These measurements of the image flux in scale space can only be performed reliably in well defined anchor points. We restrict ourselves to the anchor points introduced in Section 2, namely singular points. The reader should note that there is no theoretical objection against the use of any other point type.

Equation (20) is a sum of positive convex energies and can therefore be minimized for each component of \mathbf{k} separately. The fact that equation (20) is decoupled for each spatial and scale component readily yields

$$\mathbf{k} = \mathcal{P}_V \mathbf{j}, \quad (22)$$

with V denoting the span of the filters $\{\phi_p\}_{p \in P}$ and \mathcal{P}_V the component-wise linear projection onto V . Recall Section 3, notably equation (14).

5 Evaluation

The results of the optic flow estimation are evaluated quantitatively by means of the angular error of the estimated vector field as proposed by Barron et al. [14]. The test sequence used for evaluation is the familiar *Yosemite Sequence*. Qualitative evaluation is performed by analysis of the flow field of the *Translating Tree Sequence* and the *Hamburg Taxi Sequence*. These image sequences and their ground truths, if available, are obtained from the University of Western Ontario (<ftp://ftp.csd.uwo.ca> in the directory `/pub/vision/`) as mentioned in a paper by Barron et al. [14]. The algorithm is implemented in Mathematica [27]. For the properties of the different sequences we refer to the original article [14].

The anchor pointset P is obtained in two steps. Initial guesses of the positions of the singular points are obtained using ScaleSpaceViz [28] which basically uses a zero-crossings method for solving (2). The locations of the points returned by this program are refined by iteration over equation (8) until the estimated error is below 10^{-3} pixels. In case a singular point position cannot be refined, which can be caused by a poor initial guess of its position, the point is discarded. The result of these two steps is a highly accurate representation of P .

The dense flux field is estimated for each spatial component separately as described in equation (22). The scale component is, for the sake of simplicity, ignored as it has no

counterpart in the algorithms evaluated in the comparison paper by Barron et. al.[14] or elsewhere. More on this new aspect will be discussed in section 6.

For each of the test sequences the velocities of the singular points are estimated by inversion of equation 6. When the velocity of a singular point is unstable the point is not taken into account in the algorithm. The velocity of a singular point is considered stable as long as its stability measure (recall equation (5)) satisfies

$$\det \begin{bmatrix} \mathbf{H}(t; \mathbf{x}, s) & \nabla \frac{\partial u(t; \mathbf{x}, s)}{\partial s} \\ \vdots & \vdots \\ \nabla^T \det \mathbf{H}(t; \mathbf{x}, s) \dots \frac{\partial \det \mathbf{H}(t; \mathbf{x}, s)}{\partial s} \end{bmatrix} > \epsilon . \tag{23}$$

In the experiments ϵ is set to 10^{-5} . This settings of ϵ is an educated guess and should be subject of further research (it is likely to depend on the condition number of the linear system).

The time derivatives in equation (6) are either taken as two-point central derivatives or as Gaussian derivatives with fixed time scale τ . The Gaussian derivatives are computed with a scale component in the time direction, notably τ . The performance of the reconstruction depends on the setting of the parameter γ introduced in equation (16). Since the goal of this section is to evaluate the feasibility of the proposed algorithm only a fixed sensible setting of $\gamma = 16$ pixels, is taken into account. In theory one obtains smoother reconstructions as γ tends to infinity, all consistent with the dynamic anchor point constraints. Definition 3 is applied to resolve the desired dense optic flow field.

5.1 Qualitative Evaluation

A visualization of the angular component of the estimated flow vectors belonging to the *Translating Tree Sequence* with time scale set to $\tau = 3$ frames is presented in Figure 3. In this case problems occur at the boundaries. However only those edges that possess appearing and disappearing singular points show problems. The results could be influenced negatively by the low number of features (95 singular points with an image size of 150×150 pixels) that were used in the algorithm. It is remarkable that the translating structure of the flow field is correctly detected by our algorithm without explicitly accounting for this structure.

The optic flow field of the *Hamburg Taxi Sequence* superimposed on the image on which the optic flow estimation is performed is shown in Figure 4. It shows all vectors that possess a magnitude larger than 0.2 pixels/frame. Most notable is the fact that the pedestrian, present in the upper left of the image sequence, is detected by our algorithm. Also some false vectors are present in the scene. This could be caused by numerical errors that occur when high order derivatives are taken at fine scales. A more restricting setting of ϵ will remove these vectors but coarsens the set of anchor points too much. Boundary problems are visible at the location where the van enters the sequence.

5.2 Quantitative Evaluation

A more objective approach to the evaluation of the quality of the estimated flow field is the average angular error [14]. The results of the experiments conducted on the *Yosemite*



Fig. 3. The estimated flow field of *Translating Tree Sequence* using time scale $\tau = 3$ frames

Table 1. Summary of the performance of the proposed optic flow retrieval method applied to the *Yosemite Sequence*

| Time Scale | Average Angular Error | Standard Deviation of Angular Error | # Singular Points |
|------------|-----------------------|-------------------------------------|-------------------|
| 1 | 24.93° | 36.00° | 352 |
| 2 | 19.19° | 34.45° | 303 |
| 3 | 19.97° | 34.30° | 271 |
| 4 | 22.94° | 42.61° | 251 |
| 5 | 25.00° | 39.50° | 226 |

Sequence are summarised in Table 1. It shows the angular error, standard deviation of the angular error and the number of singular points used in the reconstruction step for different time scales τ .

When the time scale is increased less singular points can be taken into account in the reconstruction step of the algorithm. The setting of the time scale is a tradeoff between the number of features that can be taken into account (which decreases with time scale) and noise robustness (which increases with time scale).

A histogram of the angular errors of the estimated optic flow in the *Yosemite Sequence* using a time scale of $\tau = 2$ frames is shown in Figure 5. The figure shows that

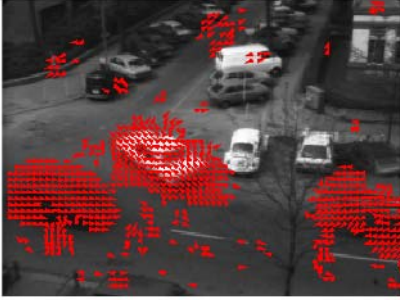


Fig. 4. The estimated flow field of *Hamburg Taxi Sequence* using time scale $\tau = 2$ frames superimposed on the image on which the calculations were applied. Only vectors that denote a velocity larger than 0.2 pixels/frame are displayed.

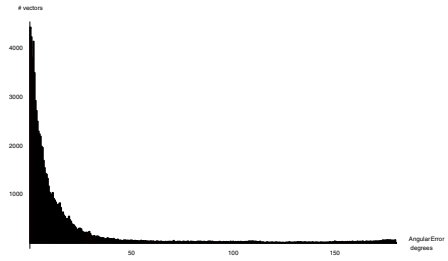


Fig. 5. Histogram of the angular error of the estimated flow field belonging to the *Yosemite Sequence*

the number of erroneous vectors decreases rapidly with the magnitude of the angular error, which is a promising result. Similar results are obtained for different time scales. Still many vectors that point to the opposite direction are present. This increases the standard deviation of the angular error that is presented in Table 1. Since we did not develop a confidence measure it is unjustified to leave these incorrect vectors out.

6 Summary and Recommendations

We have proposed an inherent multi scale optic flow estimation method that finds a flow field using the dynamic properties of so called anchor points. These anchor points can be any type of point as long as it is isolated and geometrically identifiable. We selected singular points of a Gaussian scale space image for the evaluation of our method. General reconstruction theory can be applied to optic flow retrieval from anchor points by encoding the dynamic properties of these points in a so called flux field. In order to obtain a dense flux field from the sparse set of flux vectors at multi scale anchor points, which is related to the optic flow of an image sequence, general reconstruction theory can be applied to each dimension independently.

The evaluation of zeroth order flow estimation, for which up to zeroth order dynamic features are taken into account, shows reasonable results considering the sparseness of features.

Problems that showed up are mainly caused by too sparse a set of features. Other causes of error are instability of the anchor point flow estimation, which is in general caused by occlusion and the image boundary, and numerical instability in the inversion step that is present in the reconstruction algorithm.

One of the unique properties of the proposed algorithm is that the velocity vectors possess a scale component. This can potentially be useful for segmentation and camera

motion estimation. We neglected the presence of this scale component in the reconstruction process. It does give more information about the local structure of the vector field and can therefore be used as an extra constraint for the estimation of the spatial components of the optic flow field. A “zoom”, for example, will cause singular points to translate and move downwards in scale. A method to add this extra property in the estimation of the optic flow field has yet to be proposed.

6.1 Recommendations

We suggest the following improvements to the algorithm which may lead to a significant performance increase:

- Taking into account higher order features (accelerations etc.).
- Improved reconstruction, e.g. by choosing a more appropriate space for projection.
- Using more anchor points.

Possible anchor point candidates are singular points of the laplacian of an image, so called blob points as proposed by Lindenberg [29], scale space saddles as proposed by Koenderink [30] or other singular or critical points. One can use all these point types simultaneously.

Acknowledgement

The Netherlands Organisation for Scientific Research (NWO) is gratefully acknowledged for financial support.

References

1. Horn, B.K.P., Schunck, B.G.: Determining optical flow. *Artificial Intelligence* **17** (1981) 185–203
2. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: *Proceedings of the European Conference on Computer Vision*. Volume 3024 of *Lecture Notes in Computer Science.*, Springer-Verlag (2004) 25–36
3. Nagel, H.H.: Displacement vectors derived from 2nd order intensity variations in image sequences. *Computer Vision, Graphics and Image Processing* **21** (1983) 85–117
4. Uras, S., Giroso, F., Verri, A., Torre, V.: A computational approach to motion perception. *Biological Cybernetics* **60** (1988) 79–87
5. Anandan, P.: A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision* **2** (1989) 283–310
6. Singh, A.: *Optic Flow Computation: a Unified Perspective*. IEEE Computer Society Press, Los Alamitos, CA (1991)
7. Heeger, D.: Optical flow using spatio-temporal filters. *International Journal of Computer Vision* **1** (1988) 279–302
8. Fleet, D.J., Jepson, A.D.: Computation of component image velocity from local phase information. *International Journal of Computer Vision* **5** (1990) 77–104
9. Terzopoulos, D.: Image analysis using multigrid relaxation methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8** (1986) 129–139

10. Wu, Y.T., Kanade, T., Li, C.C., Cohn, J.: Image registration using wavelet-based motion model. *International Journal of Computer Vision* **38** (2000) 129–152
11. Florack, L.M.J., Niessen, W.J., Nielsen, M.: The intrinsic structure of optic flow incorporating measurement duality. *International Journal of Computer Vision* **27** (1998) 263–286
12. Suinesiaputra, A., Florack, L.M.J., Westenberg, J.J.M., ter Haar Romeny, B.M., Reiber, J.H.C., Lelieveldt, B.P.F.: (Optic flow computation from cardiac MR tagging using a multi-scale differential method—a comparative study with velocity-encoded mri) 483–490
13. Werkhoven, P., Toet, A., Koenderink, J.J.: Displacement estimates through adaptive affinities. *IEEE Trans. Pattern Anal. Mach. Intell.* **12** (1990) 658–663
14. Barron, J.L., Fleet, D.J., Beauchemin, S.S.: Performance of optical flow techniques. *International Journal of Computer Vision* **12** (1994) 43–77
15. Bruhn, A., Weickert, J., Feddern, C., Kohlberger, T., Schnörr, C.: Variational optical flow computation in real time. *IEEE Transactions on Image Progressing* **14** (2005) 608–615
16. Unser, M., Aldroubi, A.: A general sampling theory for nonideal acquisition devices. *IEEE Transactions on Signal Processing* **42** (1994) 2915–2925
17. Lillholm, M., Nielsen, M., Griffin, L.D.: Feature-based image analysis. *International Journal of Computer Vision* **52** (2003) 73–95
18. Nielsen, M., Lillholm, M.: What do features tell about images? In Kerckhove, M., ed.: *Scale-Space and Morphology in Computer Vision: Proceedings of the Third International Conference, Scale-Space 2001, Vancouver, Canada*. Volume 2106 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin (2001) 39–50
19. Janssen, B., Kanters, F., Duits, R., Florack, L.: A linear image reconstruction framework based on Sobolev type inner products. In: *5th International Conference on Scale Space and PDE Methods in Computer Vision*. (2005)
20. Haar Romeny, B.M.t.: *Front-End Vision and multiscale image analysis*. Kluwer Academic Publishers (2002)
21. Lindeberg, T.: *Scale-Space Theory in Computer Vision*. The Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, Dordrecht, The Netherlands (1994)
22. Florack, L.M.J.: *Image Structure*. Volume 10 of *Computational Imaging and Vision Series*. Kluwer Academic Publishers, Dordrecht, The Netherlands (1997)
23. Damon, J.: Local Morse theory for solutions to the heat equation and Gaussian blurring. *Journal of Differential Equations* **115** (1995) 368–401
24. Florack, L., Kuijper, A.: The topological structure of scale-space images. *Journal of Mathematical Imaging and Vision* **12** (2000) 65–79
25. Gilmore, R.: *Catastrophe Theory for Scientists and Engineers*. Dover Publications, Inc., New York (1993) Originally published by John Wiley & Sons, New York, 1981.
26. Abramowitz, M., Stegun, I.A., eds.: *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover Publications, Inc., New York (1965) Originally published by the National Bureau of Standards in 1964.
27. Wolfram, S.: *Mathematica: A System for doing Mathematics by Computer*. Second edn. Addison-Wesley (1991)
28. Kanters, F.: *Scalespaceviz*. <http://www.bmi2.bmt.tue.nl/image-analysis/people/FKanters/Software/ScaleSpaceViz.html> (2004)
29. Lindeberg, T.: Feature detection with automatic scale selection. *International Journal of Computer Vision* **30** (1998) 79–116
30. Koenderink, J.J.: A hitherto unnoticed singularity of scale-space. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11** (1989) 1222–1224

The Effect of Presmoothing Image Sequences on the Computation of Optical Flow

J.V. Condell, B.W. Scotney, and P.J. Morrow

University of Ulster, Northland Road, Londonderry, Northern Ireland

Abstract. The computation of optical flow has been proposed as a pre-processing step for many high-level vision algorithms. One of the main approaches to the optical flow problem is the gradient-based approach which differentiates the image intensity to compute the optical flow. Often motion vectors computed using various approaches are not reliable. Spatial smoothing of the image sequence is advisable in order to improve velocity estimates in the presence of noise. That is, the application of some kind of linear operation to the images in the sequence before solving for the flow. The pre-processing typically takes the form of some type of spatial Gaussian smoothing or scale specific band-pass filtering of the input images. The main objectives of such filtering have been to lessen the effects of noise, to isolate image structure of interest and to attenuate temporal aliasing and quantization effects in the input images.

This paper investigates the effect of presmoothing on this computation of optical flow. The well known method of Horn and Schunck as well as our own finite element method are implemented and tested for improvements due to presmoothing. Discussions are provided on the effects of presmoothing for a variety of image sequences. In our experiments, smoothing is carried out on a selection of well known sequences in both time and space. Improvements are shown using presmoothing.

Keywords: Finite Volume Methods, Inverse Finite Element Methods, Motion Tracking, Optical Flow, Spatial Smoothing, Temporal Smoothing.

1 Introduction

The estimation of grey-level derivatives is of great importance to the analysis of image sequences for the computation of velocity. Many methods are available in the literature for motion tracking, each with their own characteristics and limitations [4], [9], [12]. One of the main approaches to the optical flow problem is the gradient-based approach which differentiates the image intensity to compute the optical flow. Horn and Schunck [9] presented a gradient-based approach in the form of an iterative algorithm to find the optical flow pattern in which the derivatives of image intensity are approximated using a finite volume approach. It is one of the most powerful algorithms for optical flow computation, making the simple assumption that image brightness itself does not change during motion. By assuming that image intensity is conserved, a gradient constraint equation

may be derived; approximation of the image intensity derivatives in this equation, together with smoothness constraints, forms the basis of algorithms for the computation of the optical flow field describing the transformation of one grey-level image in a sequence to the next.

The research in this paper is focused on the evaluation of the effect of presmoothing on optical flow estimation in image sequences. Here another gradient-based approach is used where finite element methods (FETBI) approximate the image intensity derivatives rather than finite volume methods. The method of Horn and Schunck [9] which uses finite volume methods (HS) is also used to evaluate presmoothing. The finite element approach has many advantages which include a rigorous mathematical formulation, speed of reconstruction, conceptual simplicity and ease of implementation via well-established finite element procedures in comparison to finite volume or finite difference techniques. A finite element approach provides a facility for adaptive partitioning of images [6].

Motion vectors computed using various approaches are often not reliable. Spatial smoothing of the image sequence is advisable in order to improve velocity estimates in the presence of noise. That is, the application of some kind of linear operation to the images in the sequence before solving for the flow. The objectives of filtering may be to reduce the effects of noise, to isolate image structure of interest and to attenuate temporal aliasing and quantization effects in the input images. In our experiments, spatial smoothing was carried out on a selection of five sequences in both time and space. The one-dimensional Gaussian function was used to generate the masks for the smoothing in time. Images were smoothed in time and then space and vice-versa to assess which approach performed better.

The finite element algorithm is described in Section 2. Section 3 discusses the presmoothing of images with results shown in Section 4. Section 5 evaluates and concludes the paper.

2 Finite Element Formulation

Implementations based on finite difference or finite volume methods become complicated to generalise when the image intensity values are not uniformly sampled because they are based on point differences along regular co-ordinate directions. In contrast, finite element methods are ideally suited for use with variable and adaptive grids, and hence provide a framework for developing algorithms to work with non-uniformly sampled images. It has previously been shown that the algorithm of Horn and Schunck (HS) for optical flow estimation may be considered as one of a family of methods that may be derived using an inverse finite element approach (FETBI) [5]. The two-dimensional Galerkin bilinear finite element technique (FETBI) has been developed [7]. This paper looks at the issues of presmoothing images before computing optical flow. Comparisons will be made with other methods available in the literature.

2.1 Two-Dimensional Formulation

We are concerned with the *inverse* problem in which the image intensity values are known, and it is the *velocity function* b , or *optical flow*, that is unknown

and is to be approximated. The image intensity at the point (x, y) in the image plane at time t is denoted by $u(x, y, t)$ which is considered to be a member of the Hilbert image space H^1 at any time $t > 0$. The optical flow is denoted by $\underline{b} \equiv (b_1(x, y, t), b_2(x, y, t))$ where b_1 and b_2 denote the x and y components of the flow \underline{b} at time $t > 0$. In the *inverse* problem with which we are concerned, the image intensity values are known and it is the *velocity function* b , or optical flow, that is unknown and is to be approximated. The optical flow constraint equation is

$$u_x b_1 + u_y b_2 + u_t = 0, \quad (1)$$

where u_x , u_y , u_t are the partial derivatives of the image intensity with respect to x , y and t respectively. Equation 1 cannot fully determine the flow but can give the component of the flow in the direction of the intensity gradient. An additional constraint must be imposed, to ensure a smooth variation in the flow across the image, which is formed by minimising the (square of) the magnitude of the gradient of the optical flow velocity components. It provides a smoothness measure, and may be implemented by setting to zero the Laplacian of b_1 and b_2 . Our approximation is based on the weak form where Ω is the image domain.

The computation of optical flow may be treated as a minimisation problem for the sum of the errors in the equation for the rate of change of image intensity and the measure of the departure from smoothness in the velocity field. This results in a pair of equations, which along with an approximation of laplacians of the velocity components, allow the optical flow to be computed. A smoothing parameter α^2 is incorporated into the motion equations and an iteration procedure with a tolerance value (*toler*) is used to ensure convergence to a solution on termination of the iteration procedure [7].

2.2 Finite Element Bilinear Method (FETBI)

For the uniform bilinear technique, consider a finite element discretisation of the image domain Ω_i based on a rectangular array of pixels. Nodes are placed at the pixel centres, and lines joining these form the edges of elements in the domain discretisation. The infinite dimensional function space H^1 cannot be used directly to provide a tractable computational technique, however a function in the image space H^1 may be approximately represented by a function from a finite dimensional subspace $S^h \in H^1$. From S^h a finite basis $\{\phi_j\}_{j=0}^N$ is selected, where the support of ϕ_i is restricted to a neighbourhood Ω_i (the domain in which its value is non-zero).

The approximation occurs within the finite dimensional subspace $S^h \in H^1$. The image u may thus be approximately represented at time t_n by a function $U^n \in S^h$, where $U^n = \sum_{j=0}^N U_j^n \phi_j$ and in which the parameters $\{U_j^n\}_{j=0}^N$ are mapped from the sampled image intensity values at time $t = t_n$. A basis for S^h may be formed by associating with each node i , a trial function $\phi_i(x, y)$ such that $\phi_i(x, y) = 1$ at node i , $\phi_i(x, y) = 0$ at node j , $j \neq i$, and $\phi_i(x, y)$ is bilinear and piecewise polynomial on each element. The basis function $\phi_i(x, y)$ is thus a *tent shaped* function with support limited to those rectangular elements that have node i as a vertex.

If the components of the optical flow are considered to be piecewise constant over each element, then a Galerkin finite element formulation will provide an approximation to the flow constraint Equation 1 in which each equation contains unknown velocity component values b_1 and b_2 . Of course, the image intensity values do not vary smoothly with time, but discretely from one image in the sequence to the next. The variables U_j^n and U_j^{n+1} are image intensity values in the n^{th} and $(n + 1)^{th}$ images respectively. A one-dimensional Euler Lagrange forward finite difference approximates the partial derivative $\frac{\partial U_j^n}{\partial t}$. The finite element Galerkin formulation with θ time-stepping leads to the approximation

$$\theta \sum_{j=0}^N U_j^n \int_{\Omega} \mathbf{b} \cdot \nabla \phi_j \phi_i \, d\Omega + (1 - \theta) \sum_{j=0}^N U_j^{n+1} \int_{\Omega} \mathbf{b} \cdot \nabla \phi_j \phi_i \, d\Omega + \sum_{j=0}^N \left[\frac{U_j^{n+1} - U_j^n}{\Delta t} \right] \int_{\Omega} \phi_j \phi_i \, d\Omega = 0, \tag{2}$$

where the image values are weighted by θ at the old time step and by $(1 - \theta)$ at the new time step. The forward difference Crank-Nicholson scheme is recovered when $\theta = \frac{1}{2}$. Approximations are necessary for the integrals involved in Equation 2 which may be computed via finite element integral approximations [5]. These Equation 2, along with the smoothing constraint, may be rearranged to provide an iteration scheme similar to the Horn and Schunck equations to approximate the optical flow for each element [5]. A smoothing parameter α is incorporated into the motion equations and an iteration procedure with a tolerance value (*toler*) is used to ensure convergence to a solution.

3 Presmoothing Images

Often motion vectors computed using various approaches are not reliable. Spatial smoothing of the image sequence is advisable in order to improve velocity estimates in the presence of noise. That is, the application of some kind of linear operation to the images in the sequence before solving for the flow. The pre-processing typically takes the form of some type of spatial Gaussian smoothing or scale specific band-pass filtering of the input images. The main objectives of such filtering have been to lessen the effects of noise, to isolate image structure of interest and to attenuate temporal aliasing and quantization effects in the input images. When temporal aliasing cannot be avoided, hierarchical coarse-to-fine methods provide better results [6].

Ong and Spann [12] carried out Gaussian smoothing in time if the sequence had more than fifteen frames. If the sequence had less than fifteen frames they used only two of these frames with Gaussian spatial smoothing. For larger motions Black and Anandan [2] used Gaussian pyramid levels of spatially filtered images. Ju et al. [10] used a four level Gauss method for smoothing. Schunck [13] claimed that averaging frames in a sequence in time could cause problems with motion inconsistencies. Simoncelli et al. [14] presmoothed with Gaussian

spatiotemporal smoothing as did many other authors [1]. Generally authors presmoothed in space and time with $\sigma = 1.5 \text{ pix}/\text{fra}$ for Gaussian smoothing. Barron et al. [1] used a 5×5 neighbourhood for spatial presmoothing with 15 frames in time and a standard deviation of 1.5 pixels in space and 1.5 frames in time sampled out to 3 standard deviations. Similarly, Zhang et al. [16] carried out Gaussian presmoothing for the *Yosemite Valley* sequence in space and time with $\sigma_{xy} = \sigma_t = 1.5$ over 15 frames. In our experiments, spatial smoothing was carried out on a selection of five sequences in both time and space: the *Rotating Rubic*, the *Hamburg Taxi*, the *Yosemite Valley* and the *Translating Square 1* sequence. A parameter set for convergence tolerance and α which gave satisfactory results for these sequences without spatio-temporal smoothing was chosen.

3.1 Spatial Smoothing

For smoothing in space the smoothed function \hat{U}_i is obtained as

$$\hat{U}_i = \sum_j C_{ji} U_j, \quad (3)$$

$$C_{ji} = \int_{\Omega} \phi_j \psi_i d\Omega, \quad (4)$$

with ϕ_j the basis function at node j with corresponding image value U_j , and Gaussian smoothing function

$$\psi_i = \frac{1}{2\pi\sigma^2} e^{-\left[\frac{x^2+y^2}{2\sigma^2}\right]}. \quad (5)$$

We have chosen $\sigma = \frac{1}{1.96}$ so that 95% of the cross-section of the Gaussian lies within one element-width of the node on which it is centred. Values of C_{ji} computed using the general bilinear ϕ_j function are calculated per local node number of this element. The 3×3 masks of C_{ji} coefficients are normalised so that their sum equals 1.

3.2 Temporal Smoothing

The one-dimensional Gaussian function

$$\psi(t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left[\frac{t^2}{2\sigma^2}\right]} \quad (6)$$

was used to generate the masks for the smoothing in time.

3.3 Smoothing Implementation

Images were smoothed in time and then space and vice-versa to assess which approach performed better. There were 21 images available for the *Square1*, *Rotating Rubic* and *Hamburg Taxi* sequences. Only 15 image frames were available for the *Yosemite Valley* sequence. All of the images were initially smoothed

spatially. Then the first half of the images were time smoothed to form a new *image 1* with the second half then time smoothed to form a new *image 2*. This completed the space and time smoothing operations.

After some analysis, it was thought that some of the sequences used in this analysis were suffering from time smoothing over too many images. The *Hamburg Taxi* sequence seemed to lose the car and van flow during this analysis which was not satisfactory and the *Yosemite Valley* did not improve but rather became less satisfactory. So a rule was developed to calculate the number of images to be included in time smoothing. Initial exploratory *Rotating Rubic* sequence analysis showed that a formula could be developed which was a function of the maximum velocity possible throughout the image based on prior knowledge of the image sequences [7]. The general rule was that the faster the flow, the fewer the frames that should be used in time averaging. A formula for the number of frames (N) to be included in the smoothing calculations was developed as $N = \lceil \frac{k}{maxvel} \rceil$ where *maxvel* denotes the maximum velocity known to occur in the image sequence in pixels per frame and k is a parameter which was calculated on analysis of initial *Rotating Rubic* sequence results where 15 image frames smoothed in time from the sequence produced a satisfactory result ($N = 15$ therefore $k = 21$ as the maximum velocity occurring during the *Rotating Rubic* sequence was 1.4 pixels/frame). The value for N must always be an odd number as we require 95% of $\frac{N-1}{2}$ of the image sequence frames of the cross-section of the Gaussian to lie within one element-width of the node on which it is centred. Thus for a 95% confidence interval, the variable σ is calculated thus:

$$\sigma = \frac{1}{1.96} \left[\frac{N-1}{2} \right]. \quad (7)$$

It was calculated (from N) that all sequences used in this part of the analysis should have their number of images used in time smoothing adjusted. The *Yosemite Valley* sequence was reduced to 3 from previously 15 images being used. Similarly the *Rotating Rubic* sequence was reduced to 15 images from 21 previously, the *Square1* sequence was reduced to 19 from 21 and the *Hamburg Taxi* was reduced to 7 images from 21 images. This reduction in the number of images used in the time smoothing process improved the *Hamburg Taxi* and *Yosemite* sequence results significantly.

3.4 Errors for Performance Analysis

Synthetic image sequences are used for which the correct 2D motion fields are known, with the *angular* and *absolute* errors measures used for quantitative comparisons. *Real* image sequences are also used for analysis where the displaced frame difference error is calculated and discussed [1], [7], [12]. Velocity is written as displacement per unit time; pixels/frame (*pix/fr*).

4 Experimental Results

For these sequence experiments the following representation is made: *OR* denotes no time or space smoothing used (i.e. OOriginal results); *SP* denotes spatially

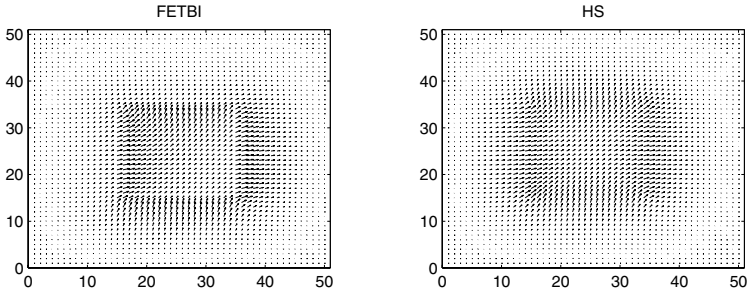


Fig. 1. *Translating Square1* FETBI and HS *Spatially Smoothed* Flow for Convergence Tolerance 0.001 and Smoothing $\alpha = 75$

Table 1. Performance of FETBI, HS and Other Algorithms with Angular Errors for *Smoothed* and *Unsmoothed: Translating Square 1, Yosemite Valley* and *Translating Tree* Image Sequences

| Algorithm (Toler/ α) | Angular Error | | Absolute Error | | Density |
|-----------------------------------|---------------|--------|----------------|-------|---------|
| | Mean | SD | Mean | SD | % |
| <i>Translating Square 1</i> | | | | | |
| FETBI (0.001/75) OR | 16.261 | 11.579 | 1.047 | 0.464 | 100.00 |
| HS (0.001/75) OR | 17.494 | 10.885 | 1.076 | 0.370 | 100.00 |
| FETBI (0.001/75) SP | 14.610 | 10.627 | 0.980 | 0.387 | 100.00 |
| HS (0.001/75) SP | 17.083 | 10.735 | 1.065 | 0.375 | 100.00 |
| <i>Yosemite Valley</i> | | | | | |
| FETBI (0.001/75) OR | 21.185 | 25.503 | 1.021 | 0.922 | 100.00 |
| HS (0.001/75) OR | 19.319 | 25.700 | 1.119 | 1.029 | 100.00 |
| FETBI (0.001/75) TISP | 18.800 | 22.915 | 0.978 | 0.936 | 100.00 |
| HS (0.001/75) TISP | 17.828 | 22.900 | 1.032 | 1.006 | 100.00 |
| Aisbett [15] | 22.78 | 20.04 | | | 100.00 |
| <i>Original Barron et al. [1]</i> | 31.69 | 31.18 | | | 100.00 |
| Black and Anandan [15] | 31.03 | 22.82 | | | 100.00 |
| Nagel [15] | 26.23 | 21.73 | | | 100.00 |
| Heeger [12] | 22.82 | 35.28 | | | 64.20 |

smoothed; *TI* denotes temporally smoothed; *SPTI* denotes spatial smoothing followed by temporal smoothing; *TISP* denotes temporal smoothing followed by spatial smoothing.

Error values results for the *Translating Square1* sequence are shown for comparison for the non-smoothed (OR) and spatially smoothed (SP) versions

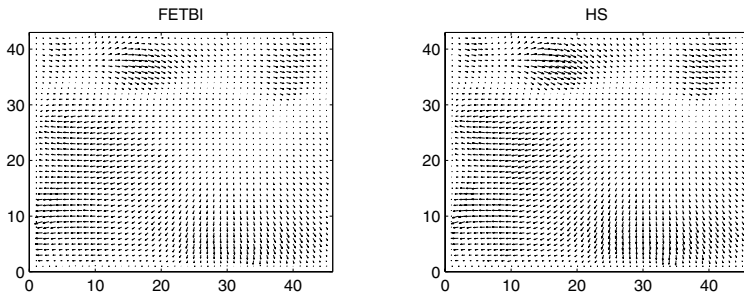


Fig. 2. *Yosemite Valley* FETBI and HS *Temporal and Spatially Smoothed* Flow for Convergence Tolerance 0.001 and Smoothing $\alpha = 75$

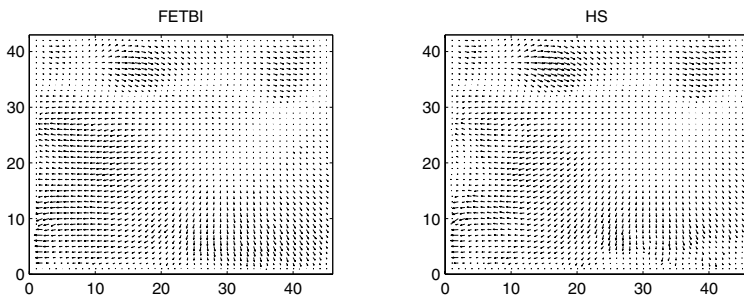


Fig. 3. *Yosemite Valley* FETBI and HS *Original* Flow for Convergence Tolerance 0.001 and Smoothing $\alpha = 75$

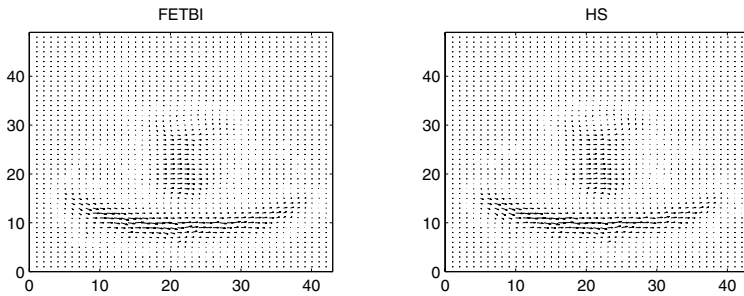


Fig. 4. *Rotating Rubic* FETBI and HS *Spatially Smoothed* Flow for Convergence Tolerance 0.001 and Smoothing $\alpha = 400$

for tolerance 0.001 and $\alpha = 75$ (see Figure 1) in Table 1. These values show a 11% improvement for the spatially-smoothed flow over the original flow for the FETBI algorithm.

Some results for *Yosemite Valley* are included which involve the computation of optical flow from bilinear-Gaussian temporal and spatially smoothed images of this sequence. Figure 2 shows the results of computing flow from these temporal

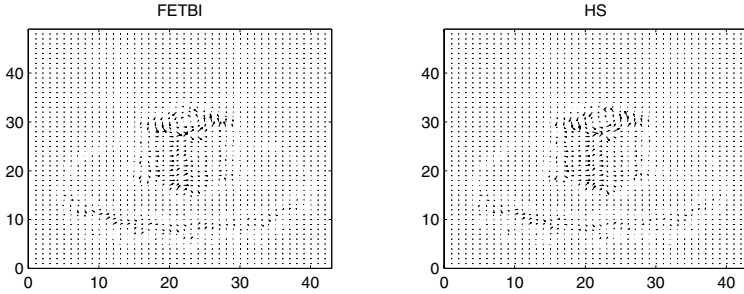


Fig. 5. *Rotating Rubic* FETBI and HS *Temporally Smoothed Flow* for Convergence Tolerance 0.001 and Smoothing $\alpha = 400$

Table 2. Displaced Frame Difference Errors for *Smoothed* and *Unsmoothed: Rotating Rubic* and *Hamburg Taxi* Image Sequences with FETBI, HS and Other Algorithms [12]

| Algorithm (Toler/ α) | Forward DFD | | Backward DFD | | Density |
|---------------------------------|-------------|--------|--------------|--------|---------|
| | Mean | SD | Mean | SD | % |
| <i>Rotating Rubic</i> | | | | | |
| FETBI (0.001/400) OR | 2.634 | 8.588 | 2.634 | 8.588 | 100.00 |
| HS (0.001/400) OR | 2.584 | 8.476 | 2.584 | 8.476 | 100.00 |
| FETBI (0.001/400) SP | 2.634 | 8.585 | 2.634 | 8.585 | 100.00 |
| HS (0.001/400) SP | 2.732 | 8.795 | 2.732 | 8.795 | 100.00 |
| FETBI (0.001/400) TI | 4.792 | 13.606 | 4.792 | 13.606 | 100.00 |
| HS (0.001/400) TI | 4.691 | 13.319 | 4.691 | 13.319 | 100.00 |
| Anandan | 5.31 | 4.99 | | | 100.00 |
| Nagel | 3.28 | 3.04 | | | 100.00 |
| Fleet and Jepson | 4.42 | 3.08 | | | 12.63 |
| <i>Hamburg Taxi</i> | | | | | |
| FETBI (0.001/75) OR | 3.631 | 11.249 | 3.737 | 11.880 | 100.00 |
| HS (0.001/75) OR | 3.499 | 10.773 | 3.482 | 10.610 | 100.00 |
| FETBI (0.001/75) TISP | 3.308 | 10.136 | 3.329 | 10.286 | 100.00 |
| HS (0.001/75) TISP | 3.222 | 9.472 | 3.223 | 9.598 | 100.00 |
| Anandan | 4.35 | 3.60 | | | 100.00 |
| Lucas and Kanade | 3.99 | 3.23 | | | 100.00 |
| Nagel | 4.21 | 3.48 | | | 100.00 |
| Fleet and Jepson | 6.20 | 4.84 | | | 27.84 |
| Odobez and Boutheymy | 3.93 | 3.17 | | | 98.87 |
| Ong and Spann | 3.84 | 3.08 | | | 95.46 |

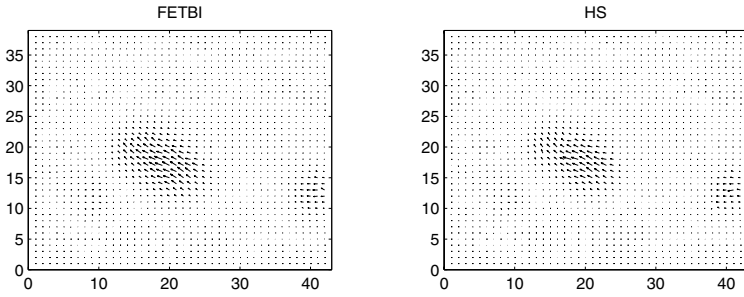


Fig. 6. *Hamburg Taxi* FETBI and HS *Original* Flow for Convergence Tolerance 0.001 and Smoothing $\alpha = 75$

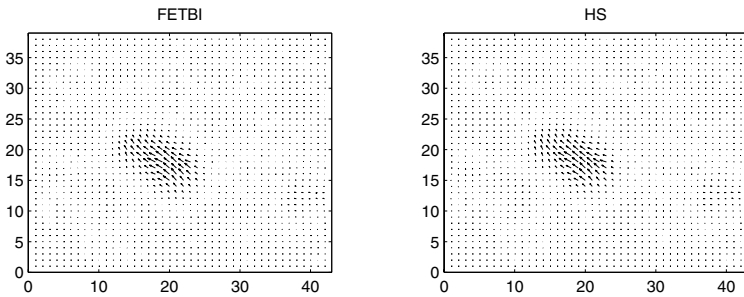


Fig. 7. *Hamburg Taxi* FETBI and HS *Temporal and Spatially Smoothed* Flow for Convergence Tolerance 0.001 and Smoothing $\alpha = 75$

and spatially smoothed images (TISP) with parameters of tolerance 0.001 and $\alpha = 75$ with error values included in Table 1. The original result (OR) for this combination of parameters is shown in Figure 3. For this sequence, 13% improvements are recorded for FETBI and HS algorithms using the temporal-spatial smoothed images instead of the original images. Some other authors in the literature have carried out spatial and temporal smoothing on this *Yosemite Valley* sequence and reported that this smoothed out errors and lowered angular and absolute error values: [3], [11]. This is consistent with the results presented here. Heeger [8] claimed smoothing helped overcome problems on the horizon boundary. A method by Zhang et al. [16] included presmoothing to overcome the aperture problem in the mountains and thereby achieved better results.

Results are shown for the spatially smoothed (SP) and temporally smoothed (TI) versions of the *Rotating Rubic* sequence for tolerance 0.001 and $\alpha = 400$ in Figure 4 and Figure 5 respectively. Table 2 shows the corresponding displaced frame difference errors, where it can be seen that the spatially smoothed version gives very similar FETBI results to the original results without any smoothing (OR - see Table 2). The temporally smoothed results shown in Figure 5 are quite interesting here. They show a generally less smooth flow result with more

definition in individual flow vectors. However with these temporally smoothed images the error increases.

Some results are included for the *Hamburg Taxi* sequence which involve the computation of optical flow from temporal and bilinear-Gaussian spatially smoothed images of this sequence. Figure 6 and Figure 7 respectively show the results of computing original (OR) and temporal-spatially smoothed (TISP) flow with parameters of tolerance 0.001 and $\alpha = 75$, with error values included in Table 2. Improvements are recorded for both FETBI and HS algorithms, with 10% and 9% improvements respectively.

5 Conclusion

This paper has provided a mathematical and experimental background for presmoothing images prior to optical flow computation. In the results presented, when comparing the two dimensional Horn and Schunck method with the FETBI technique, the FETBI technique performs well. The smoothing investigation showed improvements for the FETBI algorithm for the *Translating Square1*, *Yosemite Valley* and *Hamburg Taxi* sequences. Generally FETBI performed as well as HS on most image sequences and in some cases showed improvements over HS results. In particular, FETBI showed an 11% improvement when the *Translating Square 1* image sequence was spatially smoothed. The *Yosemite Valley* sequence showed a 13% improvement with temporal-spatially smoothed images for both the FETBI and HS algorithms. On the *Hamburg Taxi* image sequence temporal-spatially smoothed images provided a 10% and 9% improvement in errors for the FETBI and HS algorithms respectively. Generally 3 out of the 4 image sequences shown in this paper show at least a 10% improvement in error results with the FETBI algorithm. One of the image sequences did not show an improvement but yet the errors did not always increase. These results suggest the optimum amount of smoothing may depend on the particular image sequence but generally where presmoothing is used (temporal or spatial) the results do not deteriorate.

References

1. Barron, J. L., D. J. Fleet, and S. S. Beauchemin. Performance of Optical Flow Techniques . *IJCV*, **12** 1:43–77, 1994.
2. Black, M.J. and Anandan, P. The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields . *Comp. Vision and Image Understanding* , **63** 1:75–104, 1996.
3. Chin, T. M., W. C. Karl, and A. S. Willsky. Probabilistic and Sequential Computation of Optical Flow using Temporal Coherence . *IEEE Transactions on Image Process.* , **3** 6:773-788, 1994.
4. Cohen, I. and I. Herlin. Non Uniform Multiresolution Method for Optical Flow and Phase Portrait Models: Environmental Applications . *IJCV*, **33** 1:1–22, 1999.
5. Condell, J. V. Motion Tracking in Digital Images . PhD Thesis, Faculty of Informatics, University of Ulster , 2002.

6. Condell, J. V., B. W. Scotney, and P. J. Morrow. Adaptive Grid Refinement Procedures for Efficient Optical Flow Computation . *IJCV*,**61** 1:31-54, 2005.
7. Graham, J.V., B. W. Scotney, and P. J. Morrow. Evaluation of Inverse Finite Element Techniques for Gradient Based Motion Estimation. Proc. Third IMA Confer. on Imaging and Digital Image Process., 200–220, 2000.
8. Heeger, D.J. Model for the Extraction of Image Flow. *Jnl. of Optical Society of America A: Optics, Image Science and Vision*, **4** 8:1455-1471, 1987.
9. Horn, B. K. P, and B. G. Schunck. Determining Optical Flow . *AI*, **17** 185–203, 1981.
10. Ju, S. X., Black, M. J., and Jepson, A. D. Skin and Bones: Multilayer, Locally Affine, Optical Flow and Regularisation with Transparency. *IEEE Proc. Intern. Confer. on Comp. Vision and Pattern Recognition (CVPR 1996)*, 307-314, 1996.
11. Nomura, A. Spatio-Temporal Optimization Method for Determining Motion Vector Fields Under Non-Stationary Illumination. *Image and Vision Computing*, Elsevier Science, **18** 939-950, 2000.
12. Ong, E. P and M. Spann. Robust Optical Flow Computation based on Least-Median-of-Squares Regression . *IJCV*,**31** 1:51–82, 1999.
13. Schunck, B. G. The Motion Constraint Equation for Optical Flow. *IEEE Proc. Seventh Intern. Confer. on Pattern Recognition (ICPR 1984)*, **1,2**, 20–22, 1984.
14. Simoncelli, E. P., Adelson, E. H. and Heeger, D. J. Probability Distributions of Optical Flow. *IEEE Proc. Intern. Confer. on Comp. Vision and Pattern Recognition (CVPR 1991)*, 310-315, 1991.
15. Sim, D.G., Park, R.H. A Two-Stage Algorithm for Motion Discontinuity-Preserving Optical Flow Estimation. *IEEE Proc. Intern. Confer. on Comp. Vision and Pattern Recognition (CVPR 1997)*, **1** 19–37, 1997.
16. Zhang, L., Sakurai, T. and Miike, H. Detection of Motion Fields under Spatio Temporal Non-Uniform Illumination. *Image and Vision Computing*, Elsevier Science, **17** 309-320, 1999.

Optical Flow Based Frame Interpolation of Ultrasound Images

Tae-Jin Nam¹, Rae-Hong Park^{1,2}, and Jae-Ho Yun¹

¹ Department of Electronic Engineering, Sogang University,
C.P.O. Box 1142, Seoul 100-611, Korea
{tjnam, rhpark, yastoil}@sogang.ac.kr
<http://eevision1.sogang.ac.kr>

² Interdisciplinary Program of Integrated Biotechnology, Sogang University

Abstract. In this paper, we present an optical flow based frame rate up-conversion method for ultrasound images. The conventional mechanical scan method for multi-planar images has a slow frame rate, thus frame interpolation is desirable for smooth display. In the proposed frame rate up-conversion method, several new interpolated frames are inserted between two input frames, giving smooth renditions to human eyes. We employ a window-based optical flow based method to find accurate motion estimates for frame interpolation. Consequently, the proposed method can provide detailed and improved interpolated images without block artifact. Experimental results with three sets of ultrasound image sequences show the effectiveness of the proposed interpolation method.

Keywords: Optical flow, image enhancement, frame rate up-conversion, real-time 3-D, ultrasound image.

1 Introduction

Medical visualization resources encompass a broad array of different modalities, including X-ray imaging, computed tomography (CT), magnetic resonance imaging (MRI), and ultrasound. The inherent flexibility of ultrasound imaging and its moderate cost without known bio-effects give it a vital role in the diagnostic process compared with CT and MRI. Ultrasound visualization has been in routine use in almost hospitals and clinics to diagnose widely different pathology [1].

General ultrasound diagnosis equipments take in one-dimensional (1-D) signals, from which two-dimensional (2-D) ultrasound images are generated as a collection of 1-D signal information. Three-dimensional (3-D) ultrasound images are also generated as a collection of 2-D signal information. 3-D image conversion methods use a ray-casting rendering method that considers trace-light effects for image modeling, while requiring a high computational load [1][2]. 3-D ultrasound diagnosis equipments are classified into two types: those integrated in ultrasound equipments and those using external workstations. The former methods process signals using physical beam data directly in ultrasound diagnosis with fast and accurate 3-D data acquisition. The latter methods, integrated 3-D approaches, are classified into two methods: mechanical scan and hand-free scan. Mechanical scan using 1-D array

converters provides accurate and high-resolution 3-D data in real-time, thus this approach is a more promising method than the hand-free scan one that is difficult to implement in real-time.

The present 3-D scan method using mechanical 1-D arrays shows a drawback that it has a relatively slow multi-planar image update rate at about 1.5–2 volumes/sec over general regions of images. The image obtained by the 3-D scan method does not show an issue related to stationary objects. Nowadays, the image update rate has been increased much, however this speed is not still enough for smooth rendition to human eyes, in which abrupt changes are perceived due to fast motions. To increase the frame rate further, we use a frame rate up-conversion method [3] by which a number of interpolated frames are inserted between two input frames, providing more natural motions in real-time. To improve visual quality, accurate motion detection is required.

The rest of the paper is structured as follows. Section 2 presents motion estimation methods: the block matching algorithm (BMA) and the optical flow method. Section 3 describes two optical flow based motion estimation methods: anisotropic method and Lucas and Kanade's method. Section 4 proposes the temporal interpolation method using the employed optical flow method. Section 5 gives experimental results and discussions. Section 6 concludes the paper.

2 Motion Estimation Methods

We need the temporal relationship between two images (previous and current frames) for temporal frame interpolation. This information is obtained from the difference image or the intensity change caused by object motions. Accurate motion estimation is required for effective description of motion trajectory of objects in a frame. Generally, motion estimation methods can be classified into three categories: area-based, feature-based, and gradient-based. The area-based approach has been widely used for image compression. However, it cannot estimate the real motion field. The feature-based motion estimation methods were proposed to find the correct motion vectors using feature points. Since only a small number of feature points are detected, interpolation of the dense motion field is required. Gradient-based approach uses intensity changes between two input images because motion produces intensity changes.

In this paper, gradient-based approaches are appropriate for image sequence processing because dense optical flow can be obtained, under the assumption of conservation of intensity. They assume that at a point the intensity at the corresponding image point remains constant over time. If point at time t corresponds to point at time $t + \delta t$, we can obtain

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t). \quad (1)$$

Using Taylor series expansion:

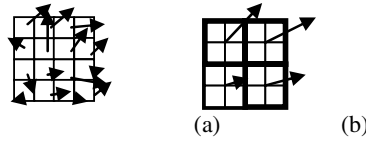


Fig. 1. Detected MVs by optical flow methods. (a) pixel-based. (b) window-based.

$$I(x + \delta x, y + \delta y, t + \delta t) = I(x, y, t) + \delta x \frac{\partial I}{\partial x} + \delta y \frac{\partial I}{\partial y} + \delta t \frac{\partial I}{\partial t} \tag{2}$$

we get

$$I_x u + I_y v + I_t = 0 \tag{3}$$

where $u = \frac{\delta x}{\delta t}$, $v = \frac{\delta y}{\delta t}$, $I_x = \frac{\partial I}{\partial x}$, $I_y = \frac{\partial I}{\partial y}$, and $I_t = \frac{\partial I}{\partial t}$.

The motion constraint line of the optical flow equation is plotted in the space, from which the normal velocity can be directly obtained. The normal velocity is the same as the perpendicular distance of the line from the origin. The problem arises on the gradient-based formulation as an underconstrained system composed of a single equation with two components, which is called an aperture problem. This formulation is acceptable in the regions that do not suffer from the aperture problem, and the slope of the motion constraint line is related to the direction of intensity gradients [4][18].

3 Optical Flow Based Motion Estimation

Various optical flow methods are classified into two types: global and local. Horn and Schunck’s method [5] and Nagel’s method [6] are global ones whereas Lucas and Kanade’s method [7] is a local one. In our previous work [8], local change detection is employed in frame interpolation using Lucas and Kanade’s method. Also we use MV information using an anisotropic diffusion method to obtain smooth MVs. It shows good interpolation results, however, has a drawback that it consists of two similar steps for consistent MV generation: anisotropic diffusion filtering and median filtering.

We simulate two optical flow frame interpolation methods: 1) we utilize MV information of neighboring windows in the anisotropic diffusion method, 2) we use Lucas and Kanade’s method using median filtering which is faster than the former method. However the latter requires hole and overlap processing.

3.1 Comparison of Pixel-Based and Window-Based Optical Flow Methods

Optical flow methods find intensity changes pixel by pixel. They detect the correct motion field with noiseless images. 3-D ultrasound images are generated by rendering a number of 2-D ultrasound images using a ray-casting method, showing blurring.

They are somewhat noisy due to the characteristic of ultrasound. The pixel-based optical flow method [5] does not give consistent MVs as shown in Fig. 1(a). It utilizes correlation of neighboring pixels with a high computational load.

To obtain consistent motion vectors as shown in Fig. 1(b) with a reduced computational load, we propose a window-based optical flow method [7], in which the computation time of the optical flow method with 2×2 windows is reduced by a factor of three compared with the pixelwise optical flow method. Also the 2×2 window-based method shows good results as the pixel based method because 2×2 windows are small.

3.2 Anisotropic Diffusion Method

The optical flow based motion estimation method using regularization shows a difficulty in obtaining a physically reasonable solution in the sense that it cannot cope well with discontinuities. Conventional methods have not completely solved this problem. Anisotropic diffusion for the adaptive Gaussian smoothing in image reconstruction was proposed [9]. Gaussian filtering with varying scale can be transformed into a form of a diffusion equation, where diffusion coefficients are computed properly by adaptive estimation and edges can be successfully localized.

The anisotropic diffusion equation is defined by

$$E_t = \text{div}(s(x, y, t)\nabla E) \quad (4)$$

where E_t represents a temporal derivative of an illumination image E , $s(x, y, t)$ signifies a diffusion coefficient which need not be a constant, and div and ∇ denote divergence and gradient operators, respectively. This equation can be rewritten as [10]

$$E_t = s(x, y, t)\nabla E + \nabla s(x, y, t)\nabla E. \quad (5)$$

We implement a weighted least squares fit of local first-order constraints to a constant model for estimation of \mathbf{v} in each local region Ω by minimizing

$$\sum_{x \in \Omega} w^2(x) [\nabla I(x, y, t) \cdot \mathbf{v} + I_t(x, y, t)]^2 \quad (6)$$

where $w(x)$ denotes a window function that gives more influence at the center of the window. Note that the velocity field \mathbf{v} is calculated in the least squares sense [11].

The anisotropic optical flow method shows better results than the BMA with a higher computational load. Therefore we need to reduce the processing time. We simulate optical flow computation, in which the 2×2 measurement window is used for 4×4 pixels (detection window). This window-based optical flow method reduces the processing time, utilizing the relationship of neighboring pixels. But interpolated frames show block artifact that degrades image quality.

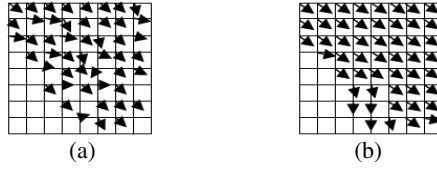


Fig. 2. Median filtering of MVs. (a) before processing. (b) after processing.

3.3 Lucas and Kanade’s Method

Lucas and Kanade’s method uses a small window, if the intensity changes in a window between two consecutive frames are large. To detect intensity changes, the intensity gradient values are computed, in which intensities at the same pixel location in the previous and current frames are used. Lucas and Kanade’s method has an advantage in feature matching, and we use median filtering for correct MV detection.

Astola *et al.* proposed vector median filters, by generalizing the definition with a suitable vector norm [12]. Let $V = \{v_i | 1 \leq i \leq N\}$ be a set of MVs in the window, with N being the number of MVs in the window and with v_{med} included in V . The median of the MVs is defined as

$$\sum_{v_i \in V} \|v_{med} - v_i\| \leq \sum_{v_i \in V} \|v_r - v_i\| \tag{7}$$

where v_k denotes the vector field and $v_r \in V^R$, with V^R representing the vector subspace containing R MVs. The norm defines the distance metric [13][14]. The median filter performs a nonlinear filtering operation, where a window moves over an image and at each point the median value of the data within the window is taken as the output. So median filtering has some desirable properties that cannot be achieved with linear filters [12]. Fig. 2 shows an example of median filtering of MVs.

4 Proposed Frame Interpolation

The motion of an imaged object can be estimated using the differentials of luminance variations in time or space, in which we employ optical flow based motion estimation. In addition, a median filter can be used for noise reduction to decrease the number of incorrect MVs, because the incorrectly detected displacement yields distortion in interpolated frames.

4.1 Frame Interpolation

Motion compensated interpolation involves motion estimation at frames to be interpolated. The motion information as well as classification of the covered and



Fig. 3. Covered and uncovered regions. (a) covered region. (b) uncovered region.

uncovered regions of the decoded frames is readily available at the receiver. K frames interpolated between two given frames I_n and I_{n+1} are called $I_{n:1}, I_{n:2}, \dots, I_{n:K}$, where $K=3$ is used in our experiments. The object label at the integer position is then taken to be the label of the closest trajectory within a radius of half-pixel width. If such a trajectory does not exist, that pixel is considered as a covered or uncovered pixel, depending on the label for that pixel location at frames $I_{n:1}, I_{n:2}$, and $I_{n:3}$. Thus we can find the covered region at frame I_n whereas uncovered region at frame I_{n+1} .

The motion field at frame $I_{n:1}$ is found as follows. For pixels belonging to an object, the MV is taken to be one fourth of the projected MV (note that $K=3$). MVs are assigned to covered or uncovered pixels once the motion field is found. The motion compensated interpolation merely involves averaging of the corresponding pixels, the pixel values at frames I_n and I_{n+1} . For covered pixels, the pixel values from frame $I_{n:1}$ are copied since no motion information is defined. In the case when a new object appears at frame $I_{n:3}$, only the pixels from frame $I_{n:1}$ are used in interpolating the image at the previous frame [3][15].

Frame interpolation is described as follows. Let $I_n(x, y)$ be intensity values at (x, y) and $\mathbf{v} = (v_x, v_y)$ denote the estimated MV. Interpolated frames are obtained as follows, depending on the region classification:

Bi-directionally interpolated the frame:

$$I_{ni}(x, y) = \frac{1}{2} \left\{ I_n \left(x + \frac{K+1-i}{K+1} \cdot v_x, y + \frac{K+1-i}{K+1} \cdot v_y \right) + I_{n+1} \left(x - \frac{i}{K+1} \cdot v_x, y - \frac{i}{K+1} \cdot v_y \right) \right\} \quad (8)$$

Covered region:

$$I_{ni}(x, y) = I_n(x, y) \quad (9)$$

Uncovered region:

$$I_{ni}(x, y) = I_{n+1}(x, y). \quad (10)$$

Holes and overlapped regions are inevitable in blockwise interpolation that uses blockwise information of previous and current frames. Also they are produced in the window-based optical flow method. Because the block (window) grid is not matched well to that of interpolated frames in the reconstruction of interpolated frames, interpolated frames have covered and uncovered regions. The covered region is the region that disappears in the interpolated frame by a moving block (window). Similarly the uncovered region is the region that is created in the interpolated frame

by the moving block. These covered and uncovered regions correspond to holes and overlapped regions, respectively, that are to be eliminated. Fig. 3 shows these regions. These regions are detected by pixelwise flags that are incremented by one when the pixel in the interpolated frame is covered by the block specified by the displacement in motion compensation (MC). The total number of occurrences covered by the motion compensated block in the interpolated frame is stored. Holes (overlapped regions) consist of pixels with the flag value equal to zero (larger than one).

To process hole regions, we use causal edge-based line-average filtering. We consider the current hole pixel based on the relationship of neighboring pixels: edge directionality. In this filtering, types of neighboring pixels of a given hole pixel are defined depending on the directions of the line: row, column, diagonal, and anti-diagonal. It calculates the difference along each line and finds the direction that yields the smallest difference. Then the average value of neighboring pixels along the selected line is used for interpolation of a hole pixel at the center [16]. We also consider pixels in both previous and current frames, and use a sum of absolute difference (SAD) as a matching function. As compared with the BMA, the window based optical flow method gives smaller numbers of holes and overlapped pixels in the interpolated frames that are unpleasant to human eyes.

4.2 Image Enhancement

Block artifact in the interpolated frame due to blockwise processing is severe near motion block boundaries. The BMA followed by lowpass filtering is used to reduce block artifact, however reducing high frequency details in a signal as well. A simple method uses a sharpening mask to solve the problem, however still producing block artifact.

We simulate image enhancement filtering using the correlation of two input frames. We can formulate a function of intensity from histograms of two input frames. We implement a function of intensity on interpolated images using a sharpening mask. Interpolated images by the BMA show good results, whereas those by the optical flow method show overshoots. Note that the optical flow method does not require postprocessing for image enhancement [17].

5 Simulation Results and Discussions

The proposed optical flow based interpolation method for 3-D ultrasound images consists of three steps: motion estimation (anisotropic method or Lucas and Kanade's method with MV smoothing), temporal interpolation, and hole and overlap region processing. To detect MVs, we use a window-based optical flow (motion flow) method. To reduce the sensitivity of Lucas and Kanade's method to noise, vector median filtering performs MV smoothing. Interpolated frames are reconstructed by motion compensation using MVs in the anisotropic method (MV's for motion compensation using smoothed MV's in Lucas and Kanade's method). However hole and overlapped regions are generated in reconstructed (interpolated) frames, so they are to be removed. We simulate the proposed algorithm using the window-based optical flow method on a Pentium IV 3.0 GHz, 1 GByte RAM, and Windows OS systems.

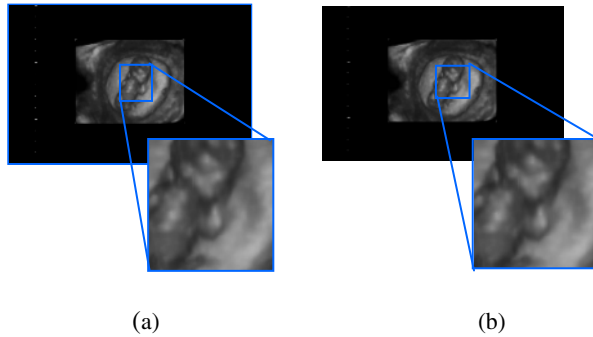


Fig. 4. Interpolated images and zoomed regions by a factor of two (test sequence 1, frame $I_{35;3}$, 4×4). (a) BMA (no deblocking filter). (b) anisotropic optical flow method.

Fig. 4 shows comparison of 636×386 interpolated images (frame $I_{35;3}$ of the test sequence 1), in which block size is 4×4 and 100×100 zoomed images are shown. Fig. 4(a) shows the interpolated image with block artifact, in which no deblocking filter is used, however degrading high frequency details. Fig. 4(b) shows the interpolated image by the proposed filter with less block artifact, providing improved high frequency details but showing overshoots.

Fig. 5 shows comparison of interpolated images using the BMA (frames I_{25} and I_{26}) of the test sequence 2. Fig. 5(a) shows the result without deblocking filtering, giving block artifact whereas Fig. 5(b) shows the result with deblocking filtering, yielding less block artifact but losing frequency details.

Fig. 6(a) shows the frames I_{35} and I_{36} of the test sequence 1, in which fast and small motions are observed. Figs. 6(b) and 6(c) show interpolated images using the BMA (block size: 8×8) and the anisotropic optical flow method (window size: 2×2), respectively, in which zoomed images (290×220) by a factor of two along the horizontal and vertical directions are shown. In the BMA, we change block size: 4×4 and 8×8 . The processing time with 8×8 blocks is smaller than that with 4×4 blocks by 37%, with a little bit lower image quality. In the anisotropic optical flow method,

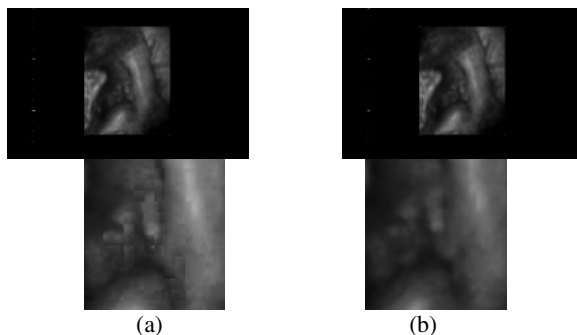


Fig. 5. Comparison of interpolated images (150×150) using the BMA (test sequence 2, frame $I_{25;3}$, 8×8). (a) without deblocking filtering. (b) with deblocking filtering.

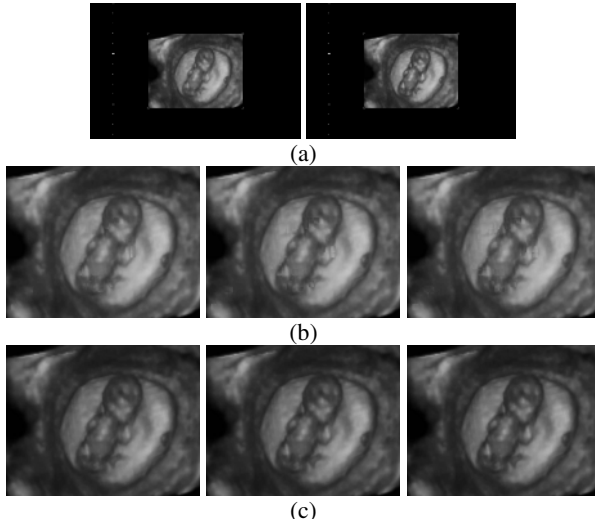


Fig. 6. Comparison of interpolated frames (test sequence 1). (a) input frames I_{35} and I_{36} . (b) zoomed interpolated frames using the BMA (frames $I_{35:1}$, $I_{35:2}$, and $I_{35:3}$, block size: 8×8). (c) zoomed interpolated frames using the anisotropic optical flow method (frames $I_{35:1}$, $I_{35:2}$, and $I_{35:3}$, window size: 2×2).

we change window size: 4×4 and 2×2 . The processing time with 4×4 windows is smaller than that with 2×2 windows, in which 4×4 windows shows block artifact as the BMA in large motion areas. The anisotropic optical flow method with 2×2 windows shows better image quality than that with 4×4 windows.

Fig. 7(a) shows the frames t_{14} and t_{15} of the test sequence 2, in which fast motions are observed. Figs. 7(b) and 7(c) show interpolated images using the BMA (block size: 8×8) and the anisotropic optical flow method (window size: 2×2), respectively. In the BMA, we change block size: 4×4 , 8×8 , and 16×16 . The block artifact with 16×16 blocks is larger than that with 8×8 blocks, with a little bit smaller processing time. Also the processing time with 8×8 blocks is smaller than that with 4×4 blocks, with a little bit lower image quality. In the anisotropic optical flow method, we change window size: 4×4 and 2×2 . The processing time of the window-based method with 4×4 windows is smaller than that with 2×2 windows. The anisotropic optical flow methods show some block artifact. The block artifact with 2×2 windows is smaller than that with 4×4 windows, with a little bit larger processing time.

Fig. 8(a) shows the frames t_{14} and t_{15} of the test sequence 3, in which small motions are observed. Figs. 8(b) and 8(c) show interpolated images using the BMA (block size: 8×8) and the anisotropic optical flow method (window size: 2×2), respectively. In the BMA, we change block size: 4×4 and 8×8 . The processing time with 8×8 blocks is smaller than that with 4×4 blocks. The processing time with 8×8 blocks is smaller than that with 4×4 blocks by 8%, with a little bit lower image quality. Similar artifact in large-motion area. The method with 2×2 windows shows better image quality than that with 4×4 windows.

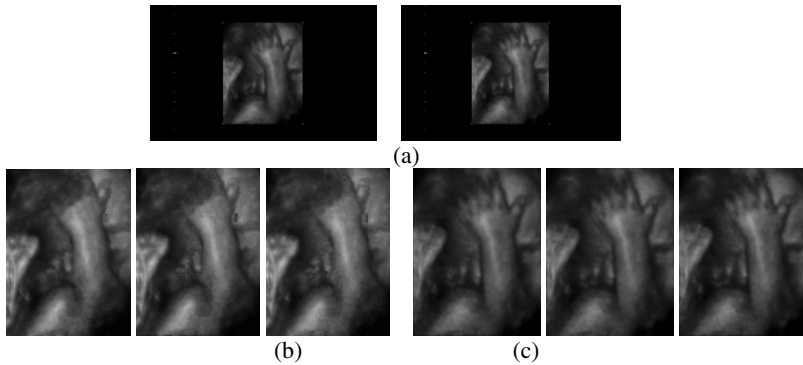


Fig. 7. Comparison of interpolated frames (test sequence 2). (a) input frames I_{14} and I_{15} . (b) zoomed interpolated frames using the BMA (frames $I_{14:1}$, $I_{14:2}$, and $I_{14:3}$, block size: 8×8). (c) zoomed interpolated frames using the anisotropic optical flow method (frames $I_{14:1}$, $I_{14:2}$, and $I_{14:3}$, window size: 2×2).

Processing time of the simulated methods with selected window sizes is compared. Fig. 9(a) shows the comparison of the computation time as a function of the frame index (40 frames) on the test sequence 1 using the BMA (block size: 8×8) and the two optical flow methods (window size: 2×2). The BMA method takes 0.29 sec (block size 8×8) on the average. Lucas and Kanade's optical flow method takes 2.64 sec and the anisotropic method 3.65 sec on the average. The average processing time with the BMA is smaller than that of the optical flow method (anisotropic optical flow method) by a factor of about 12.5. Also the anisotropic optical flow method is faster (0.6 sec) than the previous methods. Fig. 9(b) shows the comparison of the computation time as a function of the frame index (40 frames) on the test sequence 2 using the BMA (block size: 8×8) and two optical flow methods (window size: 2×2). The BMA method takes 0.27 sec (block size: 8×8). Lucas and Kanade's optical flow method takes 2.57 sec and the anisotropic optical flow method takes 3.34 sec on the average. The average processing time with the BMA is smaller than that of the optical flow method by a factor of about 12. Also the anisotropic optical flow method is faster (0.6 sec) than the previous method [8]. Fig. 9(c) shows the comparison of the computation time as a function of the frame index (40 frames) on the test sequence 3 using the BMA (block size: 8×8) and two optical flow methods (window size: 2×2). The BMA method takes 0.26 sec (block size: 8×8). Lucas and Kanade's optical flow method takes 2.8 sec and the anisotropic optical flow method takes 3.75 sec, on the average. The average processing time with the BMA is smaller than that of the optical flow method by a factor of about 14. Also the anisotropic optical flow method is faster (0.2sec) than the previous method [8]. The optical flow method uses spatial and temporal correlations. The window-based optical flow method reduces the processing time by 20% on the average compared with the previous work [8]. We present a fast optical flow method using a detection window, in which the measurement window is smaller than the detection window, utilizing the relationship of neighboring pixels.

The proposed method may be thought to be a little bit slow to process images in real-time. Note that it is much faster than direct viewing of a large amount of 3-D data and the processing time of the proposed algorithm is reduced with a high-speed CPU.

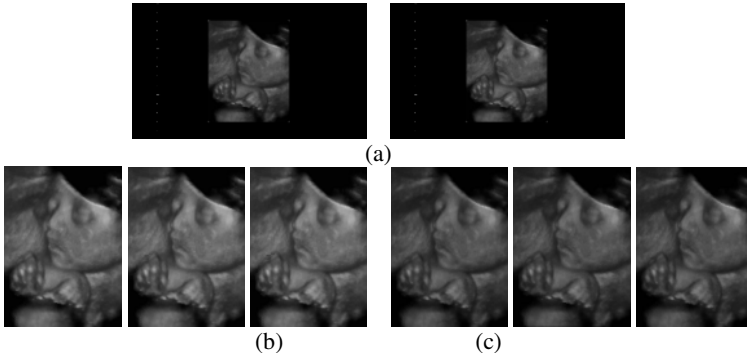


Fig. 8. Comparison of interpolated frames (test sequence 3). (a) input frames I_{14} and I_{15} . (b) zoomed interpolated frames using the BMA (frames $I_{14:1}$, $I_{14:2}$, and $I_{14:3}$, block size: 8×8). (c) zoomed interpolated frames using the anisotropic optical flow method (frames $I_{14:1}$, $I_{14:2}$, and $I_{14:3}$, window size: 2×2).

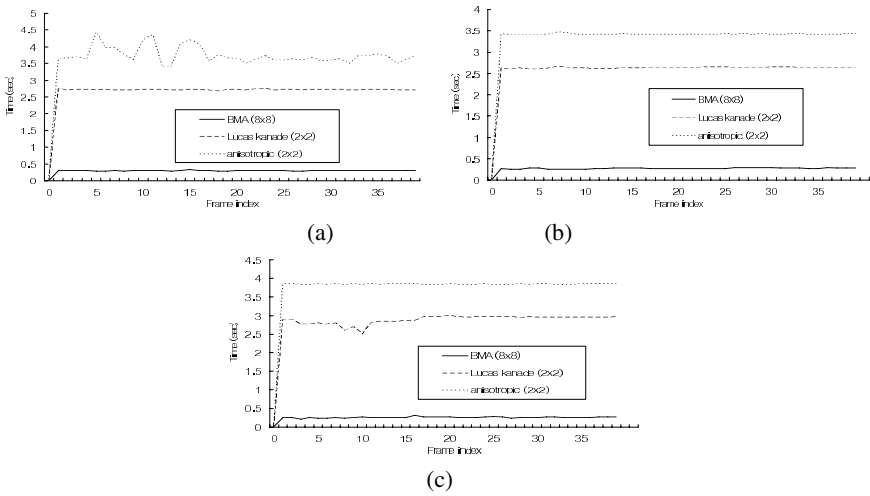


Fig. 9. Comparison of the computation time as a function of the frame index (40 frames). (a) Test sequence 1. (b) Test sequence 2. (c) Test sequence 3.

6 Conclusions

This paper proposes a window-based optical flow based motion estimation method for ultrasound image frame interpolation and postprocessing for image enhancement. Computer simulation with several sets of real ultrasound image sequences shows that the results by the proposed optical flow based interpolation method are better than those by the BMA, reducing block artifacts. Further research will focus on the development of the fast frame interpolation methods for real-time implementation.

Acknowledgement

This work was supported by Brain Korea 21 Project and ultrasound images were provided by Medison Co., Ltd.

References

1. Nelson, T. R., Downey, D. B., Pretorius, D. H., and Fenster, A.: Three-Dimensional Ultrasound. Lippincott Williams & Wilkins (1999)
2. Lee, G.-I., Park, R.-H., Song, Y.-S., Kim, C.-A., and Hwang, J.-S.: Real-Time 3-D Ultrasound Fetal Image Enhancement Techniques Using Motion Compensated Frame Rate Up-conversion. Proc. SPIE Ultrasonic Imaging and Signal Processing, Vol. 5035. San Diego CA USA (2003) 373–385
3. Tekalp, A. M., Digital Video Processing. Prentice Hall (1995)
4. Singh, A., Optical Flow Computation: A Unified Perspective. IEEE Computer Society Press (1991)
5. Horn, B. K. P., and Schunk, B. G.: Determining Optical Flow. Artificial Intelligence, Vol. 17. (1981) 185–203
6. Nagel, H. H.: Displacement Vectors Derived from Second-Order Intensity Variations in Image Sequences. Int. J. Comput. Vision Graphics Image Process, 21 2 (1983) 85–117
7. Lucas, D. B., and Kanade, T.: An Iterative Image Registration Technique with an Application to Stereo Vision. Proc. DARPA Image Understanding Workshop, Washington DC USA (1981) 121–130
8. Nam, T.-J., and Park, R.-H.: Frame Interpolation of 3-D Ultrasound Using Optical Flow. Proc. Computational Imaging II, Vol. 5299. San Jose CA USA (2004) 378–386
9. Perona, P., and Malik, J.: Scale-Space and Edge Detection Using Anisotropic Diffusion. IEEE Trans. Pattern Analysis and Machine Intelligence, 12 7 (1990) 629–639
10. Sim, D. G., and Park, R.-H.: A Two-Stage Algorithm for Motion Discontinuity-preserving Optical Flow Estimation. Computer Vision and Image Understanding, 65 1 (1997) 19–37
11. Barron, J. L., Fleet, D. J., and Beauchemin, S. S.: Performance of Optical Flow Techniques. Int. J. Computer Vision, 12 1 (1994) 43–77
12. Astola, J., Haavisto, P., and Neuvo, Y.: Vector Median Filters. Proc. IEEE, 78 (1990) 678–689
13. Alparone, L., Barni, M., Bartolini, F., and Santurri, L.: An Improved H.263 Video Coder Relying on Weighted Median Filtering of Motion Vectors. IEEE Trans. Circuits and Syst. Video Technol., 11 2 (2001) 235–240
14. Tukey, J. W., Exploratory Data Analysis. Reading, Addison-Wesley (1970).
15. Han, S. C., and Wood, J. W.: Frame-Rate Up-conversion using Transmitted Motion and Segmentation Fields for Very Low Bit-Rate Video Coding. Proc. IEEE Int. Conf. Image Processing, Vol. 1. Santa Barbara CA USA (1997) 747–750
16. Sim, D. G., and Park, R.-H.: Anisotropic Hierarchical Motion Estimation Method Based on Decomposition of the Functional Domain. Int. J. Visual Communication and Image Representation, 7 3 (1996) 259–272
17. Nam, T.-J., and Park, R.-H.: Image Enhancement of 3-D Ultrasound Interpolation Image Using BMA. Proc. Image Science System and Technol., Las Vegas NV USA (2004) 541–547
18. Spies, H., and Barron, J. L.: Evaluating the Range Flow Motion Constraint. Proc. Pattern Recognition, Vol. 3. Quebec Canada (2002) 517–520

An Iterative Multiresolution Scheme for SFM

Carme Julià, Angel Sappa, Felipe Lumbreras, Joan Serrat, and Antonio López

Computer Vision Center and Computer Science Department,
Universitat Autònoma de Barcelona,
08193 Bellaterra, Spain
{cjulia, asappa, felipe, joans, antonio}@cvc.uab.es

Abstract. Several factorization techniques have been proposed for tackling the *Structure from Motion* problem. Most of them provide a good solution, while the amount of missing and noisy data is within an acceptable ratio. Focussing on this problem, we propose to use an incremental multiresolution scheme, with classical factorization techniques. Information recovered following a coarse-to-fine strategy is used for both, filling in the missing entries of the input matrix and denoising original data. An evaluation study, by using two different factorization techniques—the *Alternation* and the *Damped Newton*—is presented for both synthetic data and real video sequences.¹

1 Introduction

Structure From Motion (SFM) consists in extracting the 3D shape of a scene as well as the camera motion from trajectories of tracked features. Factorization is a method addressing to this problem. The central idea is to express a matrix of trajectories W as the product of two unknown matrices, namely, the 3D object's shape S and the relative camera pose at each frame M : $W_{2f \times p} = M_{2f \times r} S_{r \times p}$, where f , p are the number of frames and feature points respectively and r the rank of W . The goal is to find the factors M and S that minimize

$$\|W - MS\|_F^2 \quad (1)$$

where $\|\cdot\|$ is the Frobenius matrix norm [1].

The *Singular Value Decomposition* (SVD) gives the best solution for this problem when there are not missing entries. Unfortunately, in most of the real cases not all the data points are available, hence other methods need to be used. With missing data, the expression to minimize is the following

$$\|W - MS\|_F^2 = \sum_{i,j} |W_{ij} - (MS)_{ij}|^2 \quad (2)$$

where i and j correspond to the index pairs where W_{ij} is defined.

¹ This work has been supported by the Government of Spain under the CICYT project TRA2004-06702/AUT. The second author has been supported by The Ramón y Cajal Program.

In the seminal approach Tomasi and Kanade [2] propose an initialization method in which they first decompose the largest full submatrix by the factorization method and then the initial solution grows by one row or by one column at a time, unveiling missing data. The problem is that finding the largest full submatrix is a NP-hard problem. Jacobs [3] treats each column with missing entries as an affine subspace and shows that for every r -tuple of columns the space spanned by all possible completions of them must contain the column space of the completely filled matrix. Unknown entries are recovered by finding the least squares regression onto that subspace. However it is strongly affected by noise on the data. An incremental SVD scheme of incomplete data is proposed by Brand [4]. The main drawback of that technique is that the final result depends on the order in which the data are given. Brandt [5] proposes a different technique based on the expectation maximization algorithm (EM) and although the feature points do not have to be visible in all views, the affine projection matrices in each image must be known. A method for recovering the most reliable imputation, addressing the SFM problem, is provided by Suter and Chen [6]. They propose an iterative algorithm to employ this criterion to the problem of missing data. Their aim to obtain the projection of W onto a low rank matrix to reduce noise and to fill in missing data.

A different approach to address the factorization with missing data is the *Alternation technique* [7]. One of the advantages of this method is that it converges quickly. The algorithm starts with an initial random S_0 or M_0 and solves one factor at each iteration k , until the product $M_k S_k$ converges to W . The key point of this 2-step algorithm is that, since the updates of S given M (analogously in the case of M given S) can be independently done for each row of S , missing entries in W correspond to omitted equations. Due to that fact, with a large amount of missing data the method would fail to converge.

In [7], Buchanan and Fitzgibbon summarize factorization approaches with missing data and proposes the *Alternation/Damped Newton Hybrid*, which combines the *Alternation* strategy with the *Damped Newton* method. The latter is fast in valleys, but not effective when far from the minima. The goal of introducing this hybrid scheme is to give a method that has fast initial convergence and, at the same time, has the power of non-linear optimization.

One disadvantage of the above methods is that. They give a good factorization while the amount of missing points is low, which is not common in real image sequences, unfortunately. Addressing to this problem, we recently presented an iterative multiresolution scheme [8], which incrementally fill in missing data. At the same time noisy data are filtered. The key point of that approach is to work with a reduced set of feature points along a few number of consecutive frames. Thus, the 3D reconstruction corresponding to the selected feature points and the camera motion of the used frames are obtained. The missing entries of the trajectory matrix can be recovered just multiplying the shape and motion matrices. The amount of missing data is reduced after each iteration; at the same time it increases the chances of having a better result. In the current paper

we propose improvements over the original approach, as well as the use of two different techniques under this incremental multiresolution scheme.

This paper is organized as follows. Section 2 briefly introduces the incremental multiresolution scheme. Improvements on the original version are emphasized. Section 3 presents an evaluation study of the use of two factorization techniques with the proposed scheme. Conclusions and future work are given in section 4.

2 Iterative Multiresolution Scheme

In this section the iterative multiresolution scheme, which incrementally fill in missing data, is presented. Essentially, the basic idea is to generate sub-matrices with a reduced density of missing points. Thus, any classical factorization technique could be used for factoring these sub-matrices, recovering their corresponding 3D shape and motion; at the same time missed data on the original matrix will be filled with their resulting product. Additionally, it is expected that noisy data are filtered. The technique consists of two stages, which are briefly described below. Improvements on the original version are highlighted, more details of the original technique can be found in [8].

2.1 Observation Matrix Splitting

Let $W_{2f \times p}$ be the observation matrix (also referred through the paper as input matrix) of p feature points tracked over f frames containing missing entries; it will be denoted as W . Let k be the index indicating the current iteration number.

In a first step, the input matrix W is split in a uniformly distributed set of $k \times k$ non-overlapped sub-matrices, W_k , with a size of $\lfloor \frac{2f}{k} \rfloor \times \lfloor \frac{p}{k} \rfloor$.

Then, in a second step, a multiresolution approach is followed. It consists in computing four W_{2k} overlapped sub-matrices with twice the size of W_k (only for $k > 2$). The idea of this enlargement process is to study the behavior of feature points contained in W_k when a bigger region is considered (see Fig. 1).

Since generating four W_{2k} , for every W_k , is a computationally expensive task, a simple and more direct approach is followed. It consists in splitting the input matrix W in four different ways, by shifting W_{2k} half of its size (i.e., W_k) through rows, columns or both at the same time. Fig. 2 illustrates the five partitions of matrix W —i.e., W_k and W_{2k} sub-matrices generated at the sixth iteration. When all these matrices are considered together, the overlap between the different areas is obtained, see textured cell in Fig. 1 and Fig. 2. As it can be appreciated in Fig. 2, corners and border cells are considered only twice and three times, respectively, at each iteration.

2.2 Sub-matrices Processing

At this stage, the objective is to recover missing data by applying a factorization technique at every single sub-matrix. Independently of their size hereinafter sub-matrices will be referred as W_s .

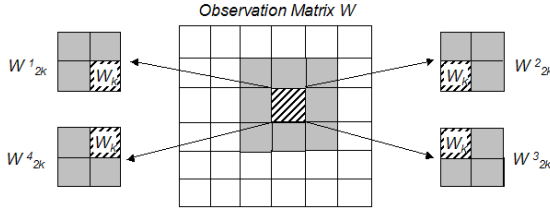


Fig. 1. W_k and W_{2k} overlapped matrices of the observation matrix W , computed during the first stage (section 2.1), at iteration $k = 6$

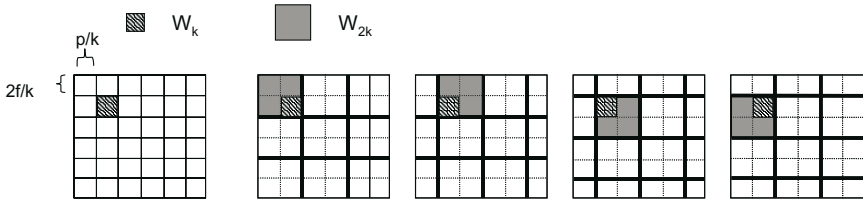


Fig. 2. Five partitions of matrix W . Note the overlap between a W_k sub-matrix with its corresponding four W_{2k} sub-matrices, computed during the first stage (section 2.1)

Given a sub-matrix W_s , its factorization using a particular technique gives its corresponding M_s and S_s matrices. Their product could be used for computing an approximation error ε_s such as equation (2). Actually, in this paper we use the root mean squared (*rms*) of this error per image point:

$$rms_s = \sqrt{\frac{\sum_{i,j} |(W_s)_{ij} - (M_s S_s)_{ij}|^2}{\frac{n}{2}}} \tag{3}$$

where i and j correspond to the index pairs where $(W_s)_{ij}$ is defined and n the amount of those points in W_s .

The main advantage of using *rms* is that it does not depend on the number of known entries. Therefore, it is a normalized measure that gives a better idea of the goodness of the solution than the error defined at equation (2), which could be confusing or provide erroneous results. For instance, by using the latter, a big error value could be obtained only due to the fact of working with a great amount of known entries. On the contrary, a small error value could correspond to a case with a few known entries.

After processing the current W_s , its corresponding *rms_s* is compared with a user defined threshold σ . In case the resulting *rms_s* is smaller than σ , every point in W_s is kept in order to be merged with overlapped values after finishing the current iteration. Additionally, every point of W_s is associated with a weighting factor, defined as $\frac{1}{rms_s}$, in order to measure the goodness of that value. These weighting factors are later on used for merging data on overlapped areas. Otherwise, the resulting *rms_s* is higher than σ , computed data are discarded. With

the rms_s error measure, a unique threshold, valid for every matrix, is defined. As mentioned above, this is the main advantage over the previous version.

Finally, when every sub-matrix W_s has been processed, recovered missing data are used for filling in the input matrix W . In case a missing datum has been recovered from more than one sub-matrix (overlapped regions), those recovered data are merged by using their corresponding normalized weighting factors. On the contrary, when a missing datum has been recovered from only one sub-matrix, this value is directly used for filling in that position. Already known entries in W could be also modified. In this case, instead of taking the new computed data directly as in [8], the mean between the initial value and the new one, obtained after the merging process, is assigned.

Once recovered missing data have been used for filling in the input matrix W , the iterative process starts again (section 2.1) splitting the new matrix W either by increasing k one unit or, in case the size of sub-matrices W_k at the new iteration stage is quite small, by setting $k = 2$. This iterative process is applied until one of the following conditions is true: a) the matrix of trajectories is totally filled; b) at the current iteration no missing data were recovered; c) a maximum number of iterations is reached.

3 Evaluation Study

Assuming both the filling missing entries and denoising capabilities, as it was presented in [8], in this section a study of using the iterative multiresolution scheme with different factorization techniques is presented. In particular, the work is focussed on the use of: Alternation and Damped Newton [7]. Experiments using both synthetic and real data are presented below. The methodology proposed to evaluate the obtained results consists in applying:

- A factorization technique over the input matrix W .
- The same factorization technique with the proposed multiresolution scheme.

3.1 Synthetic Object

Synthetic data are randomly generated by distributing 35 3D feature points over the whole surface of a cylinder, see Fig. 3 (left). The cylinder is defined by a radius of 100 and a height of 400; it rotates over its principal axis; the camera also moves. An input matrix W with 20% of known data, directly obtained taking 35 frames, is used for the evaluation, see Fig. 3 (middle). Notice that W has a banded structure; it is symmetric due to the way it has been generated. Elements of the matrix W are represented by means of a grey level scale. Feature point trajectories are plotted in Fig. 3 (right).

Factorization Using Alternation. Fig. 4 (left) shows the recovered trajectories obtained by applying the Alternation technique to the input matrix W for this synthetic example. In this case, the resulting rms is 6.43.

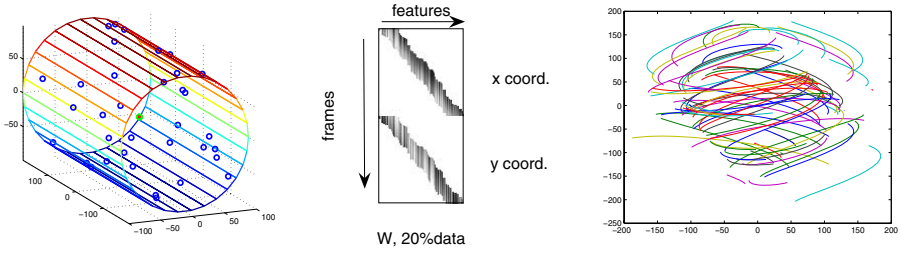


Fig. 3. (left) Synthetic cylinder. (middle) Input matrix of trajectories, with 20% of known data. (right) The same feature point trajectories represented in the image plane.

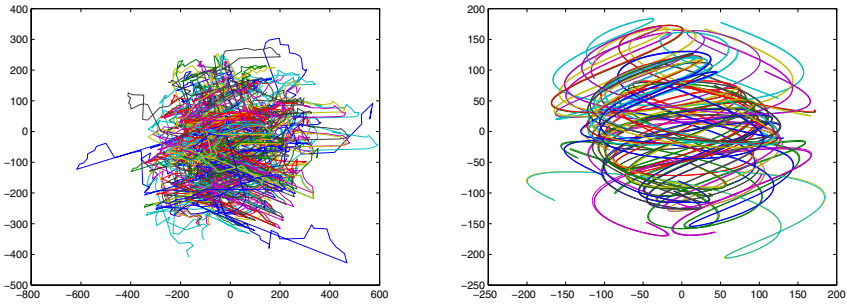


Fig. 4. (left) Recovered feature points trajectories applying Alternation to W . (right) The same, but applying Alternation with the iterative multiresolution scheme.

Fig. 5 shows intermediate results obtained after applying the Alternation technique with the multiresolution scheme to W . In order to illustrate the process, the amount of recovered data at the third, fifth and last iterations of the multiresolution scheme are presented. While the input matrix has 20% of data, the final one has 63% of data. The trajectories plotted in Fig. 4 (right) are obtained after applying the Alternation technique to this final matrix (Fig. 5 (right)). The *rms* value is 0.29. Notice that these trajectories do not form a cylinder, as one might expect, due to the fact that the camera moves.

Fig. 6 (left) and (middle) shows the recovered \mathbf{x} and \mathbf{y} camera motion axes, from both strategies and at each frame. The 3D plot corresponds to each component of the vector axes: $\mathbf{x}(x, y, z)$, and $\mathbf{y}(x, y, z)$ at each frame. The obtained 3D reconstructions of the cylinder are plotted in Fig. 6 (right). In order to show the goodness of the reconstruction, the cylinder that best fits the final data is also plotted. Fig. 6 (top) corresponds to results computed by applying Alternation to W , while Fig. 6 (bottom) presents the results obtained by applying Alternation with the proposed multiresolution scheme.

It can be seen that the results are considerably improved with the proposed multiresolution scheme, both for the recovered feature points trajectories and for the obtained shape and motion.

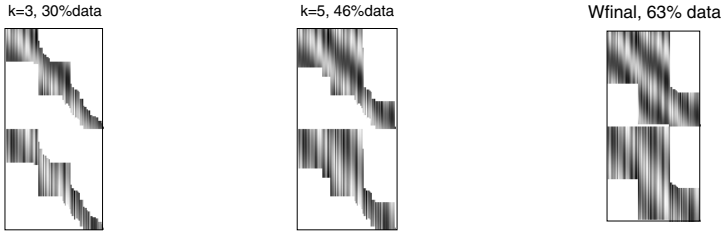


Fig. 5. Third, fifth and last iterations of the multiresolution scheme when the Alternation technique is used. A final matrix with 63% of data is obtained.

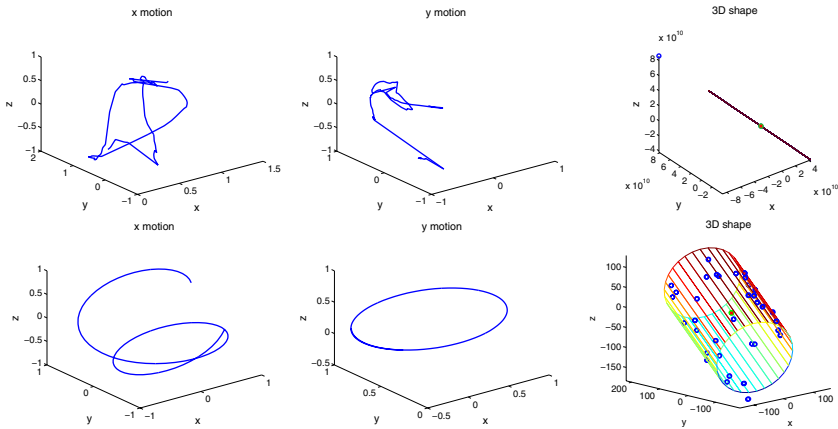


Fig. 6. (left) and (middle) Plots of the recovered \mathbf{x} and \mathbf{y} camera motion axes, at each frame. (right) 3D reconstructions of the cylinder. (top) Applying Alternation to W . (bottom) Applying Alternation with the multiresolution scheme.

Factorization Using Damped Newton. The observation matrix W presented in Fig. 3 (middle) has been also used as input for the Damped Newton factorization technique. Fig. 7 (left) plots the recovered trajectories applying the Damped Newton to W . The *rms* is 0.76.

Fig. 7 (right) shows the obtained trajectories when Damped Newton is applied with the proposed multiresolution scheme; in this case the *rms* is 22.22. As pointed out in [6], the measure of error taking only the known entries of W could be ambiguous. In this particular case, since this is a synthetic sequence, the missing entries are available. Hence, the *rms* taking into account all those entries has been computed for both results. Their values are 77.0 applying Damped Newton to W and 69.1, with the multiresolution scheme. Again, better results, not only visual but also numerical, are obtained with the multiresolution scheme.

The recovered \mathbf{x} and \mathbf{y} camera motion axes from both strategies are plotted in Fig. 8 (left) and (middle) respectively. In Fig. 8 (right) the obtained 3D reconstructions of the cylinder are shown.

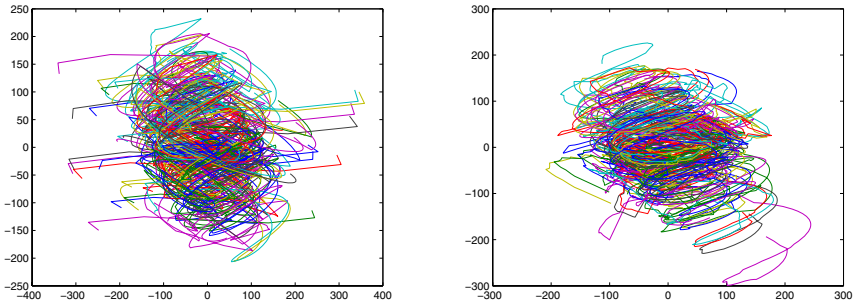


Fig. 7. (left) Recovered trajectories applying Damped Newton to W . (right) The same, but applying Damped Newton with the proposed scheme.

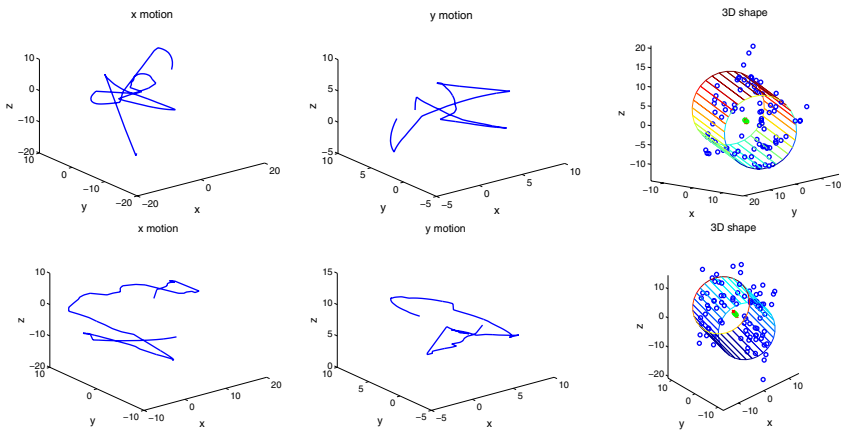


Fig. 8. (left) and (middle) Plots of the recovered x and y camera motion axes, at each frame. (right) 3D reconstructions of the cylinder. (top) Applying Damped Newton to W . (bottom) Applying Damped Newton with the multiresolution scheme.

The results obtained with the Damped Newton are not as good as the ones obtained with the Alternation, both applying it to W and with the proposed multiresolution scheme. Notice that, inspite of that, results with the proposed scheme are better than using Damped Newton directly over W .

3.2 Real Object

A real video sequence of 101 frames with a resolution of 640×480 pixels is used. The studied object is shown in Fig. 9 (left). A single rotation around a vertical axis was performed. Feature points are selected by means of a corner detector algorithm and 87 points distributed over the squared-face-box are considered. An iterative feature tracking algorithm has been used. More details about corner detection and tracking algorithm can be found in [9]. Missing data are obtained by removing data randomly. As in the previous case, an input matrix with 20%

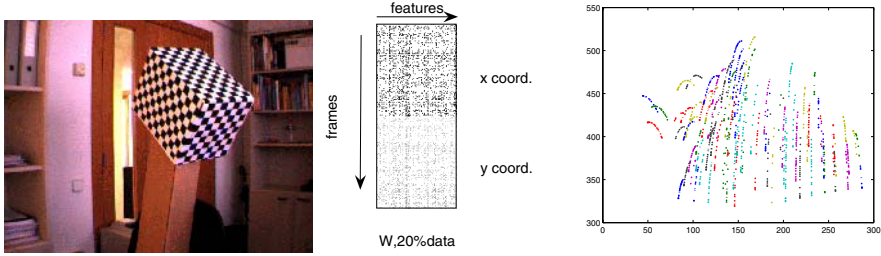


Fig. 9. (left) Object used for the real case. (middle) Input matrix of trajectories, with 20% of known data. (right) Feature point trajectories represented in the image plane.

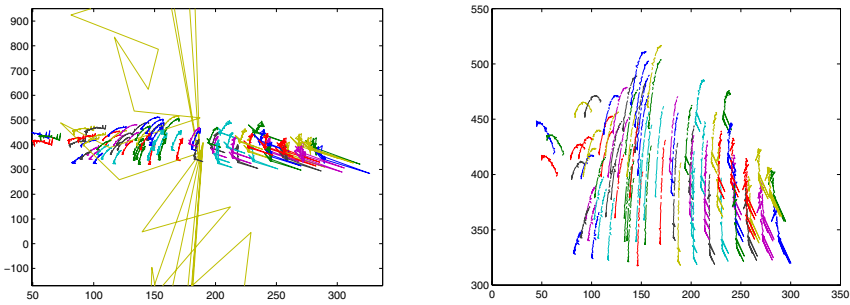


Fig. 10. (left) Recovered feature points trajectories applying Alternation to W , Fig. 9 (middle). (right) The same, but applying Alternation with the proposed scheme.

of data is considered, see Fig. 9 (middle). Notice that the input matrix W has not a band structure like in the synthetic case, but a structure that unveil the random nature of missing entries. The feature point trajectories are plotted in Fig. 9 (right). The methodology applied over the previous W is presented below.

Factorization Using Alternation. The Alternation technique is applied to the input matrix W and the obtained rms is 2.41. Fig. 10 (left) plots an enlargement of the recovered trajectories, in order to avoid a plot such as Fig. 11 (top-right).

On the contrary, when Alternation is applied with the proposed multiresolution scheme, results improve considerably. For instance, about 95% of data are contained in matrix W during last iteration (recall that the input matrix only contains about 20%), and the final resulting rms is 0.69. Fig. 10 (right) shows the trajectories recovered with the proposed scheme.

Fig. 11 (left) and (middle) show the recovered x and y camera motion axes, from both strategies. The obtained 3D reconstructions of the input object are presented in Fig. 11 (right). As in the synthetic experiment, results are improved applying Alternation with the proposed iterative multiresolution scheme.

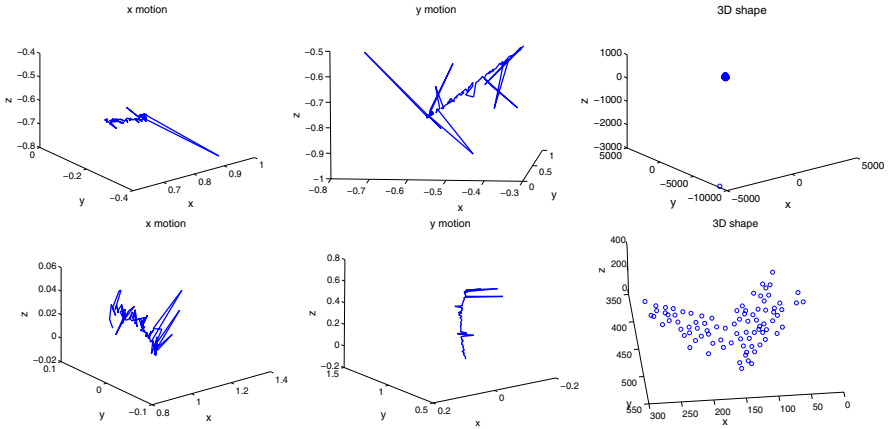


Fig. 11. (left) and (middle) Plot of the recovered x and y camera motion axes, at each frame. (right) 3D reconstruction of the input object. (top) Results obtained applying Alternation to W . (bottom) Applying Alternation with the proposed scheme.

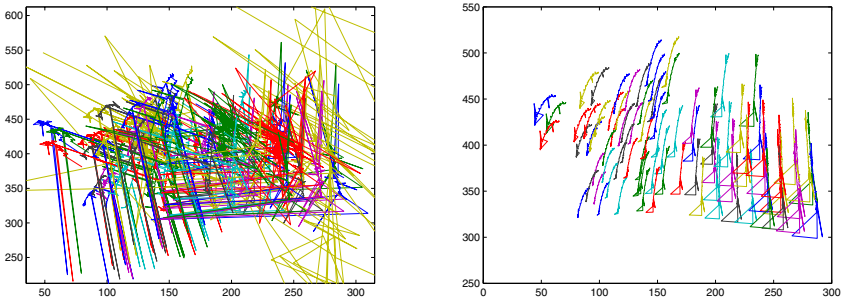


Fig. 12. (left) Recovered trajectories applying Damped Newton to W , Fig. 9 (middle). (right) The same, but applying Damped Newton with the proposed scheme.

Factorization Using Damped Newton. The recovered trajectories applying Damped Newton are plotted in Fig. 12 (left) and the rms is 0.87. Again, an enlargement has been performed in order to obtain a better visualization.

Fig. 12 (right) shows the obtained trajectories applying Damped Newton with the proposed multiresolution scheme. In this case, the rms is 7.38. Since the missing points have been obtained randomly from a full matrix, all the entries are available. Therefore, as in the synthetic case, all the points have been taken into account for computing the rms . The resulting rms errors are 29.54 applying Damped Newton to W and 8.15 with the multiresolution scheme.

Finally, the recovered x and y camera motion axes, from both strategies, are plotted in Fig. 13 (left) and (middle), respectively. The obtained 3D reconstructions of the input object are shown in Fig. 13 (right).

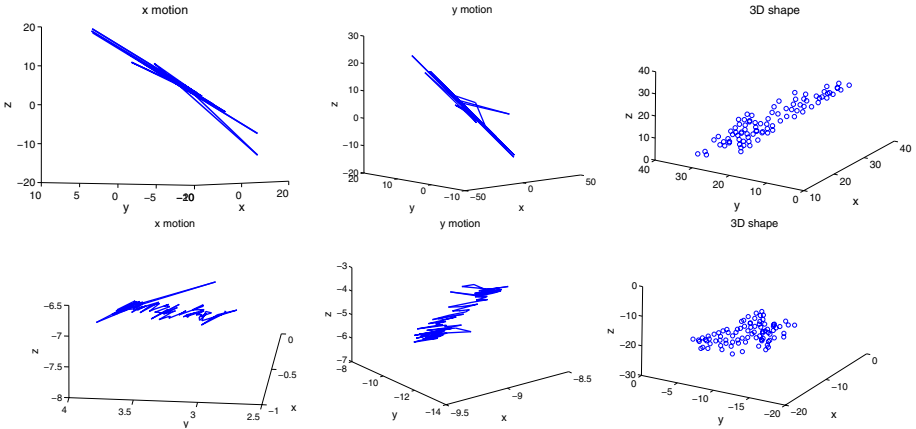


Fig. 13. (left) and (middle) Plots of the recovered x and y camera motion axes, at each frame. (right) 3D reconstructions of the input object. (top) Applying Damped Newton to W . (bottom) Applying Damped Newton with the multiresolution scheme.

As in the synthetic case, the obtained results using Damped Newton are worse than the ones obtained with Alternation. Again, results with the proposed scheme are better than the ones obtained using Damped Newton directly over W .

4 Conclusions and Future Work

This paper presents improvements on the original iterative multiresolution approach. The main contribution is the modified definition of the error. Here, the root mean of the previous error per image point is considered. As mentioned before, this is a normalized measure that gives a better idea of the goodness of the result. A less important improvement is the way in which the recovered missing data are merged in the case of data known a priori. The average between the input entries and the new computed data obtained after the merging process is assigned. Moreover, in the current paper, the attention is not only focused on the error value, but also on the obtained M and S . However, we would like to define a function to measure the goodness of these recovered factors.

Additionally, when the original multiresolution scheme was presented, its validation was done using only the Alternation technique. In this work, the Damped Newton technique is also studied and compared with the Alternation. Although any factorization technique can be applied with this multiresolution scheme, the Damped Newton method seems to be more appropriated when the input matrix contains only a few missing data. Otherwise it takes a lot of time and may be wrong results are obtained. As a future work the use of an hybrid technique will be considered.

Acknowledgments

The authors would like to thanks Aeron Buchanan for providing us with the Damped Newton code and interesting insights.

References

1. Golub, G., Van Loan, C., eds.: *Matrix Computations*. The Johns Hopkins Univ. Press (1989)
2. Tomasi, C., Kanade, T.: Shape and motion from image streams: a factorization method. Full report on the orthographic case (1992)
3. Jacobs, D.: Linear fitting with missing data for structure-from-motion. *Computer vision and image understanding, CVIU* (2001) 7–81
4. Brand, M.: Incremental singular value decomposition of uncertain data with missing values. In: *Proceedings, ECCV*. (2002) 707–720
5. Brandt, S.: Closed-form solutions for affine reconstruction under missing data. In: *Proceedings Statistical Methods for Video Processing Workshop, in conjunction with ECCV*. (2002) 109–114
6. Chen, P., Suter, D.: Recovering the missing components in a large noisy low-rank matrix: Application to sfm. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26** (2004)
7. Buchanan, A., Fitzgibbon, A.: Damped newton algorithms for matrix factorization with missing data. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* **2** (2005) 316–322
8. Julià, C., Sappa, A., Lumbreras, F., Serrat, J., López, A.: Factorization with missing and noisy data. In: *Computational Science–ICCS 2006: 6th International Conference, Proceedings, Part I*. (2006) 555 – 562
9. Ma, Y., Soatto, J., Koseck, J., Sastry, S.: *An invitation to 3d vision: From images to geometric models*. Springer-Verlang, New York (2004)

Occlusion-Based Accurate Silhouettes from Video Streams

Pedro M.Q. Aguiar, António R. Miranda, and Nuno de Castro

Institute for Systems and Robotics / Instituto Superior Técnico
Lisboa, Portugal
aguiar@isr.ist.utl.pt

Abstract. We address the problem of segmenting out moving objects from video. The majority of current approaches use only the image motion between two consecutive frames and fail to capture regions with low spatial gradient, *i.e.*, low textured regions. To overcome this limitation, we model explicitly: i) the *occlusion* of the background by the moving object and ii) the *rigidity* of the moving object across a set of frames. The segmentation of the moving object is accomplished by computing the Maximum Likelihood (ML) estimate of its silhouette from the set of video frames. To minimize the ML cost function, we developed a greedy algorithm that updates the object silhouette, converging in few iterations. Our experiments with synthetic and real videos illustrate the accuracy of our segmentation algorithm.

1 Introduction

Content-based representations for video enable efficient storage and transmission as well as powerful non-linear editing and manipulation [1]. The automatic segmentation of an image into regions that undergo different motions is a key step in the generation of content-based video representations. In this paper we address the problem of segmenting objects that exhibit a rigid motion across a set of frames.

A number of approaches to the segmentation of moving objects are found in the video coding literature. In fact, efficient video coding reduces temporal redundancy by predicting each frame from the previous one through motion compensation. Regions undergoing different movements are then compensated in different ways, according to their motion, see for example [2] for a review on very low bit rate video coding. The majority of these approaches are based on a single pair of consecutive frames and try to capture the moving object by detecting the regions that changed between the two co-registered images, see for example [3]. Since these methods were developed for image coding rather than for inferring high level representations, they often lead to inaccurate segmentation results. In particular, they fail to segment moving objects containing low textured regions because these regions are considered as unchanged, being then missclassified as belonging to the background.

The more recent interest on the so-called layered representations for video [4,5,6,7,8] has motivated further work on motion-based segmentation. A number of approaches in the computer vision literature uses other cues besides motion, such as color and edges [9], or regularization priors [10]. In general, these methods lead to complex and time consuming algorithms.

Few approaches to motion segmentation use temporal integration, see [11,12,13,14] as examples. In [11,12], the images in the sequence are averaged after appropriate registration according to motion of the object. The silhouette of the moving object is estimated by detecting the regions of the current frame that are similar to the integrated image. This method overestimates the object silhouette unless the background is textures enough to blur completely the integrated image. The method in [13,14] exploits the occlusion of the background by the moving object—it estimates the silhouette of the object by integrating over time the intensity differences between the object and the background. This method succeeds even in low textured / low contrast scenes but it requires that the background is completely uncovered in the video clip.

We propose a new segmentation algorithm that exploits *occlusion* and *rigidity* without the drawback of the one in [13,14]. As in [13,14], we formulate the segmentation problem as the Maximum Likelihood (ML) estimation of the parameters involved in the video sequence model: the motions of the background, the motions of the object, the silhouette of the object, the intensity levels of the object (the object texture), and the intensity levels of the background (the background texture). The algorithm of [13,14] minimizes the ML cost function by computing, in two alternate steps, the estimates of: i) the object silhouette and ii) the background texture. We avoid the need to compute the background intensity levels at all pixels (and thus the requirement that the background is completely uncovered) by using the closed-form expression for the ML estimate of the background texture to derive the ML cost function as a function of the object silhouette alone. We develop a greedy algorithm that updates the silhouette converging in a small number of iterations.

Although our method is particularly tailored to the segmentation of rigid objects, it turns out also very useful to handle non-rigid ones. In fact, when processing videos showing non-rigid moving objects, the tracking procedures that cope with flexible silhouettes need an adequate initialization. Our method provides such an initialization because it will compute the *best rigid interpretation* of the scene, which suffices to segment out the moving objects. Finally, we remark that although our derivations assume scalar-valued images, *e.g.*, intensity of grey-level images, they are straightforwardly extended to vector-valued images, *e.g.*, multispectral images.

1.1 Paper Organization

In section 2 we formulate the segmentation problem as ML inference. Section 3 describes the ML cost function minimization procedure. In section 4 we outline how the algorithm is initialized. Section 5 contains experiments and section 6 concludes the paper.

2 Problem Formulation: Segmentation as Maximum Likelihood Inference

We consider 2D parallel motions, *i.e.*, all motions (translations and rotations) are parallel to the camera plane. We represent those motions by specifying time varying position vectors. These position vectors code rotation-translation pairs that take values in the group of rigid transformations of the plane, *i.e.*, the special Euclidean group SE(2). The vector \mathbf{p}_f represents the position of the background relative to the camera in frame f . The vector \mathbf{q}_f represents the position of the moving object relative to the camera in frame f . The image obtained by applying the rigid motion coded by the vector \mathbf{p} to the image \mathbf{I} is denoted by $\mathcal{M}(\mathbf{p})\mathbf{I}$, *i.e.*, pixel (x, y) of the image $\mathcal{M}(\mathbf{p})\mathbf{I}$ is given by $\mathcal{M}(\mathbf{p})\mathbf{I}(x, y) = \mathbf{I}(f_x(\mathbf{p}; x, y), f_y(\mathbf{p}; x, y))$, where $f_x(\mathbf{p}; x, y)$ and $f_y(\mathbf{p}; x, y)$ represent the coordinate transformation imposed by the 2D rigid motion coded by \mathbf{p} . We denote the inverse of $\mathcal{M}(\mathbf{p})$ by $\mathcal{M}(\mathbf{p}^\#)$ and the composition of $\mathcal{M}(\mathbf{a})$ with $\mathcal{M}(\mathbf{b})$ by $\mathcal{M}(\mathbf{ab})$, *i.e.*, we have $\mathcal{M}(\mathbf{pp}^\#)\mathbf{I} = \mathbf{I}$. For more details, see [13,14].

2.1 Observation Model

We consider a scene with a moving object in front of a moving camera. The pixel (x, y) of the image \mathbf{I}_f belongs either to the background \mathbf{B} or to the object \mathbf{O} . The image \mathbf{I}_f is then modelled as

$$\mathbf{I}_f = \left\{ \mathcal{M}(\mathbf{p}_f^\#)\mathbf{B} \left[1 - \mathcal{M}(\mathbf{q}_f^\#)\mathbf{T} \right] + \mathcal{M}(\mathbf{q}_f^\#)\mathbf{O} \mathcal{M}(\mathbf{q}_f^\#)\mathbf{T} + \mathbf{W}_f \right\} \mathbf{H}, \quad (1)$$

where we make $\mathbf{I}_f(x, y) = 0$ for (x, y) outside the region observed by the camera. This is taken care of in (1) by the binary mask \mathbf{H} whose (x, y) entry is such that $\mathbf{H}(x, y) = 1$ if pixel (x, y) is in the observed image \mathbf{I}_f or $\mathbf{H}(x, y) = 0$ if otherwise. Naturally, \mathbf{H} does not depend on the frame index f , since the motion of the camera is captured as background motion. In (1), \mathbf{T} is the moving object silhouette— $\mathbf{T}(x, y) = 1$ if the pixel (x, y) belongs to the moving object or $\mathbf{T}(x, y) = 0$ if otherwise—and \mathbf{W}_f stands for the observation noise, assumed Gaussian, zero mean, and white.

2.2 Maximum Likelihood Inference

Given a set of F video frames $\{\mathbf{I}_f, 1 \leq f \leq F\}$, we want to estimate the background texture \mathbf{B} , the object texture \mathbf{O} , the object silhouette \mathbf{T} , the camera poses $\{\mathbf{p}_f, 1 \leq f \leq F\}$, and the object positions $\{\mathbf{q}_f, 1 \leq f \leq F\}$. Using the observation model in (1) and the Gaussian white noise assumption, ML estimation leads to the minimization over all parameters of the functional

$$\begin{aligned} C(\mathbf{B}, \mathbf{O}, \mathbf{T} \{ \mathbf{p}_f \}, \{ \mathbf{q}_f \}) &= \int \int \sum_{f=1}^F \left\{ \mathbf{I}_f(x, y) \right. \\ &\quad \left. - \mathcal{M}(\mathbf{p}_f^\#)\mathbf{B}(x, y) \left[1 - \mathcal{M}(\mathbf{q}_f^\#)\mathbf{T}(x, y) \right] \right. \\ &\quad \left. - \mathcal{M}(\mathbf{q}_f^\#)\mathbf{O}(x, y) \mathcal{M}(\mathbf{q}_f^\#)\mathbf{T}(x, y) \right\}^2 \mathbf{H}(x, y) dx dy, \quad (2) \end{aligned}$$

where the inner sum is over the full set of F frames and the outer integral is over all pixels. For details, see [13,14].

3 Maximum Likelihood Estimation: Greedy Algorithm

The minimization of the functional C in (2) with respect to (wrt) the set of constructs $\{\mathbf{B}, \mathbf{O}, \mathbf{T}\}$ and to the motions $\{\{\mathbf{p}_f\}, \{\mathbf{q}_f\}, 1 \leq f \leq F\}$ is a highly complex task. To obtain a computationally feasible algorithm, we decouple the estimation of the motion vectors from the determination of the constructs $\{\mathbf{B}, \mathbf{O}, \mathbf{T}\}$. This is reasonable from a practical point of view and is well supported by experimental results with real videos. We perform the estimation of the motions on a frame by frame basis by using known motion estimation methods [15]. After estimating the motions, we introduce the motion estimates into the ML cost C and minimize wrt the remaining parameters, *i.e.*, wrt the silhouette \mathbf{T} of the moving object, the texture \mathbf{O} of the moving object, and the texture \mathbf{B} of the background.

We express the estimate $\widehat{\mathbf{O}}$ of the moving object texture and the estimate $\widehat{\mathbf{B}}$ of the background texture in terms of the object silhouette \mathbf{T} . By minimizing C in (2) wrt the intensity value $\mathbf{O}(x, y)$, we obtain the average of the pixels that correspond to the point (x, y) of the object. The estimate $\widehat{\mathbf{O}}$ of the moving object texture is then

$$\widehat{\mathbf{O}} = \mathbf{T} \frac{1}{F} \sum_{f=1}^F \mathcal{M}(\mathbf{q}_f) \mathbf{I}_f. \quad (3)$$

Minimizing the ML cost (2) wrt the intensity value $\mathbf{B}(x, y)$, we get the estimate $\widehat{\mathbf{B}}(x, y)$ as the average of the observed pixels that correspond to the pixel (x, y) :

$$\widehat{\mathbf{B}} = \frac{\sum_{f=1}^F \left[\mathbf{1} - \mathcal{M}(\mathbf{p}_f \mathbf{q}_f^\#) \mathbf{T} \right] \mathcal{M}(\mathbf{p}_f) \mathbf{I}_f}{\sum_{i=f}^F \left[\mathbf{1} - \mathcal{M}(\mathbf{p}_f \mathbf{q}_f^\#) \mathbf{T} \right] \mathcal{M}(\mathbf{p}_f) \mathbf{H}}. \quad (4)$$

The estimate $\widehat{\mathbf{B}}$ of the background texture in (4) is the average of the observations \mathbf{I}_f registered according to the background motion \mathbf{p}_i , in the regions $\{(x, y)\}$ not occluded by the moving object, *i.e.*, when $\mathcal{M}(\mathbf{p}_f \mathbf{q}_f^\#) \mathbf{T}(x, y) = 0$. The term $\mathcal{M}(\mathbf{p}_f) \mathbf{H}$ provides the correct averaging normalization in the denominator by accounting only for the pixels seen in the corresponding image.

We now replace the estimates $\widehat{\mathbf{O}}$ and $\widehat{\mathbf{B}}$, given by expressions (3,4), in the cost function (2), obtaining an expression for the ML cost function C in terms of a single unknown—the moving object silhouette \mathbf{T} , $C(\mathbf{T})$. This is a huge difference from the approach in [13,14], where only the estimate $\widehat{\mathbf{O}}$ is replaced in (2), leading to an expression for the ML cost function C in terms of \mathbf{B} and \mathbf{T} , *i.e.*, $C(\mathbf{B}, \mathbf{T})$. In [13,14], the ML cost is minimized by using a two-step iterative algorithm that computes, in alternate steps, the minimum of $C(\mathbf{B}, \mathbf{T})$ wrt \mathbf{B} for fixed \mathbf{T} , and the minimum of $C(\mathbf{B}, \mathbf{T})$ wrt \mathbf{T} for fixed \mathbf{B} . This last step requires that (the previous estimate of) the background texture \mathbf{B} is known at

all pixels, in particular it imposes that all background pixels occluded by the moving object are observed at least in one frame of the video sequence. Thus, the method of [13,14] does not deal with videos where the moving object only partially un-occludes the background, *i.e.* where some region of the background is occluded at all frames. In contrast, we propose to replace the expression of the background estimate $\hat{\mathbf{B}}$ in terms of the object silhouette \mathbf{T} into the ML cost C in (2), leading to an expression for $C(\mathbf{T})$ that is suitable to minimize wrt to the moving object silhouette \mathbf{T} alone.

Replacing the estimates of \mathbf{O} and \mathbf{B} , given by expressions (3) and (4), into the ML cost function (2), we get, after simple manipulations:

$$\begin{aligned}
 C(\mathbf{T}) = & \iint \sum_{f=1}^F \left\{ \mathbf{I}_f(x, y) \right. \\
 & - \frac{\sum_{i=1}^F \left[\mathbf{1} - \mathcal{M}(\mathbf{p}_f^\# \mathbf{p}_i \mathbf{q}_i^\#) \mathbf{T} \right] \mathcal{M}(\mathbf{p}_f^\# \mathbf{p}_i) \mathbf{I}_i}{\sum_{i=f}^F \left[\mathbf{1} - \mathcal{M}(\mathbf{p}_f^\# \mathbf{p}_i \mathbf{q}_i^\#) \mathbf{T} \right] \mathcal{M}(\mathbf{p}_f^\# \mathbf{p}_i) \mathbf{H}} \left[\mathbf{1} - \mathcal{M}(\mathbf{q}_f^\#) \mathbf{T}(x, y) \right] \\
 & \left. - \mathcal{M}(\mathbf{q}_f^\#) \mathbf{T} \frac{1}{F} \sum_{i=1}^F \mathcal{M}(\mathbf{q}_f^\# \mathbf{q}_i) \mathbf{I}_i \right\}^2 \mathbf{H}(x, y) dx dy. \tag{5}
 \end{aligned}$$

We minimize this resulting cost $C(\mathbf{T})$ wrt its only argument \mathbf{T} by using a greedy approach, in the spirit of several schemes that were successfully used to segment single images according to attributes like intensity, color, or texture, see the original energy minimization formulation of [16] and approaches that use variational methods [17], levels sets [18], partial differential equations [19], snakes [20,21], or active contours [22]. In our approach, given a previous estimate $\hat{\mathbf{T}}_n$ of the moving object silhouette, the algorithm updates the estimate by including in $\hat{\mathbf{T}}_{n+1}$ the neighboring pixels of $\hat{\mathbf{T}}_n$ that lead to a decrease of the cost C and excluding the neighboring pixels that lead to an increase of C .

4 Initialization: Motion Detection

To initialize the segmentation algorithm, we need an initial guess of the silhouette of the object. Our experience has shown that the algorithm converges to the correct solution even when the initial guess of the silhouette is very far from the optimal estimate, for example when the initial guess is a single pixel in the interior of the object. However, the impact of a computationally simple initialization algorithm is high because, as it always happens with iterative algorithms, the closer is the initial guess to the correct solution, the faster is the convergence.

We compute the initial guess by using motion detection. To improve over simply detecting the motion between two frames, we merge silhouettes computed from several pairs of frames. The following example illustrates the procedure.

4.1 Synthetic Sequence 1

We synthesize a video sequence using an object texture that contains regions of almost constant intensity, *i.e.*, regions with low texture. Fig. 1 represents three frames of that synthetic video that shows a static background and a moving car. Note that, due to the low textured regions of the car, motion segmentation is not trivial for this video sequence, as referred in section 1.



Fig. 1. Synthetic video sequence 1

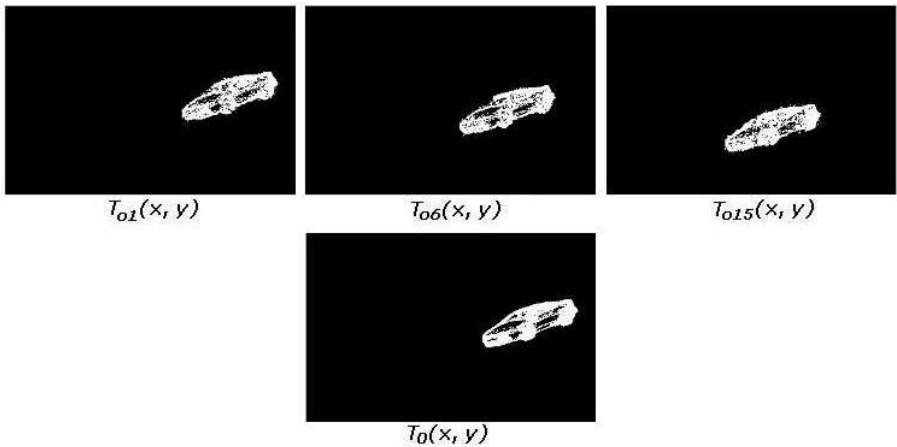


Fig. 2. Initial estimate of the silhouette of the moving car in the video in Fig. 1

In Fig. 2 we illustrate the initialization procedure for the first 20 frames of the video of Fig. 1. The top row contains pairwise estimates of the silhouette. These estimates are very incomplete due to the low texture of the car. The bottom image represents the initial guess of the silhouette obtained by merging the pairwise estimates. We see that this initial guess is more accurate than the pairwise estimates but it still misses a considerable number of pixels. Note that “filling-in” the regions that are missing in this initial guess by using spatial rules, *e.g.*, with morphological operations, is not trivial and requires the manual adaptation of several parameters in general dependent of the video sequence being processed.

5 Experiments

We report the results of our algorithm when segmenting two synthetic image sequences and one real video sequence.

In Fig. 3 we represent (left to right, top to bottom) the evolution of the estimate of the moving object silhouette, superimposed with its texture, for the synthetic video of Fig. 1. We see that, even for this low textured object, the estimate converges to the correct silhouette of the car. To better illustrate the behavior of the algorithm, we represent in Fig. 4, from left to right, the evolution of the estimate of the background texture. The left image of Fig. 4 shows the estimate at an early stage of the iterative process, *i.e.*, it shows an estimate that is blurred due to the still inaccurate estimate of the object silhouette. The right image of Fig. 4 demonstrates how the final estimate of the background texture is correct, *i.e.*, it is not blurred by the object texture. Note that, since in this

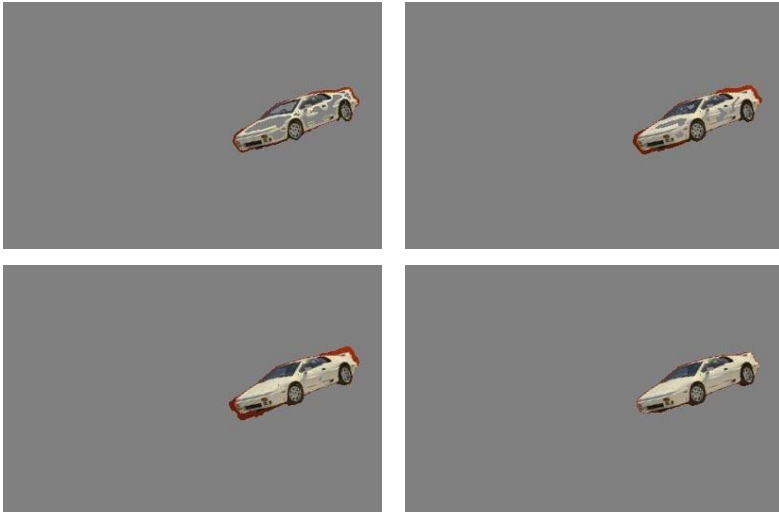


Fig. 3. Evolution of the estimate of the silhouette of the moving car for the video sequence in Fig. 1



Fig. 4. Evolution of the estimate of the background texture for the video sequence in Fig. 1



Fig. 5. Synthetic video sequence 2. Note that, since parts of the background are not seen at any frame (they are occluded by the moving object at all frames), the method of [13,14] can not be used to segment this video sequence.



Fig. 6. Background estimates for the video sequence in Fig. 5. The parts of the background that are not seen in any frame of the video sequence, are represented in black.

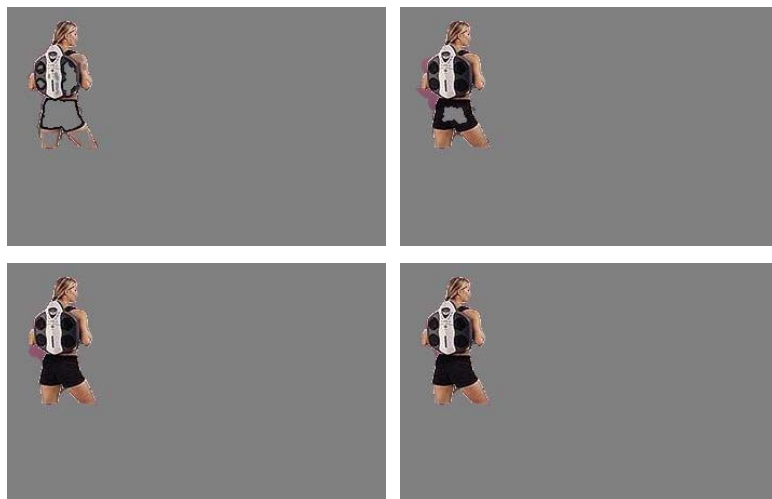


Fig. 7. Evolution of the estimate of the moving object silhouette for the video sequence in Fig. 5. In spite of the incomplete observation of the background, our method succeeded in segmenting accurately the moving object.

video clip the background was completely uncovered, the final estimate in the right image in Fig. 4 can be completely computed, *i.e.*, computed at all pixels.



Fig. 8. Real video sequence



Fig. 9. Evolution of the estimate of the background texture from the video in Fig 8



Fig. 10. Evolution of the estimate of the silhouette of the moving car from the video in Fig 8

5.1 Synthetic Sequence 2

We now synthesize a video sequence that shows a moving object with a more challenging shape. It also exhibits low textured regions. Fig. 5 shows three frames of this sequence. In this video the synthetic motion of the object is such that the background is not completely uncovered. The algorithm proposed in [13,14] to minimize the ML cost would then fail to segment this moving object. This is because the algorithm of [13,14] requires building a complete estimate of the background at intermediate steps, see discussion in section 1.

In Figs. 6 and 7 we represent the evolution of the estimates of the background texture and the moving object silhouette, respectively. Note that the background texture in the right image of Fig. 6 is not complete—we represent in black the pixels that, due to the occlusion by the moving object, were not observed in the video clip. As expected, our method is not affected by this covered background areas—we see from the bottom right image of Fig. 7 that our algorithm succeeded in accurately segmenting out the moving object in this video clip.

5.2 Real Video Sequence

We use a real video sequence that shows a moving car. Fig. 8 shows three frames from this video clip. Figs. 9 and 10 represent the evolution of the algorithm, demonstrating its good performance. See the evolution of the estimates of the background texture, in Fig. 9, and of the moving object silhouette, in Fig. 10.

6 Conclusion

We proposed a new algorithm to segment moving objects in video sequences. The algorithm exploits the *rigidity* of the object silhouette and the *occlusion* of the background by the moving object. Our experimental results illustrate the behavior of the algorithm and demonstrate its effectiveness.

References

1. Aguiar, P., Jasinschi, R., Moura, J., Pluempitiwiriyawej, C.: Content-based image sequence representation. In Reed, T., ed.: Digital Video Processing. CRC Press (2004) 7–72 Chapter 2.
2. Li, H., Lundmark, A., Forchheimer, R.: Image sequence coding at very low bitrates: A review. *IEEE Trans. on Image Processing* **3**(5) (1994)
3. Diehl, N.: Object-oriented motion estimation and segmentation in image sequences. *Signal Processing: Image Communication* **3**(1) (1991)
4. Jasinschi, R., Moura, J.: Content-based video sequence representation. In: Proc of IEEE Int. Conf. on Image Processing, Washington D.C., USA (1995)
5. Sawhney, H., Ayer, S.: Compact representations of videos through dominant and multiple motion estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **18**(8) (1996)

6. Jasinschi, R., Moura, J.: Generative Video: Very Low Bit Rate Video Compression. U.S. Patent and Trademark Office, S.N. 5,854,856 (1998)
7. Tao, H., Sawhney, H., Kumar, R.: Dynamic layer representation with applications to tracking. In: Prof. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina (2000)
8. Jovic, N., Frey, B.: Learning flexible sprites in video layers. In: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Hawaii (2001)
9. Dubuisson, M.P., Jain, A.: Contour extraction of moving objects in complex outdoor scenes. *Int. Journal of Computer Vision* **14**(1) (1995)
10. Bouthemy, P., François, E.: Motion segmentation and qualitative dynamic scene analysis from an image sequence. *Int. Journal of Computer Vision* **10**(2) (1993)
11. Irani, M., Peleg, S.: Motion analysis for image enhancement: Resolution, occlusion, and transparency. *Journal of Visual Communications and Image Representation* **4**(4) (1993) 324–335
12. Irani, M., Rousso, B., Peleg, S.: Computing occluding and transparent motions. *Int. Journal of Computer Vision* **12**(1) (1994)
13. Aguiar, P., Moura, J.: Maximum likelihood estimation of the template of a rigid moving object. In: Energy Minimization Methods in Computer Vision and Pattern Recognition. Springer–Verlag, LNCS 2134 (2001)
14. Aguiar, P., Moura, J.: Figure–ground segmentation from occlusion. *IEEE Trans. on Image Processing* **14**(8) (2005)
15. Bergen, J., et al.: Hierarchical model-based motion estimation. In: Proc of European Conf. on Computer Vision, Santa Margherita Ligure, Italy (1992)
16. Mumford, D., Shah, J.: Boundary detection by minimizing functionals. In: Prof. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, San Francisco, CA, USA (1985)
17. Morel, J., Solimini, S.: Variational Methods in Image Segmentation. Birkhäuser, Boston (1995)
18. Malladi, R., Sethian, J., Vemuri, B.: Shape modeling with front propagation: A level set approach. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **17**(2) (1995) 158–175
19. Sapiro, G.: Geometric Partial Differential Equations and Image Analysis. Cambridge University Press (2001)
20. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. *Int. Journal of Computer Vision* **1**(4) (1988) 321–331
21. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic snakes. *Int. Journal of Computer Vision* **22** (1997) 61–79
22. Chan, T., Vese, L.: Active contours without edges. *IEEE Trans. on Image Processing* **10**(2) (2001) 266–277

Towards Constrained Optimal 3D Tracking

Huiying Chen and Youfu Li

Department of Manufacturing Engineering and Engineering Management,
City University of Hong Kong, Kowloon, Hong Kong
meyfli@cityu.edu.hk

Abstract. In this paper, we present a method for optimal 3-dimensional tracking using interaction matrix. We developed two constraints to avoid the singularity and local minima of the interaction matrix. These constraints were combined with the minimization of tracking errors to identify suitable system configuration for tracking. Experiments were conducted to validate the method.

1 Introduction

Visual tracking is essential for many applications and it has received considerable attention in recent research. Depending on whether the depth information is recovered or not, visual tracking can be divided into 3-dimensional (3D) tracking and 2-dimensional (2D) tracking. In general, the objective of visual tracking is to continuously estimate and update the positions and orientations of the target [1]. The positions and orientations here refer to the relative positions and orientations between the object and the vision system. The direct linkage between the variation of the relative position and orientation and the variation of feature on the image plane is governed by a so-called interaction matrix [2].

The interaction matrix describes the projection of the 3D velocity field onto the image motion field. The interaction matrix has been extensively studied in visual servoing [3, 4]. However, as the interaction matrix suffers from singularity and local minimum, it has mostly been computed at the “equilibrium”, which is a possible solution to avoid singularities [5]. This calls for depth estimation. Recently, research on singularity of the interaction matrix has been conducted on singularity problems in visual servoing [6]. Michel and Rives [7] studied to find a particular visual feature set with 3 points where the interaction matrix had neither local minima nor singularities. Malis et al. [8] developed a triangular interaction matrix that had no singularity in the whole task space. Nevertheless, only little attention, if any, has been devoted to the singularity problem in visual tracking. Also, nonsingularity has rarely been explored in visual tracking as a constraint.

Singularity problem in visual servoing has been studied, for example in an eye-in-hand system [6]. The purpose is to ensure that the control law for the camera motion is implementable. Generally, for a monocular eye-in-hand system, at least 3 visual feature points are needed to obtain the interaction matrix. In this case, the points’ configuration is modified to avoid singularities [7]. This will be different in visual tracking where the problem is whether the object location can be reliably recovered by

the image features or not. Furthermore, instead of a set of feature points, we study the singularity of every feature point for 3D tracking.

In this paper, nonsingular constraints of the interaction matrix are defined for 3D tracking, to improve tracking performance. Also, an adaptive search window and motion prediction scheme is implemented.

2 3D Tracking Via Interaction Matrix

2.1 Visual Feature and Relative Motion

Let $\mathbf{p} = (u, v)^T$ be a vector describing the current visual feature, u and v the image coordinates of the feature. According to the definition of the interaction matrix, we have

$$\dot{\mathbf{p}} = \mathbf{L}_s \mathbf{T}_{re}, \quad (1)$$

where \mathbf{L}_s is the *interaction matrix*. $\mathbf{T}_{re} = \begin{pmatrix} \mathbf{V}(t) \\ \boldsymbol{\Omega}(t) \end{pmatrix}$ is the relative motion between the object and the camera with $\mathbf{V} = (v_x, v_y, v_z)^T$, which is the translational velocity, and $\boldsymbol{\Omega} = (\omega_x, \omega_y, \omega_z)^T$, which is the angular velocity. $\dot{\mathbf{p}}$ is the time derivative of image feature due to the object relative motion \mathbf{T}_{re} , $\dot{\mathbf{p}} = (\dot{u}, \dot{v})^T$. Therefore, the interaction matrix \mathbf{L}_s is a 2×6 matrix.

An object's position can be obtained by calculating the relative motion. Equation (1) yields

$$\mathbf{T}_{re} = \mathbf{L}_s^+ \dot{\mathbf{p}}, \quad (2)$$

where \mathbf{L}_s^+ is the pseudo-inverse matrix of \mathbf{L}_s . \mathbf{L}_s^+ can be obtained by the following calculation

$$\mathbf{L}_s^+ = \mathbf{L}_s^T (\mathbf{L}_s \mathbf{L}_s^T)^{-1}. \quad (3)$$

2.2 3D Tracking with Stereo Vision System

From the elementary stereo geometry in a canonical configuration, in which two stereo cameras have the same focal length and their optical axes are parallel with each other, assuming that a point $\mathbf{P} = (x, y, z)^T$ in the right camera frame is projected onto the right image plane as a point $\mathbf{p}_r = (u_r, v_r)^T$ and a point $\mathbf{p}_l = (u_l, v_l)^T$ on the left image plane, we have

$$z = \frac{Bf}{u_r - u_l}, \quad (4)$$

where B is the baseline of the stereo system, and f represents the focal length.

According to [4], the interaction matrix can be written in the following form as a z 's function

$$\mathbf{L}_s = \begin{bmatrix} -\frac{f}{z} & 0 & \frac{u_r}{z} & \frac{u_r v_r}{f} & -\frac{f^2 + u_r^2}{f} & v_r \\ 0 & -\frac{f}{z} & \frac{v_r}{z} & \frac{f^2 + v_r^2}{f} & -\frac{u_r v_r}{f} & -u_r \end{bmatrix}. \tag{5}$$

3 Nonsingular Constraints in Interaction Matrix Using

3.1 The Smallest Singular Value

The singular value decomposition of an interaction matrix \mathbf{L}_s is expressed as

$$\mathbf{L}_{s(2 \times 6)} = \mathbf{U}_{\Sigma} \mathbf{\Sigma} \mathbf{V}_{\Sigma}^T = \mathbf{U}_{\Sigma(2 \times 2)} \underbrace{\begin{pmatrix} \sigma_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 & 0 & 0 \end{pmatrix}}_{\Sigma} \mathbf{V}_{\Sigma(6 \times 6)}^T. \tag{6}$$

We always define that $\sigma_1 \geq \sigma_2 \geq 0$. If the interaction matrix is singular, which means that at least one of its singular values (σ_1 and σ_2) is equal to zero, the vision system will lose the projection information in certain direction(s). This will then cause 3D tracking failure. Therefore we impose a constraint on the smallest singular value (σ_2 here) on 3D tracking. The constraint is defined as

$$G_1 : \{ \sigma_2 \in \sigma \mid \sigma - \xi > 0 \}, \tag{7}$$

where ξ is a positive lower bound.

As the smallest singular value cannot have an analytic solution, we used simulations to identify its property. Simulation results shown that focal length of the vision system, as well as image feature location on the scene can affect the smallest singular value (σ_2). As shown in Fig. 1(a), the mean values of σ_2 are plotted for different

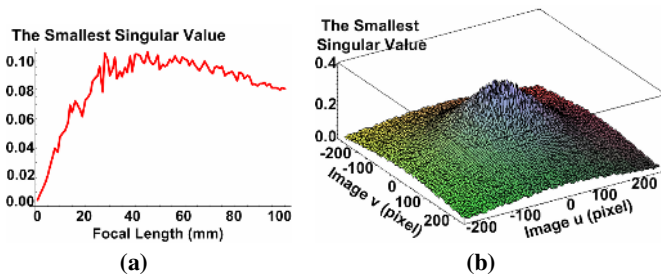


Fig. 1. The smallest singular value as a function of f (focal length), (b) r (image feature location)

values of focal length. According to the result, there is a global maximum in σ_2 , which corresponds to the best nonsingular condition of interaction matrix. Then the nonsingular constraint on focal length is

$$G_{1f} : \{f_{\min} \leq f < f_{\max}\}, \tag{8}$$

where f_{\min} and f_{\max} are the lower and upper bound of focal length value corresponding to ξ .

Simulation result in Fig. 1(b) shows that features located near to the image center provide larger singular values. Features located far away from the image center tend to give smaller singular values, which may cause singularity problem. Therefore, the nonsingular constraint on image feature location is introduced

$$G_{1r} : \{r < r_{\max}\}, \tag{9}$$

where r is the distance of the image feature point to the image centre, and r_{\max} is the maximum distance with respect to ξ .

3.2 Condition Number

The condition number $\kappa(\mathbf{L}_s)$ is defined as a ratio of the largest to smallest singular value, $\kappa(\mathbf{L}_s) = \sigma_1 / \sigma_2$. A system is said to be singular if the condition number is infinite. $\kappa(\mathbf{L}_s)$ can be calculated as follows

$$\kappa(\mathbf{L}_s) = \frac{\|\mathbf{L}_s\|}{\|\mathbf{L}_s^+\|}. \tag{10}$$

Substituting (5) into (3), we can calculate the pseudo-inverse matrix of the interaction matrix and then obtain

$$\mathbf{L}_s^+ = \eta \begin{bmatrix} -fz(v_r^2 + f^2) & fzv_r u_r \\ fzv_r u_r & -fz(u_r^2 + f^2) \\ f^2 z u_r & f^2 z v_r \\ 0 & z^2 f(v_r^2 + u_r^2 + f^2) \\ -z^2 f(v_r^2 + u_r^2 + f^2) & 0 \\ z^2 v_r(v_r^2 + u_r^2 + f^2) & -z^2 u_r(v_r^2 + u_r^2 + f^2) \end{bmatrix}, \tag{11}$$

where $\eta = 1/[f^2(v_r^2 + u_r^2 + f^2)(f^2 + z^2v_r^2 + f^2z^2 + u_r^2z^2)]$.

Substituting (11) into (10), after reduction, we can obtain a very simple form of the condition number as follows

$$\kappa(\mathbf{L}_s) = \frac{u_r^2 + v_r^2 + f^2}{f^2} = \frac{r^2 + f^2}{f^2}. \tag{12}$$

Equation (12) indicates that the condition number of the interaction matrix is only affected by focal length and the location of the image feature point.

Consequently, the nonsingular constraint on the condition number can be defined as

$$G_2 : \left\{ \frac{r^2 + f^2}{f^2} = 1 + \frac{r^2}{f^2} < \kappa_0 \right\}, \quad (13)$$

where κ_0 is a threshold for the condition number. Further, G_2 can be decomposed into two constraints for f and r as

$$G_{2f} : \{f > f_{\min} |_{\kappa_0, r}\}, \quad (14)$$

$$G_{2r} : \{r < r_{\max} |_{\kappa_0, f}\}. \quad (15)$$

4 Error Analysis

4.1 3D Tracking Error Modeling

We denote the observed feature motion with errors as \mathbf{p}' . We have

$$\mathbf{p}' = \mathbf{p} + \Delta\mathbf{p}, \quad (16)$$

where $\Delta\mathbf{p}$ is the feature motion error, $\Delta\mathbf{p} = (\Delta\dot{u}, \Delta\dot{v})^T$. Without loss of generality, let \mathbf{Q} be a 2×2 scale matrix so that $(\Delta\dot{u} \ \Delta\dot{v})^T = \mathbf{Q} (\dot{u} \ \dot{v})^T$. When $\Delta\mathbf{p}$ is projected to the object motion field, from (2), we have the estimated object motion error as follows

$$\Delta\mathbf{T}_{re} = \mathbf{L}_s^+ \Delta\mathbf{p} + \Delta\mathbf{L}_s^+ \mathbf{p} = (\mathbf{L}_s^+ \mathbf{Q} + \Delta\mathbf{L}_s^+) \mathbf{p}, \quad (17)$$

where $\Delta\mathbf{T}_{re} = \begin{pmatrix} \Delta\mathbf{V}(t) \\ \Delta\mathbf{\Omega}(t) \end{pmatrix}$ and $\Delta\mathbf{L}_s^+$ is the error of \mathbf{L}_s^+ .

If quantization error is considered, then we have

$$\Delta\mathbf{L}_s^+ = \frac{\partial\mathbf{L}_s^+}{\partial z_s} \Delta z + \frac{\partial\mathbf{L}_s^+}{\partial u_r} \Delta u_r + \frac{\partial\mathbf{L}_s^+}{\partial v_r} \Delta v_r, \quad (18)$$

where Δz is the depth estimation error, Δu_r and Δv_r are the quantization errors along u and v image coordinates of the right camera respectively. We assume that Δu_r and Δv_r are independent of each other.

According to the properties of error transition, (4) leads to

$$\Delta z = \frac{-Bf}{(u_r - u_l)^2} \Delta(u_r - u_l), \quad (19)$$

where $\Delta(u_r - u_l)$ is the disparity error. Because Δu_r and Δu_l are independent of each other, if we assume that the two cameras have the same quantization error, then according to the error transition rule we have $\Delta(u_r - u_l) = 2\Delta u_r$. Equation (19) yields

$$\Delta z = \frac{-2Bf}{(u_r - u_l)^2} \Delta u_r = \frac{-2z^2}{Bf} \Delta u_r. \tag{20}$$

Thus (18) becomes

$$\Delta \mathbf{L}_s^+ = \frac{\partial \mathbf{L}_s^+}{\partial z} \Delta z + \frac{\partial \mathbf{L}_s^+}{\partial u_r} \Delta u_r + \frac{\partial \mathbf{L}_s^+}{\partial v_r} \Delta v_r. \tag{21}$$

We denote the two partial matrices of \mathbf{L}_s^+ as \mathbf{L}_{s1}^+ and \mathbf{L}_{s2}^+ , and those of $\Delta \mathbf{L}_s^+$ as $\Delta \mathbf{L}_{s1}^+$ and $\Delta \mathbf{L}_{s2}^+$, namely

$$\mathbf{L}_{s(6 \times 2)}^+ = \begin{pmatrix} \mathbf{L}_{s1(3 \times 2)}^+ \\ \mathbf{L}_{s2(3 \times 2)}^+ \end{pmatrix}, \quad \Delta \mathbf{L}_{s(6 \times 2)}^+ = \begin{pmatrix} \Delta \mathbf{L}_{s1(3 \times 2)}^+ \\ \Delta \mathbf{L}_{s2(3 \times 2)}^+ \end{pmatrix}. \tag{22}$$

Following the denotation of (2) and (17), we have

$$\Delta \mathbf{V} = \mathbf{L}_{s1}^+ \Delta \dot{\mathbf{p}} + \Delta \mathbf{L}_{s1}^+ \dot{\mathbf{p}} = (\mathbf{L}_{s1}^+ \mathbf{Q} + \Delta \mathbf{L}_{s1}^+) \dot{\mathbf{p}}, \tag{23}$$

$$\Delta \mathbf{\Omega} = \mathbf{L}_{s2}^+ \Delta \dot{\mathbf{p}} + \Delta \mathbf{L}_{s2}^+ \dot{\mathbf{p}} = (\mathbf{L}_{s2}^+ \mathbf{Q} + \Delta \mathbf{L}_{s2}^+) \dot{\mathbf{p}}. \tag{24}$$

Then the estimated motion error in 3D object space is

$$\Delta \dot{\mathbf{P}} = (\Delta \dot{P}_x, \Delta \dot{P}_y, \Delta \dot{P}_z)^T = [\mathbf{P}_\times] \Delta \mathbf{\Omega} + \Delta \mathbf{V}, \tag{25}$$

where $[\mathbf{P}_\times]$ is a skew-symmetric matrix.

Consequently, the positioning error of tracking is

$$\varepsilon_p = \|\Delta \mathbf{P}\| \doteq \|\Delta \dot{\mathbf{P}}\| \delta t = \sqrt{\Delta \dot{P}_x^2 + \Delta \dot{P}_y^2 + \Delta \dot{P}_z^2} \delta t, \tag{26}$$

where δt is the sampling period of tracking.

4.2 Influence of System Configuration on Tracking Errors

Tracking error \mathcal{E}_p can be used as a cost function of the tracking system so that the parameters for optimal tracking can be obtained by minimization of \mathcal{E}_p .

According to our simulation study, both the baseline value B and focal length value f can affect the tracking error. As shown in Fig. 2, the tracking error is approximately inversely proportional to B and f . This result tallies with what is observed in the real camera setup. According to the projection rule, a large focal length f can provide better sensing in a wide range. As to the baseline value B , following

the triangulation rule, the larger the baseline, the more sensitive the depth estimation is. Thus a larger B value results in smaller tracking errors.

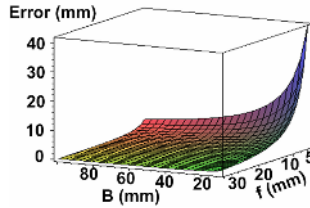


Fig. 2. Tracking error \mathcal{E}_p affected by B and f

5 Combined Constraints for Image Feature Location

When the vision system is reconfigurable, we can keep the tracking feature point targeted at a specific position (u^*, v^*) on the image to minimize the tracking error and improve tracking performance.

Assume that the image feature velocities at certain sampling instant are \dot{u}_i, \dot{v}_i . Let $\alpha_i = \dot{u}_i / \dot{v}_i$, where α_i is a scale factor. We have examined how different image feature locations affect the tracking error. As shown in Fig. 3, the distribution of tracking error on the image plane is central symmetric, and it converges to two global minima along a line $u = \alpha_i v$ on the edge of the image plane.

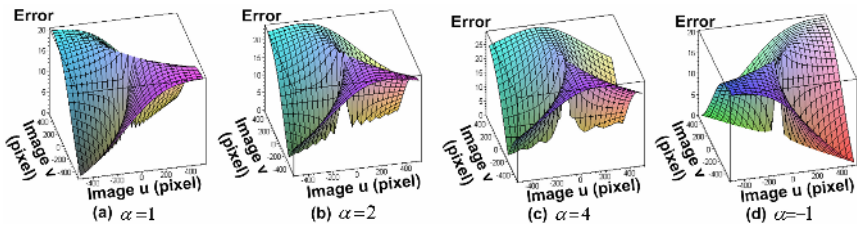


Fig. 3. Tracking error converges on line $u = \alpha v$ at different α

Fig. 3 also shows how tracking errors change due to different values of α_i . According to the discussion in Section 3, in order to satisfy nonsingular constraints, image feature points should be located close to the image centre. However, to minimize tracking error, image feature points should be located away from the image centre on the line $u = \alpha_i v$. Therefore, an optimal targeting position can be achieved by making a compromise between the satisfaction of nonsingular constraints and the minimization of the tracking error. Fig. 4 shows the area of those positions.

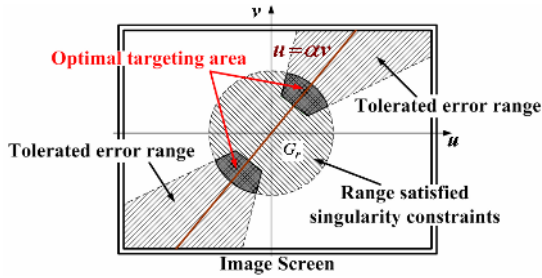


Fig. 4. The optimal targeting area on the image screen

6 Experimental Results

6.1 System Set-Up

The implementation of the proposed tracking method was conducted using our vision system, with a PC-based IM-PCI system and a variable scan frame grabber. Our algorithms were developed in VC++ programming language and run as imported functions by ITEX-CM. The system setup consists of two identical cameras (model TM-765), with a resolution of 768×582 pixels.

6.2 Tracking Implementation

We defined the tracking error as the distance between the estimated object location and the true object location at each sampling instant. In our experiment, initial tests with traditional tracking method (the segmentation-based method) showed that tracking errors tended to increase when the object moved faster. Therefore, we have developed a tracking method based on the use of adaptive search windows whose sizes are in proportional to the object velocity. This method enables us to efficiently and reliably search for a target with changing velocities within a small area of the predicted window. Assuming that the feature location \mathbf{p}_t at time t on the right camera image plane, the relative motion $\mathbf{T}_{re}(t)$, and the inter-frame time τ are given, then predicted location (centre location) of the search window can then be estimated as

$$\mathbf{p}_{t+\tau} = \mathbf{p}_t + \int_t^{t+\tau} \dot{\mathbf{p}}_t dt \doteq \mathbf{p}_t + \dot{\mathbf{p}}_t \tau = \mathbf{p}_t + \tau \mathbf{L}_s(t) \mathbf{T}_{re}(t). \tag{27}$$

The change of the window size is defined as proportional to the product of the image velocity and the tracking error as follows

$$\mathbf{s}_w(t) = \begin{pmatrix} \delta_u(t) \\ \delta_v(t) \end{pmatrix} = \begin{pmatrix} \delta_{u0} \\ \delta_{v0} \end{pmatrix} + \lambda \varepsilon_p(t) \begin{pmatrix} |\dot{u}(t)| \\ |\dot{v}(t)| \end{pmatrix}, \tag{28}$$

where λ is a positive scale factor, $(\delta_u(t), \delta_v(t))^T$ are the window sizes along u and v image coordinates respectively, and $(\delta_{u0}, \delta_{v0})^T$ defines the minimum window size. Similarly, the adaptive search window for the left camera can be derived.

6.3 Tracking in 3D

Firstly, we implemented our method to track a ball. The location of the search window for the next step was predicted using the position and velocity information currently available. The size of the search window was changed according to the predicted tracking error and the object velocity. Some examples of snap shots in the tracking are shown in Fig. 5(a). In another experiment, we implemented our method in tracking a human hand as shown in Fig. 5(b). In this case, the 3D orientation as well as the 3D position of the target has to be considered in the tracking. A tracking rate of about 10fps was achieved in the implementation.

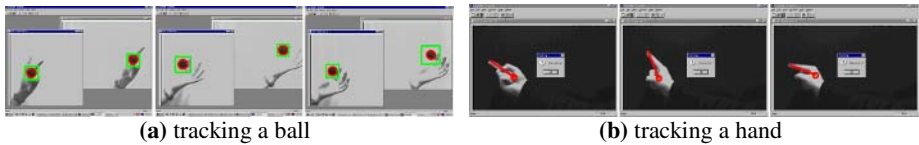


Fig. 5. Stereo tracking by our method

6.4 Nonsingular Constraints for Tracking

Two parameters can affect singularity: feature location and focal length. Since singularity can further affect tracking performance, we examined the influence of those parameters on tracking errors and verified their effects on singularity.

Image Feature Location. As discussed in Section 5, feature location itself can also affect tracking errors. Thus, a special motion pattern is adopted to eliminate such influence. We made the tracking object undergo uniform circular motion with different diameters. Using this motion, at any tracking point of the circumference, the magnitude of velocity is uniform and the direction of velocity is perpendicular to the line from the object to the centre. Thus, if the influence of singularity is not considered, then according to Fig. 3 the object on the same circumference will have the same effect on tracking errors. Furthermore, to eliminate the influence of velocity on tracking error the object was made to move along different circumferences at the same magnitude of velocity. This made it possible to extract the relatively “pure” influence of singularity on tracking errors.

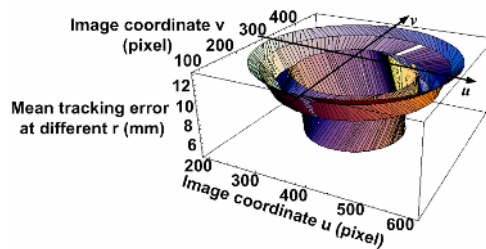


Fig. 6. The absolute tracking error as a function of image feature location

Then we analyzed the obtained result. The mean tracking errors on different concentric circles were calculated and they are shown in Fig. 6. The absolute tracking errors are affected by image feature location (r). Note that the errors caused by lens distortion may contribute to the tracking errors here. Therefore, we conducted separate experimental examinations on the static errors which include the effect of the lens distortion in the vision system. With our camera calibration, the mean static errors were found to be within 1.0mm for the setup. Thus, the influence of the lens distortion on the tracking errors can be ignored. This experimental result is consistent with the simulation result shown in Fig. 1(b) and it verified the singularity effect on tracking error. When the image feature point is close to the image centre, the smallest singular value becomes large so that better tracking performance can be achieved.

Focal Length. The mean tracking errors with their variances at different focal length values are plotted in Fig. 7.

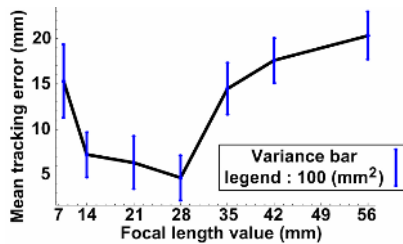


Fig. 7. Tracking error affected by f

Fig. 7 shows that there is an optimal focal length value which minimizes the tracking error. This is consistent with the result of Fig. 1(a) in Section 3, indicating that there is a global minimum in the smallest singular value at which its corresponding focal length best satisfied the nonsingular constraint G_{1f} .

However, the real situation is far more complicated. On one hand, according to (14) in Section 3, focal length affects the condition number so that a large focal length can provide small condition number and better nonsingularity. Thus, a large focal length is desired to reduce the tracking errors due to singularity. On the other hand, the focal length itself can directly affect tracking errors (see Fig. 2 in Section 4). When the focal length is too large or too small ($f < 10\text{mm}$ or $f > 35\text{mm}$ in this case), the blurring effects of target will affect the precision of tracking adversely. The result in Fig. 7 therefore is considered as a combination of the above effects.

6.5 Optimal Targeting Area

We made a point object undergo a same uniform straight-line motion along different parallel lines $u_i = -v_i + b_i$, with $i = 1 \dots 9$ and $b_3 = 0$ (see Fig. 8(a)). The direction of velocity of the object was restricted in $\dot{u} = -1 \cdot \dot{v}$. Then according to the simulation result of Fig. 3, the tracking error expected to converge on the line $u = -v$ (L5 in Fig. 8(a)).

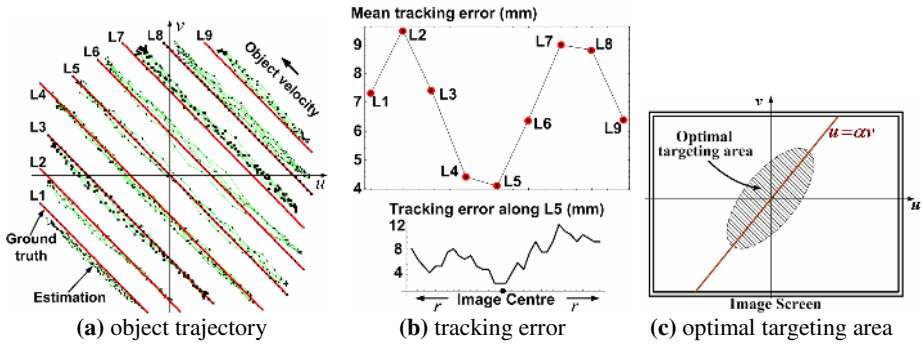


Fig. 8. Tracking error study on different parallel lines and optimal targeting area

The object moved along each of the parallel lines repeatedly and the average tracking errors were obtained at each sampling. The mean tracking errors on different lines are shown in Fig. 8(b). The smallest mean tracking error appears on **L5**. In other words, the tracking errors converge along the line $u = -v$. It should be noted that in Fig. 3, there is a global maximum in the tracking error at the image centre. However, in that simulation, the nonsingular constraint was not taken into account. In experiments, the influence of singularity will play a more important role so that around the central area, better performance can be achieved for tracking (see Fig. 8(b)). Ultimately, the optimal targeting area can be modified as shown in Fig. 8(c).

7 Conclusion

In this paper, we have developed an optimal 3D tracking method using nonsingular constraints from the interaction matrix. Influence of the system configuration on the interaction matrix has been studied and constraints have been designed to avoid singularities of the interaction matrix. Also, we have examined the system parameters to achieve better tracking performance. Experimental results verified the effectiveness of the proposed method.

Acknowledgment

This work was fully supported by a grant from the Research Grants Council of Hong Kong [Project No. CityU117605].

References

1. K. Nickels and S. Hutchinson, "Model-Based Tracking of Complex Articulated Objects," *IEEE Transactions on Robotics and Automation*, Vol. 17, No. 1, pp. 28-36, Feb. 2001.
2. C. Collewet and F. Chaumette, "Positioning a Camera with Respect to Planar Objects of Unknown Shape by Coupling 2-D Visual Servoing and 3-D Estimations," *IEEE Transactions on Robotics and Automation*, Vol. 18, No. 3, pp. 322-333, June 2002.

3. Y. Mezouar, H. H. Abdelkader, P. Martinet, and F. Chaumette, "Central Catadioptric Visual Servoing From 3D Straight Lines," *Proc. of the 2004 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pp. 343-348, Seidai, Japan, 2004.
4. B. Espiau, F. Chaumette, and P. Rives, "A New Approach to Visual Servoing in Robotics," *IEEE Trans. on Robotics and Automation*, Vol. 8, No. 6, pp. 313-326, Jun. 1992.
5. E. Cervera et al., "Improving Image-Based Visual Servoing with Three-Dimensional Features," *The International Journal of Robotics Research*, Vol. 22, No. 10-11, pp. 821-839, 2003.
6. F. Chaumette, "Potential Problems of Stability and Covergence in Image-Based and Position-Based Visual Servoing," *The Confluence of Vision and Control*, LNCIS 237, Springer-Verlag, pp. 66-78, 1998.
7. H. Michel and P. Rives, "Singularities in the Determination of the Situation of a Robot Effector from the Perspective View of Three Points," *Technical Report 1850*, INRIA, Feb. 1993.
8. E. Malis, F. Chaumette, and S. Boudet, "2-1/2-D Visual Servoing," *IEEE Transactions on Robotics and Automation*, Vol. 15, No. 2, pp. 238-250, Apr. 1999.

A Generic Approach to Object Matching and Tracking

Xiaokun Li¹, Chiman Kwan¹, Gang Mei¹,
and Baoxin Li²

¹ Signal/Image Processing Group, Intelligent Automation Inc.,
Rockville, MD 20855, USA

² Computer Science and Engineering Dept., Arizona State University,
Tempe, AZ 85287, USA

Abstract. In this paper, a generic approach to object matching and fast tracking in video and image sequence is presented. The approach first uses Gabor filters to extract flexible and reliable features as the basis of object matching and tracking. Then, a modified Elastic Graph Matching method is proposed for accurate object matching. A novel method based on posterior probability density estimation through sequential Monte Carlo method, called as Sequential Importance Sampling (SIS) method, is also developed to track multiple objects simultaneously. Several applications of our proposed approach are given for performance evaluation, which includes moving target tracking, stereo (3D) imaging, and camera stabilization. The experimental results demonstrated the efficacy of the approach which can also be applied to many other military and civilian applications, such as moving target verification and tracking, visual surveillance of public transportation, country border control, battlefield inspection and analysis, etc.

Keywords: Image analysis, feature extraction, object matching, real-time tracking.

1 Introduction

In image analysis and recognition, automatically recognizing objects of interest is always a challenging problem, and has been a research topic for many years. In recent years, detecting and tracking moving object in video is becoming a more interesting research topic and alluring more research efforts. In low level computer vision, one fundamental problem in object recognition and tracking is feature extraction as the result of extraction will directly affect the recognition performance. Another tough problem in object recognition is the matching between target and template. One reason for these difficulties is that, in real world, the object of interest always has some orientation difference and shape deformation as compared to its template in database. The goal of this paper is to develop an efficient method for object recognition and verification. The proposed method is based on Gabor filter-based Elastic Graph Matching (EGM) which has been successfully used in image texture analysis, face and fingerprint recognition [1-5]. But, by applying a new template-based matching method as the initialization of EGM, which is invariant to object rotation and size, we can overcome the limitations of conventional EGM and extend its applicability to more general cases such as stereo imaging, object tracking, and image sequence stabilization.

Another important issue discussed in this paper is object tracking. In video/image analysis, object tracking often becomes a more desirable problem after recognition. An automatic algorithm is needed to answer the questions: what is trace of the detected object? Or, is the object in the current frame the one I am looking for? Once the object is detected, people usually want to know its status and position in the subsequent frames. In the real world, the object of interest is moving in 3D space, meaning the features of the object, which are projected onto 2D image, are also changing along the temporal axis. This makes object tracking a very challenging problem. Even with the difficulties mentioned above, many new methods and exciting results have been obtained in recent years, e.g. [6-12]. Unlike the current methods, we propose to use Sequential Importance Sampling (SIS) method to track moving object in real-time, which has several important advantages in object tracking. First, SIS is based on posterior probability density estimation through sequential Monte Carlo (MC) method. The samples used for tracking are weighted properly via MC and updated with current observation while keeping track of a slowly varying change. Second, with SIS, tracking can be completed simultaneously by using the estimated posterior density.

In this paper, a generic approach for object matching and tracking is presented. The approach consists of three steps. The first step is Gabor filter-based feature extraction which provides an efficient way for selecting object features. The second step is an improved Elastic Graph Matching for object matching. The last step is a novel approach to simultaneously tracking multiple object in video/image sequence. Our method is based on posterior probability density estimation through sequential Monte Carlo methods.

The paper is organized as follows. Section 2 gives out the technical details on how object feature extraction is formulated with Gabor filter. Section 3 describes our approach on using EGM for object matching. In Section 4, one efficient solution for fast tracking is presented. Section 5 provides some experimental results both in video and image sequence.

2 Gabor Filter-Based Feature Extraction

In human visual system (HSV), research has shown that people are sensitive to both specific orientation and spatial frequencies of object of interest. For feature representation and extraction, wavelets are good at representing orientation and frequency characteristic of object of interest. A Gabor filter bank can act as a simple form of wavelet filter bank. Because of its simplicity and optimum joint spatial/spatial-frequency localization, Gabor filter has attracted many research efforts [4-5, 13-19] and has been applied in many image analysis and computer vision-based applications, e.g. face and fingerprint analysis and recognition [1-3].

Gabor filter bank is a group of 2-D filters which record the optimal jointed localization properties of region of interest in both spatial and spectral domain. Typically, an image is filtered with a set of Gabor filters which have different or preferred orientations and spatial frequencies. To be specific, an image $I(\vec{x})$ is filtered with a set of Gabor wavelets as follows,

$$(wI)(\bar{k}, \bar{x}_0) = \int \phi_{\bar{k}}(\bar{x}_0 - \bar{x})I(\bar{x})d\bar{x} \tag{1}$$

where $\phi_{\bar{k}}$ is the Gabor wavelet (filter) defined by

$$\phi_{\bar{k}}(\bar{x}) = \frac{\bar{k}}{\sigma^2} \exp(-\frac{\bar{k}^2 \bar{x}^2}{2\sigma^2}) [\exp(i\bar{k}\bar{x}) - \exp(-\frac{\sigma^2}{2})] \tag{2}$$

with $\bar{k} = k_v e^{i\phi_\mu}$ controlling the orientation and the scale of the filters. By varying v and μ , we can get different Gabor filters with different orientations and scales. In our implementation, μ controls the orientation and is assigned by any value of 0, 1, 2, to 7 and v controls the spatial frequency and is assigned from 0, 1, and 2 with $k_v = (\pi/2)/\sqrt{2^v}$ and $\phi_\mu = (\mu\pi)/8$. After filtering with a set of Gabor filters (24 filters from the above choice of v and μ), the outputs on each pixel in the image form a 24-dimensional vector called ‘‘jet’’. The amplitude of the jet represents whether a pixel has significant gradient value in both orientation and frequency. Thus, it can be used to determine if this pixel is a good feature for object matching and tracking.

3 Matching

In order to correspond two images from two different sensors, called as image level matching, or find the correspondence from target to template of reference image/database for object recognition and verification, called as object level matching, we have to solve the feature correspondence (matching) problem. With the feature points detected in the previous section, we propose to use an improved Elastic Graph Matching method to solve the matching by finding the corresponding features in the target frames. Some more detailed description of EGM can be found in [5, 20]. In most cases, due to the possible arbitrary relative positioning of the sensors with different field of view (FOV), conventional EGM method may never converge to the correct position because of the position, orientation, and scale difference between target and template, and thus we propose a coarse-to-fine strategy to perform robust matching. We first roughly match the two images (target image and template image) or find the object of interest in target image by searching with template, and then use EGM method to tune the matching result. The template matching with unknown rotation and size can be formulated using a non-orthogonal image expansion approach [21]. In this method, image or object of interest will be recovered by a delta function at the template location. The convolution equation can be expressed as:

$$g(r) = f(r; \theta^0) * \delta(r - \bar{r}^0) + n(r) \tag{3}$$

where the position vector is $r^T = [r_x, r_y]$, $*$ is 2-D convolution, and f is the template (image or object of interest) at \bar{r}^0 . The orientation and size differences between target and template are represented by a vector θ (where θ^0 is the true parameter set).

$$\theta^T = [s, \phi] \tag{4}$$

where s is the size and ϕ is the rotation, a rotated and resized template can be given as

$$f(r; \theta) = f\left(\frac{1}{s}M(\phi)r\right) \tag{5}$$

where $M(\phi) = \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix}$. In this coarse step, maximum likelihood (ML) can be used to estimate the parameter set θ and use delta restoration method [22] for location estimation \hat{r} . The cost function of ML can be described as

$$l(\theta, r | g) = \|f(r; \theta) * \delta(r - \bar{r}) - g(r)\|^2 \tag{6}$$

The maximum likelihood solution is then obtained by minimizing Eq. (6) as

$$\{\hat{r}, \hat{\theta}\} = \operatorname{argmin}l(\theta, r | g) \tag{7}$$

To solve the optimization problem, a Linear Least Square Estimate (LLSE) of the delta function can be considered to use. More details can be found in [22].

Even with the method mentioned above, two images or two objects may never be able to correlate with each other exactly due to local structural and depth variations. However, this is addressed naturally by the elasticity of the matching graph in the algorithm. In this paper, we present one improved EGM method which uses Gabor jets as inputs. Its main steps are given as follows:

Algorithm: Enhanced Elastic Graph Matching (EGM)

Step 1: Find approximate position: We use the novel template matching with unknown rotation and size parameter to identify the initial correspondence/matching between target and template of reference image/database. From the correspondences, some corresponding pairs of pixels from target and template are selected as features whose magnitudes of the jets are obviously larger than that of other pixels.

Step 2: Verify position: We first average the magnitudes of the jets of each feature point. The jets to each pixel are termed as “bunch”. Then, we assign the average value to the processed bunch and compute the similarity function S_a without phase comparison.

$$S_a(J, J') = \frac{\sum_j a_j a'_j}{\sqrt{\sum_j a_j^2 \sum_j a_j'^2}}$$

where a_j is the average value of the j th bunch. Alternatively, we can compute the similarity function S_ϕ with phase.

$$S_\phi(J, J') \approx \frac{\sum_j a_j a'_j \cos(\phi_j - \phi'_j)^2}{\sqrt{\sum_j a_j^2 \sum_j a_j'^2}}$$

If the similarity is larger than the predefined threshold, the result by template matching is acceptable. Otherwise, error message will be generated and the EGM process is stopped.

Step 3: Refine position and size: To the current bunch graph, we vary its position and size to tune the correspondence. For each bunch, check the four different pixels $(\pm 3, \pm 3)$ displaced from its corresponded position in the target image. At each position, we check two different sizes with a factor of 1.2 smaller or larger the bunch graph.

Step 4: Refine aspect ratio: A similar relaxation process as described in Step 3 is performed. But at this time, we apply the operation only to x and y dimensions independently.

4 Tracking

In the previous section, we discuss the feature correspondence between target and template, or two input images, or an image pair of two video sequences. When people want to know the status of object of interest in a single image/video sequence, target tracking becomes an interesting research topic. Since object is located in 3D space and projected onto 2D image, some features of the object will appear and some will disappear when target is moving or sensor is moving. This is an inevitable challenge facing any conventional method of feature tracking.

Under a weak perspective camera model, the motion of a planar rigid object can be approximated by a 2D affine group. Although the set of jets is defined on an object of interest, e.g. human face, which is definitely not an ideal planar object. But, if deformation of each feature point is allowed, one can still get a good approximation to the jet motions. Therefore, we model the jet motions as a 2D affine transformation plus a local deformation. We also assume the motion change between two subsequent frames is small. Unlike conventional methods, we propose to use Markov chain Monte Carlo techniques [23] for tracking. Specifically, the Sequential Importance Sampling (SIS) algorithm is used as motion predictor to find the correspondence features in two subsequent frames (t frame and $t+1$ frame) and on-line select features by updating new weights. In the SIS approach, object motion is formulated as the evaluation of the conditional probability density $p(X_t | Z_t)$. At time t , $p(X_t | Z_t)$ is approximated by a set of its samples, and each sample is associated with a weight reflecting its significance in representing the underlying density (importance sampling). The basic steps of SIS are given as follows:

SIS Algorithm

Let $S_t = \{X_t^{(j)}, j = 1, \dots, m\}$ denote a set of random draws that are properly weighted by the set of weights $W_t = \{w_t^{(j)}, j = 1, \dots, m\}$ with respect to the distribution π_t . At each time step t ,

Step 1. Random draw $x_{t+1}^{(j)}$ from the distribution $g_{t+1}(x_{t+1} | x_t^{(j)})$;

Step 2. Compute

$$u_{t+1}^{(j)} = \frac{\pi_{t+1}(x_{t+1}^{(j)})}{\pi_t(x_t^{(j)})g_{t+1}(x_{t+1}^{(j)} | x_t^{(j)})}$$

$$w_{t+1}^{(j)} = u_{t+1}^{(j)}w_t^{(j)}.$$

Then $(x_{t+1}^{(j)}, w_{t+1}^{(j)})$ is a properly weighted sample of π_{t+1} .

In this algorithm, $g(\cdot)$ is called the trial distribution or proposal distribution and computed by $g_{t+1}(X_{t+1} | X_t) = \frac{1}{\sqrt{2\pi\sigma_1}} \exp\left\{-\frac{(x_{t+1} - x_t)^2}{\sigma_1^2}\right\}$. Thus, the SIS can be applied recursively for $t=1, 2, \dots$, to accommodate an ever-changing dynamical system

In our tracking algorithm, after obtaining a predicted $x_{t+1}^{(j)}$, we check it with the measured value in $t+1$ frame. Based on the measured feature points from the frame at $t+1$, a matching error is computed for the mapped set and the measured set. According to the matching error, $u_{t+1}^{(j)}$ is computed and $w_{t+1}^{(j)}$ is then updated. Note that we do not specify any uncertainty model for individual feature points, which may be too complex to be modeled by a simple function, since it needs to account for inaccuracies in 2D approximation, uncertainty due to noise, non-rigidity of object of interest, etc. In our method, the local deformation at each jet is used to account for these factors.

Another issue during our implementation is we reduce the motion parameter space from 2-dimension (x and y directions) to one-dimension (θ). Here, we can consider a rigid object subject to motion which can be modeled by a transformation f parameterized by a parameter vector θ . Let X_0 denote an original parameterization of the object. X_0 can be a set of jets. Let $X = f(\theta, X_0)$ denote the transformation of X_0 into X . Under a small and continuous motion assumption, X would be similar to X_0 and θ would be very close to θ_0 .

$$X = f(\theta, X_0) = f(\theta_0, X_0) + J_\theta(\theta - \theta_0) + o(\cdot) \tag{8}$$

$$\approx X_0 + J_\theta(\theta_0)(\theta - \theta_0)$$

where $o(\cdot)$ denotes higher-order terms and $J_0(\cdot)$ is the Jacobian matrix with respect to θ . Consider the 2-D affine motion $f(\theta, \cdot)$ as

$$f(\theta, \cdot) = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} (\cdot) + \begin{pmatrix} T_x \\ T_y \end{pmatrix}$$

Vector $\{a_{11}, \dots, a_{22}\}$ represents 2D affine rotation and $\{T_x, T_y\}$ represents translation.

We can compute the Jacobian matrix using

$$J_{\theta}(\theta_0) = \left. \frac{\partial X}{\partial \theta} \right|_{\theta_0} = \begin{bmatrix} x_0 & y_0 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_0 & y_0 & 0 & 1 \end{bmatrix} \quad (9)$$

In our tracking algorithm, we take $\Delta\theta$ as X and use X to find the new position by computing the Jacobian matrix.

Algorithm: SIS-based Tracking

Initialization: The relative camera/sensor motion with a transformation group is modeled first. The motion parameters constitute a state vector distributed according to a density function $\pi(t)$. Then, we track the evolution of $\pi(t)$ over time t using the SIS algorithm, by which $\pi(t)$ is represented by a set of samples $x(t,j)$ with proper weights $w(t,j), j=1,2,\dots$

Step 1. Find a set of feature points of object of interest in the first frame ($t=0$).

Step 2. For time $t>0$, track the set of (feature) points from t to $t+1$ by performing the following:

- a) Each sample $x(t+1,j)$ of $\pi(t+1)$ is used to map/predict the set of feature points to time $t+1$.
- b) Based on the measured feature points from the frame at time $t+1$, a matching error is computed from the mapped set.
- c) The matching error is used to control the updating of $w(t,j)$ to get $w(t+1,j)$.

Step 3. At time $t+1$, we compute the expectation (the weighted mean) of the samples $x(t+1,j)$ to get the motion at that moment, and the points corresponding to the mean give the feature set at that time.

Step 4. For the frame at time $t+1$, use the EGM algorithm with small elasticity to fine-tune the result.

5 Experiments and Applications

Many tests on image/video sequences have been performed with our proposed algorithm. In this section, three tests are selected to illustrate the efficiency and possible applications of the algorithm.

5.1 Dancer Matching and Tracking

The test data sets were acquired from public domain (the website of Microsoft Research Labs). The dynamic scenes were captured by eight cameras at different views with a common synchronization control. The data set from each camera consists of 100 images (24-bits true color, 1024 x 768) at 3 frames per second. We performed two different tests on them. One is finding feature correspondence between two

images acquired from two different cameras. Another one is target tracking in a single video stream.

Fig. 1 shows the feature correspondence results of two images captured from two cameras with different view points. The pixels are correctly corresponded.

Fig. 2 shows the tracking of a dancer in using video sequence from a single camera. Again our algorithm worked very well and we were able to track the dancer even though her movements were very drastic.

- **Matching**

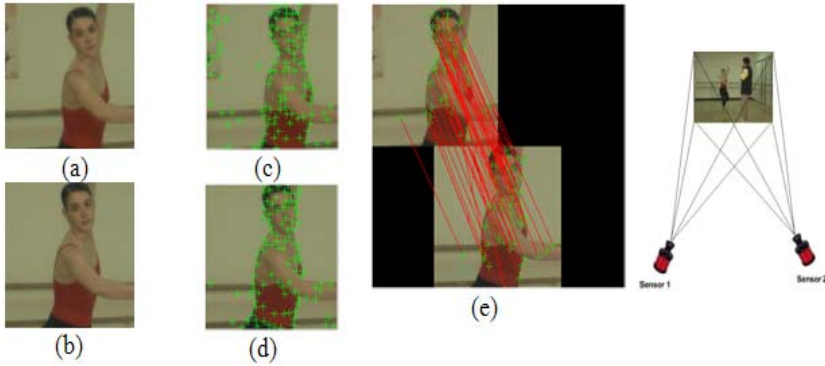


Fig. 1. Finding feature correspondence from two different-view images. (a) and (b) are two different-view images captured at the same time. (c) and (d) show the extracted feature points. (e) is the result of feature correspondence of (c) and (d).

- **Tracking**

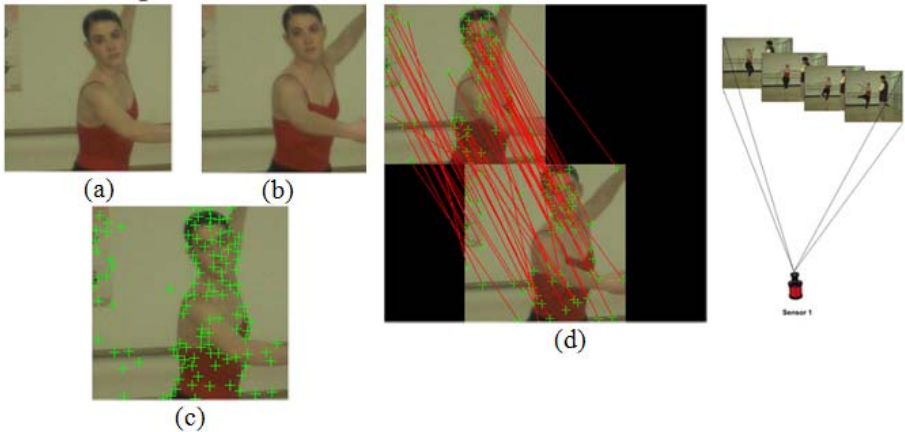


Fig. 2. Object tracking: (a) and (b) are two subsequent images acquired from same camera at different time instances. (c) shows the selected feature points from (a). The tracking result is given in (d).

5.2 Stereo (3D) Image Generation

One important application of image matching is stereo imaging. After finding the feature correspondence between two different-view images, we can use the theories of multi-view geometry [24] to generate stereo image. The testing video sets for stereo imaging were collected by two video cameras with the resolution of 640 x 480 and the frame rate of 10 f/s. We first find the correspondence of the two first frames of the two input video to create a stereo image pair. Then, the features of next stereo pairs for correspondence were tracked by using our proposed tracking method in each video stream independently. Fig. 3 shows the corresponded images and the stereo image.

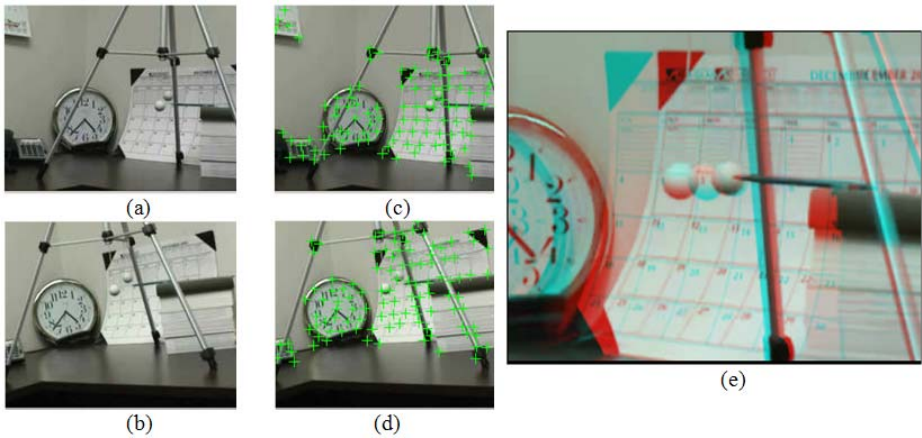


Fig. 3. Stereo imaging: (a) and (b) are two different-view images. (c) and (d) display the selected feature points for feature correspondence. (e) shows a red-cyan stereo image created from the feature correspondence (Reader can watch the 3D image with any red-cyan 3D glasses).

5.3 Target Tracking and Stabilization

One experiment was performed to show how the proposed algorithm can be applied to small target tracking and image sequence stabilization (also known as sensor motion

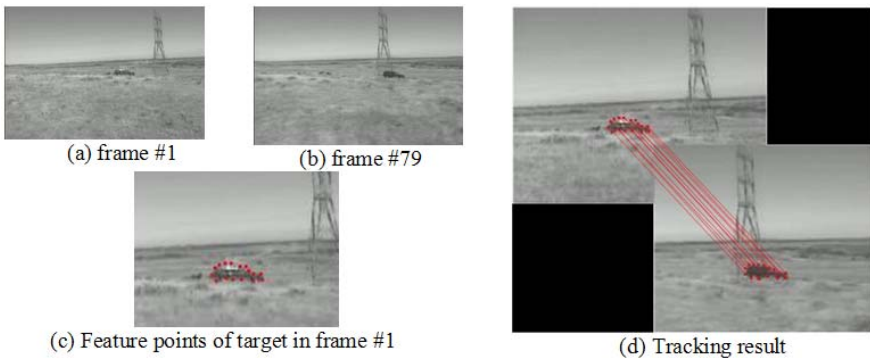


Fig. 4. Target tracking

compensation). The testing data has the resolution of 336 x 244 and the frame rate of 30 f/s. In the test, we first manually located a target of interest, e.g. a moving vehicle shown in Fig. 4 (a). Then, Gabor filter-based feature extraction and SIS-based tracking algorithms were performed to track the moving target in image sequence. The result is given in Fig. 4 (d). As we can see from the result, the small moving target can be successfully tracked in cluttered environment.

For sensor motion compensation, we modeled the camera motion as a 2D affine transformation. Stabilization is then achieved in the following steps. First, we extracted a set of feature points from the first frame. Next, we used the algorithm to track the feature set in the sequence. Stabilization was then done by warping the current frame with respect to the reference frame (the first frame) using the estimated motion parameter.

6 Summary

In this work, we have examined the problem of target matching and tracking in video/image sequence including the data acquired in noisy environment. We proposed a Gabor attribute matching and tracking algorithm based on an improved EMG and statistical sampling method. As described in the introduction section, there are many methods for object matching and tracking. Our algorithm differs from other matching methods in that we use Gabor attribute as features and extend the typical EMG method by introducing an efficient template matching method. . Consequently, our method is suitable for more applications with target rotation and size variations. Most importantly, we develop SIS-based method for real-time tracking. The advantage of our tracking algorithm is that we track target without requiring any assumptions to input data, such as Gaussian distribution, motion type and direction, rigid and non-rigid target, and do not need to predict the motion speed and direction (even allow rotation in depth) as our approach continually select good feature for tracking by updating weight at each computation. Another advantage is our tracking is intended for verification – the model templates by Gabor filter can be easily incorporated into the tracker because of its formulation and parameterization. Low computational cost is also one advantage of the algorithm. In our experiments, the tracker processes input data in real-time on an ordinary PC (CPU: Intel P4 - 2GHz, and Memory: 1024 MB).

Acknowledgements

This work was supported in part by a grant from the U. S. Missile Defense Agency, HQ0006-05-C-7277. The authors would like to thank Mr. Richard Felner and Mr. Steve Waugh for their helpful suggestions and comments.

References

1. Biometric Systems Technology, Design and performance Evaluation, Wayman, J.; Jain, A.; Maltoni, D.; Maio, D. (Eds.), Springer, (2005)
2. Handbook of Fingerprint Recognition, Maltoni, D.; Maio, D.; Jain, A.; Prabhakar, S. Springer, (2003)

3. Handbook of Face Recognition, Li, Stan Z, Jain, A.; (Eds.), Springer, 2005.
4. D. A. Clausi and H. Deng: Fusion of Gabor filter and co-occurrence probability features for texture recognition," *IEEE Trans. of Image Processing*, Vol. 14, No. 7, (2005) 925-936
5. L. Wiskott, J. Fellous, N. Kruger, C. Malsburg: Face recognition by Elastic Bunch Graph Matching, *Intelligent Biometric Techniques in Fingerprint and Face Recognition*, CRC Press, Chapter 11, (1999) 355-396
6. A. Baumberg and D. Hogg: An efficient method for contour tracking using active shape models, In *Proc. IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, (1994) 194-1999
7. M. Black and A. Jepson: Eigen Tracking: Robust matching and tracking of articulated objects using a view-based representation, *Int. J. computer Vision*, vol. 20, (1998) 63-84
8. D. Freedman and M. Brandstein: A subset approach to contour tracking in clutter, In *Proc. Int. Conf. Computer Vision* (1999)
9. D. Huttenlocher, J. Noh, and W. Rucklidge: Tracking nonrigid objects in complex scenes, In *Proc. Int. Conf. Computer Vision* (1993)
10. M. Isard and A. Blake: Contour tracking by stochastic propagation of conditional density, in *Proc. Eur. Conf. Computer Vision* (1996)
11. C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland: Pfunder: Real-time tracking of the human body, *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 19 (1997) 780-785
12. Y. Yacoob and L. Davis: Tracking rigid motion using a compact-structure constraint, in *Proc. Int. Conf. Computer Vision* (1999)
13. J. G. Daugman, "Complete discrete 2D Gabor transform by neural networks for image analysis and compression," *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol. 36(7) (1988) 1169-1179
14. R. L. DeValois and K. K. DeValois, *Spatial Vision*, Oxford Press (1988)
15. J. P. Jones and L. A. palmer: An evaluation of the 2D Gabor filter model of simple receptive fields in cat striate cortex, *J. of Neurophysiology*, Vol. 58 (1987) 1233-1258
16. A. J. Bell and T. J. Sejnowski: The independent components of natural scenes are edge filters, *Vision Research*, Vol. 37(23) (1997) 3327-3338.
17. B. A. Olshausen and D. J. Field: Emergence of simple-cell receptive field properties by learning a sparse code for natural images, *Nature*, Vol. 381 (1996) 607-609
18. M. Potzsch, N. Kruger, and V. D. Malsburg: Improving object recognition by transforming Gabor filter responses, *Network: Computation in Neural Systems*, Vol. 7(2) (1996) 341-347.
19. S. E. Grigorescu, N. Petkov, and P. Kruizinga: Comparison of texture features based on Gabor filters, *IEEE Trans. on Image Processing*, Vol. 11(10) (2002)
20. D. Beymer: Face recognition under varying pose, In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, (1994) 756-761
21. R. M. Dufour, E. L. Miller, N. P. Galatsanos: Template matching based object recognition with unknown geometric parameters, *IEEE Trans. on Image Processing*, Vol.11(12) (2002)
22. A. Abu-Naser, N. P. Galatsanos, M. N. Wernick, and D. Schonfeld: Object recognition based on impulse restoration using the expectation maximization algorithm, *J. Opt. Soc. Amer. A: Opt. Image Sci.*, Vol. 15 (1998)
23. J. Liu and R. Chen: Sequential Monte Carlo methods for dynamic systems, *J. Am. Stat. Assoc.* Vol. 93 (1998) 1031-1041
24. R. Hartley and A. Zisserman: *Multiple View Geometry in Computer Vision*, Second Edition, Cambridge Press (2003)

Image Based Visual Servoing: A New Method for the Estimation of the Image Jacobian in Dynamic Environments

L. Pari¹, J.M. Sebastián¹, C. González¹, and L. Angel²

¹Departamento de Automática, Ingeniería Electrónica e Informática Industrial (DISAM)
Escuela Técnica Superior de Ingenieros Industriales, Universidad Politécnica de Madrid
C/ José Gutiérrez Abascal, 2, 28006 Madrid, Spain

²Facultad de Ingeniería Electrónica, Universidad Pontificia Bolivariana
Km. 7 Via de Piedecuesta, Bucaramanga, Colombia
{jsebas, lpari, cgpascual, langel}@etsii.upm.es

Abstract. This paper describes a innovative algorithm for the on-line estimation of the image jacobian based on previous movements. It has been utilized for the tracking of an uncalibrated 3 DOF joint system, using two weakly calibrated fixed cameras. The algorithm incorporates the epipolar restriction of the two cameras conveyed in the fundamental matrix, and a reliability factor has been defined for the previous movements. The tests carried out show that the proposed algorithm is more robust in presence of noise than those obtained from already existing algorithms in specialized literature.

Keywords: Visual servoing, Jacobian estimation, Fundamental matrix.

1 Introduction

Visual servoing consists in the use of visual information given by visual sensors (i.e. cameras) to control a robotic system. This kind of control turns out to be very useful in dynamic environments, as it allows us to know which objects are present in the scene with high accuracy, as well as their position, orientation and velocity. It makes possible to use robots in new domains where the workspace is not known a priori. However it also proposes to solve new problems (in terms of accuracy and robustness) currently unresolved [6]. The propose algorithm in this paper increases the accuracy and robustness without requiring a thorough knowledge of the involved systems: calibration of the joint system and kinematic calibration of the vision system, both common sources of accumulative errors.

Visual servoing has been widely studied in the lastest years, there are in the literature interesting surveys [5] and [7]. Among the existing clasifications of visual servoing, one of the most known is the way visual information is used to define the signal error to control the system: position based visual servoing (PBVS or 3D) and the image based visual servoing (IBVS or 2D). In position based visual servoing, features are extracted from the image and used to reconstruct the 3D position of the target, whereas in image based visual servoing the task is defined in the image plane directly

through image features. In the latter a matrix is defined called the Image Jacobian, which linearly relates changes in image features and changes in Cartesian coordinates or changes in joints (in this case, image jacobian is called visual-motor jacobian).

This work, belongs to IBVS and contributes a new method based on the use of the epipolar restriction for the estimation of the Image Jacobian: this estimation can be carried out without the need for an accurate calibration, in these conditions the analytical calculation of the Jacobian is impossible. An uncalibrated 3 DOF joint system with two fixed cameras has been used. The tests carried out consisted of tracking curve trajectories in the 3D space and comparig them with other existing algorithms. In previous works, [11], tests of static positioning have been described with very positive results. In both cases a increase of robustness in presence of noise is achieved.

This paper is organized as follows: after the present introduction, section two details the terminology and theoretical concepts used in the paper. Section three puts forward the innovative algorithm proposed, whilst section four describes the applied workspace and the results, and finally section five reflects our conclusions.

2 Image Jacobian

Assume that a robot or positioning system is observed from one or various fixed views. Let $\mathbf{r} = [r_1 \ r_2 \ \dots \ r_p]^T$ be the p -dimensional vector that represents the position of the end effector in a Cartesian coordinate system. Let $\mathbf{q} = [q_1 \ q_2 \ \dots \ q_n]^T$ be the n -dimensional vector that represents the joint position of the robot. Let $\mathbf{s} = [s_1 \ s_2 \ \dots \ s_m]^T$ be the m -dimensional vector that represents the image features (for example the coordinates of a point in one or both images).

The relation between joint velocity of the robot $\dot{\mathbf{q}} = [\dot{q}_1 \ \dot{q}_2 \ \dots \ \dot{q}_n]^T$ and its corresponding velocity in task space, $\dot{\mathbf{r}} = [\dot{r}_1 \ \dot{r}_2 \ \dots \ \dot{r}_p]^T$, is captured in terms of the robot Jacobian, \mathbf{J}_{rq} , as $\dot{\mathbf{r}} = \mathbf{J}_{rq} \dot{\mathbf{q}}$. The relation between feature velocities $\dot{\mathbf{s}} = [\dot{s}_1 \ \dot{s}_2 \ \dots \ \dot{s}_m]^T$ and task space velocities, is given by $\dot{\mathbf{s}} = \mathbf{J}_{sr} \dot{\mathbf{r}}$.

The velocity of the image features can be directly related to joint velocities in terms of a composite Jacobian, also known as the full visual-motor Jacobian [2], [12]

$$\dot{\mathbf{s}} = \mathbf{J}_{sq} \dot{\mathbf{q}} = \begin{bmatrix} \frac{\partial s_1}{\partial q_1} & \dots & \frac{\partial s_1}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial s_m}{\partial q_1} & \dots & \frac{\partial s_m}{\partial q_n} \end{bmatrix} \dot{\mathbf{q}} \quad ; \quad \text{where } \mathbf{J}_{sq} = \mathbf{J}_{sr} \mathbf{J}_{rq} = \mathbf{J} \tag{1}$$

Determining this matrix analytically is not simple. It is necessary to remark that in its calculation there must be considered: the intrinsic parameters of the camera calibration (focal distance, image center coordinates), the 3D reconstruction of the point or an approximation (Z coordinate), the kinematic calibration of the camera (relation between camera coordinates and joint space origin), and the kinematic calibration of

the robot. Most of the previous works on visual servoing assume that the system structure and the system parameters are known, or the parameters can be identified in an off-line process. A control scheme with off-line parameter identification is not robust for disturbance, change of parameters, and unknown environments. One approach to image-based visual servoing without calibration is to dynamically estimate the full visual-motor Jacobian during motion.

2.1 Estimation of the Jacobian

Specialized literature ([2], [12]) gathers four methods for estimating the jacobian described in equation (1). In all cases an initial jacobian is obtained by making n linearly independent small movements.

Estimation Based on the Last Moves. If the change in image features and the change in joint position at moment k are represented respectively by $\Delta \mathbf{s}_k = \mathbf{s}_k - \mathbf{s}_{k-1}$ and by $\Delta \mathbf{q}_k = \mathbf{q}_k - \mathbf{q}_{k-1}$, and the image jacobian is assumed to be constant for n movements, it can then be estimated as the matrix that simultaneously satisfies n or more movements:

$$\begin{bmatrix} \Delta \mathbf{s}_{k-n+1}^T \\ \vdots \\ \Delta \mathbf{s}_k^T \end{bmatrix} = \begin{bmatrix} \Delta \mathbf{q}_{k-n+1}^T \\ \vdots \\ \Delta \mathbf{q}_k^T \end{bmatrix} \mathbf{J}_k^T \tag{2}$$

where \mathbf{J}_k is the image jacobian at moment k .

This method may not work well when several subsequent joint motions are generated in the same direction, a not unusual situation. In these cases a badly-behaved matrix is obtained, and the consequent problems arise. Sutanto [12] solves the problem by detecting when this situation will occur and adding small exploratory motions.

Broyden Method. In this method the jacobian is estimated recursively, combining the information supplied by the last movement with the previous jacobian. Regarding the former method, it has the advantage of gathering information from all the movements. The equation for the Broyden method [2], [9] is:

$$\mathbf{J}_k = \mathbf{J}_{k-1} + \frac{\left(-\Delta \mathbf{e}_k + \frac{\partial \mathbf{e}_k}{\partial t} \Delta t - \mathbf{J}_{k-1} \Delta \mathbf{q}_k \right) \Delta \mathbf{q}_k^T}{\Delta \mathbf{q}_k^T \Delta \mathbf{q}_k} \tag{3}$$

being $\mathbf{e}_k = \mathbf{s}_k^* - \mathbf{s}_k$ image features error, and \mathbf{s}_k^* the desired features

Recursive Least Squares (RLS) Method. In this method the jacobian is estimated recursively by a least squares algorithm [9], [1], the equations of which, for a positioning task, are:

$$\mathbf{J}_k = \mathbf{J}_{k-1} + \frac{\left(-\Delta \mathbf{e}_k + \frac{\partial \mathbf{e}_k}{\partial t} \Delta t - \mathbf{J}_{k-1} \Delta \mathbf{q}_k \right) \Delta \mathbf{q}_k^T \mathbf{P}_{k-1}}{\lambda + \Delta \mathbf{q}_k^T \mathbf{P}_{k-1} \Delta \mathbf{q}_k} \tag{4}$$

where

$$P_k = \frac{1}{\lambda} \left(P_{k-1} - \frac{P_{k-1} \Delta q_k \Delta q_k^T P_{k-1}}{\lambda + \Delta q_k^T P_{k-1} \Delta q_k} \right) \tag{5}$$

The behaviour of this method depends on parameter λ , which varies in a range from 0 to 1, and ponders previous movements. λ settles a compromise between the information provided by old data from previous movements and new data, possibly corrupted by noise. In the presence of moderate noise, values of λ close to 0.9 are often used.

Kalman Method. Another way to estimate the jacobian recursively is by using the Kalman filter [10]: the system is modelled through its state variables as follows:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \eta_k \\ \mathbf{y}_k = \mathbf{C}_k \mathbf{x}_k + u_k \end{cases} \tag{6}$$

where η_k, u_k are the state noise and observer noise respectively, \mathbf{x} is an $(mn \times 1)$ vector formed by concatenations of the row elements of the jacobian:

$$\mathbf{x} = \left[\begin{matrix} \frac{\partial s_1}{\partial \mathbf{q}} & \frac{\partial s_2}{\partial \mathbf{q}} & \dots & \frac{\partial s_m}{\partial \mathbf{q}} \end{matrix} \right]^T \tag{7}$$

where $\frac{\partial s_i}{\partial \mathbf{q}}$ is the i -th row of the jacobian \mathbf{J}_k . The measurement vector is

$$\mathbf{y}_{k+l} = \mathbf{s}_{k+l} - \mathbf{s}_k \tag{8}$$

and $(m \times mn)$ matrix \mathbf{C}_k is the measured joint increase matrix:

$$\mathbf{C}_k = \begin{bmatrix} \Delta \mathbf{q}_k^T & & 0 \\ & \ddots & \\ 0 & & \Delta \mathbf{q}_k^T \end{bmatrix} \tag{9}$$

The Kalman equations are:

$$\begin{aligned} P_{k+1}^- &= P_k + R_\eta \\ K_{k+1} &= P_{k+1}^- C_k^T [C_k P_{k+1}^- C_k^T + R_\nu]^{-1} \\ P_{k+1} &= [I - K_{k+1} C_k] P_{k+1}^- \\ \hat{\mathbf{x}}_{k+1} &= \hat{\mathbf{x}}_k + K_{k+1} [y_{k+1} - C_k \hat{\mathbf{x}}_k] \end{aligned} \tag{10}$$

where P_k is the error covariance matrix at moment k , P_{k+1}^- is the prediction of P_k , R_η, R_ν are the state and output covariance matrices respectively and K_{k+1} is the Kalman gain.

2.2 Multiple-View Jacobian

When several views are used, the full visual-motor Jacobian can be defined as the concatenation of the partial Jacobians of each view (more detailed in [8], [1]). All the Jacobians share the same joint increments, although visual features are managed independently. Here lies main contribution of this paper: if the features are points, the epipolar constraint is incorporated in the estimation of the Jacobian. The notation we used is detailed in section 3.2.

In previous works [11], we carried out experiments comparing the results given using one of the cameras and those given using both cameras: our results showed that using two cameras instead of one improved the behaviour of all the methods we tested. In many applications, improvement in the performance more than justifies the possible disadvantages: increased equipment cost or calculation time. Therefore, the experiments described in this work have been performed using both cameras.

2.3 Control Law

A control law have been utilized for tracking a twisted quadratic trajectory. It adds to the proportional control law [9] a predictive term from the last two references according to the equation:

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \mathbf{J}^+ \left(\mathbf{s}_k^* - \mathbf{s}_k + \mathbf{s}_k^* - \mathbf{s}_{k-1}^* \right) \quad (11)$$

being $\mathbf{J}^+ = \left(\mathbf{J}^T \mathbf{J} \right)^{-1} \mathbf{J}^T$ the pseudoinverse for the image jacobian.

3 Proposed Algorithms

The research line we have followed is the estimation of the jacobian based on already performed movements, not necessarily the n last ones. We use the visual information supplied by two cameras. We contribute two innovations: on one hand, each movement has been endowed with a certain reliability, so that the most adequate or reliable movements can be used. On the other hand, the epipolar constraint has been taken into account in the calculation of the jacobian, which significantly increases the robustness of the method, as will be seen in section four. It must be remarked that although we propose the joint application of both innovations, using them independently of each other also improves the results.

The visual features we use are centroids of points. The proposed task consists in making these visual features follow a trajectory set in advance.

3.1 Reliability Estimation

The jacobian matrix, equation (1), strongly depends on the position of the point in the image (u, v) , so the assumption that it is constant will only be valid in the surroundings of the point in the image. For this reason, a first possibility would be to promote the consideration of those already performed movements with a short path in image features. However, these movements are too sensitive to noise, so an agreement must be reached between both effects. Movements performed in the joint surroundings of the

desired movement also seem more adequate. The proposed algorithms rank the already performed movements according to a reliability which depends on two factors that assemble these concepts:

$$reliability_{ki} = Factor1_i / Factor2_{ki} \tag{12}$$

Where subindex k represents the last movement performed and subindex i will vary amongst those already performed movements which have been stored.

$Factor1_i$ promotes movements in which features vary within a range: they are not too large so that the jacobian can be considered a constant, nor too small so they will not be too sensitive to noise. The formula we used is:

$$Factor1_i = \frac{I}{\frac{abs(\|\Delta s_i\| - F)}{F} + R} \tag{13}$$

where R represents the estimated error in feature calculation (for example, for 5 points detected with subpixel precision, R would have a value close to 1.0 pixels), and F represents the most reliable feature increment (for example, 20 pixels). We must remark that $Factor1_i$ does not depend on the last performed movement, so it will be calculated and stored together with its corresponding movement.

$Factor2_{ki}$ promotes those already stored movements produced in the joint surroundings of the desired movement, comparing for this purpose the current joint position with the mid point of already reached joint positions. Its expression is:

$$Factor2_{ki} = \left\| \mathbf{q}_k - \left(\frac{\mathbf{q}_i + \mathbf{q}_{i-1}}{2} \right) \right\| \tag{14}$$

Greater reliability will be given to those performed movements that have greater value for the expression (12).

3.2 Adding the Epipolar Constraint

A point in space $\tilde{\mathbf{S}} = [X \ Y \ Z \ 1]^T$, where $\tilde{}$ represents homogeneous coordinates, is projected in the first image in point $\tilde{\mathbf{s}}' = [x' \ y' \ 1]^T = \mathbf{P}'\tilde{\mathbf{S}}$, where \mathbf{P}' represents the (3x4) projective matrix of the first camera. This point will likewise be projected in the second image in point $\tilde{\mathbf{s}}'' = [x'' \ y'' \ 1]^T = \mathbf{P}''\tilde{\mathbf{S}}$ where \mathbf{P}'' represents the (3x4) projective matrix of the second camera. Points are expressed in homogeneous coordinates and an image acquisition model without distortion is assumed. The Euclidean calibration of both cameras implies the knowledge of matrices \mathbf{P}' , \mathbf{P}'' .

The projection of the same point on both images must satisfy the equation:

$$\tilde{\mathbf{s}}''^T \mathbf{F} \tilde{\mathbf{s}}' = 0 \tag{15}$$

where \mathbf{F} is a (3x3) matrix known as the fundamental matrix or epipolar constraint. Its knowledge is known as weak or projective calibration, and its detection is much more

robust than Euclidean calibration. A more detailed description can be found in [3] and [4].

Our method considers the epipolar constraint, equations (15) in the calculation of the image jacobian, equation (1). If the considered visual features are centroids of points, and if we note a point in the first camera by ‘, and by “ in the second camera, we have the following model:

$$\mathbf{s}_k = \begin{bmatrix} \mathbf{s}'_k \\ \mathbf{s}''_k \end{bmatrix} ; \quad \mathbf{s}_{k-1} = \begin{bmatrix} \mathbf{s}'_{k-1} \\ \mathbf{s}''_{k-1} \end{bmatrix} ; \quad \mathbf{J} = \begin{bmatrix} \mathbf{J}' \\ \mathbf{J}'' \end{bmatrix} \tag{16}$$

$$\mathbf{s}'_k = \mathbf{s}'_{k-1} + \mathbf{J}' \Delta \mathbf{q}_k ; \quad \mathbf{s}''_k = \mathbf{s}''_{k-1} + \mathbf{J}'' \Delta \mathbf{q}_k \tag{17}$$

$$\begin{bmatrix} \mathbf{s}''_{kT} & I \end{bmatrix} \mathbf{F} \begin{bmatrix} \mathbf{s}'_k \\ I \end{bmatrix} = 0 \tag{18}$$

$$\begin{bmatrix} \mathbf{s}''_{k-1T} & I \end{bmatrix} \mathbf{F} \begin{bmatrix} \mathbf{s}'_{k-1} \\ I \end{bmatrix} = 0 \tag{19}$$

Substituting in (18) the values obtained in (17) and considering that equation (19) must be satisfied, we have the following non-linear equation for (J', J''):

$$\Delta \mathbf{q}_k^T \mathbf{J}''^T \mathbf{F} \mathbf{J}' \Delta \mathbf{q}_k + \Delta \mathbf{q}_k^T \mathbf{J}''^T \mathbf{F} \begin{bmatrix} \mathbf{s}'_{k-1} \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{s}''_{k-1T} & 1 \end{bmatrix} \mathbf{F} \mathbf{J}' \Delta \mathbf{q}_k = 0 \tag{20}$$

The linear equations in (2) and the non-linear equations in (20) have been jointly solved applying the Levenberg-Marquadt method.

3.3 Glossary of Tested Algorithms

In the implementation described in the present article, the following algorithms have been tested:

- 3LAST: Last three movements performed. Since our joint system has three degrees of freedom, equation (2) can be solved using the information on joint and feature velocities provided by three movements.
- 3+REL: Three most reliable movements amongst the last ten performed.
- 10LASTW: Last ten movements performed, weighted by their reliability. Weighting is introduced multiplying each row in equation (2) by its corresponding reliability.
- 10+REL: Ten most reliable movements performed.
- FUNDMAT: Ten most reliable movements performed, adding the epipolar constraint.
- BROYDEN: Broyden method.
- RLS: Recursive least squares method.
- KALMAN: Kalman filter estimation.

Algorithms 3+REL, 10LASTW, 10+REL and FUNDMAT are original, innovative methods founded on the estimation of the jacobian based on the last moves.

4 Experiments

In this section we describe our experimental equipment and results.

4.1 Experimental Setup

The system used in the experiments consists of:

- A joint system composed of a high precision positioning device and its controller, model Newport MM3000 (see Fig. 1). The system has 3 DOF with a prismatic and two revolute joints, and its theoretical precision is of a thousandth of a millimeter and a thousandth of a degree. The visual control object, made out of five black dots on a white background, the projection of which on the image will be the control features, has been attached to the last link of the joint system.

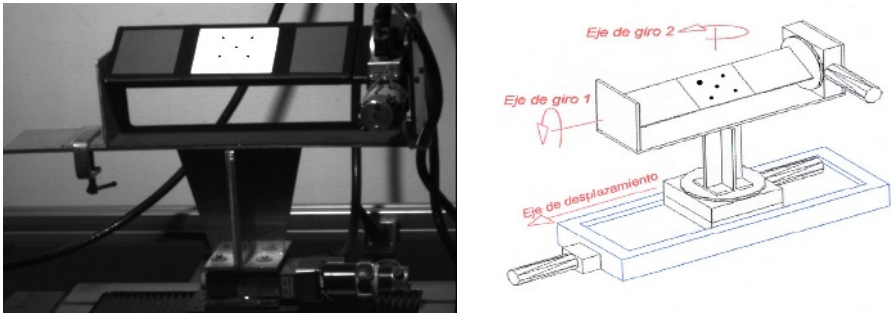


Fig. 1. Experimental setup

- An image acquisition and processing system composed by two CV-M50 analogic cameras and a Matrox Meteor II-MC image acquisition board, which allows simultaneous acquisition from both cameras. The cameras, fixed in the working environment, are separated by about 300 millimeters, both their axes converge towards the joint system, and they are separated from it by about 700 millimeters. Visual features are detected with subpixel precision, and given the simplicity of the image, the error is estimated to be of less than 0.2 pixels. Communication with the joint system controller is established through a serial RS-232C cable.

We must remark that the joint system is a closed system: once a position has been selected, control will not be handed over until it is reached. This course of action is unacceptable for a visual control system that must use the information supplied by the cameras as soon as possible. In order to simulate a more adequate behaviour, we have chosen to limit joint motion according to the estimated time it would take the vision system to acquire and process images. On the other hand, the high accuracy of the employed system allows us to perform a thorough comparative analysis of the proposed algorithms.

4.2 Control Objective

The task entrusted to the system is to achieve the tracking of a trajectory set in advance (see Fig. 2) by using the information obtained on detection of certain image features (centroids of projected dots). We intend to contrast the performance of the proposed methods of estimation of the image jacobian using the control law (section 3.2.)

Visual features must be reachable and the visual object must be visible from both points of view. To ensure coherence, we decided to obtain the desired visual features previously by acquiring images in reference joint positions, chosen randomly within the workspace (*teach-by-showing*). Due to the dependency of the jacobian on visual features and on the point in joint space, we have chosen to link a high number of trajectories (50) in order to obtain more representative results for the comparative study of the proposed algorithm.

4.3 Evaluation Indices

To evaluate the effectiveness of each method, we consider three indices, defined as follows:

- Index 0: Sum of Euclidean distances between desired and current visual features. Weighted by number of points, number of cameras and number of trajectories.
- Index 1: Sum of Euclidean distances in joint space for all of the performed movements, divided by one thousand. Weighted by number of trajectories.
- Index 2: Sum of Euclidean distances between desired and current joint positions. Weighted by number of points.

Table 1. Values for three indices, with and without added noise. Predictive law.

| | ALGORITHM | WITHOUT NOISE | | | | WITH NOISE | | | |
|---------|-----------|---------------|----------|----------|----------|------------|----------|----------|----------|
| | | 2 POINTS | 3 POINTS | 4 POINTS | 5 POINTS | 2 POINTS | 3 POINTS | 4 POINTS | 5 POINTS |
| INDEX 0 | 3LAST | 16.6 | 17.9 | 18.6 | 18.5 | 20.4 | 21.8 | 21.5 | 22.1 |
| | 3+REL | 16.8 | 18.2 | 18.0 | 17.8 | 17.2 | 21.0 | 19.6 | 22.0 |
| | 10LASTW | 16.5 | 17.5 | 17.9 | 17.7 | 18.4 | 17.8 | 20.7 | 19.9 |
| | 10+REL | 16.5 | 17.7 | 17.9 | 17.7 | 18.5 | 20.9 | 19.4 | 21.4 |
| | FUNDMAT | 16.5 | 17.5 | 17.7 | 17.6 | 16.6 | 17.7 | 17.9 | 17.7 |
| | BROYDEN | 16.4 | 18.2 | 18.7 | 17.8 | 16.9 | 19.6 | 22.8 | 19.0 |
| | RLS | 16.5 | 17.7 | 18.1 | 17.7 | 16.7 | 18.8 | 18.4 | 18.4 |
| | KALMAN | 16.0 | 17.3 | 17.7 | 17.6 | 17.4 | 18.0 | 18.7 | 18.5 |
| INDEX 1 | 3LAST | 3.43 | 3.48 | 3.68 | 3.63 | 2.22 | 2.55 | 2.29 | 2.39 |
| | 3+REL | 3.68 | 3.47 | 3.44 | 3.42 | 3.60 | 2.45 | 3.66 | 2.40 |
| | 10LASTW | 3.40 | 3.37 | 3.37 | 3.37 | 3.39 | 3.57 | 2.92 | 3.40 |
| | 10+REL | 3.37 | 3.32 | 3.34 | 3.35 | 3.18 | 2.42 | 3.44 | 2.49 |
| | FUNDMAT | 3.35 | 3.33 | 3.34 | 3.35 | 3.36 | 3.34 | 3.31 | 3.34 |
| | BROYDEN | 3.37 | 3.43 | 3.66 | 3.38 | 3.34 | 2.86 | 3.84 | 3.32 |
| | RLS | 3.34 | 3.31 | 3.38 | 3.34 | 3.44 | 3.59 | 3.49 | 3.44 |
| | KALMAN | 3.37 | 3.41 | 3.48 | 3.42 | 3.65 | 3.53 | 3.39 | 3.46 |
| INDEX 2 | 3LAST | 0.77 | 0.98 | 1.84 | 0.99 | 7.08 | 5.70 | 5.34 | 5.55 |
| | 3+REL | 0.94 | 0.67 | 0.58 | 0.41 | 1.88 | 5.66 | 1.97 | 5.36 |
| | 10LASTW | 0.59 | 0.33 | 0.39 | 0.34 | 4.11 | 0.78 | 4.99 | 3.65 |
| | 10+REL | 0.46 | 0.69 | 0.53 | 0.37 | 4.14 | 6.01 | 2.97 | 5.14 |
| | FUNDMAT | 0.42 | 0.50 | 0.38 | 0.29 | 0.45 | 0.60 | 0.45 | 0.29 |
| | BROYDEN | 0.38 | 0.80 | 1.60 | 0.45 | 1.42 | 3.64 | 6.19 | 2.38 |
| | RLS | 0.50 | 0.43 | 0.58 | 0.34 | 1.17 | 2.81 | 1.06 | 1.17 |
| | KALMAN | 0.21 | 0.42 | 0.62 | 0.56 | 3.45 | 1.63 | 1.73 | 1.58 |

4.4 Results

A comparative study was conducted on the proposed algorithms. The comparison covers visual features calculated with an estimated error of 0.2 pixels, therefore considered without noise, as well as visual features with artificially added Gaussian noise with a standard deviation of 0.5 pixels. Also, the effect of increasing the number of points considered as visual features from 2 to 5 is analyzed.

Table 1 displays the results obtained with the control predictive law, with or without added noise for all algorithms analyzed according to the three defined indexes and varying the number of points taken into account. The methods which give better results are KALMAN, RLS, FUNDMAT and 10LASTW. With added noise the most robust method is FUNDMAT, especially on index 2, although RLS and KALMAN methods also show a good behavior. Increasing the number of points has two opposite effects: we have more information to control the system, but also stricter requirements to fulfill, which causes variations in the indices.

Fig. 2 (FUNDMAT), Fig. 3 (RLS) and Fig. 4 (KALMAN) represent the evolution in joint space for each method of estimation of the jacobian. The red points represent points to be reached in the reference trajectory, and the blue line shows the evolution of the joint system. For the two laws considered, with and without added noise, the method that performs the best tracking is the one that considers the epipolar restriction (FUNDMAT).

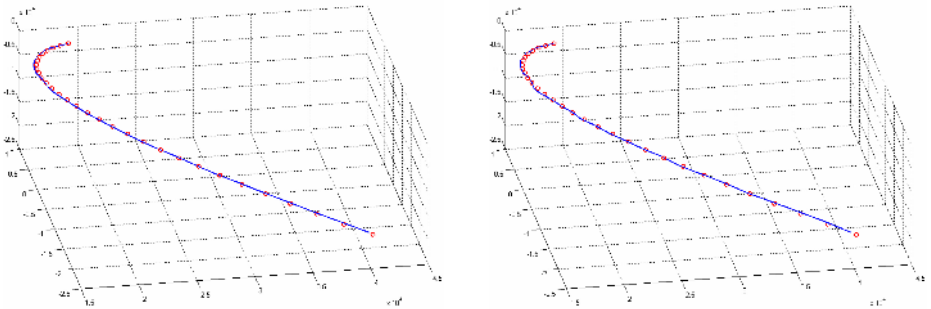


Fig. 2. Evolution for five points, FUNDMAT algorithm without and with noise

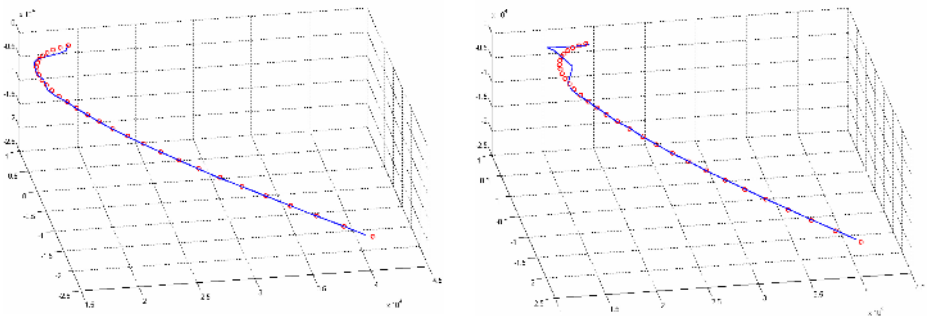


Fig. 3. Evolution in joint space for five points, RLS algorithm without and with noise

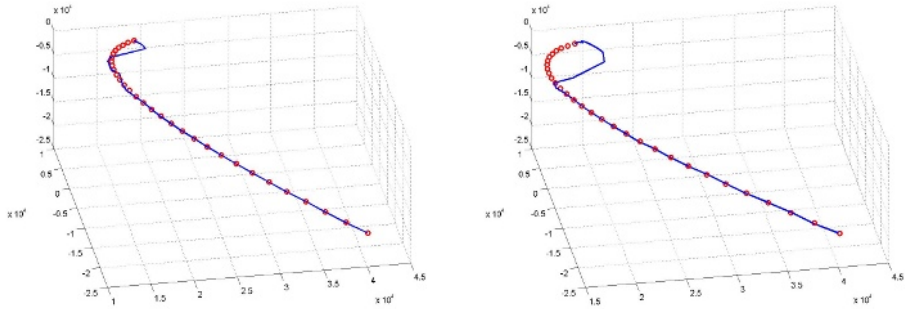


Fig. 4. Evolution in joint space for five points, KALMAN algorithm without and with noise

5 Conclusions

The on-line estimation of the image jacobian is a flexible and versatile method for the visual control of a joint structure, since it isolates the obtained results from errors in the calibration of the camera and the joint system. This paper contributes the definition of a reliability to calculate the jacobian, and the inclusion of the epipolar constraint or fundamental matrix in its calculation. This aspect is not considered in specialized literature, and it improves the results significantly when the noise level in feature detection increases. The knowledge of the fundamental matrix is no objection, as its calculation has been proven to be much more simple, robust and reliable than that of the complete calibration of the cameras and joint system.

Some aspects not dealt with in the present paper which are being currently studied are the analysis of the system stability with a control law generated from the jacobian estimation, and the analytic calculation of the image jacobian. Regarding the first aspect, we must remark that algorithm FUNDMAT has never made the system unstable throughout the many tests performed.

This work was supported by the Comisión Interministerial de Ciencia y Tecnología of the Spanish Government under the Project DPI2004-07433-C02-02.

References

- [1] M. Asada, T. Tanaka, K. Hosoda, Adaptive Binocular Visual Servoing for Independently Moving Target Tracking, Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'00), (2000) 2076-2081.
- [2] Z. Deng, M. Jägersand, Evaluation of Model Independent Image-Based Visual Servoing, Canadian Conference on Computer and Robot Vision, (2004), 138-144
- [3] O. Faugeras, Q.T. Luong, The Geometry of Multiple Images, The Massachusetts Institute of Technology Press, 2001
- [4] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision. Cambridge University Press, 2000
- [5] S.A. Hutchinson, G.D. Hager, P.I. Corke, A tutorial on visual servo control, IEEE Trans. Robotics and Automation, 12 (1996) 651-670.

- [6] D. Kragic, H.I. Christensen, Advances in robot vision, *Robotics and Autonomous Systems*. 52 (1) (2005) 1-3
- [7] D. Kragic, H.I. Christensen, Survey on visual servoing for manipulation, Technical Report ISRN KTH/NA/P-02/01-Sen, CVAP259, 2002
- [8] B. Lamiroy, B. Espiau, N. Andreff and R. Horaud, Controlling Robots With Two Cameras: How to Do it Properly, *Proceedings of the International Conference on Robotics and Automation, ICRA'00*, (2000), 2100-2105
- [9] J.A. Piepmeier, G.V. McMurray, H. Lipkin, Uncalibrated Dynamic Visual Servoing, *IEEE Transactions on Robotics and Automation*, 20 (1) (2004) 143-147.
- [10] J. Qian, J. Su, Online estimation of Image Jacobian Matrix by Kalman-Bucy filter for uncalibrated Stereo Vision Feedback, *International Conference on Robotics & Automation (ICRA'02)*, (2002), 562-567
- [11] J.M. Sebastián, L. Pari, C. González, L. Ángel, A New Method for the Estimation of the Image Jacobian for the Control of an Uncalibrated Joint System, *Lecture Notes in Computer Science*, Vol. 3522, 2005, pp. 631-638.
- [12] H. Sutanto, R. Sharma, V. Varma, The role of exploratory movement in visual servoing without calibration, *Robotics and Autonomous Systems* 23 (1998) 153-169.

QP_TR Trust Region Blob Tracking Through Scale-Space with Automatic Selection of Features

Jingping Jia, Qing Wang, Yanmei Chai, and Rongchun Zhao

School of Computer Science and Engineering
Northwestern Polytechnical University
Xi'an 710072, P.R. China

Abstract. A new approach of tracking objects in image sequences is proposed, in which the constant changes of the size and orientation of the target can be precisely described. For each incoming frame, a likelihood image of the target is created according to the automatically chosen best feature, where the target's area turns into a blob. The scale of this blob can be determined based on the local maxima of differential scale-space filters. We employ the QP_TR trust region algorithm to search for the local maxima of orientational multi-scale normalized Laplacian filter of the likelihood image to locate the target as well as to determine its scale and orientation. Based on the tracking results of sequence examples, the novel method has been proven to be capable of describing the target more accurately and thus achieves much better tracking precision.

1 Introduction

Object tracking in image sequences is the key issue in many computer vision applications such as video surveillance, perceptual user interfaces, object-based video compression and so on. Two major components can be distinguished in a typical visual tracker: target representation and target localization. Gray scale values, shape information, colors, textures, velocity and acceleration are the commonly used object's features. Target localization is the procedure to locate the area that matches the object's feature best, and it can be addressed by particular optimization methods.

It is well known that the success or failure of object tracking is primarily dependent on how distinguishable the feature of an object is from its surroundings and background. In literatures, Shi and Tomasi [1] have pointed out that good features are as important as good tracking algorithms. The degree and discriminability to which a tracker can discriminate the object and background is directly related to the image features used. It is the ability to distinguish between object and background that is most important. In [2] a fixed set of candidate features are assessed in terms of the variance ratio and the best N ones are chosen to produce the likelihood images for tracking. Bohyung Han[3] used PCA to extract the most discriminative feature from the feature set composed of every subset of the RGB and rgb color channels. Stern and Efros [4] chose the best features from 5 features spaces and switch amongst them to improve the tracking performance. All these methods focused on the determination of the best feature from a predefined feature set with finite number of features.

The tracking precision also has much to do with the description of the target's scaling and rotation. Most previous related work [5,6,7] addressed the rotation and scaling by working on some presumably possible discrete values of rotation angle and scale. However, discrete values cannot fit the complex movements of the target. Tyng-Luh and Hwann-Tzong[8] proposed to use a covariance ellipse to characterize an object and adopt a bivariate normal within the ellipse as the weighting function for the features. They dealt with the issue in a continuous manner but the features they used are just predefined color probability and edge density information, which are not guaranteed to be the best one.

In this paper we extend the discrete feature set to a continuous feature space. The best feature is chosen out of this space in terms of the target-background class variance ratio. We also adapt Lindeberg's theory [9] of feature scale selection based on the local maxima of differential scale-space filters to deal with the description of the target and describe the rotation and scaling in the continuous space.

Compared with the popular mean shift [6] method, trust region methods are more effective and can yield better performances [8]. Based on the traditional trust region method, QP_TR algorithm [10] improves the way to get the object function's Hessian matrix and gradient, and achieves even better performance. In this paper, we combine the QP_TR method with the scale space theory and propose a new tracking algorithm in which tracking is implemented as searching for local maxima of the orientational multi-scale normalized Laplacian filter function with the QP_TR method.

2 QP_TR Trust Region Algorithm

Optimization methods based on iterative schemes can be divided into two classes: line-search methods and trust-region methods [8]. For a line-search one, the iterations are determined along some specific directions, e.g., steepest descent locates its iterations by considering the gradient directions. However, a trust region method derives its iterations by solving the corresponding optimization problem in a bounded region iteratively (The optimization problem in the bounded region is called the trust region sub-problem). In fact, line search methods can be considered as special cases of trust-region methods.

2.1 The Basic Trust Region Algorithm

Trust regions methods can be used to solve the unconstrained minimization problem $\min_{\mathbf{x} \in V} f(\mathbf{x})$, where V is a vector space and f is the objective function to be minimized. There are three key elements in any trust region method: 1) trust region radius, to determine the size of a trust region; 2) trust region sub-problem, to approximate the objective function in the region by a model and 3) trust region fidelity, to evaluate the accuracy of an approximating solution.

Given the initial point $\mathbf{x}_0 \in R^n$ and initial trust region radius Δ_0 , the basic trust region method proceeds as following:

1. Initialization: Initialize the constants $\eta_1, \eta_2, \gamma_1, \gamma_2$, satisfying $0 < \eta_1 \leq \eta_2 < 1$, $0 < \gamma_1 \leq \gamma_2 < 1$. Compute $f(\mathbf{x}_0)$ and set $k = 0$;

2. Trust region sub-problem: Choose the norm $\|\cdot\|_k$, determine the trust region $B_k = \{x \in R^n \mid \|x - x_k\|_k \leq \Delta_k\}$ and define a model m_k in B_k which approximates $f(x)$;
3. Step computation: Compute a step s_k which sufficiently reduces m_k and such that $x_k + s_k \in B_k$;
4. Trust region fidelity: Compute $f(x_k + s_k)$ and $r_k = \frac{f(x_k) - f(x_k + s_k)}{m_k(x_k) - m_k(x_k + s_k)}$. If $r_k \geq \eta_1$ then the trial point is accepted and set $x_{k+1} = x_k + s_k$, otherwise set $x_{k+1} = x_k$. Since η_1 is a small positive number, the above run favors a trial point only when m_k approximates f well and yields a large r_k ;

5. Trust region radius update: Set $\Delta_{k+1} \in \begin{cases} [\Delta_k, \infty) & \text{if } r_k \geq \eta_2 \\ [\gamma_2 \Delta_k, \Delta_k) & \text{if } r_k \in [\eta_1, \eta_2) \\ [\gamma_1 \Delta_k, \gamma_2 \Delta_k) & \text{if } r_k < \eta_1 \end{cases}$. That is, when m_k approximates f well and yields a large r_k , and if $r_k \geq \eta_2$ the trust region radius will be expanded for the next iteration. On the other hand if $r_k < \eta_1$ or r_k is negative, it suggests that the objective function f is not well approximated by the model m_k within the current trust region B_k . So s_k is not accepted, the trust region radius will be shrunk to derive more appropriate model and sub-problem for the next iteration.
6. Exit conditions: If Δ_k is smaller than the predefined Δ_{end} or k exceeds the predefined MAX_{iter} , terminate; otherwise set $k = k + 1$ and go to step 2

2.2 QP_TR an Improved Trust Region Method

Based on the basic trust region algorithm, the QP_TR method [10] improves the way to get the model m_k . In QP_TR the object function is approximated by an n dimension quadratic polynomial m_k in B_k :

$$m_k(x_k + s) = m_k(x_k) + \langle g_k, s \rangle + \frac{1}{2} \langle s, H_k s \rangle \quad m_k(x_k) = f(x_k) \tag{1}$$

where m_k is generated by Multivariate Lagrange Interpolation. The Hessian Matrix H_k at x_k is extracted from m_k instead of using the ‘‘BFGS-update’’. This avoids the poor approximation of H_k when the curvature of f is changing fast. Furthermore, g_k is also extracted from m_k , $g_k = \nabla_x m_k(x_k)$. That avoids the introduced error when using forward finite differences. Thus the trust region sub-problem is to compute an $s_k \in B_k$ such that $s_k = \arg \min_{\|s\|_k \leq \Delta_k} \psi_k(s) \equiv \langle g_k, s \rangle + \frac{1}{2} \langle s, H_k s \rangle$.

In our tracking context, we set $\eta_1 = 0.1$, $\eta_2 = 0.7$, $\gamma_1 = \gamma_2 = 0.5$.

3 Target Model

Our goal in this section is to develop an efficient method that automatically chooses the best feature for tracking. Features used for tracking only need be locally discriminative, in that the object only needs to be clearly separable from its immediate surroundings. We represent target appearance using histograms of color filter bank responses applied to R, G, and B pixel values within local image windows. This representation is chosen since it is relatively insensitive to variation in target appearance due to viewpoint, occlusion and non-rigidity. In [2] a fixed set of candidate features are evaluated and the best one is chosen to produce the likelihood image. Based upon [2] but different from it, we extend the candidate features in a continuous manner.

In this paper, the candidate features are composed of linear combinations of R, G, B pixel values. Specifically, we have chosen the following candidate feature:

$$F = \omega_1 R + \omega_2 G + \omega_3 B \quad \omega_i \in \mathbf{R} \quad (2)$$

where F is the linear combination of R, G, and B with real coefficients except for $(\omega_1, \omega_2, \omega_3) = (0, 0, 0)$. Many common features from the literatures are included in the candidate space, such as the raw values of R, G, and B, intensity R+G+B, approximate chrominance features such as R-B, and so-called excess color features such as 2G-R-B. All features are normalized into the range 0 to 255 and discretized into histograms of length 255.

We follow [2] to use a “center-surround” approach to sampling pixels covering the object and background. A rectangular set of pixels covering the object is chosen to represent the object pixels, while a larger surrounding ring of pixels is chosen to represent the background. For an inner rectangle of dimensions $h_i \times w_i$ pixels, an outer margin of width $\max(h_i, w_i)$ pixels forms the background sample. We use the object pixels to get the target histogram H_{obj} for a candidate feature and the background pixels to get the background histogram H_{bg} . We form an empirical discrete probability distribution $p(i), i = 1 \dots 255$ for the object, and $q(i)$ for the background by normalizing each histogram by the number of elements in it.

From any candidate features, we can create a new feature that is “tuned” to discriminate between object and background pixels. The tuned feature is formed as the log likelihood ratio of the class conditional candidate feature distributions. The log likelihood ratio [3] of a feature value i is given by

$$L(i) = \max \left(-1, \min \left(1, \log \frac{\max(p(i), \delta)}{\max(q(i), \delta)} \right) \right) \quad (3)$$

where δ is a small value that prevents dividing by zero or taking the log of zero (we choose it to 0.001). The nonlinear log likelihood ratio maps object/background distributions into positive values for colors distinctive to the object and negative for colors associated with the background. Colors shared by both object and background tend towards zero. Back-projecting these log likelihood ratio values into the image produces a likelihood image suitable for tracking, as shown in Fig 1.

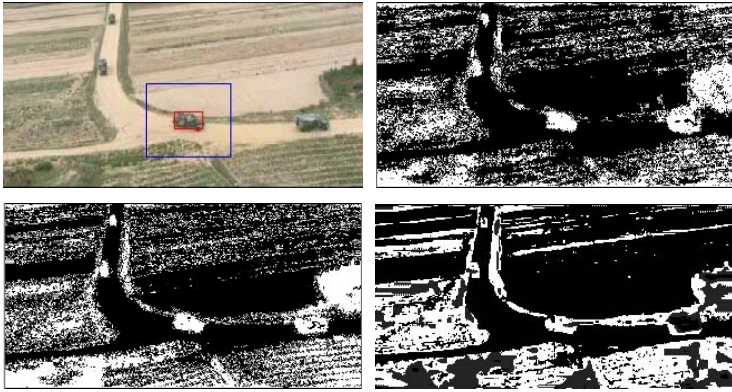


Fig. 1. Likelihood images by three methods

The separability that $L(i)$ induces between object and background classes can be measured using the two-class variance ratio. The variance of $L(i)$ with respect to object class distribution $p(i)$ is calculated as:

$$\text{var}(L; p) = E[L^2(i)] - (E[L(i)])^2 = \sum_i p(i)L^2(i) - \left[\sum_i p(i)L(i) \right]^2 \tag{4}$$

and similarly for background class distribution $q(i)$ [2]. Then the variance ratio can be defined as:

$$VR(L; p, q) \equiv \frac{\text{var}(L; (p+q)/2)}{\text{var}(L; p) + \text{var}(L; q)} \tag{5}$$

The higher the ratio is, the wider the object and background are separated [2]. This means the triple of $(\omega_1, \omega_2, \omega_3)$ which produces the highest $VR(L; p, q)$, corresponds to the best feature. To get this best triple we can define an object function $\psi(\omega_1, \omega_2, \omega_3)$ that takes a triple $(\omega_1, \omega_2, \omega_3)$ as the parameter, calculates $p(i), q(i), L(i)$, and returns $-VR(L; p, q)$. Apply the QP_TR method on $\psi(\omega_1, \omega_2, \omega_3)$ and we can get $(\omega_1, \omega_2, \omega_3)_{best}$. Introduce $(\omega_1, \omega_2, \omega_3)_{best}$ into (2) and we can get the best feature. Using the best feature we follow (3) to get the best “tuned” feature $L_{best}(i)$ which can be back-projected to produce the likelihood image.

Figure 1 shows the weight images calculated with three methods. The upper right one is by our method, the lower left one is by the method in [2], while the lower right is by that in [11].

Different from [2], we use real ω_i instead of integer ones. And we choose the best ω_i in the continuous space rather than from a finite set. This improvement enables us to get more approximate feature. For example, in Fig.1 the likelihood image using our method has a $VR=1.89425$, which is larger than that of the likelihood image

produced by method in [2], which is 1.60777 . The corresponding triple $(\omega_1, \omega_2, \omega_3)$ is $(-5.02673, 4.9104, 0.959966)$.

4 Scale-Space Blob

During tracking, the size of the target will change with the distance to the camera. The target itself may rotate. Tracking algorithm should adapt to these kinds of change and describe them precisely. Most previous related work [5,6,7] addresses rotation and scaling by working on some presumably possible discrete values of rotation angle and scale. For example, for each frame the tracker in [6] tries two sizes of plus and minus ten percent of the previous size to guess the current size. However, it's difficult to adapt to the changes in size by using only these discrete values (We will show this later). Aiming at this shortcoming, we propose to solve the problem in the continuous scale space and describe the scaling and rotation in a continuous manner.

The work of Lindeberg [8] provided an elegant theory for selecting the best scale for describing features in an image. Given any continuous function $f : R^D \rightarrow R$ and a Gaussian kernel with scale t , $g : R^D \times R_+ \rightarrow R$, $g(\mathbf{x};t) = \frac{1}{(2\pi t)^{N/2}} e^{-(x_1^2 + \dots + x_D^2)/(2t)}$, the scale-space representation of f is its convolution with g , i.e., $L : R^D \times R_+ \rightarrow R$, $L(\cdot;t) = g(\cdot;t) * f(\cdot)$ with various scale t . The γ -normalized derivative operator is defined as $\partial_\xi = t^{\gamma/2} \partial_x$. A very good property of the γ -normalized derivative of L is the perfect scale invariance as follows:

Consider two functions f and \tilde{f} related by $f(\mathbf{x}) = \tilde{f}(\mathbf{x})$ and define the scale-space representation of f and \tilde{f} in the two domains by

$$\begin{aligned} L(\cdot;t) &= g(\cdot;t) * f(\cdot) \\ \tilde{L}(\cdot;\tilde{t}) &= g(\cdot;\tilde{t}) * \tilde{f}(\cdot) \end{aligned} \tag{6}$$

where the spatial variables and the scale parameters are transformed according to

$$\begin{aligned} \tilde{\mathbf{x}} &= s\mathbf{x} \\ \tilde{t} &= s^2t \end{aligned} \tag{7}$$

Then L and \tilde{L} are related by $L(\mathbf{x};t) = \tilde{L}(\cdot;\tilde{t})$ and the m -th order spatial derivatives satisfy $\partial_{x^m} L(\mathbf{x};t) = s^m \partial_{\tilde{x}^m} \tilde{L}(\tilde{\mathbf{x}};\tilde{t})$. For γ -normalized derivatives defined in the two domains by

$$\begin{aligned} \partial_\xi &= t^{\gamma/2} \partial_x \\ \partial_{\tilde{\xi}} &= \tilde{t}^{\gamma/2} \partial_{\tilde{x}} \end{aligned} \tag{8}$$

we have $\partial_{\xi^m} L(\mathbf{x};t) = s^{m(1-\gamma)} \partial_{\tilde{\xi}^m} \tilde{L}(\tilde{\mathbf{x}};\tilde{t})$. From this relation it can be seen that, when $\gamma = 1$, $\partial_{\xi^m} L(\mathbf{x};t) = \partial_{\tilde{\xi}^m} \tilde{L}(\tilde{\mathbf{x}};\tilde{t})$. That is, if the γ -normalized derivative of f

assumes a local maximum at $(x_0; t_0)$ in the scale-space, the γ -normalized derivatives of \tilde{f} will also assume a local maximum at $(s^2x_0; s^2t_0)$. Based on this property we can choose appropriate combination of the γ -normalized derivatives to determine the scale of some structure in the data.

A gray image can be seen as a two dimensional function. That is, $D = 2$, and $f : R^2 \rightarrow R$. When $\gamma = 1$, $(t^\gamma \nabla^2 L)^2 = (t \nabla^2 L)^2 = (t(L_{xx} + L_{yy}))^2$ reflects the details of blobs in an image. We call $\phi(x, y, t) = (t(L_{xx}(x, y) + L_{yy}(x, y)))^2$ the multi-scale normalized Laplacian filter function. With different scale values t , $\phi(x, y, t)$ achieves local maxima at blobs with different sizes. The gray image in Fig 2 contains two blobs. With $t = 90$, $\phi(x, y, t)$ assumes the local maximum at the center of the smaller one, while with $t = 391$, $\phi(x, y, t)$ assumes another local maximum at the center of the larger one. So the locations of the blobs as well as their scales can be determined by examining the local maxima of $\phi(x, y, t)$ at various positions and scales.

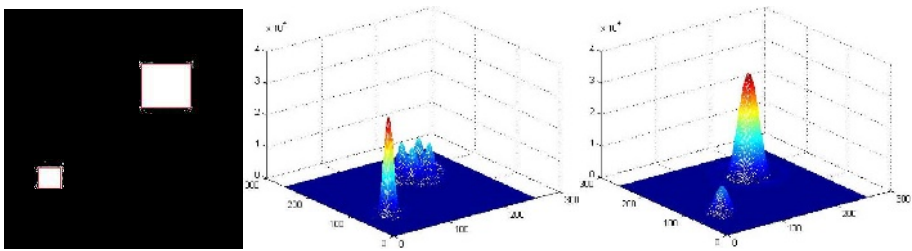


Fig. 2. For blobs with different sizes, $\phi(x, y, \alpha, t)$ assumes local maxima at different positions with different scales, Left: Original image Middle: The response of $\phi(x, y, \alpha, t)$ with $t = 90$ Right: The response of $\phi(x, y, \alpha, t)$ with $t = 391$

When the blob’s orientation is under the consideration we define the orientational multi-scale normalized Laplacian filter function $\phi(x, y, \alpha, t)$ as follows:

$$\phi(xc, yc, \alpha, t) = \left(t \left(L_{x'x'}(xc, yc) + L_{y'y'}(xc, yc) \right) \right)^2$$

$$x' = (x - xc) \cos \alpha + (y - yc) \sin \alpha + xc$$

$$y' = -(x - xc) \sin \alpha + (y - yc) \cos \alpha + yc$$

$$0 \leq x \leq image \ width, 0 \leq y \leq image \ height \tag{10}$$

$\phi(x, y, \alpha, t)$ will assume a maximum only when α reflects the correct orientation of the blob. As shown in Fig 3, for the blob in (a) only when $\alpha = 0$ (which means no rotation), does $\phi(x, y, \alpha, t)$ assume the maximum. So we can use $\phi(x, y, \alpha, t)$ to determine the positions, orientations of blobs as well as their scales.

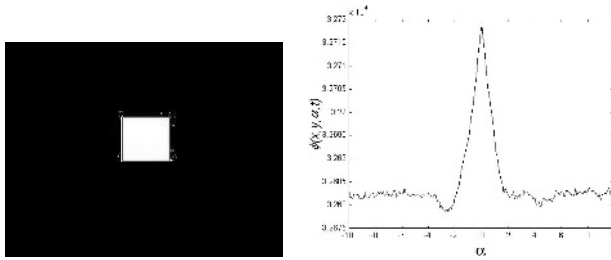


Fig. 3. $\phi(x, y, \alpha, t)$ with different α

5 Algorithm Summary

To summarize, we track targets by calculating the likelihood image of target for each incoming frame and search for the local maximum of $\phi(x, y, \alpha, t)$ in the scale space of the likelihood image by QP_TR trust region method. The algorithm proceeds as follows:

1. Target initialization: determine the width and height of the target, denoted as w_t and h_t . The width and height of the frame are w and h respectively. Using the method in section 3 to determine the best feature to use, record the corresponding triple $(\omega_1, \omega_2, \omega_3)_{best}$ and $L_{best}(i)$.
2. Resize the frame so that the ratio of width to height will be $\rho = wh_t / h / w_t$. If we hold w fixed the height of the frame will be w / ρ .
3. Compute the likelihood image for the resized frame using $(\omega_1, \omega_2, \omega_3)_{best}$ and $L_{best}(i)$.
4. Initialize a vector $\mathbf{x}_0 = (x_{prev}, y_{prev}, \alpha_{prev}, t_{prev})^T$, where (x_{prev}, y_{prev}) is the center of the target in the previous frame, α_{prev} the previous orientation parameter and t_{prev} the scale parameter in the previous frame. Set the initial trust region radius $\Delta_0 = 9$, the minimum radius $\Delta_{end} = 0.1$ and the maximum iteration $MAX_{iter} = 1000$.
5. Run the QP_TR algorithm for $f(\mathbf{x}) = -\phi(x, y, \alpha, t)$ and get $\mathbf{x}_{opt} = (x_{opt}, y_{opt}, \alpha_{prev}, t_{opt})^T$ which minimizes $f(\mathbf{x})$, where (x_{opt}, y_{opt}) is the center of the target in the current frame with α_{opt} the orientation parameter and t_{opt} its scale parameter. Go to step 2) until the track is over.

6 Examples

To verify the efficiency of our method, we apply it to many sequences and compare with other tracking algorithms. At first, we evaluated the performance of feature selection of our novel method with [10], as shown in Fig 4. The number of evaluated features by our method is reduced a lot whereas the number is a fixed one in ref [10].

At the same time, the discriminability between the object and background has been increased compared with ref [10]. From the results, we find that our method is more effective and efficient than that of method in [10] in term of the computation of feature selection and discriminability of feature for object description, e.g., VR.

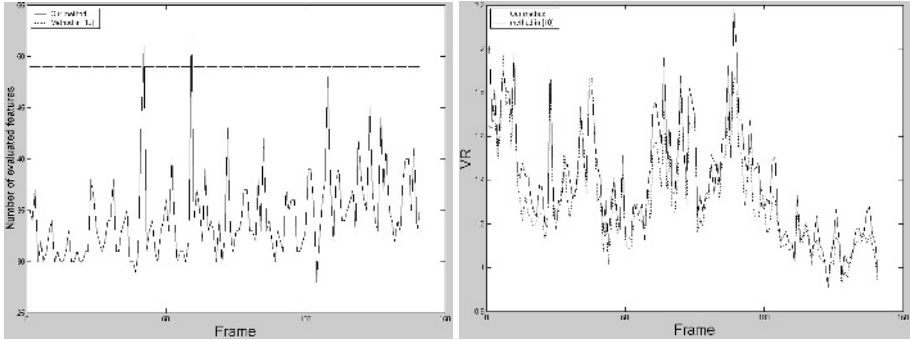


Fig. 4. The curves of the number of evaluated features and VR of the “car” sequence (from 1 to 142 frames)

Besides the above-mentioned good performance of feature extraction for object, the proposed method is proven to be accurate to describe the object and can yield better tracking result than those of refs [3, 5, 8, 12] by the following experimental results. Fig 5 shows the tracking result of the “pedestrian” sequence and performance comparison with the bandwidth mean shift tracker [3]. Since the target in this sequence does not rotate, we hold the orientation parameter α fixed. In the bandwidth mean shift tracker, the scale parameter can take only three discrete values so that there is



Fig. 5. Tracking results of a pedestrian. Upper row: Using the method presented in this paper. Lower row: Using the bandwidth mean shift tracker. Only the 3rd, 217th and 317th frame.

much error in describing the target's size, which results in error of the target localization (as shown in the lower row). The upper row shows our method's result. It can be seen that the rectangle shrinks with the target and describes the target size accurately.

In Fig 6 we verify our method's ability to cope with the rotation of target. The car rotates throughout the sequence. With our method the car is consistently tracked, both in location and in angle.



Fig. 6. Tracking results of the “car” sequence. Only the 3, 27, 52, 75, 99 and 124 frame are shown.

In Fig 7 we verify our method's ability to cope with the zoom of target. The cup rotates and zooms in throughout the “cup2” sequence. As can be seen, our method (upper row) of optimizing over a four dimensional continuous space to capture the changes in the cup's scale and orientation is much more effective than the multi-degree of freedom mean shift tracker [12] (in the lower row) which tries three discrete angles for each frame to deal with the rotation. Meanwhile the changing curves of the scale parameter and rotation angle during tracking are shown in Fig 8.

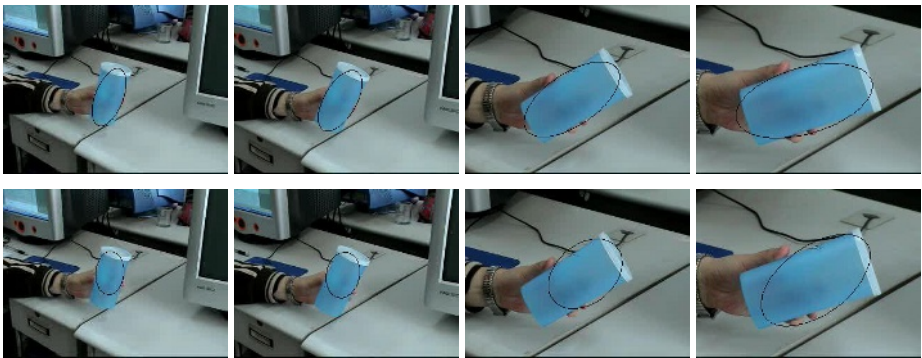


Fig. 7. Tracking results of the “cup2” sequence. Upper row: Using the method presented in this paper. Lower row: Using the multi-degree of freedom mean shift tracker. Only the 172nd, 204th, 317th and 364th frame are shown.

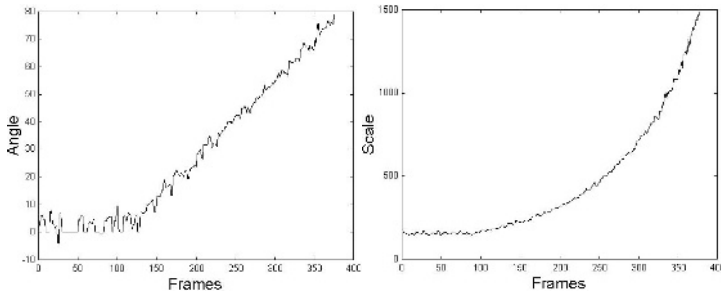


Fig. 8. The curves of the rotation angle and scale parameter of the “cup2” sequence

7 Conclusion

In this paper, we proposed a new target tracking algorithm to solve the problem of determining the scale and orientation, by the combination of Lindeberg’s scale space theory with the QP_TR trust region method. First, the best feature that best discriminates the target and background is automatically determined out of the continuous candidate feature space. Each target corresponds to a specific blob in the likelihood image. Then we introduce the orientational multi-scale normalized Laplacian filter function to detect the blobs in gray images. Conceptually, the scale space is generated by convolving the likelihood image with a filter bank of spatial LOG filters. Explicit generation of these convolutions would be very expensive and only able to evaluate at finite discrete parameters. However, by using the QP_TR trust region method, we can search for the local maximum of the object function in the continuous scale space much more efficiently, where the local maximum corresponds to a blob, i.e. the target. In this way we fulfill the precise tracking of targets. Experimental results demonstrate our new algorithm’s ability to adapt to objects’ complex movements with much better tracking precision.

Acknowledgment

The work described in the paper was supported by National Natural Science Fund (60403008), Aerospace Innovation Fund, and Aviation Science Fund (03I53065), P.R. China.

Reference

- [1] J. Shi and C. Tomasi. “Good features to track”, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, 1994, pp.593-600.
- [2] Robert T. Collins, Yanxi Liu, Marius Leordeanu, “Online selection of discriminative Tracking features”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.27, No.10, October 2005, pp1631-1643.
- [3] Bohyung Han, Larry Davis, “Object tracking by adaptive feature extraction”, *Proceedings of the 2004 International Conference on Image Processing*, Singapore, 2004, Vol.3, pp.1504-1504.

- [4] H. Stern and B. Efron, "Adaptive color space switching for face tracking in multi-colored lighting environment", *Proceedings of the International Conference on Automatic Face and Gester Recognition*, Washington DC, USA, 2002, pp.249-254.
- [5] Comaniciu, D., Ramesh, V. and Meer, P., "Real-time tracking of non-rigid objects using mean shift", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Vol II, 2000, pp.142-149.
- [6] Comaniciu, D., Ramesh, V. and Meer, P., "Kernel-based object tracking", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.25, No.5, May 2003, pp.564-577.
- [7] Jingping Jia, Rongchun Zhao, "Tracking of objects in image sequences using bandwidth matrix mean shift algorithm", *Proceedings of 7th International Conference on Signal Processing*, vol 2, August 2004, pp918-921
- [8] Tyng-Luh Liu, Hwang-Tzong Chen, "Real-time tracking using trust-region methods", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.26, No.3, March 2004, pp397-402.
- [9] Lindeberg, T., "Feature detection with automatic scale selection", *International Journal of Computer Vision*, Vol.30, No.2, Nov 1998, pp.79-116.
- [10] Frank Vanden Berghen, "Intermediate report on the development of an optimization code for smooth, continuous objective functions when derivatives are not available", http://www.optimization-online.org/DB_HTML/2003/08/704.html.
- [11] Bradski, G.R., "Computer vision face tracking for use in a perceptual user interface", *IEEE Workshop on Applications of Computer Vision*, Princeton, NJ, 1998, pp.214-219.
- [12] Jingping Jia, Yanning Zhang, etc, "Tracking of objects in image sequences using multi-freeness mean shift algorithm", *Proceedings of the 4th International Conference on Machine Learning and Cybernetics*, Vol.8, August 2005, pp5133-5138.

Human Posture Analysis Under Partial Self-occlusion

Ruixuan Wang and Wee Kheng Leow

School of Computing, National University of Singapore,
3 Science Drive 2, Singapore 117543, Singapore
{wangruix, leowwk}@comp.nus.edu.sg

Abstract. Accurate human posture estimation from single or multiple images is essential in many applications. Two main causes of difficulty to solve the estimation problem are large number of degrees of freedom and self-occlusion. Tree-structured graphical models with efficient inference algorithms have been used to solve the problem in a lower dimensional state space. However, such models are not accurate enough to formulate the problem because it assumes that the image of each body part can be independently observed. As a result, it is difficult to handle partial self-occlusion. This paper presents a more accurate graphical model which can implicitly model the possible self-occlusion between body parts. More important, an efficiently approximate inference algorithm is provided to estimate human posture in a low dimensional state space. It can deal with partial self-occlusion in posture estimation and human tracking, which has been shown by the experimental results on real data.

1 Introduction

Human posture estimation and human tracking try to accurately recover human posture from single or multiple images [8][9][13]. Accurately recovering posture is essential in many applications such as human-computer interaction, vision-based sport coaching, and physical rehabilitation.

Top-down approach is often used to estimate human posture, in which a 3D or 2D human body model is often required to generate a synthetic 2D image of the corresponding human posture. By measuring the similarity between the synthetic 2D image and the input image, the posture estimation can be updated iteratively. In general, there are many local minima in such an optimization problem, such that continuous local optimization methods are not effective [22]. To deal with the difficulty, prior motion models are often used to constrain the search space during optimization [17], although it is limited to estimating postures similar to those in the motion models. Another way is to find multiple local minima and choose the best one from them [3][23], but it requires more computation and also cannot guarantee to find the global minimum. In comparison, sampling method [12] may find the global minimum in a low state space, but directly sampling in the state space of body posture is infeasible because of the large number of degrees of freedom (e.g., 30) of human body.

By observing that the human body is in fact tree-structured, researchers often formulate the estimation problem by a tree-structured graphical model [24][11][26][7][18]. In the model, every body part is encoded by one node in the graph, and every edge connecting two nodes indicates that there are relationships between the two parts. Efficient

inference algorithms exist (e.g., BP [28]) to recover the low dimensional (e.g., 6) pose of every body part. More importantly, sampling methods [7][24][11][26] can be used in the low dimensional pose space of each body part.

However, it is not accurate enough to formulate the problem by a tree-structured graphical model. In this model, it assumes that the image of each body part can be independently observed, while self-occlusion between body parts often happens in human motion. In such case, the image of one body part can not be independently observed because it may be partially or fully occluded by other body parts. Sigal *et al.* [20] tried to deal with partial self-occlusion by learning the likelihood of the observed image conditioned on the pose state of each body part. But learning is often a complex process and it is not easy to collect training images. What is more, such learned likelihood functions are limited to approximately estimating a small set of postures. Lee *et al.* [14] and Hua *et al.* [9] used detected part candidates to obtain proposal distributions for some body parts, which are then used to help approximately estimate postures even under partial self-occlusion. Good proposal distributions are essentially important in their methods. Sudderth *et al.* [25] explicitly modelled self-occlusion using factor graph in which one binary hidden variable is required for each image pixel. However, the large number of hidden variables inside the model make the inference algorithm more complicated.

In order to deal with partial self-occlusion in posture estimation, we use a more accurate graphical model by explicitly inserting a set of hidden variables between the state of human posture and the input image observation. Each hidden variable represents the 3D shape and the appearance of one body part, and the image observation of every body part depends on all the hidden variables. The possible self-occlusion between body parts can be implicitly modelled by the relative position between the 3D shapes of parts. In addition, the non-penetration between body parts can be explicitly modelled in the middle level of the model. More important, based on the new model, a novel and efficient approximate inference algorithm is developed to accurately estimate each body part's pose in a lower (i.e. 6) dimensional space. This algorithm is an annealed iteration process. In each iteration, conditional marginal distribution of each body part is estimated based on the estimation results of previous iteration. The relationships between body parts' states and the relationships between parts' states and the image observation are updated by an annealing factor in each iteration. Such annealed process can help to find the true posture with more probability even if the initial posture is far from the truth. This inference algorithm, without any learning process, can deal with partial self-occlusion in 2D posture estimation and human tracking, which has been shown by the experimental results on real data.

2 Related Work

In general there are two types of approaches to the related human posture estimation and articulated human tracking problems: top-down and bottom-up. Compared with top-down approach introduced above, bottom-up approach can avoid the need for explicit initialization and 3D or 2D body modelling and rendering. It directly recovers human posture from images by exemplar based method or non-linear mapping based method.

The exemplar based method [16][2] searches for exemplar images similar to the input image from a set of stored exemplars, and uses the known 3D posture of the exemplar as the estimated posture. Since multiple body postures may have very similar corresponding images, this method often outputs multiple 3D body posture estimations for the input image. Much computation can be saved by constructing a distance-approximating embedding [2], such that the similarity measurement between images can be efficiently computed in the embedded low space. Because the exemplars record only a limited number of body postures, this method may not obtain good posture estimations if the body posture in the input image is different from those in the exemplars.

The non-linear mapping based method learns a nonlinear mapping function that represents the relationships between body image features and the corresponding 3D body postures. During learning, a rich set of image features (e.g., silhouette [6], histogram of shape context [1]) are extracted from each training image as the input, and the output is the known 3D posture in the corresponding training image. Agarwal and Triggs [1] used relevance vector machine to learn a nonlinear mapping function that consists of a set of weighted basis functions. Rosales et al. [19] used a special combination of sigmoidal and linear functions to learn a set of forward mapping functions by one EM technique. In addition, by embedding the manifold of one type of human motion into a lower dimensional space, and learning the two non-linear mappings between the embedded manifold and both visual input (e.g., silhouette) space and 3D body pose space, 3D body pose can be estimated from each input image by the two mapping functions [6]. The mapping based method can directly estimate body posture from a single input image, but it is often limited to recovering the body postures which are similar to the 3D postures in the training images.

Recently, the combination of top-down and bottom-up approaches has also been used to estimate postures [10][14][9][15]. In general it firstly applies low-level feature detectors (e.g., rectangle detectors [10]) to generate a set of candidates of body parts, then applies some prior knowledge or constraints (e.g., kinematic constraints) to search for good candidates and find the best 2D posture. To list a few, Mori [15] used superpixels as the element to represent the input image. Based on the boundaries of superpixels and constraints (appearance and width consistency, kinematic constraints) between body parts, a rough 2D posture configuration was obtained. Hua [9] used the detected candidates of some body parts to form importance function for later belief propagation.

Note that both types of approaches can be used in human tracking problem. Compared to CONDENSATION [12] which efficiently combines top-down approach into a probabilistic framework for human tracking, Sminchisescu [21] recently proposed a probabilistic framework in which conditional density can be propagated temporally in discriminative (bottom-up), continuous chain models.

3 Problem Formulation

A human skeleton model (Figure 1(a)) is used to represent body joints and bones, and a triangular mesh model (Figure 1(b)) is used to represent the body shape. Each vertex in the mesh is attached to the related body part. The relative bone length and part width to a standard human model are used to represent each body part's shape size.

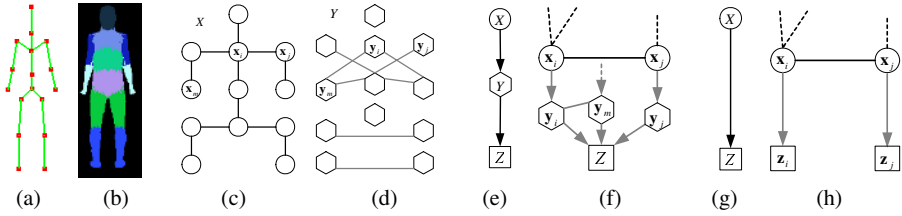


Fig. 1. Human body model and graphical model. (a) Human skeleton model. (b) Each vertex in the mesh model is assigned to one specific body part. (c) A tree-structured graph represents human posture \mathcal{X} . Each node represents one body part \mathbf{x}_i and the edge between two nodes represents the potential relationship between them. (d) Each node in \mathcal{Y} represents one 3D body part \mathbf{y}_i and the edge between nodes represents the non-penetration relationship between them. (e) and (f) represent the graphical model we used. (g) and (h) represent the tree-structured graphical model.

Human body posture \mathcal{X} is represented by a set of body parts' poses $\mathcal{X} = \{\mathbf{x}_i | i \in \mathcal{V}\}$ (Figure 1(c)), where \mathcal{V} is the set of body parts. The pose $\mathbf{x}_i = (\mathbf{p}_i, \boldsymbol{\theta}_i)$ represents the i^{th} body part's 3D position \mathbf{p}_i and 3D orientation $\boldsymbol{\theta}_i$. Given the shape size of each body part, a synthetic 3D body part $\mathbf{y}_i = f(\mathbf{x}_i)$ (Figure 1(d)) is generated for each part's pose \mathbf{x}_i , where f represents (but is not limited to) a deterministic process. By projecting the synthetic 3D body $\mathcal{Y} = \{\mathbf{y}_i | i \in \mathcal{V}\}$, a synthetic image observation can be generated. During posture estimation, each synthetic image observation will be used to compare with a real input image observation \mathcal{Z} . The relationship between \mathcal{Y} and \mathcal{Z} is represented by the observation function $\phi(\mathcal{Y}, \mathcal{Z})$. In addition due to the articulation, every pair of adjacent body parts \mathbf{x}_i and \mathbf{x}_j must be connected. Such kind of kinematic constraint is enforced by the potential function $\psi_{ij}(\mathbf{x}_i, \mathbf{x}_j)$. Denote \mathcal{E} as the set of adjacent body parts \mathbf{x}_i and \mathbf{x}_j , i.e., $(i, j) \in \mathcal{E}$. Another kind of constraint is that body parts cannot penetrate each other, which can be enforced by potential function $\varphi_{im}(\mathbf{y}_i, \mathbf{y}_m)$. Denote \mathcal{E}' as the set of part pair \mathbf{y}_i and \mathbf{y}_m , i.e., $(i, m) \in \mathcal{E}'$.

A graphical model (Figure 1(e) and 1(f)) is used to represent all the relationships introduced above. Note that this model is different from the tree-structured graphical model that is generally used by other researchers [7][9]. In the tree-structured model (Figure 1(g) and 1(h)), it assumes that the image \mathbf{z}_i of each body part i can be independently observed such that the relationship between \mathbf{x}_i and \mathbf{z}_i can be easily evaluated using local observation. However in general, self-occlusion between body parts often happens in human motion. In such a case, local observation \mathbf{z}_i can not be observed independently and only the whole body's image observation \mathcal{Z} can. In our graphical model, a middle level (\mathbf{y}_i) is inserted between \mathbf{x}_i and \mathcal{Z} in order to precisely model the image generation process. Each hidden variable \mathbf{y}_i represents the 3D shape and appearance of one body part, and the image observation of every body part depends on all the hidden variables. This is different from tree-structured model in which every part's observation depends only on the part's state. In our graphical model, the possible self-occlusion between body parts can be implicitly modelled by the relative position between the 3D shapes of parts. In addition, the non-penetration relationship between 3D body parts can be enforced by potential function $\varphi_{im}(\mathbf{y}_i, \mathbf{y}_m)$, while such relationship cannot be modelled in tree-structured graphical model.

The problem is to infer \mathcal{X} and corresponding \mathcal{Y} from \mathcal{Z} . From the structure of the graphical model (Figure 1(e) and 1(f)), the posterior distribution $p(\mathcal{X}, \mathcal{Y} | \mathcal{Z})$ can be factorized as

$$\begin{aligned} p(\mathcal{X}, \mathcal{Y} | \mathcal{Z}) &\propto p(\mathcal{Z} | \mathcal{Y}) p(\mathcal{Y} | \mathcal{X}) p(\mathcal{X}) \\ &\propto \phi(\mathcal{Y}, \mathcal{Z}) \prod_{(i,m) \in \mathcal{E}'} \varphi_{im}(\mathbf{y}_i, \mathbf{y}_m) \prod_{i \in \mathcal{V}} \delta(f(\mathbf{x}_i) - \mathbf{y}_i) \prod_{(i,j) \in \mathcal{E}} \psi_{ij}(\mathbf{x}_i, \mathbf{x}_j), \end{aligned} \quad (1)$$

where $\delta(\cdot)$ is the Dirac's delta function because \mathbf{y}_i is a deterministic function of \mathbf{x}_i . Now the objective is to find the maximum *a posteriori* estimation \mathcal{X}^* and corresponding \mathcal{Y}^* which make $p(\mathcal{X}, \mathcal{Y} | \mathcal{Z})$ maximum.

4 Inference Algorithm

Instead of directly inferring \mathcal{X} and \mathcal{Y} from (1), we calculate the conditional marginal distribution $p(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$. Unfortunately, due to the complex structure of the graphical model, the generally used efficient belief propagation algorithm cannot be used to calculate $p(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$. Here we develop an approximate inference algorithm to calculate the maximum $p(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$ by introducing into it the idea of simulated annealing. This algorithm is an annealed iteration process. In each iteration, every $p(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$ is estimated based on the estimation of the other body parts from the previous iteration and the real input image \mathcal{Z} . Since the estimation is not accurate in the first several iterations, the relationships between different body parts are relaxed and loose at first, and then become more and more restricted with respect to iteration. The update of relationships is realized by an annealing factor. In the following, we first explain how annealing factor is introduced to the iterations. After that, we will design the potential functions and observation functions.

Denote $\tilde{p}^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$ as the estimation of the true $p^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z}) = \{p(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})\}^{\lambda_n}$ at iteration n , where $n = 0, \dots, N-1$ and $\lambda_{N-1} > \dots > \lambda_1 > \lambda_0$. When λ_n increases (linearly or exponentially) with respect to iteration n , the MAP estimation \mathbf{x}_i^* and \mathbf{y}_i^* will emerge more and more clearly, because $\{p(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})\}^{\lambda_n}$ is much larger at \mathbf{x}_i^* and \mathbf{y}_i^* than at other \mathbf{x}_i values. For two adjacent iterations, we have the following approximations:

$$\begin{aligned} \tilde{p}^{(n+1)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z}) &\approx \{p^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})\}^{\lambda_{n+1}/\lambda_n} \\ &\approx \{\tilde{p}^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \hat{\mathcal{X}}_{-i}^n, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z})\}^{\lambda_{n+1}/\lambda_n}, \end{aligned} \quad (2)$$

$$\begin{aligned} p^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z}) &\approx \int_{\mathcal{X}_{-i}, \mathcal{Y}_{-i}} \{\tilde{p}^{(n)}(\mathbf{x}_i, \mathbf{y}_i, \mathcal{X}_{-i}, \mathcal{Y}_{-i} | \mathcal{Z})\} \\ &= \int_{\mathcal{X}_{-i}, \mathcal{Y}_{-i}} \{\tilde{p}^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{X}_{-i}, \mathcal{Y}_{-i}, \mathcal{Z}) p^{(n)}(\mathcal{X}_{-i}, \mathcal{Y}_{-i} | \mathcal{Z})\} \\ &\approx \int_{\mathcal{X}_{-i}, \mathcal{Y}_{-i}} \{\tilde{p}^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \hat{\mathcal{X}}_{-i}^n, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z}) \tilde{p}^{(n)}(\mathcal{X}_{-i}, \mathcal{Y}_{-i} | \mathcal{Z})\} \\ &= \tilde{p}^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \hat{\mathcal{X}}_{-i}^n, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z}), \end{aligned} \quad (3)$$

where \mathcal{X}_{-i} is the set of body parts' poses except \mathbf{x}_i , and \mathcal{Y}_{-i} is the set of 3D body parts except \mathbf{y}_i . $\hat{\mathcal{X}}_{-i}^n$ and $\hat{\mathcal{Y}}_{-i}^n$ are the corresponding estimations at iteration n . In (2), $p^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$ is approximated by $\tilde{p}^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \hat{\mathcal{X}}_{-i}^n, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z})$. Although it needs to be theoretically explored for such approximation, the approximation (3) may be reasonable at least due to the following observations. During the first several iterations, the relationship between part \mathbf{x}_i and the other parts \mathcal{X}_{-i} are so loose that they are independent. The second observation is that when iteration n is large enough, $p^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$ will become a Dirac's delta like function. In both cases, the $\tilde{p}^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \hat{\mathcal{X}}_{-i}^n, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z})$ can be used to exactly represent $p^{(n)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$.

From (1) and (2), we can get

$$\begin{aligned} &\tilde{p}^{(n+1)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z}) \tag{4} \\ &\propto \alpha \phi^{(n+1)}(\mathbf{y}_i, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z}) \prod_{m \in \Gamma'(i)} \varphi_{im}^{(n+1)}(\mathbf{y}_i, \hat{\mathbf{y}}_m^n) \delta(f(\mathbf{x}_i) - \mathbf{y}_i) \prod_{j \in \Gamma(i)} \psi_{ij}^{(n+1)}(\mathbf{x}_i, \hat{\mathbf{x}}_j^n) \\ &\propto \alpha \{\phi^{(n)}(\mathbf{y}_i, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z}) \prod_{m \in \Gamma'(i)} \varphi_{im}^{(n)}(\mathbf{y}_i, \hat{\mathbf{y}}_m^n) \delta(f(\mathbf{x}_i) - \mathbf{y}_i) \prod_{j \in \Gamma(i)} \psi_{ij}^{(n)}(\mathbf{x}_i, \hat{\mathbf{x}}_j^n)\}^{\lambda_{n+1}/\lambda_n}, \end{aligned}$$

where $\Gamma(i) = \{k | (i, k) \in \mathcal{E}\}$ is the neighbor of body part i , and similarly for $\Gamma'(i)$. α is a normalizing factor including potential functions related to the other body parts. From (4), conditional marginal distribution can be updated iteratively. Also, we can get

$$\phi^{(n+1)}(\mathbf{y}_i, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z}) \propto \{\phi^{(n)}(\mathbf{y}_i, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z})\}^{\lambda_{n+1}/\lambda_n}, \tag{5}$$

$$\varphi_{im}^{(n+1)}(\mathbf{y}_i, \hat{\mathbf{y}}_m^n) \propto \{\varphi_{im}^{(n)}(\mathbf{y}_i, \hat{\mathbf{y}}_m^n)\}^{\lambda_{n+1}/\lambda_n}, \tag{6}$$

$$\psi_{ij}^{(n+1)}(\mathbf{x}_i, \hat{\mathbf{x}}_j^n) \propto \{\psi_{ij}^{(n)}(\mathbf{x}_i, \hat{\mathbf{x}}_j^n)\}^{\lambda_{n+1}/\lambda_n}. \tag{7}$$

Observation functions $\phi^{(n+1)}(\mathbf{y}_i, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z})$ and potential functions $\varphi_{im}^{(n+1)}(\mathbf{y}_i, \hat{\mathbf{y}}_m^n)$ and $\psi_{ij}^{(n+1)}(\mathbf{x}_i, \hat{\mathbf{x}}_j^n)$ will be updated based on (5) (6) and (7) in the $(n + 1)^{th}$ iteration.

4.1 Potential Functions

Potential function $\psi_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ can be used to enforce relationships between body parts i and j . In our work, $\psi_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ is used to enforce kinematic constraints and angle constraints between two adjacent body parts. In this case, assuming part i is one neighbor of part j , we can get

$$\psi_{ij}^{(n)}(\mathbf{x}_i, \mathbf{x}_j) \propto \psi_{ij1}^{(n)}(\mathbf{x}_i, \mathbf{x}_j) \psi_{ij2}^{(n)}(\mathbf{x}_i, \mathbf{x}_j), \tag{8}$$

$$\psi_{ij1}^{(n)}(\mathbf{x}_i, \mathbf{x}_j) = \mathcal{N}(T(\mathbf{x}_i) - \mathbf{p}_j; 0, \Lambda_{ij}^n), \tag{9}$$

$$\psi_{ij2}^{(n)}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 1 & \text{if } \theta_{ij} \in \Theta_{ij} \\ a_{ij}^n & \text{otherwise} \end{cases}, \tag{10}$$

where $\psi_{ij}^{(n)}(\mathbf{x}_i, \mathbf{x}_j)$ represents the probability of \mathbf{x}_i given \mathbf{x}_j . $\psi_{ij1}^{(n)}(\mathbf{x}_i, \mathbf{x}_j)$ is used to enforce kinematic constraints, where T is a rigid transformation that is obtained from position \mathbf{p}_i and orientation θ_i in the pose \mathbf{x}_i and the size information of the i^{th} body

part, and $\Lambda_{i,j}^n$ is the variance matrix of the gaussian function \mathcal{N} in the n^{th} iteration. $\psi_{ij2}^{(n)}(\mathbf{x}_i, \mathbf{x}_j)$ is used to enforce angle constraints, where θ_{ij} is the angle between the two body parts' orientation $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_j$, Θ_{ij} is the valid angle range between body part i and j , and a_{ij}^n is a value between 0 and 1. Note that Λ_{ij}^n and a_{ij}^n are tuned based on (7).

Potential function $\varphi_{im}(\mathbf{y}_i, \mathbf{y}_m)$ is used to enforce non-penetration constraints between two related body parts i and m where

$$\varphi_{im}^{(n)}(\mathbf{y}_i, \mathbf{y}_m) = \begin{cases} 1 & \text{if } d_{im} > D_{im} \\ b_{im}^n & \text{otherwise} \end{cases}, \tag{11}$$

d_{im} is the minimum distance between part \mathbf{y}_i and \mathbf{y}_m , and D_{im} is the allowable minimum distance between the two parts. b_{im}^n is a value between 0 and 1. b_{im}^n is tuned according to (6) and becomes smaller with respect to the iteration, which means the non-penetration constraints will be more and more enforced.

4.2 Observation Functions

Observation function $\phi(\mathcal{Y}, \mathcal{Z})$ measures the likelihood of \mathcal{Z} given \mathcal{Y} . In order to measure the likelihood, the 3D body \mathcal{Y} is projected, and then the similarity between the projected image and the real input image observation \mathcal{Z} is computed to estimate the likelihood. Since $\phi(\mathcal{Y}, \mathcal{Z})$ is estimated by the similarity of the two whole images, it can deal with self-occlusion where one body part is partially occluded by others.

In our work, edge and silhouette were used as the features for the similarity measurement. Chamfer distance was used to measure the edge similarity. For the silhouette similarity, in addition to the overlapping area of the projected image and the human body image region in the input image, the chamfer distance from the projected image region to the body image region in the input image was also used. The relative weight between edge and silhouette similarity is experimentally determined. Note that the edge similarity was a value between 0 and 1 by normalizing the chamfer distance, such that that the scaling problem between the edge similarity and the silhouette similarity was avoided.

4.3 Nonparametric Implementation

Because of the non-Gaussian property of potential functions and observation functions, analytic computation of the functions is intractable. We use Monte Carlo method to search for each body part's state by iteratively updating conditional marginal distribution $\tilde{p}^{(n+1)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$, called Annealed Marginal Distribution Monte Carlo (AMDMC). In our algorithm, each distribution $\tilde{p}^{(n+1)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$ is represented by a set of K weighted samples,

$$\tilde{p}^{(n+1)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z}) = \{(\mathbf{s}_i^{(n+1,k)}, \pi_i^{(n+1,k)}) | 1 \leq k \leq K\} \tag{12}$$

where $\mathbf{s}_i^{(n+1,k)}$ is the k^{th} sample of the i^{th} body part state \mathbf{x}_i in the $(n + 1)^{th}$ iteration and $\pi_i^{(n+1,k)}$ is the weight of the sample. Note that $\mathbf{y}_i = f(\mathbf{x}_i)$ is a deterministic function and so it is not necessary in the nonparametric representation.

In each iteration, every $\tilde{p}^{(n+1)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$ is updated based on (4). The update process based on the Monte Carlo method is described in the following:

1. Update potential functions and observation functions based on (5)–(11).
2. Compute estimation $\hat{\mathcal{X}}_{-i}^n$ of the other body parts from initialization or previous iteration result, and get $\hat{\mathcal{Y}}_{-i}^n = f(\hat{\mathcal{X}}_{-i}^n)$.
3. Use importance sampling to generate new samples $\mathbf{s}_i^{(n+1,k)}$ from related marginal distributions of previous iteration. The related marginal distributions include the neighbors' and its own marginal distributions of previous iteration. The new samples are to be weighted in the following step to represent marginal distribution.
4. Update marginal distribution. For each new sample $\mathbf{s}_i^{(n+1,k)}$, compute $\mathbf{y}_i^{(n+1,k)} = f(\mathbf{s}_i^{(n+1,k)})$ and then calculate the weight $\pi_i^{(n+1,k)}$, where

$$\begin{aligned} \pi_i^{(n+1,k)} &= \phi^{(n+1)}(\mathbf{y}_i^{(n+1,k)}, \hat{\mathcal{Y}}_{-i}^n, \mathcal{Z}) \prod_{m \in \Gamma'(i)} \varphi_{im}^{(n+1)}(\mathbf{y}_i^{(n+1,k)}, \hat{\mathbf{y}}_m^n) \\ &\times \prod_{j \in \Gamma(i)} \psi_{ij}^{(n+1)}(\mathbf{s}_i^{(n+1,k)}, \hat{\mathbf{x}}_j^n). \end{aligned} \quad (13)$$

$\pi_i^{(n+1,k)}$ is then re-weighted and normalized because we use importance sampling to generate sample $\mathbf{s}_i^{(n+1,k)}$. The updated marginal distributions will be used to update marginal distributions in the next iteration.

Human body posture can be estimated from the set of marginal distributions. The sample with the maximum weight in $\tilde{p}^{(n+1)}(\mathbf{x}_i, \mathbf{y}_i | \mathcal{Z})$ can be used to represent the estimation $\hat{\mathbf{x}}_i^{n+1}$ of i^{th} body part's pose.

Our algorithm can be viewed as the generalization of annealed particle filtering [5]. When body posture \mathcal{X} is a single high dimensional state rather than a set of body parts' states, our AMDMC algorithm will become exactly the annealed particle filtering. Another related algorithm is the modified nonparametric belief propagation (mNBP) [27]. mNBP can also deal with partial self-occlusion, but it is based on the tree-structured model and there is no theoretical foundation on the modification. While mNBP is tested on synthetic image for posture estimation, our AMDMC is on real image sequences.

5 Experimental Results

Evaluation of our AMDMC algorithm and comparison with some related work were performed using real image sequences. The first experiment evaluated the algorithm's ability to accurately estimating 2D human posture under partial self-occlusion from a single image. The second experiment evaluated its ability to tracking 2D human body from a monocular image sequence. Note that the algorithm can be applied to estimate 3D human postures when multiple images or image sequences are available.

5.1 Preprocess

In the real image sequences in which a TaiChi motion was recorded, the static background inside each image was removed by a statistical background model. And the human model was modified to fit the human body size in every image sequence.

Note that the edge information plays an important role in posture estimation especially when limbs (e.g., arms) fall inside the torso image region. However, because of

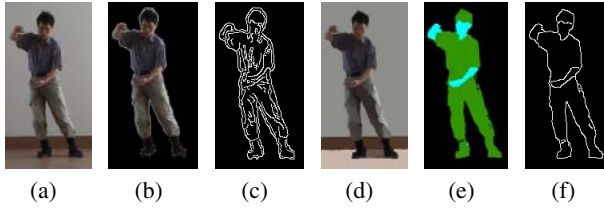


Fig. 2. Removing noisy edges inside each body part. (a) Input image. (b) Input image after background removal. (c) The extracted edge without removing noisy edges. (d) Over-segmentation result of (a) by mean shift. (e) Cluster every segment inside the body image region into one of the two clusters. (f) The edge extracted from (e).

the variability in the appearance of human body, many noisy edges inside each body part will probably happen (Figure 2(c)). Mean shift [4] was used here to remove the noisy edges. Firstly from the first input image, the body image was manually divided into several clusters according to their appearance, and the histogram of each cluster was obtained. Then for subsequent input images, over-segmentation of the foreground image region was obtained by mean shift (Figure 2(d)), and every segment was classified into one cluster by comparing the histogram similarity between the segment and each cluster (Figure 2(e)). Of course because of similar appearance between symmetric body limbs, several body parts are often clustered together. In our experiment, just two clusters were used. One was for lower arms and the face, the other for the remained body parts. The noisy edges can be effectively removed by the above process (Figure 2(f)).

5.2 Posture Estimation Under Partial Self-occlusion

Because of the depth ambiguity in a single image, 3D joint position cannot be estimated correctly. Therefore not 3D but 2D joint position error $E_{2D} = \frac{1}{nh} \sum_{i=1}^n \|\hat{\mathbf{p}}_{2i} - \mathbf{p}_{2i}\|$ was computed to assess the performance of our algorithm, where $\hat{\mathbf{p}}_{2i}$ and \mathbf{p}_{2i} are the estimated and true 2D image position of the i^{th} body joint. \mathbf{p}_{2i} was obtained by manually setting the image position of each body joint. h is the articulated body height and it is about 190 pixels in each 320×240 image.

In this experiment, 150 weighted samples were used to represent each conditional marginal distribution. The annealing related factor λ_{n+1}/λ_n was simply set to 0.9 in every iteration, and the total iteration number is 50. For each iteration, around 12 seconds was spent, most of which was used to generate synthetic image from mesh model and compute observation functions.

The skeletons in Figures 3(b) and 3(c) illustrate the 2D true posture and initial posture respectively. The skeletons in Figures 3(d), 3(e) and 3(f) illustrate the estimated posture using BPMC, mNBP, and our algorithm respectively. Both BPMC and mNBP [9] are efficient inference algorithms based on tree-structured model, and they have the same computation complexity as our AMDMC due to the similar structure of the graphical models. Note that the annealing factor was also used in BPMC and mNBP such that they can be compared with ours. Even the annealed BPMC was expected to obtain better result than the original BPMC, the pose estimation of some body parts (i.e., arms in Figure 3(d)) was not accurate when there was partial self-occlusion between body

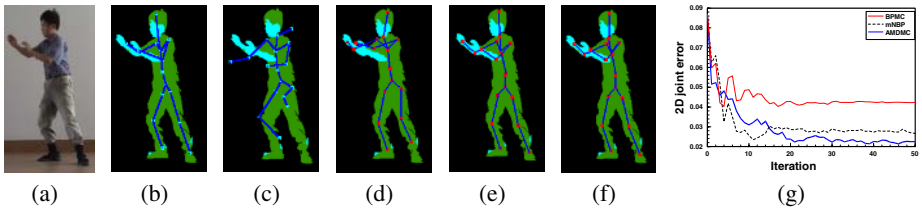


Fig. 3. Posture estimation from a single image. (a) Input image. (b) 2D true posture represented by skeleton. (c) initial posture. Estimated posture by (d) BPMC, (e) mNBP and (f) our AMDMC. (g) AMDMC is better than BPMC and comparable to mNBP in posture estimation.

parts because local body image observation was used in BPMC's observation function. In comparison, the posture estimation was very close to the truth when using the mNBP and our AMDMC, in which the global body image observation was used in the observation function.

For this input image and the initial posture, Figure 3(g) represents the 2D joint position error E_{2D} with respect to the iteration number when using different inference algorithms. It shows that, after 10 to 15 iterations, the error has decreased to a relatively small value (i.e., 2.5% of body height h , or 5 pixels when h is 190 pixels) for both mNBP and AMDMC, but there was a higher error for BPMC.

Similar results have been reported on estimating body posture from a single image. Lee and Cohen [14] reported a higher 2D joint error which was 12 pixels when h was 150 pixels. Hua *et al.* [9] also reported a similar higher joint error. Both algorithms can deal with partial self-occlusion by detecting the possible positions of some body parts before posture estimation, while our AMDMC does not require any part detection in dealing with partial self-occlusion. In addition, Sigal *et al.* [20] provided an NBP algorithm which requires a complex learning process. They tested their algorithm on a simple walking posture using a multi-camera system. In comparison, our algorithm does not require any learning process and can deal with more complex postures.

5.3 Articulated Human Tracking

In the second experiment, a sequence of 88 real images were used for human tracking. The sequence was extracted from one original sequence of 350 images by sampling one from every four consecutive images. This will make human tracking more challenging because large posture difference between two adjacent frames will probably happen. In the tracking process, the initial posture for each image came from the estimated posture of previous frame image, while for the first frame image we manually set the initial posture. The annealing factor λ_{n+1}/λ_n was set to 0.85 for all 30 iterations. Because of the unknown true postures in the real images, the tracking result was visually compared with related algorithms. From Figure 4, we can see that both AMDMC and mNBP can accurately track human motion even under severe self-occlusion between two arms, while BPMC failed to track some body parts after a short sequence. The reason is clear. Because local image observation was used in BPMC to estimate each body part's pose, it is easy to fall into a local minimum in MAP estimation of the marginal distributions.

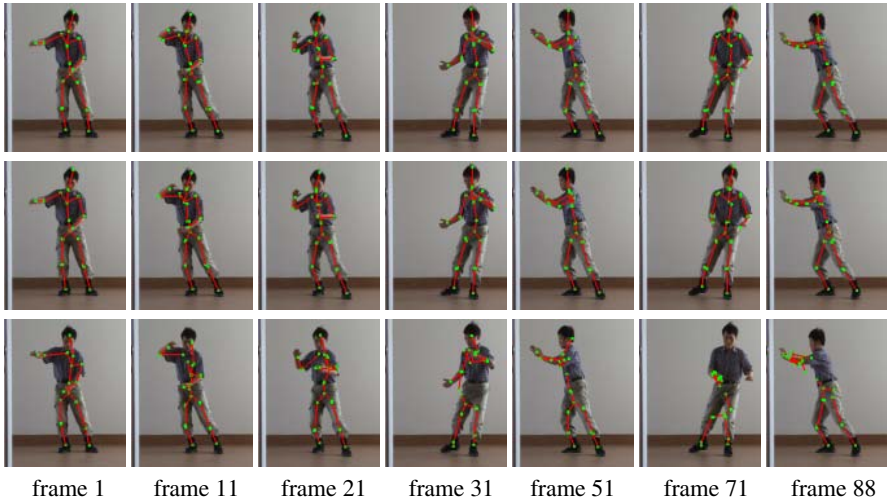


Fig. 4. Results of 2D articulated human tracking. The first row is the result by AMDMC; The second row is by mNBP; The third row is by BPMC.

6 Conclusion

We presented a new graphical model and developed an efficient inference algorithm (AMDMC) to accurately estimate human postures under partial self-occlusion. The AMDMC algorithm can estimate human posture in a low dimensional state space by iteratively updating a set of body parts' marginal distributions. Experiments showed that the AMDMC algorithm can accurately estimate 2D human posture from a single image even if the initial posture was far from the truth and if there was partial self-occlusion between body parts. The AMDMC can be easily extended to articulated human tracking, which has been shown by the successful 2D articulated human tracking from a monocular image sequence. Compared to the existing techniques for posture estimation under self-occlusion, our AMDMC does not require any learning process or part detection beforehand. However in a monocular image sequence, it is difficult to discriminate the left body limbs from the right ones when human body viewed from the side. In such a case, prior motion knowledge or constraints must be explored in advance. Our future work is to extend the AMDMC algorithm to deal with more general cases in human posture and tracking by exploring motion models and human body constraints.

References

1. A. Agarwal and B. Triggs. 3d human pose from silhouettes by relevance vector regression. In *Proc. IEEE Conf. on CVPR*, pages 882–888, 2004.
2. V. Athitsos, J. Alon, S. Sclaroff, and G. Kollios. Boostmap: A method for efficient approximate similarity rankings. In *Proc. IEEE Conf. on CVPR*, pages 268–275, 2004.
3. T. Cham and J. Rehg. A multiple hypothesis approach to figure tracking. In *Proc. IEEE Conf. on CVPR*, pages 239–245, 1999.

4. D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Machine Intell.*, 24:603–619, 2002.
5. J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *Proc. IEEE Conf. on CVPR*, pages 126–133, 2000.
6. A. Elgammal and C. Lee. Inferring 3d body pose from silhouettes using activity manifold learning. In *Proc. IEEE Conf. on CVPR*, pages 681–688, 2004.
7. P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *Int. Journal of Computer Vision*, 61(1):55–79, 2005.
8. D. M. Gavrilu. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding: CVIU*, 73(1):82–98, 1999.
9. G. Hua, M. H. Yang, and Y. Wu. Learning to estimate human pose with data driven belief propagation. In *Proc. IEEE Conf. on CVPR*, pages 747–754, 2005.
10. S. Ioffe and D. Forsyth. Finding people by sampling. In *Proc. IEEE Conf. on ICCV*, pages 1092–1097, 1999.
11. M. Isard. Pampas: Real-valued graphical models for computer vision. In *Proc. IEEE Conf. on CVPR*, pages 613–620, 2003.
12. M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. ECCV*, pages 343–356, 1996.
13. X. Lan and D. P. Huttenlocher. Beyond trees: common-factor models for 2D human pose recovery. In *Proc. IEEE Conf. on ICCV*, pages 470–477, 2005.
14. M. W. Lee and I. Cohen. Proposal maps driven mcmc for estimating human body pose in static images. In *Proc. IEEE Conf. on CVPR*, pages 334–341, 2004.
15. G. Mori. Guiding model search using segmentation. In *Proc. IEEE Conf. on ICCV*, 2005.
16. G. Mori and J. Malik. Estimating human body configurations using shape context matching. In *Proc. ECCV*, pages 666–680, 2002.
17. A. H. R. Urtaun, D. J. Fleet and P. Fua. Priors for people tracking from small training sets. In *IEEE Int. Conf. ICCV*, 2005.
18. D. Ramanan, D. A. Forsyth, and A. Zisserman. Strike a pose: Tracking people by finding stylized poses. In *Proc. IEEE Conf. on CVPR*, pages 271–278, 2005.
19. R. Rosales, V. Athitsos, and S. Sclaroff. 3D hand pose reconstruction using specialized mappings. In *Proc. IEEE Conf. on ICCV*, pages 378–385, 2001.
20. L. Sigal, S. Bhatia, S. Roth, M. Black, and M. Isard. Tracking loose-limbed people. In *Proc. IEEE Conf. on CVPR*, pages 421–428, 2004.
21. C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Discriminative density propagation for 3D human motion estimation. In *Proc. IEEE Conf. on CVPR*, pages 390–397, 2005.
22. C. Sminchisescu and B. Triggs. Kinematic jump processes for monocular 3D human tracking. In *Proc. IEEE Conf. on CVPR*, pages 69–76, 2003.
23. C. Sminchisescu and B. Triggs. Building roadmaps of minima and transitions in visual models. *Int. Journal of Computer Vision*, 61(1):81–101, 2005.
24. E. Sudderth, A. Ihler, W. Freeman, and A. Willsky. Nonparametric belief propagation. In *Proc. IEEE Conf. on CVPR*, pages 605–612, 2003.
25. E. Sudderth, M. Mandel, W. Freeman, and A. Willsky. Distributed occlusion reasoning for tracking with nonparametric belief propagation. In *NIPS*, 2004.
26. E. Sudderth, M. Mandel, W. Freeman, and A. Willsky. Visual hand tracking using nonparametric belief propagation. In *IEEE CVPR Workshop on Generative Model based Vision*, 2004.
27. R. Wang and W. K. Leow. Human body posture refinement by nonparametric belief propagation. In *IEEE Conf. on ICIP*, 2005.
28. J. S. Yedidia, W. T. Freeman, and Y. Weiss. Constructing free energy approximations and generalized belief propagation algorithms. Technical report, MERL, 2002.

Particle Filtering with Dynamic Shape Priors^{*}

Yogesh Rathi, Samuel Dambreville, and Allen Tannenbaum

Georgia Institute of Technology, Atlanta, GA

{yogesh.rathi, samuel.dambreville, tannenba}@bme.gatech.edu

Abstract. Tracking deforming objects involves estimating the global motion of the object and its local deformations as functions of time. Tracking algorithms using Kalman filters or particle filters have been proposed for tracking such objects, but these have limitations due to the lack of dynamic shape information. In this paper, we propose a novel method based on employing a locally linear embedding in order to incorporate dynamic shape information into the particle filtering framework for tracking highly deformable objects in the presence of noise and clutter.

1 Introduction

The problem of tracking moving and deforming objects has been a topic of substantial research in the field of active vision; see [1,2] and the references therein. There is also an extensive literature with various proposals for tracking objects with static shape prior [3]. This paper proposes a novel method to incorporate dynamic shape priors into the particle filtering framework for tracking highly deformable objects in the presence of noise and clutter.

In order to appreciate this methodology, we briefly review some previous related work. The possible parameterizations of planar shapes described as closed contours are of course very important. Various finite dimensional parameterizations of continuous curves have been proposed, perhaps most prominently the B-spline representation used for a “snake model” as in [2]. Isard and Blake (see [1] and the references therein) use the B-spline representation for contours of objects and propose the CONDENSATION algorithm [1] which treats the affine group parameters as the state vector, learns a prior dynamical model for them, and uses a particle filter [4] to estimate them from the (possibly) noisy observations. Since this approach only tracks affine parameters, it cannot handle local deformations of the deforming object.

Another approach for representing contours is via the level set method [5,6] where the contour is represented as the zero level set of a higher dimensional

^{*} This research was supported by grants from NSF, NIH (NAC P41 RR-13218 through Brigham and Women’s Hospital), AFOSR, ARO, MURI, and MRI-HEL, and the Technion, Israel Institute of Technology. This work was done under the auspices of the National Alliance for Medical Image Computing (NAMIC), funded by the National Institutes of Health through the NIH Roadmap for Medical Research, Grant U54 EB005149.

function, usually the signed distance function [5]. For segmenting an object, an initial guess of the contour (represented using the level set function) is deformed until it minimizes an image-based energy functional. Some previous work on tracking using level set methods is given in [3,7,8,9,10].

Shape information is quite useful when tracking in clutter, especially if the object to be tracked gets occluded. Hence, a number of methods have been proposed [3] which incorporate a static shape prior into the tracking framework. The approach of these works is based on the idea that the object being tracked does not undergo a deformation (modulo a rigid transformation). Another method to obtain a shape prior is using PCA (principal component analysis) [11]. In this case, it is assumed that the shape can undergo small variations which can be captured by doing linear PCA. However, linear PCA is quite inadequate in representing the shape variations if the object being tracked undergoes large deformations (as will be explained in detail in the subsequent sections).

The authors in [12] use a particle filtering algorithm for geometric active contours to track highly deformable objects. The tracker however fails to maintain the shape of the object being tracked in case of occlusion. The present work extends the method proposed in [12] by incorporating dynamic shape priors into the particle filtering framework based on the use of a Locally Linear Embedding (LLE). LLE [13,14] attempts to discover the nonlinear structure in high dimensional data by exploiting the local symmetries of linear reconstructions. To the best of our knowledge, this is the first time LLE has been used for shape analysis and tracking. Another approach closely related to our work was proposed in [15], wherein exemplars were used to learn the distribution of possible shapes. A different method in [16] separates the space of possible shapes into different clusters and learns a transition matrix to transition from one patch of shapes to the next. Our approach is different from those in [15,16] in that we do not learn the dynamics of shape variation *a priori*. The only knowledge required in our method is a possible set of shapes of the deforming object.

The literature reviewed above is by no means exhaustive. Due to paucity of space we have only quoted a few related works. The rest of the paper is organized as follows: Section 2 gives the motivation and briefly describes the concepts of LLE, shape similarity measures, and curve evolution. Section 3 develops the state space model in detail and Section 4 describes the experiments conducted to test the proposed method. Some conclusions and further research directions are discussed in Section 5.

2 Preliminaries

Principal component analysis (PCA) is one of the most popular forms of dimensionality reduction techniques. In PCA, one computes the linear projections of greatest variance from the top eigenvectors of the data covariance matrix. Its first application to shape analysis [11] in the level set framework was accomplished by embedding a curve C as the zero level set of a signed distance function Φ . By doing this, a small set of coefficients can be utilized for a shape prior in various

segmentation tasks as shown in [11,17]. However, linear PCA assumes that any required shape can be represented using a linear combination of eigen-shapes, i.e., any new shape $\tilde{\Phi}$ can be obtained by [17], $\tilde{\Phi} = \bar{\Phi} + \sum_{i=1}^N w_i \Phi_i$, where w_i are weights assigned to each eigenshape Φ_i and $\bar{\Phi}$ is the mean shape. Thus, PCA assumes that the set of training shapes lie on a linear manifold.

More specifically, let us consider shapes of certain objects with large deformations, for example, Figure 1 shows a set of few shapes of a man. PCA was performed on 75 such shapes (embedded in a signed distance function). Figure 2 shows the original and the reconstructed shape. Thus, linear PCA cannot be used to obtain a shape prior if the training set lies on a non-linear manifold.



Fig. 1. Few shapes of a man from a training set. Note the large deformation in shape.

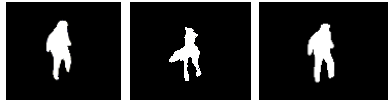


Fig. 2. Left: Original shape, Middle: projection in the PCA basis, Right: LLE (2 nearest neighbors)

In [18], the authors proposed an unsupervised Locally Linear Embedding (LLE) algorithm that computes low dimensional, neighborhood preserving embeddings of high dimensional data. LLE attempts to discover nonlinear structure in high dimensional data by exploiting the local symmetries of linear combinations. It has been used in many pattern recognition problems for classification. In this work, we use it in the particle filtering framework for providing dynamic shape prior.

2.1 Locally Linear Embedding for Shape Analysis

The LLE algorithm [14] is based on certain simple geometric intuitions. Suppose the data consists of N vectors Φ_i sampled from some smooth underlying manifold. Provided there is sufficient data, we expect each data point and its neighbors to lie on or close to a locally linear patch of the manifold. We can characterize the local geometry of these patches by linear coefficients that reconstruct each data point from its neighbors. In the simplest formulation of LLE, one identifies k nearest neighbors for each data point. Reconstruction error is then measured by the cost function: $E(W) = \left(\Phi - \sum_j w_j \Phi_j \right)^2$. We seek to minimize the reconstruction error $E(W)$, subject to the constraint that the weights w_j that lie

outside the neighborhood are zero and $\sum_j w_j = 1$. With these constraints, the weights for points in the neighborhood of Φ can be obtained as [13]:

$$w_j = \frac{\sum_{m=1}^k R_{jm}}{\sum_{p=1}^k \sum_{q=1}^k R_{pq}}, \text{ where } Q_{jm} = (\Phi - \Phi_j)^T (\Phi - \Phi_m), \quad R = Q^{-1} \quad (1)$$

In this work, we assume that a closed curve C_j is represented as the zero level set of a signed distance function Φ_j . Stacking all the columns of Φ_j one below the other, one can obtain a vector of dimension D^2 , if Φ_i is of dimension $D \times D$. (In the rest of the paper, we use Φ interchangeably to represent a vector of dimension D^2 or a matrix of dimension $D \times D$. The appropriate dimension can be inferred from the context.) Figure 2 shows a particular shape being represented by 2 of its nearest neighbors.

2.2 Finding the Nearest Neighbors

The previous section showed how to represent a shape Φ_i by a linear combination of its k neighbors. Here we consider the key issue of how to find the nearest neighbors. One might be tempted to use the Euclidean 2-norm to find distance between shapes, i.e., if $d^2(\Phi_i, \Phi_j)$ is the (squared) distance between Φ_i and Φ_j , then $d^2(\Phi_i, \Phi_j) = \|\Phi_i - \Phi_j\|^2$. However, this norm does not represent distance between shapes, but only distance between two vectors. Since we are looking for the nearest neighbors of C_i in the shape space, a similarity measure between shapes is a more appropriate choice. Many measures of similarity have been reported; see [19,3,20]. In this paper, we have chosen the following distance measure [21]:

$$d^2(\Phi_i, \Phi_j) = \int_{p \in Z(\Phi_i)} EDT_{\Phi_j}(p) dp + \int_{p \in Z(\Phi_j)} EDT_{\Phi_i}(p) dp \quad (2)$$

where, EDT_{Φ_i} is the Euclidean distance function of the zero level set of Φ_i (one can think of it as the absolute value of Φ_i), and $Z(\Phi_i)$ is the zero level set of Φ_i . We chose this particular distance measure because it allows for partial shape matching which is quite useful for occlusion handling. More details about this measure may be found in [21]. We should note that the development of the remaining algorithm does not depend on the choice of the distance measure. Thus, once the distance measure between each Φ_i and the rest of the elements in the training set is known, one can find the nearest neighbors of Φ_i .

2.3 Curve Evolution

There is a large literature concerning the problem of separating an object from its background [3,9]. Level sets have been used quite successfully for this task. In [22], the authors have proposed a variational framework for segmenting an object using the first two moments (mean and variance) of image intensities. In the present work, we have used the energy functional given in [22]

$$\begin{aligned}
 E_{image} = & \int_{\Omega} \left(\log \sigma_u^2 + \frac{(I(x) - u)^2}{\sigma_u^2} \right) H(\Phi) dx \\
 & + \int_{\Omega} \left(\log \sigma_v^2 + \frac{(I(x) - v)^2}{\sigma_v^2} \right) (1 - H(\Phi)) dx + \nu \int_{\Omega} \|\nabla H(\Phi)\| dx,
 \end{aligned} \tag{3}$$

which upon minimization gives the following PDE:

$$\frac{\partial \Phi}{\partial t} = \delta_{\epsilon}(\Phi) \left(\nu \operatorname{div} \frac{\nabla \Phi}{\|\nabla \Phi\|} + \log \frac{\sigma_v^2}{\sigma_u^2} - \frac{(I(x) - u)^2}{\sigma_u^2} + \frac{(I(x) - v)^2}{\sigma_v^2} \right). \tag{4}$$

Here $I(x)$ is the image, u, v are the mean intensities inside and outside the curve C (corresponding to Φ) respectively, σ_u^2, σ_v^2 are the respective variances and $\delta_{\epsilon}(\Phi) = \frac{dH}{d\Phi}$ is the Dirac delta function and H is the Heaviside function as defined in [22]. Note that, one could use any type of curve evolution equation in the algorithm being proposed. We have made this particular choice because it is simple yet powerful in segmenting cluttered images.

3 The State Space Model

This section describes the state space model, the prediction model, and the importance sampling concept used within the particle filtering framework for tracking deformable objects. We will employ the basic theory of particle filtering here as described in [4].

Let S_t denote the state vector at time t . The state consists of parameters T that models the rigid (or affine) motion of the object (e.g., $T = [x \ y \ \theta]$ for Euclidean motion) and the curve C (embedded as the zero level set of Φ) which models the shape of the object, i.e., $S_t = [T_t \ \Phi_t]$. The observation is the image at time t , i.e., $Y_t = Image(t)$. Our goal is to recursively estimate the posterior distribution $p(S_t|Y_{1:t})$ given the prior $p(S_{t-1}|Y_{1:t-1})$. This involves a time update step and a measurement update step as described in the next section.

In general, it is quite difficult to obtain a model for predicting the position and shape of the deforming object. More specifically, in the current case, it is very difficult to obtain samples from the infinite dimensional space of closed curves (shapes). This problem can be solved using Bayesian *importance sampling* [23], described briefly below: Suppose $p(x)$ is a probability density from which it is difficult to draw samples (but for which $p(x)$ can be evaluated) and $q(x)$ is a density which is easy to sample from and has a heavier tail than $p(x)$ (i.e. there exists a bounded region R such that for all points outside R , $q(x) > p(x)$). $q(x)$ is known as the *proposal density* or the *importance density*. Let $x^i \sim q(x)$, $i = 1, \dots, N$ be samples generated from $q(\cdot)$. Then, an approximation to $p(\cdot)$ is given by $p(x) \approx \sum_{i=1}^N \omega^i \delta(x - x^i)$, where $\omega^i \propto \frac{p(x^i)}{q(x^i)}$ is the normalized weight of the i -th particle. So, if the samples, $S_t^{(i)}$, were drawn from an importance density, $q(S_t|S_{1:t-1}, Y_{1:t})$, and weighted by $\omega_t^{(i)} \propto \frac{p(S_t^{(i)}|Y_{1:t})}{q(S_t^{(i)}|S_{1:t-1}, Y_{1:t})}$, then $\sum_{i=1}^N \omega_t^{(i)} \delta(S_t^{(i)} - S_t)$ approximates $p(S_t|Y_{1:t})$. The choice of the importance

density is a critical design issue for implementing a successful particle filter. As described in [24], the proposal distribution $q(\cdot)$ should be such that particles generated by it, lie in the regions of high observation likelihood. Another important requirement is that the variance of the weights ω^i should not increase over time. Various algorithms have been proposed [24] to achieve this objective. One way of doing this is to use an importance density which depends on the current observation. This idea has been used in many past works such as the unscented particle filter [25] where the proposal density is a Gaussian density with a mean that depends on the current observation. In this work, we propose a possible importance density function $q(S_t|S_{t-1}, Y_t)$ and show how to obtain samples from it. Note that, the space of closed curves from which we want to obtain samples is infinite dimensional.

3.1 Time Update

The prediction \hat{S}_t at time t is given by: $\hat{S}_t = f_t(S_{t-1}, i_t, n_t)$ where n_t is random noise vector, i_t is any user defined input data (in our case, it is the set of training data) and f_t is possibly a nonlinear function. The problem of tracking deforming objects can be separated into two parts [8]:

1. Tracking the global rigid motion of the object;
2. Tracking local deformations in the shape of the object, which can be defined as any departure from rigidity.

Accordingly, we assume that the parameters that represent rigid motion T_t and the parameters that represent the shape Φ_t are independent. Thus, it is assumed that the shape of an object does not depend on its location in the image, but only on its previous shape and the location of an object in space does not depend on the previous shape. Hence, the prediction step consists of predicting the spatial position of the object using $\hat{T}_t = T_{t-1} + n_t^{(T)}$ where $n_t^{(T)}$ is random Gaussian noise vector with variance σ_T^2 . The prediction for shape $\hat{\Phi}_t$ is obtained as follows:

$$\hat{\Phi}_t = p_0\Phi_{t-1} + p_1\Phi_{t-1}^{(N_1)} + p_2\Phi_{t-1}^{(N_2)} + \dots + p_k\Phi_{t-1}^{(N_k)} \tag{5}$$

where p_0, p_1, \dots, p_k are user defined weights such that $\sum_i p_i = 1$ and $\Phi_{t-1}^{(N_i)}, i = 1..k$ are the k nearest neighbors of Φ_{t-1} . The nearest neighbors are obtained as described in Section 2.2. A more generalized formulation of the prediction step above can be obtained by sampling the weights p_i from a known distribution (for example, an exponential distribution) to obtain a set of possible shapes and then choosing the predicted shape from this set, based on certain criteria.

We should note that, one of the main contributions of this paper is the formulation of a scheme that allows to dynamically predict the shape of the object without learning the sequence in which they occur (unlike the methods in [15,16]). Thus, the only knowledge required in this prediction step is a training set of shapes. In particular, one does not need to sample from an infinite-dimensional space of shapes (curves) but only from a set containing the linear combination of k nearest neighbors of Φ_{t-1} . This not only reduces the search

space dramatically, but also allows to sample from a finite set of possible shapes. Once the latest observation Y_t is obtained, one can update the prediction based on this information as explained in the following section.

3.2 Measurement Update

At time t , for each particle i , generate samples as described in the prediction step in (5). Using the image at time t (Y_t), a rigid transformation is applied to each $\hat{\Phi}_t^{(i)}$ (in particular $\hat{C}_t^{(i)}$) by doing L_r iterations of gradient descent on the image energy E_{image} with respect to the rigid transformation parameters T . The curve is then deformed by doing a few (L_d) iterations of gradient descent (“curve evolution”) on the energy, E , i.e., we generate

$$\tilde{\Phi}_t^{(i)} = f_{R^r}^{L_r}(\hat{\Phi}_t^{(i)}, Y_t), \quad \Phi_t^{(i)} = f_{CE}^{L_d}(\tilde{\Phi}_t^{(i)}, Y_t) \tag{6}$$

where $f_{R^r}^{L_r}(\Phi, Y)$ is given by (for $j = 1, 2, \dots, L_r$)

$$r^0 = r, \quad r^j = r^{j-1} - \alpha^j \nabla_r E_{image}(r^{j-1}, \Phi, Y), \quad T = r^{L_r}, \quad f_{R^r}^{L_r}(\Phi, Y) = T\Phi \tag{7}$$

and $f_{CE}^{L_d}(\mu, Y)$ is given by (for $j = 1, 2, \dots, L_d$)

$$\mu^0 = \mu, \quad \mu^j = \mu^{j-1} - \alpha^j \nabla_\mu E(\mu^{j-1}, Y), \quad f_{CE}^{L_d}(\mu, Y) = \mu^{L_d} \tag{8}$$

where $E = E_{image} + \beta E_{shape}$. The energy E_{image} is as defined in equation (3) and E_{shape} is defined by [3]: $E_{shape}(\Phi) = \int_\Omega \bar{\Phi}(x) dx$, where $\bar{\Phi}(x)$ is the contour obtained from a linear combination of the nearest neighbors of Φ , with weights obtained using LLE from equation (1). The corresponding curve evolution equation is given by

$$\frac{\partial \Phi}{\partial t} = \bar{\Phi}(x) \|\nabla \Phi\|. \tag{9}$$

This PDE tries to drive the current contour shape Φ towards the space of possible shapes and equation (8) tries to drive the current contour towards the minimizer of energy E which depends on the image and shape information. The parameter β is user defined and weights the shape information with the image information. The use of LLE to provide shape information for contour evolution is another main contribution of this paper.

Details about equation (7) can be obtained from [17]; and equation (8) may be implemented by summing the PDE’s (4) and (9). We perform only L (L_d or L_r) iterations of gradient descent since we do not want to evolve the curve until it reaches a minimizer of the energy, E_{image} (or E). Evolving to the local minimizer is not desirable since the minimizer would be independent of all starting contours in its domain of attraction and would only depend on the observation, Y_t . Thus the state at time t would lose its dependence on the state at time $t - 1$ and this may cause loss of track in cases where the observation is bad. In effect, choosing L to be too large can move all the samples too close to the current observation, while a small L may not move the particles towards the desired

region. The choice of L depends on how much one trusts the system model versus the obtained measurements. Note that, L will of course also depend on the step-size of the gradient descent algorithm as well as the type of PDE used in the curve evolution equation.

For each i , the sample $S_t^{(i)}$ thus obtained is drawn from the importance density $q(S_t^{(i)}|S_{t-1}^{(i)}, Y_t) = \mathcal{N}(S_t^{(i)}, \Sigma)$, where we assume a Gaussian fit for the density $q(\cdot)$ centered at each $S_t^{(i)}$. We further assume that the variance Σ is very small and constant for all particles, i.e., $q(S_t^{(i)}|S_{t-1}^{(i)}, Y_t) = \text{constant}$. We should note that, this methodology, even though sub-optimal (to the best of our knowledge, an optimal method to sample from an infinite dimensional space of curves does not exist) allows to obtain samples that lie in region of high likelihood. The above mentioned step of doing gradient descent can also be interpreted as an MCMC move step, where particles are “moved” to region of high likelihood by any available means, as given in [24].

3.3 Setting the Importance Weights

In this paper, the state process is assumed to be Markov, and the observations are conditionally independent given the current state i.e., $p(Y_t|S_{0:t}) = p(Y_t|S_t)$. This gives the following recursion for the weights [23]: $\omega_t^{(i)} \propto \omega_{t-1}^{(i)} \frac{p(Y_t|S_t^{(i)})p(S_t^{(i)}|S_{t-1}^{(i)})}{q(S_t^{(i)}|S_{t-1}^{(i)}, Y_t)}$.

The probability $p(Y_t|S_t)$ is defined as $p(Y_t|S_t) \propto e^{-\frac{E(S_t, Y_t)}{\sigma_{tot}^2}}$. We define $p(S_t|S_{t-1}) = p(T_t|T_{t-1}) p(\Phi_t|\Phi_{t-1})$ with

$$p(T_t|T_{t-1}) \propto e^{-\frac{|T_t - T_{t-1}|}{\sigma_T^2}}, \quad p(\Phi_t|\Phi_{t-1}) \propto e^{-\frac{d^2(\Phi_t, \Phi_{t-1})}{\sigma_d^2}} + a e^{-\frac{d^2(\Phi_t, \check{\Phi}_{t-1})}{\sigma_d^2}} \tag{10}$$

where d^2 is the (squared) distance measure defined above in (2), and $\check{\Phi}_{t-1}$ is the MAP (maximum a-posteriori) estimate of the shape at time $t-1$. We should note that, using the MAP shape information available from time $t-1$ is quite essential, since it adds weights to particles which are closer to the previous best estimate than particles that are far away. This is quite useful in case of occlusion wherein particles which look like the previous best shape are given higher probability, despite the occlusion. The parameter a is user defined.

Based on the discussion above, the particle filtering algorithm can be written as follows:

- Use equation (5) to obtain $\hat{T}_t, \hat{\Phi}_t$.
- Perform L_r steps of gradient descent on rigid parameters using (7) and L_d iterations of curve evolution using (8).
- Calculate the importance weights, normalize and resample [4], i.e.,

$$\tilde{\omega}_t^{(i)} \propto p(Y_t|S_t^{(i)})p(T_t^{(i)}|T_{t-1}^{(i)})p(\Phi_t^{(i)}|\Phi_{t-1}^{(i)}), \quad \omega_t^{(i)} = \frac{\tilde{\omega}_t^{(i)}}{\sum_{j=1}^N \tilde{\omega}_t^{(j)}}. \tag{11}$$

4 Experiments

The proposed algorithm was tested on 3 different sequences and the results are presented in this section. We certainly do not claim that the method proposed in this paper is the best one for every image sequence on which it was tested, but it did give very good results with a small number of particles on all of the image sequences. We should add that to the best of our knowledge this is the first time dynamic shape prior in a level set framework has been used in conjunction with the particle filter [4] for tracking such deforming objects.

In all of the test sequences, we have used the following parameters which gave good results:

1. Choosing k , the number of nearest neighbors (for obtaining $\bar{\Phi}(x)$ in eqn (9)): k will depend on the number of similar shapes available in the training set [18]. In our experiments, $k = 2$ gave acceptable results.
2. Choosing σ_d^2 : A classical choice [20] is $\sigma_d^2 = c \frac{1}{N} \sum_{i=1}^N \min_{j \neq i} d^2(\Phi_i, \Phi_j)$. For all the test sequences, $c = 1/20$ was used.
3. σ_T^2 models the motion dynamics of the object being tracked. In all the test sequences, since the spatial motion of the object was not large, we used $\sigma_T^2 = 1000$. Also, only translational motion was assumed, i.e., $T = [x \ y]$.

4.1 Shark Sequence

This sequence has very low contrast (object boundaries in some images are barely visible even to human observers) with a shark moving amid a lot of other fish which partially occlude it simultaneously in many places. This results in a dramatic change in the image statistics of the shark if a fish from the background

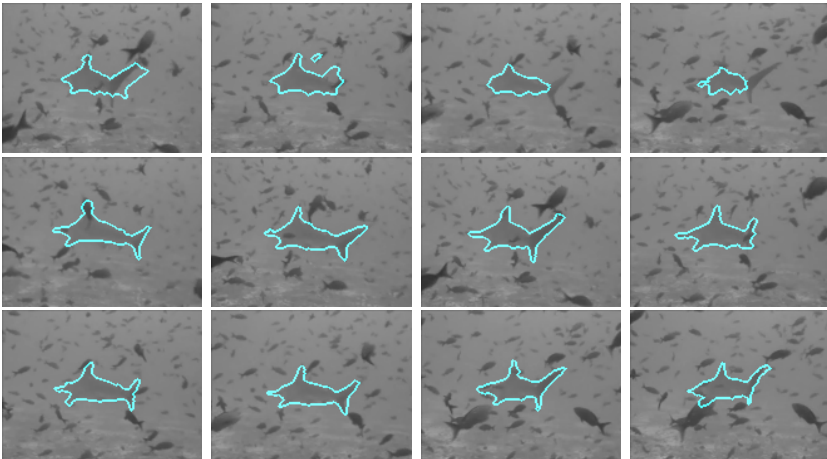


Fig. 3. First row shows tracking results with no shape information. Next two rows show results using the proposed algorithm.

occludes the shark. The training set was obtained by hand segmenting 10% of the images from the image sequence. Tracking results without shape information using the algorithm in [12] is shown in Figure 3. As can be seen, even though the algorithm tracks the shark, it is unable to maintain the shape. Results using the proposed algorithm are shown in Figure 3. This sequence demonstrates the robustness of the proposed algorithm in the presence of noise and clutter. The following parameters were used in tracking this sequence: $L_r = 1$, $L_d = 10$, particles = 40.

4.2 Octopus Sequence

As seen in Figure 4, the shape of the octopus undergoes large changes as it moves in a cluttered environment. It gets occluded for several frames by a fish having the same mean intensity. Tracking this sequence using equation (4) or any other method without shape information may result in the curve leaking to encompass the fish. Figure 4 shows tracking results using the proposed method. The following set of parameters were used in tracking this sequence: $L_r = 3$, $L_d = 10$, particles = 50, training set included 9% of possible shapes.

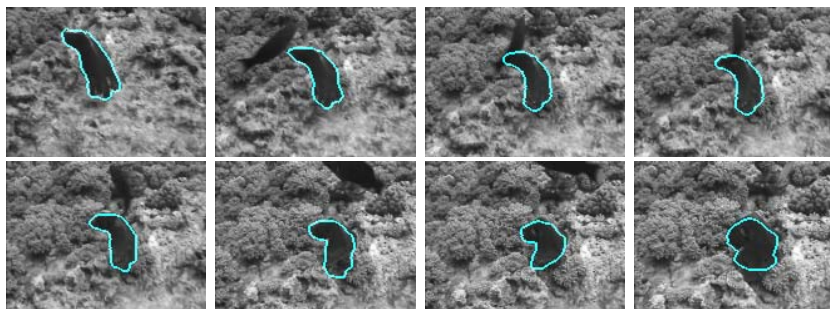


Fig. 4. Octopus Sequence: Results using the proposed algorithm. Notice that a fish with the same mean intensity occludes the octopus.

4.3 Soccer Sequence

This sequence tracks a man playing soccer. There is large deformation in the shape due to movement of the limbs (hands and legs) as the person tosses the ball around. The deformation is also great from one frame to next when the legs occlude each other and separate out. There is clutter in the background which would cause leaks if geometric active contours or the particle filtering algorithm given in [12] were used to track this sequence (see Figure 5). Results of tracking using the proposed method are shown in Figure 5. The following set of parameters were used to track this sequence: $L_r = 5$, $L_d = 18$, particles = 50, and 20% of the possible shapes were included in the training set (see Figure 1).

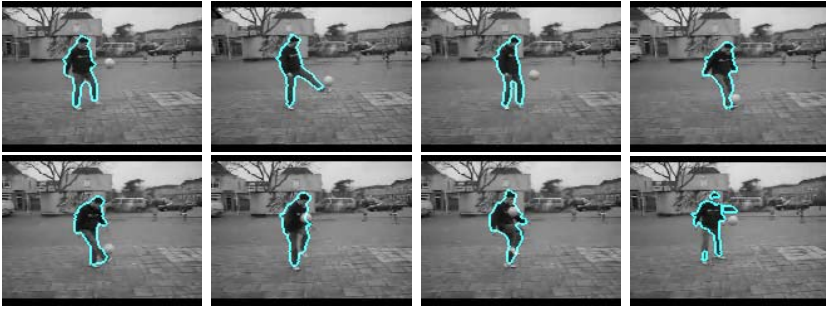


Fig. 5. Results of tracking using the proposed method. Last image at the bottom right is the segmentation using equation (4) without any shape information.

5 Conclusions and Limitations

In this paper, we have presented a novel method which incorporates dynamic shape prior information into a particle filtering algorithm for tracking highly deformable objects in presence of noise and clutter. The shape prior information is obtained using Locally Linear Embedding (LLE) for shapes. No motion or shape dynamics are required to be known for tracking complex sequences, i.e., no learning is required. The only information needed is a set of shapes that can appropriately represent the various deformations of the object being tracked.

Nevertheless, the current algorithm has certain limitations. First, it is computationally very expensive, as each particle has to be evolved for many iterations. Second, the training set should contain sufficient number of possible shapes of the object being tracked so that LLE can be used.

References

1. Blake, A., Isard, M., eds.: Active Contours. Springer (1998)
2. Terzopoulos, D., Szeliski, R.: Tracking with Kalman Snakes. In: Active Vision. MIT Press (1992) 3–20
3. Zhang, T., Freedman, D.: Tracking objects using density matching and shape priors. In: Proceedings of the Ninth IEEE International Conference on Computer Vision. (2003) 1950–1954
4. Gordon, N., Salmond, D., Smith, A.: Novel approach to nonlinear/nongaussian bayesian state estimation. IEE Proceedings-F (Radar and Signal Processing) (1993) 140(2):107–113
5. Sethian, J.A.: Level Set Methods and Fast Marching Methods. 2nd edn. Cambridge University Press (1999)
6. Osher, S.J., Sethian, J.A.: Fronts propagation with curvature dependent speed: Algorithms based on hamilton-jacobi formulations. Journal of Computational Physics **79** (1988) 12–49
7. Paragios, N., Deriche, R.: Geodesic active contours and level sets for the detection and tracking of moving objects. Transactions on Pattern analysis and Machine Intelligence **22**(3) (2000) 266–280

8. Yezzi, A., Soatto, S.: Deformation: Deforming motion, shape average and the joint registration and approximation of structures in images. *International Journal of Computer Vision* **53**(2) (2003) 153–167
9. Dambreville, S., Rath, Y., Tannenbaum, A.: Shape-based approach to robust image segmentation using kernel pca. In: *IEEE, CVPR*. (2006)
10. Dambreville, S., Rath, Y., Tannenbaum, A.: Tracking deforming objects with unscented kalman filtering and geometric active contours. In: *American Control Conference*. (2006)
11. Leventon, M., Grimson, W.L., Faugeras, O.: Statistical shape influence in geodesic active contours. In: *Proc. CVPR*. (2000) 1316–1324
12. Rath, Y., Vaswani, N., Tannenbaum, A., Yezzi, A.: Particle filtering for geometric active contours with application to tracking moving and deforming objects. In: *Proc. CVPR*. (2005)
13. D.Ridder, Duin, R.: Locally linear embedding for classification. Technical Report PH-2002-01, Pattern Recognition Group, Delft University of Technology (2002)
14. Saul, L.K., Roweis, S.: (An introduction to locally linear embedding)
15. Toyama, K., Blake, A.: Probabilistic tracking in a metric space. In: *ICCV*. (2001) 50–59
16. Heap, T., Hogg, D.: Wormholes in shape space: Tracking through discontinuous changes in shape. In: *ICCV*. (1998) 344
17. Tsai, A., Yezzi, A., Wells, W., Tempany, C., et. al.: A shape based approach to the segmentation of medical imagery using level sets. In: *IEEE Tran. On Medical Imaging*. Volume 22. (2003)
18. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. *Science* **290**(5500) (2000) 2323–2326
19. Cremers, D., Soatto, S.: A pseudo-distance for shape priors in level set segmentation. In: *IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision*. (2003)
20. D. Cremers, S. Osher, S.S.: Kernel density estimation and intrinsic alignment for knowledge-driven segmentation: Teaching level sets to walk. *Pattern Recognition, Tbingen, Springer LNCS* **3175**(3) (2004) 36–44
21. Funkhouser, T., Kazhdan, M., Shilane, P., Min, P., Kiefer, W., Tal, A., Rusinkiewicz, S., Dobkin, D.: Modeling by example. In: *ACM Transactions on Graphics (SIGGRAPH 2004)*. (2004)
22. Rousson, M., Deriche, R.: A variational framework for active and adaptative segmentation of vector valued images. *Motion, Workshop 2002* **00** (2002) 56
23. Doucet, A., deFreitas, N., Gordon, N.: *Sequential Monte Carlo Methods in Practice*. Springer (2001)
24. Doucet, A.: On sequential monte carlo sampling methods for bayesian filtering. In: *Technical Report CUED/F-INFENG/TR. 310, Cambridge University Department of Engineering*. (1998)
25. van der Merwe, R., de Freitas, N., Doucet, A., Wan, E.: The unscented particle filter. In: *Advances in Neural Information Processing Systems 13*. (2001)

The OBSERVER: An Intelligent and Automated Video Surveillance System

Duarte Duque, Henrique Santos, and Paulo Cortez

University of Minho, Department of Information Systems, Campus de Azurém,
4800-058 Guimarães, Portugal
{duarteduque, hsantos, pcortez}@dsi.uminho.pt

Abstract. In this work we present a new approach to learn, detect and predict unusual and abnormal behaviors of people, groups and vehicles in real-time. The proposed OBSERVER video surveillance system acquires images from a stationary color video camera and applies state-of-the-art algorithms to segment and track moving objects. The segmentation is based in a background subtraction algorithm with cast shadows, highlights and ghost's detection and removal. To robustly track objects in the scene, a technique based on appearance models was used. The OBSERVER is capable of identifying three types of behaviors (normal, unusual and abnormal actions). This achievement was possible due to the novel N-ary tree classifier, which was successfully tested on synthetic data.

Keywords: Moving object detection, Tracking, Behavior detection.

1 Introduction

In the 70's, Tickner and Poulton, two researchers from the psychology field published a study [1] about the efficacy of human surveillance when dealing with a large number of cameras. The study has demonstrated that the level of attention and the accuracy of abnormal event detection decreases along the time. The authors also verified that human factors, such age and sex, can influence the reliability of incident detection.

A recent work [2] from the Department of Experimental Psychology, at the University of Bristol, was dedicated to study if potentially antisocial or criminal behavior could be predicted by humans when viewing real images recorded in CCTV. The outcome of a signal-detection analysis established that the estimates of the perceived magnitude of the evidence in favor of an incident were very similar for surveillance experts and novices, with an average true positive detection rate of 81%. The study suggests that prediction of future events is indeed possible, although it is imperfect. Moreover, the authors pointed out to the possibility of developing automatic units to detect abnormal events.

Some attempts to automatically detect and predict abnormal behaviors have been already presented. The ADVISOR system [3], aiming to detect vandalism acts, crowding situations and street fights, was one of the most relevant works in this field. It made use of a 3D model of a monitored area, where abnormal actions were previously defined by security experts, who described those acts using a predefined description language. This sort of approach led to a context dependent detection system,

where all the objects from the scene and relations between those objects need to be defined.

Alessandro Mecocci in [4] introduces an architecture for an automatic real-time video surveillance system, capable of autonomously detecting anomalous behavioral events. The proposed system automatically adapts to different scenarios without any human intervention, and uses self-learning techniques to learn the typical behavior of the targets in each specific environment. Anomalous behaviors are detected if the observed trajectories deviate from the typical learned prototypes. Despite the significant contribution of Mecocci’s work, it does not accomplish the particular features of a surveillance system, where typically the monitored area comprises different levels of security, i.e. the existence of restricted and public areas. Also, Mecocci made use of only spatial information, which is a significant lack of attributes for describing actions.

To overcome the identified problems, a project called OBSERVER, which is described in this paper, was started in 2004, aiming to automatically learn, detect and predict abnormal behaviors in public areas.

The paper is organized as follows. In section 2, the segmentation process, based on an adaptive background subtraction algorithm, which includes shadow, highlight and ghost detection techniques, is described. In section 3, we present the tracking algorithm and describe the appearance model approach. Then, in section 4, the main features of the automatic dataset generation are explained. The proposed method to predict abnormal behaviors and the creation of the N-ary tree classifier are also described. Finally, experimental results are presented in section 5 and, in section 6, we draw some conclusions.

2 The Segmentation Process

Segmenting moving objects is the first process of a computer based video surveillance system. It should provide accurate segmentation results in real-time. A good segmentation process should handle image source errors, such as white noise from the camera, cast shadows and highlights. To accomplish that, the OBSERVER system uses an adaptive background subtraction algorithm, combined with shadow, highlight and ghost detection and removal methods.

In the first step of the proposed segmentation process, given an image I and a background image B , both in the RGB color space, we select the moving pixels based on a fixed threshold. As outcome, a binary mask, called Primary Motion Mask (PMM) is generated.

$$PMM_n(x, y) = \begin{cases} 1 & \text{if } \arg \max_{R,G,B} |I_n^{R,G,B}(x, y) - B_n^{R,G,B}(x, y)| > T \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

This primary mask may be affected by error sources as we can see in Figure 1 (c). Therefore, the process continues with the detection of shadows and highlights from the PMM . This is achieved by a color space transformation of the acquired image from RGB to HSV, from which the shadows and highlight binary masks are produced, by the following operations:

$$SM_n(x, y) = \begin{cases} 1 & \text{if } \alpha \leq \frac{I_n^V(x, y)}{B_n^V(x, y)} \leq \beta \wedge |I_n^S(x, y) - B_n^S(x, y)| \leq \tau_S \wedge |I_n^H(x, y) - B_n^H(x, y)| \leq \tau_H \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$HM_n(x, y) = \begin{cases} 1 & \text{if } \frac{1}{\beta} \leq \frac{I_n^V(x, y)}{B_n^V(x, y)} \leq \frac{1}{\alpha} \wedge |I_n^S(x, y) - B_n^S(x, y)| \leq \tau_S \wedge |I_n^H(x, y) - B_n^H(x, y)| \leq \tau_H \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The $I_n^H(x, y)$, $I_n^S(x, y)$ and $I_n^V(x, y)$ represent respectively the hue, saturation and value components of a pixel at coordinate (x, y) from the input image I at time n . The same notation was applied to the background image B .

In the above expressions, $0.65 \leq \alpha \leq 0.80$, $0.90 \leq \beta \leq 0.95$, $\tau_S \leq 15\%$ of digitizer saturation range and $\tau_H \leq 60$ degrees of color. This is explained in detail in previous work [5]. The PMM is then subtracted by SM and HM , generating a Motion Mask (MM).

$$MM_n = PMM_n \cap \neg SM_n \cap \neg HM_n \quad (4)$$

Concerning noise removal, the use of separated masks for shadow and highlight detection allows a better result than was obtained with a common mask. The noise is removed from each mask by a morphologic open operator followed by a close, using a 3x3 structuring element.

The resulting foreground regions from the MM are then filled in, in order to eliminate holes inside segmented blobs. The resulting blobs are therefore filtered by an area threshold, where small regions are considered as noise and removed from the MM . Using this mask, the blobs are labeled and grouped into objects to be tracked, as described in the next section.

Some of the detected objects may be ghosts, i.e. false positives originated by displacements of background objects. In those cases, ghosts regions need to be detected in the MM , in order to adapt the background model accordingly.

As stated before, an object can be composed by several segmented blobs.

$$Object_j = \bigcup_{i=0}^N Blob_i, \quad N \in \mathbb{N} \quad (5)$$

We identify a segmented object as a ghost, only if all the blobs that constitute the object are classified as ghosts. This classification is determined as follows:

$$Blob_i = \begin{cases} ghost & \text{if } \frac{Perimeter(\phi(I_n) \cap \phi(Blob_i))}{Perimeter(\phi(Blob_i))} < 10\% \\ motion & \text{otherwise} \end{cases} \quad (6)$$

where Φ denotes the edge detection operator.

The choice of the technique to be applied in the edge detection is critical. It must present good results and low response times. For this purpose the Canny, Sobel and Perwitt algorithms were tested.

Despite the result quality, the computational requirements for those algorithms exceeded significantly the time requirements for our application. To overcome this drawback, a faster and simple, yet efficient edge detection algorithm was used. It compares each pixel value with its four connected neighbor pixels. If the absolute

difference between a pixel and one of its neighbors is higher than a given threshold, that pixel is marked as an edge. Based on this description, the following operator will then produce a binary image with all the edges detected.

$$\Phi(D(x, y)) = \begin{cases} 1 & \text{if } |D(x, y) - D(x-1, y-1)| > T \vee |D(x, y) - D(x-1, y+1)| > T \vee \\ & |D(x, y) - D(x+1, y-1)| > T \vee |D(x, y) - D(x+1, y+1)| > T \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

Finally, the background image is updated in the RGB and HSV color spaces. Pixels belonging to shadows, highlights and objects remain unchanged. Background pixels of segmented regions classified as ghosts are set with the actual image value. The remaining pixels are updated by an infinite impulse response filter in order to adapt the reference image to slight luminosity changes. The background updating can therefore be expressed by:

$$B_{n+1}(x, y) = \begin{cases} B_n(x, y) & \text{if } motion \vee shadow \vee highlight \\ I_n(x, y) & \text{if } ghost \\ \alpha \cdot B_n(x, y) + (1 - \alpha) \cdot I_n(x, y) & \text{otherwise} \end{cases} \quad (8)$$

where α controls how fast are the changes introduced.

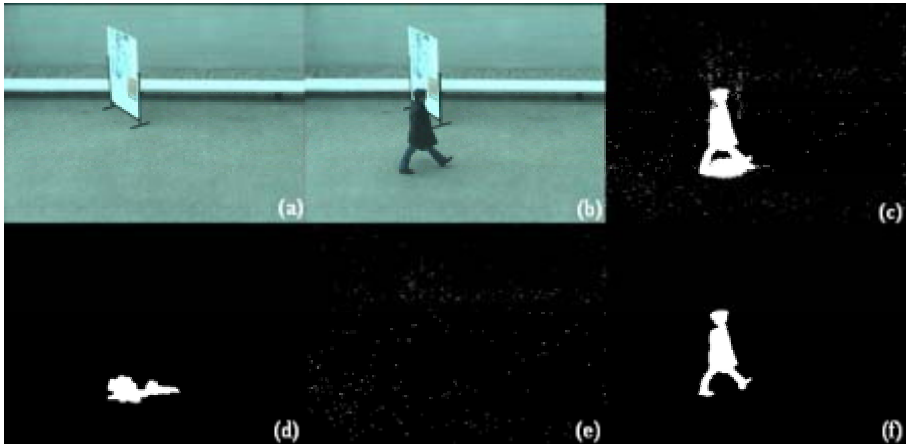


Fig. 1. (a) Background Image; (b) Current Image; (c) Primary Motion Mask; (d) Shadow Mask; (e) Highlight Mask; (f) Filtered Motion Mask

3 Tracking of Moving Objects

After segmenting moving objects, it is necessary to track them over the time, as long as they remain on the scene. For this task, a method based on appearance models was chosen.

Appearance models can be seen as dynamic memory representations of objects that are updated, at each frame, during the track operation. They are composed by two masks: the Appearance Image (*AI*), which maintains a color model of the object; and

the Probability Mask (*PM*), that represents the most likely shape that the object can assume. An example of an appearance model can be seen in Figure 2.

At each frame we will have a list of objects and, if in the presence of tracks, a list of tracks, respectively:

$$O = \{O_1, O_2, \dots, O_J\} \text{ with, } O_j = \{BB, M\}$$

$$T = \{T_1, T_2, \dots, T_K\} \text{ with, } T_k = \{ID, Type, BB, AI, PM\}$$

Where *BB* is the bounding box of the object, *M* is the object mask, *ID* is the track identifier, and *Type* indicates the object’s class, i.e. “person”, “group” or “vehicle”. The tracking algorithm is divided in four steps: track setup; track alignment; pixel assignment; and track update.

3.1 Track Setup

In the track setup, a correspondence matrix *C*, which associates each object *O_j* to a track *T_k*, is created. This matrix with *JxK* dimension, where *J* is the number of segmented objects and *K* the number of detected tracks, is a binary matrix with value one in an element *C_{i,j}*, when the object *O_j* is associated with a track *T_k*.

This association is established if the currently segmented object *O_j* is overlapped by a previous track *T_k*. An example of a correspondence matrix is:

$$C_{j,k} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

In this example, there are three objects and four tracks. The first segmented object (first line) is a merge of the first and second tracks, i.e. first and second columns, respectively. Also, the first and second objects derived from a split from the second track. The third object is single tracked by track three, and the track four was not found at the current frame.

If an object can’t be assigned to at least one track, then a new track must be created for that object. In that case, the *AI* of the track is initialized with the color image of the segmented object, the *PM* is initialized with an intermediate value, and a new *ID* is assigned to the track. The correspondence matrix *C* is then processed, and objects that are associated to a same track are joined together, forming a macro-object (*MO*).

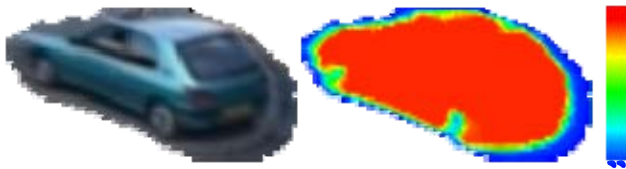


Fig. 2. Example of an Appearance Model. At left the appearance image, and at right the Probability Mask with scale.

3.2 Track Alignment

After performing the track setup, a list of MO and a list of tracks (T_k) are presented. In order to align the tracks, for each observed MO we will try to find the track displacement that best fits the appearance model of T_k with the MO . This is done for all the tracks assigned to a MO , and starting from those tracks whose objects are closest to the camera. The displacement (δ) is obtained by maximizing the fitting function P_{FIT} :

$$\delta = (\delta x, \delta y) = \arg \max_{\delta} (P_{FIT}(MO, T_k, \delta)) \tag{8}$$

where,

$$P_{FIT}(MO, T_k, \delta) = \frac{\sum_{x,y \in MO} P_{APP}(I^{R,G,B}(x, y), AI_k(x - \delta x, y - \delta y)) \cdot PM_k(x - \delta x, y - \delta y)}{\sum_{x,y \in T_k} PM_k(x, y)} \tag{9}$$

Considering that color components are uncorrelated and have equal variance, then:

$$P_{APP}(I^{R,G,B}, \mu^{R,G,B}) = \frac{1}{\sqrt{(2\pi)^3} \cdot \sigma^3} \cdot e^{-\frac{1}{2} \left[\left(\frac{I^R - \mu^R}{\sigma} \right)^2 + \left(\frac{I^G - \mu^G}{\sigma} \right)^2 + \left(\frac{I^B - \mu^B}{\sigma} \right)^2 \right]} \tag{10}$$

The displacement is initialized with the values of the previous track and continues the search by a gradient descent approach. When the displacement that maximizes the fitting function is achieved, the pixels of the macro-object that match to the track are excluded for the following tracks fitting.

3.3 Pixel Assignment

The pixel assignment is used to decide when a track is finished or not. Once the tracks are aligned, each pixel of a MO with a PM value greater than zero is assigned to one of the tracks that compete for that pixel. If the ratio between the number of assigned pixels and the number of probability mask pixels with value different than zero is lower than a predefined threshold, the track is removed.

The association of each pixel to a track is made based on the product of the pixel probability to belong to the appearance image by the pixel probability to belong to the track, which is $(x, y) \in T_k$ for:

$$\arg \max_k (P_{APP}(I^{R,G,B}(x, y), AI_k(x - \delta x, y - \delta y)) \cdot PM_k(x - \delta x, y - \delta y)) \tag{11}$$

A MO pixel can be in one of three states. If it is assigned to the track T_k then we say it belongs to the set of the assigned pixels (AP_k). If the pixel doesn't have a value in the probability mask, then it is labeled as new. Otherwise, it is classified as missed pixel.

3.4 Track Update

The final step in the tracking procedure is the track update. In this step, the appearance image (AI) and the probability mask (PM) of single tracked objects are updated. Merged tracks will update only the position of the actual bounding box.

The appearance models are updated by the following rules:

$$PM_k^{t+1} = \begin{cases} \alpha \cdot PM_k^t(x, y) + (1 - \alpha) & \text{if } (x, y) \in AP_k \\ 0.5 & \text{if } \textit{new} \\ \alpha \cdot PM_k^t(x, y) & \textit{otherwise} \end{cases} \quad (12)$$

$$AI_k^{t+1} = \begin{cases} \alpha \cdot AI_k^t(x, y) + (1 - \alpha) \cdot I(x, y) & \text{if } (x, y) \in AP_k \\ I(x, y) & \text{if } \textit{new} \\ AI_k^t(x, y) & \textit{otherwise} \end{cases} \quad (13)$$

where, $0 \leq \alpha \leq 1$.

The appearance model of a tracked object remains in memory while the object is in the scene. After an object leaves the monitored area, the appearance model is deleted from the computer memory. As effect, if an object reenters in the scene, it will be detected as a new object, and not an extension of the previously observed movement. To overcome this issue, an approach that maintains, for a predefined period of time, the appearance models of departed objects, can be used. However, the RAM memory becomes a critical resource when dealing with a huge number of objects.

4 Behavior Detection and Prediction

The final goal of the OBSERVER system is to predict abnormal behaviors. For this propose the monitored temporal sequences of object attributes (object position, area, perimeter, velocity, etc) must be processed by a classifier in order to infer the type of action performed by an object. Such classifier will need to learn, from a dataset, the typical patterns of normal and abnormal events.

4.1 Automatic Dataset Generation

Typically, in an area under surveillance, the focus of attention falls into two types of events: unusual behaviors (e.g. a person running in a hotel lobby, or a car in wrong-way driving in a motorway) and violation of a restricted area (for example with pedestrians crossing a highway). The OBSERVER system was developed to successfully handle both types of events.

While the first type of events can be solved entirely by a classification algorithm, the violation of restricted areas requires human intervention in the system configuration stage. To accomplish that, the user only needs to mark the scene's restricted areas.

After the definition of the restricted areas, if there is any, the system starts to track objects moving in the scene. The observed tracks, along with the attributes of the objects, are recorded in real-time into a database. Tracks that overlap restricted areas are flagged as alarm events, the others are signaled as normal events. When the number of recorded tracks reaches to a predefined value, considered sufficient to the learning phase, then the classifier engine starts.

4.2 Constructing the N-Ary Tree Classifier

Before explaining the process of building the N-ary tree classifier, used to classify object behaviors, we will start by clarifying the idea behind the proposed classification system and the architecture of the classifier.

The behavior of an object can be described by a set of actions that it performs in a certain environment. From the security point of view we could expect, at least, three kinds of observations: *normal*, *unusual* or *abnormal* actions. Normal actions are those which are frequently observed and do not origin the violation of any restricted area. Unusual actions are those that are not common or have never occurred. When an action leads to a violation of a restricted area, then it should be classified as an abnormal action.

Besides this basic function, the OBSERVER will be able to predict those events from the registered object path. This takes us to a classifier that should respond to the following question: if an object, with specific properties, travels until a certain point, what is its probability to follow a path that will cause an abnormal event?

We propose such a classifier, using an N-ary tree, whose nodes are multivariable Gaussian distributions $N(\mu, \Sigma)$ in the object attribute's hyperplane, and have an abnormal probability (P_{ab}). To simplify the computation, we assume that the variances, in the covariance matrix Σ , are not correlated.

Has we can see in Figure 3, the classifier's tree is organized by layers. Each layer defines a period of time, so every track should be described by a sequence of nodes, each one from a different layer, starting in "layer 1", when the observed object enters the scene. The nodes present clusters of related points, i.e. Gaussian distributions, and subsequent layers are separated by a fixed time slice. For example, we can define that each layer will have a time period of ten frames.

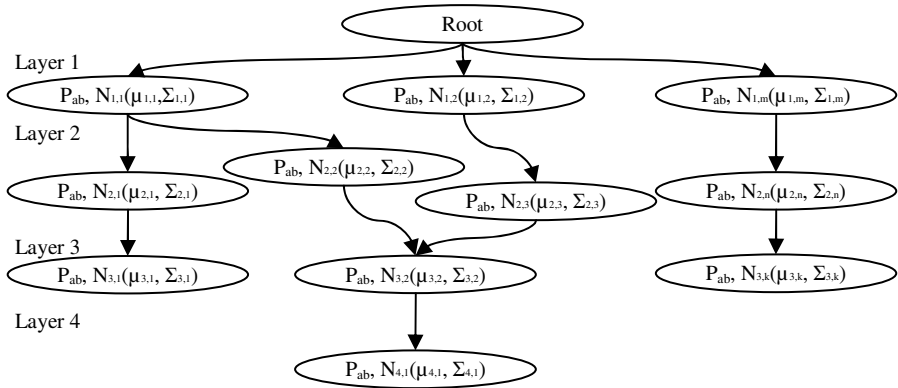


Fig. 3. Example of an N-ary tree classifier

To build the classifier, we use as input the set of prerecorded tracks. Each track is defined by a sequence of attribute vectors, describing the properties of an object at each sample. The following object attributes are considered: the 2D coordinates center-of-gravity; velocity; area; type descriptor (i.e. human, group or vehicle); and an alarm flag.

The first step consists on the partitioning of the data into equal time slices, i.e. layers. Then, for each layer of a track and for each attribute, the sampled data is processed in order to compute the mean value of the attribute for that layer. When this process finishes, the clustering is performed.

Since there is no information about the number of clusters in a layer, a clustering algorithm capable to infer the number of expected distributions should be used. For this

propose, an Expectation-Maximization algorithm with automatic cluster number detection based in k-fold cross-validation [6] was implemented, with k = 10, as described next.

4.3 Clustering the Data

Consider X a d-component column vector of object attributes, μ the d-component mean vector, Σ the d-by-d covariance matrix, and $|\Sigma|$ and Σ^{-1} its determinant and inverse, respectively.

For each layer there are N track samples, and initially the number of classes (C) is set to one. The first step of the k-fold cross-validation is to divide the dataset into ten fractions. Next, nine fractions are selected to compute the expectation and maximization steps. The remaining fraction is used to compute the log-likelihood. This procedure is executed ten times, in order to compute the log-likelihoods of all the fractions.

The final log-likelihood is calculated as the mean of the ten log-likelihoods. The value of classes (C) will be incremented while the log-likelihood (L) is not stabilized. After setting the number of clusters (C), the computation of the normal distributions is performed by the Expectation-Maximization algorithm, as described bellow.

Starting conditions:

$$\begin{aligned}
 & \text{foreach } j \in C \text{ do} \\
 & \quad P(W_j) = 1/C \\
 & \quad \mu_j = \text{random}(X) \\
 & \quad \bar{\mu}_j = \sum_{i=1}^N X_i / N \\
 & \quad \Sigma_j = \sum_{i=1}^N (X_i - \bar{\mu}_j)^2 / N
 \end{aligned}$$

Maximization step:

$$\begin{aligned}
 & \text{foreach } j \in C \text{ do} \\
 & \quad \hat{P}(W_j) = \frac{\sum_{i=1}^N P(W_j | X_i)}{N} \\
 & \quad \hat{\mu}_j = \frac{\sum_{i=1}^N [X_i \cdot P(W_j | X_i)]}{N \cdot \hat{P}(W_j)} \\
 & \quad \hat{\Sigma}_j = \frac{\sum_{i=1}^N [P(W_j | X_i) \cdot (X_i - \hat{\mu}_j) \cdot (X_i - \hat{\mu}_j)^T]}{N \cdot \hat{P}(W_j)}
 \end{aligned}$$

Expectation step:

$$\begin{aligned}
 P(X) &= \sum_{j=1}^K [P(X | W_j) \cdot P(W_j)] \\
 & \text{foreach } j \in C \text{ do} \\
 P(W_j | X) &= \frac{P(X | W_j) \cdot P(W_j)}{P(X)} \\
 & \text{where,} \\
 P(X | W_j) &= \frac{1}{(2\pi)^{d/2} \cdot |\Sigma_j|^{1/2}} \cdot e^{-\left(\frac{(X - \mu_j)^T \cdot \Sigma_j^{-1} \cdot (X - \mu_j)}{2}\right)}
 \end{aligned}$$

Stopping criteria:

$$\begin{aligned}
 & \text{do} \\
 & \quad \text{Expectation_Step}() \\
 & \quad \text{Maximization_Step}() \\
 & \quad L = \sum_{i=1}^N \ln(P(X_i)) \\
 & \quad \text{while}(|L - L_{old}| < \varepsilon)
 \end{aligned}$$

When the stopping condition is satisfied, the mean and variance values of the distributions of a layer are merged. After executing the clustering over the entire dataset,

a set of clusters for each layer is obtained. The next step consists on building the classification tree, defining links between clusters of different layers.

4.4 Linking the N-Ary Tree Classifier

To build the classification tree, it is necessary to go through all the track sequences of the dataset. For each track, we will try to find a path (sequences of clusters, each cluster in a different layer) that represents it. When a dissimilar path is observed, a new branch is created. During this process of tree nodes linking, the information about the event type (i.e. normal or abnormal event) of each track and at each time slice is recorded within the clusters. At the end, all nodes of the tree will have the information about: mean and variance value of the multivariable normal distribution; and the number of normal and abnormal events observed by the cluster. In this way, it is possible to infer the probability of an object to exhibit an abnormal behavior, associated with a certain path.

5 Experimental Results

In this section we present the preliminary experiments to evaluate the proposed N-ary tree classifier. The dataset used to evaluate the classifier was artificially generated. For this propose, a semiautomatic track generator software, designated OBSERVER-TG, was developed. This tool allows the user to generate tracks in a scene, with configurable Gaussian noise over the object position, velocity, area and perimeter.

A dataset of 180 tracks with different lengths was created, with a restricted area, as shown in Figure 4. The set of generated tracks represent six different paths. Ten of the tracks violated the restricted area, originating alarm events. All the abnormal tracks started at bottom left of the scene, cross the road and turn left into the protected area.

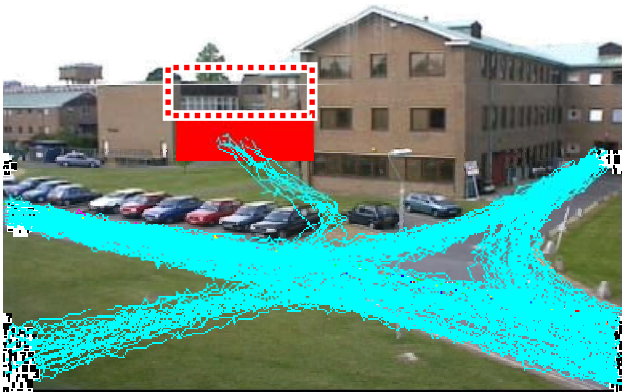


Fig. 4. Background overlapped by 180 tracks from the dataset and a restricted area (red box)

To evaluate the accuracy of the proposed classifier, the dataset was randomly fractioned in three parts. To access the system performance, a hold-out validation scheme [6] was adopted, where 2/3 of the simulated data was used for training (e.g. learning phase) and the remaining 1/3 for testing. Tracks that have probability of abnormal behavior greater than zero are classified as abnormal events, and sequences that have unobserved paths are classified as unusual events. The test results obtained in the three simulations (A, B, C) are presented in the table bellow.

Table 1. Test results of the behavior classifier

| Test Set | | | OBSERVER Test Results | |
|----------|----------|----------|-----------------------|----------|
| | N°Tracks | Abnormal | Unusual | Abnormal |
| A | 60 | 5.00% | 1.66% | 3.33% |
| B | 60 | 8.33% | 5.00% | 3.33% |
| C | 60 | 3.33% | 1.66% | 1.66% |

For instance, we can see a reasonable performance of the OBSERVER in the simulation A, with a test set that has 5% of abnormal tracks. The system has successfully detected 3.33% of tracks as abnormal events, and 1.66% as unusual events. Summating the abnormal with the unusual results, we obtain the total of 5%.

When analyzing the results, we detected that all the normal tracks from the test dataset were correctly classified. This outcome is a consequence of the similarity of the generated tracks. In a real situation, when the type of observed paths is more chaotic, we can expect that a considerably amount of safety tracks will be classified as unusual events. In such situation, the system user should be prompted to classify unusual events as normal or abnormal behaviors, for use in future and refined classifiers.

6 Conclusions

In this work we presented a new approach to automatically detect and predict abnormal behaviors. The proposed solution is based on an adaptive background subtraction algorithm, an appearance model tracking technique and an N-ary tree classifier which has performed well with artificial test data. Moreover, the N-ary tree classifier possesses complementary features that can be exploited to find the most used paths and to analyze the occupancy of the monitored areas.

The classifier prototype was constructed as a stand alone application due to the expensive computation required by the clustering process and the necessary evaluation tasks. However, once finished, it seems to be sufficiently fast to meet the real-time constraints of a surveillance system.

Despite the confidence level on the results obtained, the validation of this solution still lacks of an extensive test with real data. Therefore, our main concern in the near future is to extend the experiments of the proposed classifier to different scenarios, with a greater number of paths, both from simulated and real-world data.

References

1. Tickner A.H., Poulton E.C.: Monitoring up to 16 synthetic television pictures showing a great deal of movement. *Ergonomics*, Vol. 16. (1973) 381-401
2. Troscianko T., Holmes A., Stillman J., Mirmehdi M., Wright D., Wilson A.: What happens next? The predictability of natural behaviour viewed through CCTV cameras. *Perception*, Vol. 33 (1). (2004) 87-101
3. INRIA, ADVISOR – Annotated Digital Video for Intelligent Surveillance and Optimised Retrieval, 2006, URL: <http://www-sop.inria.fr/orion/ADVISOR/>
4. Mecocci A., Pannozzo M., Fumarola A.: Automatic detection of anomalous behavioural events for advanced real-time video surveillance. *International Symposium on Computational Intelligence for Measurement Systems and Applications*. Lugamo (2003) 187-192
5. Duque D., Santos H., Cortez P.: Moving object detection unaffected by cast shadows, highlights and ghosts. *IEEE International Conference on Image Processing*. Genoa (2005) 413-416
6. Kohavi R.: A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*. San Francisco (1995) 1137-1143.

A Novel Spatio-temporal Approach to Handle Occlusions in Vehicle Tracking

Alessandro Bevilacqua and Stefano Vaccari

DEIS - ARCES (Advanced Research Center on Electronic Systems)

University of Bologna,

Via Toffano, 2/2 – Bologna, Italy, 40125

abevilacqua@arces.unibo.it

Abstract. In Intelligent Transportation Systems (ITS's), the process of vehicle tracking is often needed to permit higher-level analysis. However, during occlusions features could “jump” from an object to another one, thus resulting in tracking errors. Our method exploits second order statistics to assess the correct membership of features to respective objects, thus reducing false alarms due to splitting. As a consequence, object's properties like area and centroid can be extracted stemming from feature points with a higher precision. We firstly validated our method on toy sequences built ad hoc to produce strong occlusions artificially and subsequently on sequences taken from a traffic monitoring system. The experimental results we present prove the effectiveness of our approach even in the presence of strong occlusions. At present, the algorithm is working in the daytime in the Traffic Monitoring System of the city where we have been living.

1 Introduction

Nowadays many Traffic Monitoring Systems (TMS's) rely on computer vision techniques in order to analyze statistics on traffic flows or to detect traffic congestion. To this purpose, these systems must be able to detect and track vehicles. Often, they rely on a distributed network of monocular cameras and the presence of occlusions among objects is one of the most challenging problems to deal with. As a matter of fact, when one camera is used and two or more vehicles overlap in a 2-D view tracking them separately is a daunting task because the 2-D scene viewpoint could not permit a correct separation between such moving objects. Some approaches coping with occlusion handling include a priori knowledge by using 2-D or shape-based models. However, when a car turns in depth, we can have self-occlusion: 2-D model-based trackers will fail in this case, while shape-based methods are likely to lose track of the car because of the fast shape changing speed. In fact, when vehicles are even partially occluded, some of their feature points are still visible. Nevertheless, during occlusions feature points may “jump” between vehicles and mislead a feature tracker. A probable solution could be exploiting more information or employing higher level methods. As a matter of fact, some other research works are based on 3-D models. However, they require camera calibration and a priori knowledge about objects that will move on the scene. Also, algorithms are often computationally expensive and they can cope with real-time applications regarding urban traffic scenes with difficulty.

The work we present has been developed within a feature-based algorithm that tracks every object as a set of feature points. The system does not rely on any a priori knowledge about the targets within the scene. The method we devised exploits second order statistics to improve the reliability of objects tracking during occlusions by disambiguating uncertain features. That is, we calculate the probability of features, whose ownership is doubtful, to belong to each occluding object. Although our algorithm has been developed to deal mainly with vehicles, it does not rely on rigid body constraints. Rather, using second order statistics allows to relax these constraints. Each uncertain feature is assigned to the vehicle which the feature has the maximum likelihood; this happens for the vehicle which is the “most rigid” *with respect to that feature*. In fact, during an occlusion between (one or more) vehicles, the more a vehicle follows rigid body constraints, the stronger the relationship between its features. However, it is worth remarking that, since this is a statistical approach, each feature has always a probability to belong to a given object, although it does not follow the rigid body constraint. For this reason the algorithm works well even in the presence of objects that do not follow rigid body constraints. The probability is simply as high as the vehicle’s uncertain features adhere to the rigid body constraint. This relaxation allows us to cope with vehicles rotating in depth and, in general, with vehicles (cars, bicycles and motorcycles) independently of either their trajectory or the angle of shooting, as shown in the experiments. Practically speaking, our method is employed within a tracking algorithm to reduce the presence of non-existent objects (false alarms) derived from splitting and thus improving the accuracy of object’s features (e.g.: centroid, area) needed for higher-level analysis (e.g.: trajectory analysis, area based object classification). The experiments accomplished by using challenging sequences showing urban road intersections allow us to assess the quality and the robustness of our method even in the presence of difficult occlusions. It is also worth remarking that this work represents the first research exploiting second order statistics to disambiguate uncertain features.

As for the paper’ scheme, Section 2 analyses previous works regarding with object tracking. Section 3 outlines the principles of the overall tracking algorithm. Section 4 constitutes the core of the work and explains how statistics are used during occlusions. In Section 5 extensive experiments accomplished both on challenging sequences built to stress algorithm and sequences taken from urban traffic scenes are discussed. Finally, Section 6 draws conclusion and gives some lines for future work.

2 Previous Work

Many methods are known to track vehicles which use features and manage occlusions. In [1] features are used to track vehicles in highways, by grouping sets of features with the same motion. Homography combined with camera calibration is used in the grouping process to take road perspective into consideration. Features are tracked in consequent frames by a Kalman filter. This is feasible due to the roughly constant speed of vehicles in a highway. Features are grouped if relative distances are under a given threshold, or by speed similarity. Grouping thresholds could cause errors in object tracking due to over-grouping or unnecessary segmentations, which create tracking misses

and non-existent objects respectively. Besides, the proposed method can work only with highway traffic, so it would be applicable with great difficulties in urban intersections.

The authors in [2] propose a technique to remove outliers from the trajectories of feature points fitting a subspace and removing those points that have large residuals. They generate and track features through the entire video stream using the KLT[10] algorithm, then apply the RANSAC method to detect trajectories that does not follow the rigid movement constraint. This constraint is applicable when it is reasonable to assume an orthographic view with a weak perspective. So this algorithm fails when movements involve strong perspective changes, thus limiting its effectiveness only to short sequences. However, short sequences could not permit to detect correct trajectories, providing too few data for the RANSAC algorithm to converge. Also, the computational time needed to remove outliers is high and not compliant with real time processing.

In [3] tracking is performed by template matching, with a probabilistic model based on a robust error norm. Matching is performed by finding the image region that yields the maximum likelihood with respect to the calculated probability distribution. During occlusions, a pixel is regarded as an outlier if the measurement error exceeds a pre-defined threshold. Templates cannot be updated during occlusion, so it is necessary to detect when overlapping ends. Although this algorithm can manage complete occlusions, it cannot withstand severe occlusions lasting more than 25 frames, due to wrong Kalman filter estimation.

In [5] a SOM neural network is exploited to remove feature tracking errors during occlusions. The speeds of all the features belonging to an object are inspected by the neural network searching for outliers that do not follow the rigid body assumption. The algorithm can cope with scaling and rotations of objects in a perspective view because using the neural network allows to relax the rigid body constraint. As a drawback, features with similar speeds that jump from one object to another could not be detected as outliers, because the two objects are interpreted as a single “relaxed” rigid body. The method we propose in this paper can cope with those situations through performing more precise feature labeling and object segmentation, accordingly.

A graph-cut based method is used in [8], where corner points are extracted by means of the Harris algorithm to create an initial seed for the segmentation process. The KLT tracking algorithm is used to extract image regions presenting corners with the same affine transformation. Seed regions are then expanded propagating the region’s front, identifying pixels of the same layer by means of a bi-partitioning graph-cut method. Occlusions are analyzed in a small temporal interval. Pixels are labeled as “occluded” if they obey to some temporal constraint. This multi-frame motion segmentation problem can be formulated as an energy minimization problem, introducing other links in the segmentation graph. Computing the graph-cut on all the image pixels brings computation time up to 30 seconds per frame, preventing this method to be applied in real-time applications.

Finally, the problem of finding reliable feature clusters in tracking algorithms has been faced by means of the eigenvector decomposition of the affinity matrix, based on the distribution of the distance vectors between frames. An interesting overview is reported in [4]. However, such approaches can be used just for clustering but cannot be employed as methods to resolve uncertainty. In fact, the aim of the eigenvector decomposition-based methods is that of extracting the most representative component

of a class, after the outliers (that is, the uncertain values) have been removed. At the opposite, our approach deals with sort of outliers aiming at disambiguishing them by assigning a unique label instead of removing them.

3 The Tracking Algorithm

The method presented in this paper has not been conceived as being a tracking algorithm. Rather, it works on features coming from a robust tracking algorithm we have developed [5] and it must be seen just as a False Positive Reduction (FPR) step. That is, our approach enforces the overall robustness of the existent tracking algorithm, summarized in the present section, and aims at reducing tracking errors *only during occlusions* by working *just on* a limited region near to the vehicle occlusion borders. At present, the method we present is embedded in the tracking system working in an experimental stage in the TMS of the city where we live.

Our tracker works on moving regions (“blobs”) previously extracted by a motion detector [6] through a background difference approach [7]. Inside each blob we extract corner points by applying the Shi-Tomasi method [9] and then we track those features by using the pyramidal KLT approach [10]. We derived the Lucas-Kanade equation as in [11], even though a more efficient method is proposed in [12]. In fact, it cannot be applied to our case because we estimate only translation parameters to track corner points, while optimizations presented in [12] focus on feature trackers dealing with rotation and scaling as well. As a matter of fact, given sufficiently high a frame rate, those transformations between consecutive frames are negligible for small features.

However, some features may be lost during tracking. Those features are replaced by using a minimum density approach: that is, each blob is divided in small squares of the same size and each square is tested to check whether the number of currently tracked features is over a threshold. If not, we perform feature extraction in every square that does not reach the threshold value.

3.1 From Features to Objects

To track *objects* starting from *features*, sets of corner points belonging to the same object need to be grouped and labeled. Possible solutions include either grouping by speed or by statistics given by spatial feature position. In the first case the grouping criterion is based on speed vectors similarity, while in the second case features are grouped together showing the same relative distances over a period of time. However, both methods suffer from camera perspective: vehicles rotating in depth present features with different motion, that depends from the shooting point of view. Also, thresholds used for grouping cannot be known a priori. Homography, as in [1], could be used to build a planar view of the scene but only when object heights are negligible. To overcome the drawbacks listed above, we perform a blob-based grouping on a spatio-temporal basis, that is using second order statistics. We can now distinguish 3 different situations:

- **a new object enters the scene:** as soon as an object appears, we extract features f_j assigning a unique number $L(f_j)$ (the *label*) to all of them, that identifies the object during the whole tracking process. If a new blob enters the camera field of view in

consequent frames, new features are extracted just from the last entered part. Those new features acquire the same label of the others already existent within the same blob;

- **two or more objects enter the scene sharing the same blob:** initially they are recognized as a single moving object and share the same label. But when they split, thus resulting in two or more distinct blobs, the algorithm creates new labels and renames all the blobs;
- **occlusions:** when two or more objects previously distinct merge in one blob, more than one label can be found inside this new blob. To maintain minimum feature density in the blob area (as explained in Section 3) new features have to be extracted and labeled reliably during all the occlusion process.

Finally, to identify objects we group features having the same label. At visualization level each group of features is delimited by a bounding box showing also the label of the tracked object.

Among the different situations, after due consideration handling occlusions is the hardest problem to face.

4 Occlusion Handling

During occlusions, we can distinguish mainly between two kinds of features, according to their spatial position. Figure 1 shows such features regarding the simplest case of two

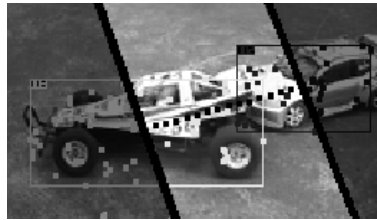


Fig. 1. The unreliable area during occlusions

tracked objects involved in the occlusion, without lacking in generality. We can have features far from occlusion boundaries which can be labeled reliably (darker regions of Figure 1). However, a region with unreliable features can be found near the occlusion boundaries (lighter area of Figure 1), since here features with different labels are close together, thus reducing labeling reliability. As a matter of fact, in the uncertainty region where objects overlap, some features often jump on the “wrong” object. For instance, this happens when the KLT algorithm finds the correspondence between features in consecutive frames but it fails to track the right spot on the image. Features which were assigned to the “right” object before occlusion are assigned to the “wrong” ones during occlusion. Each of these objects owns (at least) two different labels and it is erroneously split causing a *false object* when occlusion ends (see Figs. 2(b) and 3(a) in Section 5).

Therefore, a *feature* tracking error becomes an *object* tracking error, thus resulting in false alarms and unreliable estimation of geometric properties.

Due to the above reasons, the tracking algorithm assigns a label only to the features far from the occlusion boundaries, in order to avoid wrong label assignments. The uncertainty area in which labels are not assigned is defined by the centroids of the tracked objects inside the blob, calculated by averaging the coordinates of all the features sharing the same label. For each pair of objects involved in the occlusion the segment joining the centroids is considered (see Figure 1). Two parallel lines which are tangent and perpendicular with respect to this segment define the boundaries of the area in which features are chosen to elaborate statistics. Our statistical approach deals right with the features inside the uncertainty region. Practically speaking, feature labels are re-assigned when necessary by using second order statistics, to provide a subsequent correct segmentation between tracked objects. Hereinafter, we refer to this step as “relabeling”.

4.1 Second Order Statistics

Second order statistics is widely used in texture analysis to extract informations in still images. To this purpose, co-occurrence matrices are built to test correlation between couples of pixels at different distances. In order to better understand how our matrices are used let us start by describing the well-known co-occurrence matrices used in texture analysis.

A gray level co-occurrence matrix is defined as a sample of the joint probability density of the gray levels of two pixels separated by a given displacement. In Cartesian coordinates the displacement of the co-occurrence can be chosen as a vector $(\Delta x, \Delta y)$. According to the notation given in [13] the gray level co-occurrence matrix $N(\Delta x, \Delta y, a, b)$ of an image I is defined as

$$N(\Delta x, \Delta y, a, b) = \{\#(a, b) \mid I(x, y) = a \wedge I(x + \Delta x, y + \Delta y) = b\} \quad (1)$$

where a and b are gray levels. In our implementation we evaluate x and y distances for every pair of features $f_p, f_q \in F$ where F is the entire set of features. We sample feature coordinates in a sliding time window of size T thus obtaining:

$$C(f_p, f_q, x, y) = \{\#(x, y) \mid d_x(f_p, f_q) = x \wedge d_y(f_p, f_q) = y\} \quad (2)$$

where d_x, d_y are distances between features in Cartesian coordinates. We call each matrix C the “Spatio Temporal Co-occurrence Matrix” (STCM). We have $\frac{1}{2}\#F \times (\#F - 1)$ matrices, one STCM for each pair of tracked features. For each matrix C we have

$$\sum_{y=1}^N \sum_{x=1}^M C(f_p, f_q, x, y) = T \quad (3)$$

where N and M are image width and height.

Most of the targets we cope with are vehicles. Although vehicles are rigid bodies, the shooting angle of a monocular camera can yield 2-D views where the rigid body constraints do not hold any more. For instance, this happens when objects rotate in

depth or change their scale due to a movement perpendicular to the camera view plane. Of course, reducing the sampling period (the time window T) results in smaller object changes. However, this yields fewer data to analyze that even cannot be enough. On the contrary, enlarging the time window will improve data accuracy at the expense of possible heavier changes of a vehicle appearance. We have seen experimentally that choosing $T = 1\text{sec}$ fits most of the traffic scenes at urban intersections while a longer period could be chosen for highway traffic scenes, where vehicle's trajectory is mostly linear.

4.2 Application of Statistics

Statistics are built by evaluating the distances between every pair of features inside each blob during the whole tracking process. As soon as a new feature f is extracted by the KLT algorithm, STCM matrices are calculated. If for a pair of features (f, g) the distance (d_x, d_y) kept constant we will find a high value of $C(d_x, d_y, f, g)$, thus meaning a high likelihood for the features f and g to be a rigid part of the same (rigid or not rigid) object. On the contrary, a scattered matrix reveals changes in the distance of the feature pair, thus pointing out a membership to quite a non rigid part of a body or even to different bodies. As we stated before, our approach only serves to reduce features unreliability. Only few corner points are included in the relabeling process. Now, let us see which features are taken into consideration by the algorithm:

- **features along the occlusion borders:** experimentally we have seen that features prone to tracking errors are mainly located in the overlapping zone along the occlusion borders. This consideration has been used to apply statistics only to couples of features (f, g) for which

$$L(f) \neq L(g) \wedge \Gamma(f, g) < \epsilon \quad (4)$$

where $L(x)$ represents the label of a feature x and $\Gamma(f, g) = \sqrt{d_x(f, g)^2 + d_y(f, g)^2}$. The value of ϵ is chosen by taking into consideration the maximum translation of a feature point;

- **features extracted in the uncertainty zone:** in this are it is possible to have features tracked correctly which have not been provided with a label yet. In fact, tracking points that are lost by the KLT algorithm are replaced with subsequent extractions of new corners, whose assignment is delayed when they fall in the uncertainty zone. Meanwhile those features are tracked, so that statistical data could be acquired.

4.3 Probability Estimation

Camera vibrations and approximations inside the KLT algorithm (that result in small fluctuations of feature placements) may generate noise in distance measurements. Therefore, in order to reduce noise before using STCM matrices we perform a Gaussian smoothing (with $\sigma = 1.0$) as a discrete convolution of STCM matrices with a square mask M of size $q = 5$:

$$G(f, f_j, x, y) = \frac{\sum_{n=1}^q \sum_{m=1}^q C\left(f, f_j, x - \frac{q}{2}, y - \frac{q}{2}\right) \cdot M(m, n)}{\sum_{n=1}^q \sum_{m=1}^q M(m, n)} \tag{5}$$

then probability matrices can be computed as:

$$P(f, f_j, x, y) = \frac{G(f, f_j, x, y)}{\sum_{n=1}^N \sum_{m=1}^M G(f, f_j, x, y)} \tag{6}$$

where M and N are the width and the height of the image.

Let O be the set of objects $O_k, k = 1 \dots N_O$ involved in an occlusion, with $N_O = \#O$. Features are then $f_j^k \in O_k, j = 1 \dots N_k$ where N_k is the number of features of O_k . Therefore the probability for a feature f to belong to an object O_k is given by Eq (7):

$$P(f \in O_k) = \frac{\frac{1}{N_k} \sum_{j=1}^{N_k} P(f, f_j^k, d_x, d_y)}{\sum_{k=1}^{N_O} \frac{1}{N_k} \sum_{j=1}^{N_k} P(f, f_j^k, d_x, d_y)} \tag{7}$$

where distances d_x and d_y are measured along the Cartesian axes between f and f_j^k . We provide f with the label of the object the feature most probably belongs to, according to Eq (8):

$$L(f) = L(O_{\hat{k}}), O_{\hat{k}} \mid \max_k P(f \in O_k) \tag{8}$$

Therefore, when the feature f has been tracked for one entire time-window T , then statistical data is analyzed. As a result, f will get the label of the object O_k whose features have changed least their distance from f . That is, f will get the label of the object that better fits the rigid body constraint. Again, it is worth remarking that this does not require for the body to be rigid or appear as being rigid (for example, a vehicle rotating in depth). Simply, the label $L(f)$ will be that of the *mostrigid* body, among the ones involved, with respect to the feature f .

5 Experimental Results

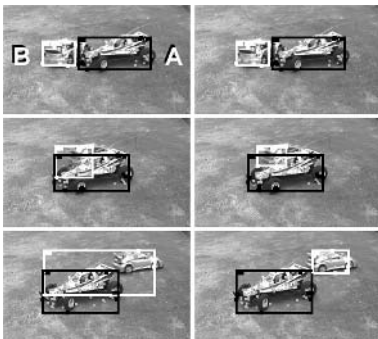
Before testing our algorithm with real sequences, we validated our algorithm by acquiring a sequence T0 by using RC-cars, to create a wide number of situations where statistics can be used. Besides, these cars appear as being real vehicles having yet stronger interactions. After that, we tested the algorithm using some video sequences (T1 and T2) from the TMS of our city, showing an urban environment. Sequences have been acquired at 12.5 f/sec and are of 1514, 9245 and 37913 frames, respectively. All of them

have been elaborated with the following parameters, which have been kept unchanged also in the experimental stage of our system in the TMS:

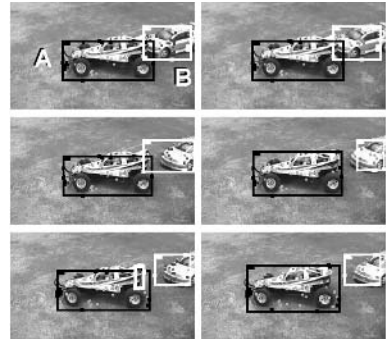
$$\begin{cases} T = 9 & \text{frames} \\ \epsilon = 20 & \text{pixels} \end{cases}$$

T0 contains 28 occlusion cases, for a total amount of 311 frames where cars overlap. All figures show different cases of occlusion and the results referring same frames are shown without (left) and with (right) the use of our method. Besides, hereinafter we refer to frame i as the couple of frames starting from top.

In Figure 2(a) vehicle B is performing an in-depth turn while covered by vehicle A (frame 1). One feature $f \in B$ shifts onto A (frame 2), producing a dilation of the bounding box (frame 3) and a consequent error in computing all the related features (centroid, etc.). However, when using our method almost immediately $P(f \in A)$ becomes higher than $P(f \in B)$, so our algorithm asserts that $L(f) = L(A)$. In the next two figures we show an example of how a far heavier error can be avoided through using our method. In Fig. 2(b) vehicle A occludes vehicle B and both are stopped (frame 1).



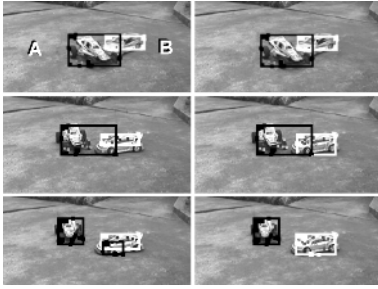
(a) Left, without statistics; right, with statistics



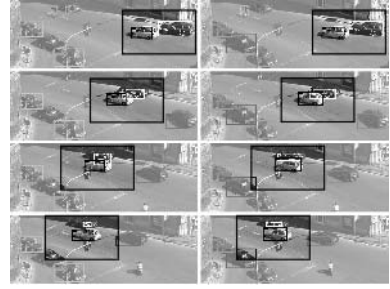
(b) Left, without statistics; right, with statistics

Fig. 2. Two occlusion examples in sequence T0

Then some features of B, in the overlapping area, remain attached to A while B moves backward (frame 2). When blob splits (frame 3) those features appears as belonging to a new objects and get a new. That is, a non-existent object (false alarm) is generated and erroneously tracked for some frames, thus propagating the error until the false object leaves the scene. With the application of our method, the features erroneously ended up on B are correctly re-labeled as A thus preventing this problem to occur. As for Fig. 3(a), here A cuts across B’s path and some features jump from A to B, as proved by the large black bounding box of frame 1 and frame 2. However, in frame 2 you can see that the bounding box of vehicle A better adheres when using statistics, since the “jumping” features are correctly assigned to vehicle A. Also, at the end of the occlusion after that A went in reverse, the wrong assignment of the uncertain features yields a



(a) Left, without statistics; right, with statistics



(b) Left, without statistics; right, with statistics

Fig. 3. Occlusions in T0 and T1 sequences

split which produces a false new object (frame 3). At the opposite, our method prevent the features to jump on vehicle A, thus handling correctly this occlusion. It is worth noticing that splitting is one of the most common in the tracking systems and one of the most difficult to cope with. Let us come to the sequence T1 extracted from the TMS. In the traffic intersection of Fig. 3(b), vehicles move at slightly different speeds, thus requiring a very precise estimate of the probability of the uncertain features, according to Equation 7. Here, the two highlighted cars move in the same direction. For all the occlusion period (70 frames) statistics is useful to both detect features jumped on the wrong car and assign labels to features extracted in the uncertainty region. Often, occlusions in sequences of real world can be simpler than those built artificially in sequence T0. As a matter of fact, in T1 tracking errors are less frequent than in T0, mainly because vehicles are smaller. As a consequence, statistics is useful also to confirm the

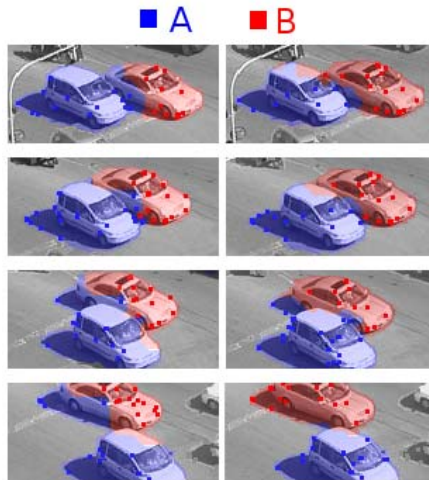


Fig. 4. Left, without statistics; right, with statistics

labels assigned to features. Finally, in Fig. 4 you can see a couple of occluding cars, after having been tracked separately. You can see some features belonging to A near the occlusion border which jump on the slower car (B, in frame 1 and 2, left) when no statistics is used. Without using statistics, these features are erroneously given to A car in all of the 4 frames. Even though cars are tracked correctly, this results in an unreliable separation which alters both blob's properties. We would also like to stress how such errors tend to propagate because once a feature has been erroneously assigned to an object (frame 3, left), this assignment makes the erroneous bounding box, as well as all the features included, reliable. Therefore, after being assigned to the A, the features of B are tracked as really belonging to A (frame 4, left).

6 Conclusion and Future Work

In this paper we have proposed a method to improve the performance of a feature based tracking algorithm during occlusion. Our approach exploits second order statistics to detect and remove feature tracking errors occurred during occlusions. Each uncertain feature is given the label of the object to which it has shown the maximum "attraction power", that is the maximum probability derived from second order statistic. This results in a better defined separating borders among occluding objects, thus improving the outcome of the tracking system in terms of reliability of properties of the objects being tracked. The algorithm has been tested on different sequences containing different cases of occlusion and always it has proved to disocclude vehicles with a high accuracy. Future work could be focused on considering to extend our approach to some more properties besides the feature position, such as to some values related to the texture analysis of a small region centered on the feature being tracked. It is worth noticing that the overall tracking algorithm has been working in real time in the daytime at an experimental stage in the TMS of the city where we have been living.

References

1. Beymer D., McLauchlan P., Coifman B., and Malik J., "A Real-Time Computer Vision System for Measuring Traffic Parameters", *Proc. IEEE CVPR*, San Juan, Puerto Rico, pp. 495-501, 1997.
2. Sugaya Y. and Kanatani K., "Outlier Removal for Motion Tracking by Subspace Separation", *8th Symposium on Sensing via Image Information (SSII2002)*, Yokohama, Japan, pp. 603-608, July 2002.
3. Nguyen H.T., Smeulders A.W.M., "Fast Occluded Object Tracking by a Robust Appearance Filter", *IEEE Transactions on PAMI*, pp. 1099-1104, August 2004.
4. Weiss Y., "Segmentation using eigenvectors: a unifying view", *Proc. IEEE ICCV*, pp. 975-982, 1999.
5. Bevilacqua A., Di Stefano L., and Vaccari S., "Occlusion Robust Vehicle Tracking based on SOM (Self-Organizing Map)", *7th IEEE Workshop on Motion and Video Computing (Motion'05)*, October 5-7, 2005, Breckenridge, CO, USA., Vol.2, pp.84-89.
6. Bevilacqua A., Di Stefano L., and Lanza A., "An efficient motion detection algorithm based on a statistical non parametric noise model", *17th IEEE International Conference on Image Processing (ICIP 2004)*, October 24-27, 2004, Singapore, pp.2347-2350.

7. Bevilacqua A., Di Stefano L., and Lanza A., "A Simple Self-Calibration Method To Infer A Non-Parametric Model Of The Imaging System Noise", *7th IEEE Workshop on Motion and Video Computing (Motion'05)*, January 5-7, 2005, Breckenridge, CO, USA., Vol.2, pp. 229-234.
8. J. Xiao and M. Shah, "Motion Layer Extraction in the Presence of Occlusion Using Graph Cut", *Proc. IEEE CVPR*, Vol.2, Washington, DC, USA, pp. 972-979, 2004.
9. Shi J. and Tomasi C., "Good features to track", *Proc. IEEE CVPR*, pp. 593-600, 1994.
10. Lucas B. and Kanade T., "An iterative image registration technique with an application to stereo vision", *DARPA Image Understanding Workshop*, pp. 121-130, 1981.
11. Birchfield S., "Derivation of Kanade-Lucas-Tomasi tracking equation", *Unpublished*, January 1997.
12. Baker S. and Matthews I., "Lucas-Kanade 20 Years On: A Unifying Framework", *IJCV*, 56, 3, pp. 221-255, 2004.
13. Chang P. and Krumm J., "Object Recognition with Color Cooccurrence Histograms", *Proc. IEEE CVPR*, Fort Collins, CO, pp. 498-504, 1999.

A Robust Particle Filter-Based Face Tracker Using Combination of Color and Geometric Information

Bogdan Raducanu¹ and Y. Jordi Vitrià^{1,2}

¹ Centre de Visió per Computador, Edifici O - Campus UAB

² Departament de Ciències de la Computació, Universitat Autònoma de Barcelona
08193 Bellaterra, Barcelona, Spain
{bogdan, jordi}@cvc.uab.es

Abstract. Particle filtering is one of the most successful approaches for visual tracking. However, so far, most particle-filter trackers are limited to a single cue. This can be a serious limitation, since it can reduce the tracker's robustness. In the current work, we present a multiple cue integration approach applied for face tracking, based on color and geometric properties. We tested it over several video sequences and we show it is very robust against changes in face appearance, scale and pose. Moreover, our technique is proposed as a contextual information for human presence detection.

1 Introduction

Face tracking is required for a large number of computer applications: HCI, surveillance, biometry, video compression, human-robot communication, etc. There are many approaches proposed to solve this problem, but among them a very common technique is represented by particle filter. Particle filter has been introduced in [7] and its basic idea consists of propagating a probability distribution through time, based on sample sets. Its powerfulness is represented by its ability to track simultaneously non-linear/non-gaussian multi-modal distributions, solving in this way the limitations imposed by Kalman filter [24]. Since it was introduced, it was used mainly for object-tracking purposes, but it also has been proposed in robotics for the problem of self-localization for mobile robots [20].

Particle filtering has been mainly used to track objects based on their color histograms [26], [12], [15], [13] or geometrical shape [7], [6]. The use of single cue proved to offer good results, but since the most applications are intended for dynamic environments, with continuous changes in illumination conditions, shape, partial or total occlusion of the object being tracked, this could pose a limitation in the robustness of these methods. The solution is represented by the integration of several cues, like in [16], [21], [14].

In the current paper, we propose a novel method for face tracking based on the fusion of color and geometrical structure of the face. In other words, the likelihood between the model and the hypotheses ("particles") is expressed in terms

of a linear combination between color similarity, difference in size and euclidean distance. The novelty is represented by the fact that in order to initialize the tracker and to update the confidence, a model-based approach for face detection is used. The face evidence is confirmed using Haar-like features trained with the Ada-Boost algorithm. In case of self-occlusions (head is rotated), when the detector fails despite a person continues to be present in front of the camera, a "CamShift" algorithm is employed, in order to confirm the face presence based on skin color. This could represent a great advantage, since some approaches are using either manual initialization or a learned histogram of the object to be tracked. We try to argue the relevance of our method and the improvement it introduces, by comparing it with other recent publications:

- instead of working with the RGB color model (like in [12]), we prefer HSV, since it is less sensitive to changes in illumination conditions
- the "mean-shift" algorithm from [17] was replaced by the "CamShift" algorithm. "Camshift" [3] stays for "continuous adaptive mean-shift", and unlike the "mean-shift", which is designed for static distributions, "CamShift" can deal with dynamically changing distributions.

The paper is structured as follows: section 2 contains a brief recall to particle filter; section 3 focuses upon observation model; in section 4, the whole system is presented and the integration of several cues is explained; some experimental results are reported in section 5; finally, section 6 contains our conclusions and the guiding lines for future work.

2 Particle Filtering

The tracking problem can be formulated in the framework of partially observable Markov chains [5] that considers the evolution of a state vector sequence x_k over time. This implies the estimation of the *a posteriori* density over the state x_k from all available sensor measurements $z_{0:k} = z_0 \dots z_k$. A solution to this problem is given by Bayes filters [8], which computes this *a posteriori* density recursively (we make here a first-order Markov assumption, i.e. the state x_k depends only on x_{k-1}):

$$p(x_k | z_{0:k}) = K \cdot p(z_k | x_k) \cdot \int p(x_k | x_{k-1}) \cdot p(x_{k-1} | z_{0:k-1}) dx_{k-1} \quad (1)$$

If the *a posteriori* distribution at any moment is Gaussian and the distributions $p(x_k | x_{k-1})$ and $p(z_k | x_k)$ are linear in its arguments, than the equation (1) reduces to the Kalman filter. If one of these conditions cannot be met, than a possible alternative is to use some approximation methods in order to linearize the actuation and measurements models. There are two extensions of the Kalman filter that can be used in these situations, known as Extended Kalman Filter [10] and Unscented Kalman Filter [9]. However, most of the real-world processes are non-linear and these approximations don't hold. The solution to the general case is given by Particle Filtering.

The basic idea of a Particle Filter is to maintain a multiple hypothesis (*particles*) about the object being tracked. This means that the distribution $p(x_k|z_{0:k})$ is represented using a set $\{s_k^n, w_k^n\}$, where $n = 1..N$ (N is the number of particles). The weights should be normalized such that $\sum_n w_n = 1$. In other words, each particle has associated a weight that reflects its relevance in representing the object. The initial set of particles can be randomly generated or using the output of a detector. The *a posteriori* probability is updated in a recursive way, according to the following steps:

- from the previous set of particles $\{s_{k-1}^n, w_{k-1}^n\}_{n=1..N}$, draw one particle through importance sampling, so s_k^m will correspond to some s_{k-1}^j :

$$s_k^j \sim w_{k-1}^j \tag{2}$$

- the chosen particle is propagated according to the model's dynamics:

$$s_k^n \sim p(x_k|x_{k-1} = s_{k-1}^n) \tag{3}$$

- assign a new weight to the recent generated particle equal to the likelihood of the observation, i.e.:

$$w_k^n \sim p(z_k|x_k = s_k^n) \tag{4}$$

then normalize again the resulting weights and store the new obtained set of particles.

Once the N particles have been constructed, an estimate for the current object state can be obtained from the following formula:

$$\xi[f(x_k)] = \sum_{n=1}^N w_k^n f(s_k^n) \tag{5}$$

where $f(x) = x$.

From time to time, it is necessary to resample the particles, otherwise could appear degeneracy problems, which is the concentration of the whole weight on a single particle. The resampling procedure duplicates the particles with higher weights, while discards those with lower weights. In consequence, without resampling the performance of the algorithm will be considerably degraded. A more complete discussion about the degeneracy problem can be found in [2].

In regard with the dynamical model, we chose a simple one, described by a linear stochastic differential equation:

$$x_k = Ax_{k-1} + Bu_k \tag{6}$$

where A is the state transition matrix, u_k is a vector of standard normal random variates and BB^T is the process noise covariance.

3 Model-Based Approach

In order to estimate and improve the accuracy in the prediction of the particle filter, two other independent methods have been used. The integration between a motion-based approach and a model-based approach for tracking is not new and was previously proposed in [11] and more recently in [25]. The purpose of integrating these two approaches is twofold. First, the face detector is used to initialize the face tracker. There are other methods to initialize the tracker (either automatically, starting with a random position and expecting that after a number of frames it will converge towards the solution, or manually). In our case we opted for an automatic method which guarantees the best initialization for the tracker. On the other hand, our system is aimed for long-term use (i.e. hours, maybe even days). We plan later on to use it for automatic acquisition of faces under different poses. In this situation, the reason for using a face detector would be to reinforce and to help the tracker to recover from possible failures when it gets distracted. Without this confirmation from the detector, the system won't run properly for more than a few hundred frames. The consequence will be that the system will fail quite often and will require repetitive initialization.

As first method, we employed a face detector based on Haar-like features trained with the Ada-Boost. This works very well for frontal faces. In case of self-occlusion (head is rotated), which is the most common source of failures for the face detector, we substitute this evidence with a, color-exclusive method, represented by the "CamShift" algorithm. The use of these two approaches is complementary, since the use of a color-exclusive method (without having any other independent information of facial features) could degrade the performance of the tracker. Passing a hand in front of the face (which has the same color distribution), could fool the tracker and thus loose the real target. We always prefer the use of the first method, over the second.

3.1 Face Detection Using Haar-Like Features

Ideally, techniques for face detection are desired to show robustness against changes in illumination, scale and head poses. They should be able to detect faces despite variation in facial expression and the presence or absence of natural or artificial accessories like beards and glasses. The face detection techniques can be divided in two main classes: holistic and local ones. The weakness of the holistic approaches is that they represent the images as raster vectors without any information about the 2D topology of facial features. On the other hand, face detectors based on local information have the advantage of providing more accurate information about the presence of face parts, by classifying image patches in 'face' or 'non-face' vectors.

Following this second path, we implemented a boosted face detector similar to those proposed in [22]. The idea is to use a cascade of weak classifiers in order to produce a stronger one. Each classifier is trained with a few hundreds samples of a particular object (a face), using Haar-like wavelet features, depicted in figure 1.

After each training epoch, the wrong classified features are retrained. Each classifier outputs "1" if the corresponding sample corresponds to a face and "0"

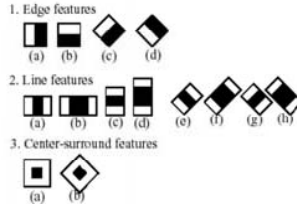


Fig. 1. Haar-like wavelet features used to train the weak classifiers

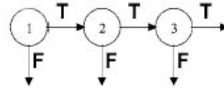


Fig. 2. The cascade representation of weak classifiers. The features that are wrong classified are rejected through the 'F' output.

1. Input: Training examples (x_i, y_i) , $i = 1..N$ with positive ($y_i = 1$) and negative ($y_i = 0$) examples.
2. Initialization: weights $w_{1,i} = \frac{1}{2^m}, \frac{1}{2^l}$ with m negative and l positive examples
3. For $t=1, \dots, T$:
 - (a) Normalize all weights
 - (b) For each feature j train classifier h_j with error $\epsilon_j = \sum_i w_{t,i} |h_j(x_i - y_i)|$
 - (c) Choose h_t with lowest error ϵ_t
 - (d) Update weights: $w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$ with $e_i = \begin{cases} 0 & : x_i \text{ correctly classified} \\ 1 & : \text{otherwise} \end{cases}$
and $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$
4. Final strong classifier: $h(x) = \begin{cases} 1 & : \sum_{t=1}^T \alpha_t h_t(x) \geq 0.5 \sum_{t=1}^T \alpha_t \\ 0 & : \text{otherwise} \end{cases}$
with $\alpha_t = \log(\frac{1}{\beta_t})$

Fig. 3. The Ada-Boost algorithm

otherwise. The final strong classifier is obtained as a linear combination of weak classifiers, followed by a threshold. This is depicted in figure 2.

The classifier is designed in such a way that allows detection of faces at different scales. This is a more convenient technique than downsampling the image itself. A very popular algorithm to train the weak classifiers in order to obtain the strong one is represented by Ada-Boost. The algorithm is summarized in figure 3.

3.2 "CamShift" Algorithm

In this section we give a brief review of this algorithm. It has been proposed as an extension of the mean-shift algorithm. Mean-shift works at one distribution scale, but is not suitable for another scale as the object moves towards and away from the camera. Small fixed-sized windows may get lost entirely for large object translation in the scene. Large fixed-sized windows may include distractors (other

people or hands) and too much noise. "CamShift" uses continuously adaptive probability distributions (that is, distributions that are updated for each frame) while mean-shift is based on static distributions, which are not updated. Since "CamShift" does not maintain static distributions, spatial moments are used to iterate towards the mode of the distribution. This is in contrast to the conventional implementation of the mean-shift algorithm where target and candidate distributions are used to iterate towards the maximum increase in density using the ratio of the current (candidate) distribution over the target.

The probability distribution image may be determined using any method that associates a pixel value with a probability that the given pixel belongs to the target. A common method is known as histogram back-projection [19] which associates the pixel values in the image with the value of the corresponding histogram bin. The back-projection of the target histogram with any consecutive frame generates a probability image where the value of each pixel characterizes probability that the input pixel belongs to the histogram that was used. Formally, the method can be described as follows. The ratio histogram, for a given color c is given by:

$$r_c(I, Q) = \min \left\{ \frac{H(Q)}{H(I)}, 1 \right\} \quad (7)$$

where $H(Q)$ represents the histogram of the model and $H(I)$ is the histogram of the image. It also represents the likelihood of a pixel $p \in I$ to belong to the target. The contribution of a subimage $I_p \subset I$ is given by:

$$\Pi_p(I, Q) = \sum_{q \in I_p} r_{I(q)}(I, Q) \quad (8)$$

Then the location of the target is given by:

$$\arg \max_p \Pi_p(I, Q) \quad (9)$$

4 Face Tracking System

4.1 Dynamic Model

In our problem about face tracking, a particle is represented by a combination of spatial and color information: $s_k = \{p_x, p_y, p_\sigma, H(p\Phi)\}$. The first three parameters represent the state vector $\{x_k = p_x, p_y, p_\sigma\}$ where (p_x, p_y) is the center of the particle and p_σ - the size of the particle. The last term $H(p\Phi)$ is the color histogram of the area 'under' the particle. We use the HSV color space, because it is less sensitive to illumination conditions. It separates the original RGB channels into H - hue (color), S - saturation and V - brightness. The color histogram is created by taking 1D histograms from the H (hue) channel in HSV space.

The tracker is initialized using the face detector based on Haar-like features. The face measurement is represented also as a feature vector $z_k =$

$\{f_x, f_y, f_\sigma, H(f\Phi)\}$, where (f_x, f_y) is the center of the face, f_σ - the size of the face and $H(p\Phi)$ is the color histogram of the face. At each frame, we reaffirm the confidence in our prediction, by invoking the feature-based face detector (in case of failure, face evidence is searched using the color-based detector implementing the "CamShift" algorithm). For this reason, we have to define a likelihood measure for our observations.

4.2 Observation Likelihood

In the particle filter, we have to update at each frame, the weight of the particles. This is achieved by computing the likelihood of the predicted parameters. Our observation model consists of a combination between color and geometrical information, thus we will use a scheme to fuse these cues into a global likelihood.

Color Cue. A common similarity measure between distributions is given by the Bhattacharyya distance [1]. When representing a color object through its histogram, we can express it as a discrete density, i.e $H(Q) = \{q^u\}_{u=1..n}$, where q_i represent the normalized bins of the histogram. Considering we have two discrete densities, $H(Q) = \{q^u\}_{u=1..n}$ and $H(R) = \{r^u\}_{u=1..n}$, we define the following coefficient:

$$\rho[Q, R] = \sum_{u=1}^m \sqrt{q^u r^u} \tag{10}$$

The larger ρ is, the more similar the color distributions are. Finally, the Bhattacharyya distance is defined as:

$$d = \sqrt{1 - \rho[Q, R]} \tag{11}$$

Based on this similarity measure, the likelihood between a candidate particle and the actual face model is given by:

$$p(z_k^c | x_k^i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{d^2}{2\sigma^2}} \tag{12}$$

where the c superscript designates the color cue. The interpretation of the above equation is that large weights correspond to small Bhattacharyya values.

In our approach we don't use a previously learned histogram for skin color distribution. We prefer instead to use an adaptive strategy, extracting the initial histogram from the very first instance of the detected face from the video stream. The update of the target model is done with the following equation:

$$H(m_t^u) = (1 - \alpha) H(m_{t-1}^u) + \alpha H(f\Phi^u) \tag{13}$$

for each histogram bin u of the model $H(M)$ and α represents the 'forgetting' term. As time passes, the relevance of the previously learned histograms tend to decrease and they are substituted by the most recent instances of face histograms.

Geometric Cues. The color cue is only one element to be taken into consideration when we appreciate the likelihood between the target and the model. Sometimes, color only cannot be sufficient. For this reason, we also make use of other elements, like size and position to the detected face. In other words, we want to be sure that not only the particles which are similar in color terms, but also has similar size and are situated in the neighborhood of the real face are the most confident candidates. The likelihood in size is expressed like:

$$p(z_k^s|x_k^i) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{d^2}{2\sigma^2}} \tag{14}$$

where $d = p_\sigma - f_\sigma$. The s superscript designates the size cue.

Meanwhile, the likelihood for position is given by the euclidean distance between the center of the predicted face and the face center provided by face detector in the previous frame. Thus,

$$p(z_k^p|x_k^i) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{d^2}{2\sigma^2}} \tag{15}$$

where $d = \sqrt{(p_x - f_x)^2 + (p_y - f_y)^2}$. The p superscript stays for the position cue.

Multiple Cue Integration. Several methods for multiple cue integration have been proposed. In [21] they propose a novel architecture for adaptively integrating different cues in a self-organized manner. They argue that this approach is sustained with results from psychophysical experiments. In Democratic Integration different cues agree on a result, and each cue adapts toward the result agreed on. In particular, discordant cues are quickly suppressed and recalibrated, while cues having been consistent with the result in the recent past are given a higher weight in the future. The same idea is further developed in [18] in order to allow the tracker to deal with multiple objects. Their idea exploits two methodologies for the adaptation to different conditions: in the first place, the integration scheme can be changed and in the second place, the visual cues themselves can be adapted. Adapting the visual cues allows to adapt to different environmental changes directly. Changing the integration scheme reflects the underlying assumption that different cues are suitable for different conditions.

In our case, we propose a simpler solution, based on a linear combination of the computed likelihoods for each cue. Thus, the global likelihood will be given by:

$$p(z_k|x_k^i) = w_k^i = \beta_1p(z_k^c|x_k^i) + \beta_2p(z_k^s|x_k^i) + \beta_3p(z_k^p|x_k^i) \tag{16}$$

where $\beta_j, j = 1, 2, 3$ are some constants. The final estimation position of the face is computed by maximizing the total likelihood:

$$x_{k \max} = \arg \max_i w_k^i \tag{17}$$

5 Experimental Results

The experimental results intend to show the improvements obtained by the introduction of the "CamShift" approach. In figure 4 we depict the results obtained using the particle filter method only. When the face evidence from the face detector is missing, the predicted position remains the same as in the last frame (we made the assumption that the proposed method is suitable for human-computer interaction, so in this case, the only possible occlusions are the ones due to head rotation - "self-occlusions"). Thus, the tracker is unable to cope with small head movements. On the other hand, when we introduced the "CamShift" approach, the tracker is able to robustly follow the face region, despite of the lack of face evidence provided by the Viola-Jones detector. This case is illustrated in figure 5.



Fig. 4. Face tracking results obtained using only the particle filter. When the detector misses the face, the predicted face is shown at the last detected position.

In order to confirm the robustness of our approach, we tested it on several video sequences and compared the results with two others methods: particle filter and "CamShift" algorithm. For this purpose, we compared the results obtained from each of these algorithms with ground-truth values. As ground-truth we considered the hand-labelled position and size of the face in a video sequence consisting of almost 700 frames (we marked the upper-left and bottom-right corners of the face region). The video sequence considered for analysis was a more complex one, in which the person was moving without restriction (in front of a cluttered background) in the camera's field of view. In figure 6 we plotted the error both in terms of position (left graphic) and size (right graphic) between the tracked face and the manual picked one. The error in position was expressed as the euclidean distance between the center of the tracked-face and the center of the manual-labelled one. From the left graphic, it can be appreciated that when the person is moving in a transversal plane (the camera perceives the face from profile) with respect to the camera's optical axes (that corresponds to the interval between the frames 100-300 and again between the frames 400-480), the particle filter alone is getting stuck, because no evidence from the face detector is provided. That's the interpretation of the high peaks in the graph. However,



Fig. 5. Face tracking results obtained through the combination of the particle filter and the "CamShift". In this case, no face is missed. The improvements obtained by the use of "CamShift" are obvious.

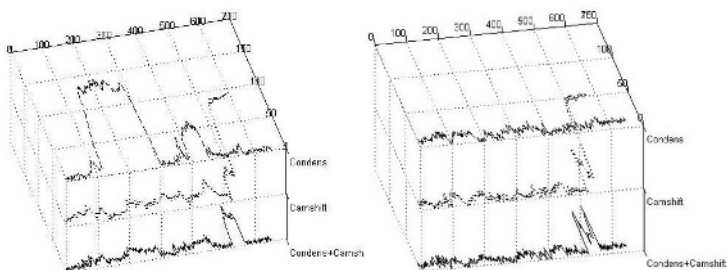


Fig. 6. Comparison between our approach and two others: particle filter and "CamShift". The plots represent the error in distance (left) and size (right) between the face reported by each of the three methods and ground truth. 'Condens' label states for particle filter.

"CamShift" and our approach perform well. Another situation can be found when the person passes his hand in front of the face (between the frames 530-580). This situation can be very well observed on the both graphs. Our system, despite its brief distraction, is able to recover after this occlusion. The particle filter is the most robust one in this case, because it is tracking the real position of the face. On the contrary, "CamShift" has lost the track forever.

6 Conclusions and Future Work

In this paper we proposed a solution for multiple cue integration in the framework of particle filters, with application to face tracking. We saw that the use of a single cue (either color or shape) is not sufficient to cope with dynamic environments. Thus, our method will offer an increased robustness against changes in

appearance, scale and pose. The successful framework of particle filter was combined with evidence about face presence obtained from a model-based approach. We tested our method on several video sequences and the results obtained are highly encouraging. In the near future, we plan to adapt our solution and implement it on an AIBO robot. Its primary task will be thus to use facial information as a contextual information for human-presence. For the beginning, the contextual information will be limited only to assess the user presence and distance to the robot. Later on, we will expand the contextual information, by including information about person's gender and identity. Regarding person identity, we are currently investigating a novel technique known as "unsupervised face recognition". That's it, the robot starts with an empty database, and will gradually 'get to now' new people.

Acknowledgements

This work is supported by MCYT Grant TIC2003-00654, Ministerio de Ciencia y Tecnología, Spain. Bogdan Raducanu is supported by the Ramon y Cajal research program, Ministerio de Educación y Ciencia, Spain.

References

1. Aherne, F., Thacker, N., Rockett, P.: The Bhattacharya Metric as an Absolute Similarity Measure for Frequency Coded Data. *Kybernetyka* **32** (1997) 1-7
2. Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T.: A Tutorial on Particle Filters for Online Nonlinear/Non-gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing* **50** (2002) 174-188
3. Bradsky, G.R.: Computer Vision Face Tracking for Use in a Perceptual User Interface. Technical Report 2001.2, Microcomputer Research Lab, Intel Corporation (2001)
4. Dey, A.K.: Understanding and Using Context. *Personal and Ubiquitous Computing Journal* **5** (2001) 4-7
5. Doucet, A., de Freitas, J.F.D., Gordons, N.J.: *Sequential Monte Carlo Methods in Practice*. Springer (2001)
6. Gatica-Perez, D., Lathoud, G., McCowan, I., Odobez, J.-M., Moore, D.: Audio-Visual Speaker Tracking with Importance Particle Filters. *Proceedings of IEEE International Conference on Image Processing (ICIP)*, Barcelona, Spain (2003) pp.N/A
7. Isard, M., Blake, A.: CONDENSATION - Conditional Density Propagation for Visual Tracking. *International Journal of Computer Vision* **29** (1998) 5-28
8. Jazwinsky, A.M.: *Stochastic Processes and Filtering Theory*. Academic Press (1970)
9. Julier, S.J., Uhlmann, J.K.: A New Extension of the Kalman Filter to Nonlinear Systems. *Proceedings of SPIE on Signal Processing, Sensor Fusion and Target Recognition VI* **3068** (1997) 182-193
10. Maybeck, P.: *Stochastic Models, Estimation and Control (I)*. Academic Press (1979)

11. McKenna, S., Gong, S.: Tracking Faces. Proceedings of 2nd International Conference on Automatic Face and Gesture Recognition (AFGR), Vermont, USA (1996) 271-276
12. Nummiaro, K., Meier, E.K., Gool, L.v.: An Adaptive Color-Based Particle Filter. *Image and Vision Computing* **21** (2003) 99-110
13. Pantrigo, J.J., Montemayor, A.S., Cabido, R.: Scatter Search Particle Filter for 2D Real-Time Hands and Face Tracking. Proceedings of International Conference on Image Analysis and Processing, Lecture Notes in Computer Science, Springer, Berlin Heidelberg, **3617** (2005) 953-960
14. Peihua, L., Chaumette, F.: Image Cues Fusion for Contour Tracking Based on Particle Filter. Proceedings of International Workshop on Articulated Motion and Deformable Objects. Lecture Notes in Computer Science, Springer, New York, **3332** (2004) 99-107
15. Pérez, P., Hue, C., Vermaak, J., Cangnet, M.: Color-based Probabilistic Tracking. Proceedings of the 7th European Conference on Computer Vision, Lecture Notes in Computer Science, Springer, New York, **2350** (2002) 661-675
16. Pérez, P., Vermaak, J., Blake, A.: Data Fusion for Visual Tracking with Particles. Proceedings of IEEE **92** (2004) 495-513
17. Shan, C., Wei, Y., Tan, Y.T., Ojardias, F.: Real Time Hand Tracking by Combining Particle Filtering and Mean Shift. Proceedings of the 6th Int. Conf. on Automatic Face and Gesture Recognition (AFGR), Seoul, Korea (2004) pp.N/A
18. Spengler, M., Schiele, N.: Towards Robust Multi-Cue Integration for Visual Tracking. Proceedings of International Conference on Computer Vision Systems, Lecture Notes in Computer Science, Springer, Berlin Heidelberg, **2095** (2001) 93-106
19. Swain, M., Ballard, D.: Color Indexing. *International Journal of Computer Vision* **7** (1991) 11-32
20. Thrun, S.: Particle Filters in Robotics. Proceedings of Uncertainty in AI (2002) pp.N/A
21. Triesch, J., Malsburg, C.v.d.: Democratic Integration: Self-Organized Integration of Adaptive Cues. *Neural Computation* **13** (2001) 2049-2074
22. Viola, P., Jones, M.J.: Robust Real-Time Face Detection. *International Journal of Computer Vision*, **57** (2004) 137-154
23. Weiser, M.: The Computer for the Twenty-First Century. *Scientific American* (1991) 94-10
24. Welch, G., Bishop, G.: An Introduction to Kalman filter. Technical Report TR95-041, Dept. of Comp. Science, Univ. of North Carolina (2002)
25. Williams, O., Blake, A., Cipolla, R.: Sparse Bayesian Learning for Efficient Visual Tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27** (2005) 1292-1304
26. Zajdel, W., Zivkovic, Z., Kröse, B.J.A.: Keeping Track of Humans: Have I Seen This Person Before?. Proceedings of International Conference on Robotics and Automation (ICRA), Barcelona, Spain (2005) 2093-2098

Author Index

- Abate, Andrea Francesco II-297,
II-353
Acha, Begoña II-502
Agis, Rodrigo I-196
Aguiar, Pedro M.Q. I-721, I-816
Ahuja, Narendra II-146
Alattar, Ashraf I-226
Alén, S. II-853
Alexander, Simon K. I-41
Alexandre, Luís A. II-612
Angel, L. I-850
Arcay, B. II-491
Asano, Tetsuo II-286
Ashourian, Mohsen I-580
Ávila, Bruno Tenório II-886
Azzari, Pietro II-764
- Baltazar, Joel II-832
Bandeira, Lourenço II-691
Bao, Zhiqiang II-102
Barata, Teresa II-691
Barreira, N. I-272
Basir, Otman II-398
Ben Hamadou, Abdelmajid I-414
Benshair, A. II-23
Bernardino, Alexandre II-181
Berretti, Stefano I-636
Bevilacqua, Alessandro I-910,
II-764, II-906
Bhutta, Adeel A. I-94
Bielskis, Antanas Andrius II-47
Bodansky, Eugene II-468
Borges, Janete S. II-700
Botha, Charl P. II-636
Bouaziz, Bassem I-414
Bourgeois, Steve II-250
Boutemedjet, Sabri I-685
Bóveda, C. II-491
Brandão, Tomás I-587
Bres, Stéphane II-776
Bruni, V. I-755
Burkhardt, Hans I-661, II-1
Busch, Andrew II-844
- Cabello, Enrique II-305, II-317
Cai, Xiang II-660
Camarena, Joan-Gerard I-359
Campaigne, Wesley R. I-41
Campilho, Aurélio C. I-248, II-513,
II-612, II-624
Caridade, Cristina M.R. II-814
Carter, John N. II-262
Casado, Gregorio de Miguel I-260
Castelán, Mario II-134
Castilho, Hugo Peres II-790
Castro, A. II-491
Cernadas, E. II-853
Cerri, Andrea II-410
Chai, Yanmei I-862
Chamizo, Juan Manuel García I-260
Chang, Chia-Hong II-122
Charles, J.J. I-283
Chekroun, Mickael II-386
Chen, Gao I-559
Chen, Huiying I-827
Chen, Jing-Fung I-535
Chen, Tse-Shih I-468
Cheng, Qiansheng II-81
Chianese, Angelo II-274
Cho, Hyungje II-193
Cho, Kyungeun II-193
Choi, Euncheol II-722
Choi, Soo-Mi II-602
Choi, Yoo-Joo II-602
Christo, C. II-225
Ciril, Igor II-386
Conde, Cristina II-305, II-317
Condell, J.V. I-780
Cooke, Eddie II-802
Costa, Jean-Pierre Da II-730
Cortez, Paulo I-898
Ćulibrk, Dubravko I-547
Cuminato, José Alberto I-126
Cunha, João Paulo Silva II-680
Cyganek, Boguslaw I-156
- Dambreville, Samuel I-173, I-886
Darbon, Jérôme II-386

- De Canditiis, D. I-755
 de Castro, Nuno I-816
 de Toro, Francisco I-196
 De Witte, Valérie I-381
 Debayle, Johan I-29
 Debure, Kelly I-648
 Del Bimbo, Alberto I-636
 Demir, Zafer II-35
 Demirkol, Askin II-35
 Denisov, Vitalij II-47
 Dhome, Michel II-250
 Dias, Jorge II-69
 Dias, José M. Bioucas II-700
 Díaz, Javier I-196
 Djibril, Mohamed Ould II-865
 Domínguez, R. II-853
 Dong, Xiao II-535
 Dornaika, Fadi II-205
 du Buf, J.M. Hans II-329
 Duits, Remco I-743, I-767
 Duque, Duarte I-898
- Ebrahimi, Mehran I-493
 Eglin, Véronique II-776
 El-Saadany, E.F. II-589
 Emptoz, Hubert II-776
 Emre, Erol II-35
- Fanany, Mohamad Ivan II-157
 Fang, H. I-206
 Fernandes, José Maria II-680
 Ferraz, Carolina Toledo I-126
 Ferreiro-Armán, Marcos II-730
 Ferrer, Alberto II-752
 Fieguth, Paul I-41, I-339
 Florack, L.M.J. I-743, I-767
 Fondón, Irene II-502
 Formella, A. II-853
 Formiga, Andrei de Araújo II-886
 Foroosh, Hassan I-94
 Furht, Borko I-547
- Gaceb, Djamel II-776
 Galmar, Eric I-236
 García-Pérez, David I-636
 Gavet, Yann I-29
 Gentric, Stéphane II-341
 Gerónimo, David II-205
 Giannarou, Stamatia I-184
 Giorgi, Daniela II-410
- Gonçalves, José II-742
 Gonçalves, Paulo J. Sequeira II-225
 González, C. I-850
 González, Jordi I-731
 Gregori, Valentín I-138, I-359
 Gribov, Alexander II-468
 Grigat, Rolf-Rainer II-479
 Groumpos, P.P. II-110
 Guan, Ling I-1
 Guerreiro, Rui F.C. I-721
- Ha, Yeong-Ho I-426
 Hadi, Youssef II-865
 Halawani, Alaa II-1
 Hale, Scott I-648
 Han, Bing II-102
 Han, Chan-Ho I-146, I-526
 Han, Dongfeng I-215
 Hancock, Edwin R. II-57, II-134, II-169,
 II-375
 Hensel, Marc II-479
 Homayouni, Saeid II-730
 Huang, Bing Quan II-897
 Huet, Benoît I-236
 Hwang, Wen-Jyi I-535
- Ibáñez, O. I-272
 Irianto, S.Y. I-673
 Ivaldi, William II-341
- Jang, Jong Whan I-226
 Jang, Soo-Wook I-526
 Janssen, Bart I-743, I-767
 Jeong, Jechang I-458, I-516
 Jeong, Kwang-Min I-426
 Jia, Jingping I-862
 Jiang, J. I-206, I-673
 Jodoin, Pierre-Marc I-370
 Julià, Carme I-804
 Jung, Gwang I-571
- Kalva, Hari I-547
 Kalviainen, H. II-877
 Kamarainen, J.-K. II-877
 Kamel, Mohamed II-398
 Kang, Moon Gi II-722
 Kanters, Frans I-743
 Katsaggelos, A.K. II-559
 Kechadi, M.-T. II-897
 Kelly, Philip II-802
 Kerre, Etienne E. I-381

- Kim, Daijin II-457
 Kim, Donghyung I-458, I-516
 Kim, Dong Kyue I-146
 Kim, Eun-Su I-526
 Kim, Gwang-Ha II-547
 Kim, Hansung II-237
 Kim, Jin-Soo I-709
 Kim, Jongho I-458, I-516
 Kim, Jong-Nam I-571
 Kim, Kwang-Baek II-547
 Kim, Min-Jeong II-524, II-602
 Kim, Myoung-Hee II-524, II-602
 Kim, Sang-Woon I-15
 Kim, Seungjong I-458
 Kim, Shin-Hyounge I-226
 Kim, Sungshin II-547
 Kitahara, Itaru II-237
 Kogure, Kiyoshi II-237
 Kohandani, Afsaneh II-398
 Konrad, Janusz I-370
 Korchagin, Danil I-329
 Krylov, Andrey I-329
 Kumazawa, Itsuo II-157
 Kuncheva, L.I. I-283
 Kutics, Andrea I-612
 Kwan, Chiman I-839
 Kwolek, Bogdan I-295
 Kwon, Ki-Ryong I-146
 Kwon, Seong-Geun I-146
- Lampasona, Constanza II-580
 Leal, Alberto II-680
 Ledda, Alessandro I-104, I-381
 Lee, Gwang-Soon I-526
 Lee, Jing-Huei II-660
 Lee, Joon-Jae I-426
 Lee, Suk-Hwan I-146
 Lensu, L. II-877
 Leow, Wee Kheng I-874
 Li, Baoxin I-839
 Li, Hao-Jie II-823
 Li, Wenhui I-215
 Li, Xiaokun I-839
 Li, Youfu I-827
 Li, Z. I-206
 Liang, Xuefeng II-286
 Lim, I.S. I-283
 Limin, Luo II-672
 Lin, Huei-Yung II-122
 Lin, Shou-Xun I-559, II-823
- Lins, Rafael Dueire II-886
 Liu, Danhua I-318
 Liu, Mei-Hwa I-535
 López, A. II-559
 López, Antonio I-804, II-205
 López, Fernando II-752
 Lu, Xiaosuo I-215
 Lumbreras, Felipe I-804
 Luong, Hiêp I-104
 Lyden, S. II-877
- Magliveras, Spyros S. I-547
 Mahdi, Walid I-414
 Marçal, André R.S. II-700, II-814
 Marín-Jiménez, Manuel J. II-13
 Marques, Jorge S. I-351, I-436
 Marques, Manuel I-436
 Marques, Oge I-547
 Martín-Herrero, Julio II-730
 Martín, J.M. II-559
 Martínez, Pedro I-731
 Marzal, Andrés I-624
 Mayer, Gregory S. I-82
 Mei, Gang I-839
 Mendonça, Ana Maria II-513, II-612
 Mignotte, Max I-370
 Milgram, Maurice II-341
 Milios, Evangelos I-697
 Miranda, António R. I-816
 Mohamed, S.S. II-589
 Mohebi, Azadeh I-339
 Molina, R. II-559
 Monteiro, Fernando C. I-248
 Mora, Higinio Mora I-260
 Moreno, Plinio II-181
 Morillas, Samuel I-138, I-359
 Morrow, P.J. I-780
 Moscato, Vincenzo II-274
 Mosquera, Antonio I-636
 Mota, Sonia I-196
 Musé, Pablo II-410
- Nakagawa, Akihiko I-612
 Nam, Tae-Jin I-792
 Nappi, Michele II-297, II-353
 Naudet-Collette, Sylvie II-250
 Neves, Alexandre I-436
 Nikiforidis, G.N. II-110
 Nixon, Mark S. II-262
 Nonato, Luis Gustavo I-126

- O'Connor, Noel II-802
 Odeh, Suhail M. II-648
 Oliveira, Alexandre Cesar Muniz de
 II-570
 Oommen, B. John I-15
 Ortiz, Francisco I-163
 Ortmann, Wolfgang II-434

 Paalanen, P. II-877
 Padilha, A.J. II-365
 Paiva, Anselmo Cardoso de II-570
 Palazón, Vicente I-624
 Palomares, Jose M. II-648
 Pan, Chunhong II-710
 Papageorgiou, E.I. II-110
 Pari, L. I-850
 Paris, A. II-225
 Park, Rae-Hong I-792
 Park, Seong Jun I-709
 Penedo, M.G. I-272
 Peng, Ning-Song I-394
 Peng, Zhigang II-660
 Penta, Antonio II-274
 Pereira, Carlos S. II-612
 Pereira, Fernando II-832
 Pérez de la Blanca, Nicolás II-13
 Peris-Fajarnés, Guillermo I-74, I-359
 Petrakis, Euripides G.M. I-697
 Philips, Wilfried I-104, I-381
 Picariello, Antonio II-274
 Pina, Pedro II-691
 Pinho, Pedro II-832
 Pinoli, Jean-Charles I-29
 Pinto, João Rogério Caldas II-90,
 II-225, II-790
 Post, Frits H. II-636
 Pralow, Thomas II-479
 Prats, José Manuel II-752
 Pu, Jie-Xin I-394

 Qi, Feihu I-116
 Qihua, Chen II-217
 Qiu, Huaijun II-375
 Qiu, K.J. I-673
 Queluz, Paula I-587

 Raducanu, Bogdan I-922
 Ragheb, Hossein II-169
 Rakotomamonjy, A. II-23
 Rathi, Yogesh I-173, I-886

 Raudys, Sarunas II-47
 Reisert, Marco I-661
 Rett, Jörg II-69
 Ricciardi, Stefano II-353
 Riccio, Daniel II-297
 Rodrigues, João II-329
 Rodríguez-Aragón, Licesio J. II-305,
 II-317
 Roig, Bernardino I-74
 Rojas, Ignacio II-648
 Roller, Dieter II-580
 Ros, Eduardo I-196, II-648
 Russell, Adam I-648
 Ryu, Taekyung I-571

 Sabatino, Gabriele II-353
 Saborido-Rey, F. II-853
 Sadovnikov, A. II-877
 Salama, M.M.A. II-589
 Salas, Joaquín I-731
 Sanches, João M. I-351, I-436
 Santos, Henrique I-898
 Santos, J. I-272
 Santos-Victor, José II-181
 Sapena, Almanzor I-138
 Sappa, Angel D. I-804, II-205
 Saraiva, José II-691
 Schulte, Stefan I-381
 Schutte, Sander II-636
 Scotney, B.W. I-780
 Sebastián, J.M. I-850
 Sebastião, R. II-365
 Sener, Sait II-445
 Seo, Kwang-Deok I-709
 Serafim, António Limas II-790
 Serrano, Carmen II-502
 Serrat, Joan I-804
 Shao, Wenze I-53
 Shi, Guangming I-318
 Shijie, Wang II-672
 Signes Pont, María Teresa I-260
 Silva, Aristófanés Corrêa II-570
 Silva, Jorge A. II-365
 Silva Jr., Valdeci Ribeiro da II-570
 Simonsz, Huib J. II-636
 Smeaton, Alan II-802
 Smolka, Bogdan I-307
 Socek, Daniel I-547
 Sohn, Kwanghoon II-237
 Sohng, Kyu-Ik I-526

- Song, Liang I-318
 Song, Samuel Moon-Ho I-709
 Song, Zongying II-710
 Sosa, Manuel II-502
 Sousa, António Verejão II-513
 Sousa, João M.C. II-90, II-225
 Spyridonos, P. II-110
 Stathaki, Tania I-184
 Stewman, John I-648
 Suard, F. II-23
 Suesse, Herbert II-434
 Sun, Xichen II-81
 Sung, Jaewon I-57
 Sur, Frédéric II-410
 Suvonvorn, Nikom II-422

 Tamimi, Hashem II-1
 Tannenbaum, Allen I-173, I-886
 ter Haar Romeny, B.M. I-767
 Thami, Rachid Oulad Haj II-865
 Tomassini, Federico II-410
 Tsang, Ing Jyh I-600
 Tsang, Ing Ren I-600

 Um, Kyhyun II-193
 Unel, Mustafa II-445

 Vaccari, Stefano I-910
 Valiente, José Miguel II-752
 van Zwieten, Joris E. II-636
 Vartiainen, J. II-877
 Veres, Galina V. II-262
 Vidal, Anna I-74
 Vieira, Susana M. II-90
 Vinhais, Carlos II-624
 Vitrià, Y. Jordi I-922
 Vitulano, D. I-755
 Voss, Klaus II-434
 Voutsakis, Epimenides I-697
 Vrscaj, Edward R. I-82, I-446, I-493

 Wang, Ching Wei I-404
 Wang, Hongfang II-57
 Wang, Qing I-862
 Wang, Ruixuan I-874
 Wang, Tianzhu I-215
 Wang, Yi I-215
 Wang, Zhaozhong I-116
 Wee, William II-660
 Wei, Zhihui I-53
 Weiping, Zhou II-672
 Wells, B. I-283
 Willekens, Ben II-636
 Wu, Shunjun II-102

 Xiang, Pan II-217
 Xiao, G. I-206, I-673
 Xie, Xuemei I-318
 Xu, Chao I-480

 Yang, Ching-Nung I-468
 Yang, Q. II-710
 Yang, Yongming I-480
 Yi, Sooyeong II-146
 Yim, Changhoon I-63
 Yoo, Kook-yeol I-507
 Youssef, A.M. II-589
 Yun, Hyun-Joo II-524
 Yun, Jae-Ho I-792

 Zavidovique, Bertrand II-422
 Zell, Andreas II-1
 Zhang, Hui II-286
 Zhang, Yong-Dong I-559, II-823
 Zhao, Rongchun I-862
 Zheng, Guoyan II-535
 Zhi, Liu II-217
 Zhou, Fugen I-116
 Ziou, Djemel I-685