Klaus Schoeffmann
Bernard Merialdo
Alexander G. Hauptmann
Chong-Wah Ngo
Yiannis Andreopoulos
Christian Breiteneder (Eds.)

LNCS 7131

Advances in Multimedia Modeling

18th International Conference, MMM 2012
Klagenfurt, Austria, January 2012
Proceedings

Springer

Lecture Notes in Computer Science 7131

Klaus Schoeffmann Bernard Merialdo
Alexander G. Hauptmann Chong-Wah Ngo
Yiannis Andreopoulos Christian Breiteneder (Eds.)

Advances in Multimedia Modeling

18th International Conference, MMM 2012
Klagenfurt, Austria, January 4-6, 2012
Proceedings

Springer

Volume Editors

Klaus Schoeffmann
Alpen-Adria-Universität, Klagenfurt, Austria
E-mail: ks@itec.aau.at

Bernard Merialdo
EURECOM, Sophia Antipolis, France
E-mail: bernard.merialdo@eurecom.fr

Alexander G. Hauptmann
Carnegie Mellon University, Pittsburgh, PA, USA
E-mail: alex@cs.cmu.edu

Chong-Wah Ngo
City University of Hong Kong, Kowloon, Hong Kong
E-mail: cwngo@cs.cityu.edu.hk

Yiannis Andreopoulos
University College London, UK
E-mail: iandreop@ee.ucl.ac.uk

Christian Breiteneder
Vienna University of Technology, Austria
E-mail: cb@ims.tuwien.ac.at

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-27354-4 e-ISBN 978-3-642-27355-1
DOI 10.1007/978-3-642-27355-1
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011943758

CR Subject Classification (1998): H.5.1, H.3-4, I.4-5, H.2.8

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

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

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The 18^{th} International Conference on Multimedia Modeling (MMM 2012) was held in Klagenfurt, Austria, January 4–6, 2012, and hosted by the Institute of Information Technology (ITEC), Alpen-Adria Universität Klagenfurt (Klagenfurt University). MMM is a leading international conference for researchers and industry practitioners to share their new ideas, original research results and practical development experiences from all multimedia-related areas.

It was a great honor for ITEC to host MMM 2012, one of the most longstanding multimedia conferences, in Klagenfurt, Austria. ITEC comprises two research groups with two full professors and currently about 30 scientific, technical and administrative staff. The institute's research focuses on distributed multimedia systems and multimedia communication. The conference venue was the Lindner Seepark Hotel Congress & Spa, which is directly located on the lovely Lake Wörthersee, one of the most famous tourism destinations in Austria. We hope that our venue made MMM 2012 a memorable experience for all participants.

MMM 2012 featured a comprehensive program including three keynote talks, ten oral presentation sessions, a poster session, a demo session, four special sessions, and the Video Browser Showdown, which was an interactive video search competition and a new event in the history of the MMM conference. The 142 submissions from authors of 32 countries included a large number of high-quality papers in multimedia content analysis, multimedia signal processing and communications, and multimedia applications and services. We thank our 167 Technical Program Committee members and reviewers who spent many hours reviewing papers and providing valuable feedback to the authors. From the 109 submissions to the main track and based on three (for some exceptions two) reviews per submission, the Program Chairs decided to accept 38 regular papers (34.9%) and 15 poster papers (13.8%). In total, 24 papers were received for four special sessions, with 12 being selected, and 9 submissions were received for a demo session, with 6 being selected. Video browsing systems of eight teams were selected for participation in the Video Browser Showdown. Authors of accepted papers came from 19 countries. This volume of the conference proceedings contains the abstracts of three invited talks and all the regular, poster, special session and demo papers, as well as special demo papers of the Video Browser Showdown. One regular paper was awarded with The Best Paper Award, which was sponsored by the FascinatE project (FascinatE – Format-Agnostic SCript-based INterAcTive Experience, FP7-248138).

The technical program is an important aspect but only provides its full impact if surrounded by challenging keynotes.

We are extremely pleased and grateful to have had three exceptional keynote speakers, Alan Hanjalic, John R. Smith, and Ralf Steinmetz, accept our invitation and present interesting ideas and insights at MMM 2012.

We are also heavily indebted to many individuals for their significant contribution. We thank the MMM Steering Committee for their invaluable input and guidance on crucial decisions. We wish to acknowledge and express our deepest appreciation to the Local Organizing Co-chair (and Finance Chair), Laszlo Böszörmenyi, the Special Session Chairs, Mathias Lux and Marco Bertini, the Demo Chairs, Cees Snoek and Frank Hopfgartner, the Publicity Chairs, Christian Timmerer and Tao Mei, the Publication Chairs, Werner Bailer and Hermann Hellwagner, US Liaisons, Oge Marques and Ketan Mayer-Patel, Asian Liaisons, Tat-Seng Chua and Tao Mei, European Liaisons, Cathal Gurrin and Harald Kosch, and – last but not least – the Local Arrangements team (and Webmaster), Martina Steinbacher, Margit Pertl, Mario Taschwer, and Rudolf Messner. Without their efforts and enthusiasm, MMM 2012 would not have become a reality. Moreover, we want to thank our sponsors: FascinatE project, Förderverein Technische Fakultät, Lakeside Labs, MediaEval. Finally, we wish to thank all committee members, reviewers, session chairs, student volunteers and supporters. Their contribution is much appreciated.

January 2012

Klaus Schoeffmann
Bernard Merialdo
Alexander Hauptmann
Chong-Wah Ngo
Yiannis Andreopoulos
Christian Breiteneder

Conference Organization

General Chairs

Klaus Schoeffmann	Klagenfurt University, Austria
Bernard Merialdo	Eurecom, Sophia-Antipolis, France
Alexander Hauptmann	Carnegie Mellon University, USA

Program Co-chairs

Chong-Wah Ngo	City University of Hong Kong
Yiannis Andreopoulos	UCL, London, UK
Christian Breiteneder	Vienna University of Technology, Austria

Local Organization Chair

Laszlo Böszörmenyi	Klagenfurt University, Austria

Special Session Co-chairs

Mathias Lux	Klagenfurt University, Austria
Marco Bertini	University of Florence, Italy

Demo Co-chairs

Cees Snoek	University of Amsterdam, The Netherlands
Frank Hopfgartner	University of California, Berkeley, USA

Publicity and Sponsorship Co-chairs

Tao Mei	Microsoft Research Asia, China
Christian Timmerer	Klagenfurt University, Austria

Publication Co-chairs

Werner Bailer	Joanneum Research, Austria
Hermann Hellwagner	Klagenfurt University, Austria

Finance Chair

Laszlo Böszörmenyi	Klagenfurt University, Austria

US Liaisons

Oge Marques	Florida Atlantic University, USA
Ketan Mayer-Patel	UNC-Chapel Hill, USA

Asian Liaisons

Tat-Seng Chua	National University of Singapore, Singapore
Tao Mei	Microsoft Research Asia, China

European Liaisons

Cathal Gurrin	Dublin City University, Ireland
Harald Kosch	University of Passau, Germany

Webmaster and Local Support Team

Martina Steinbacher	Klagenfurt University, Austria
Margit Pertl	Klagenfurt University, Austria
Mario Taschwer	Klagenfurt University, Austria
Rudolf Messner	Klagenfurt University, Austria

Steering Committee

Yi-Ping Phoebe Chen	La Trobe University, Australia
Tat-Seng Chua	National University of Singapore, Singapore
Tosiyasu L. Kunii	University of Tokyo, Japan
Wei-Ying Ma	Microsoft Research Asia, China
Nadia Magnenat-Thalmann	University of Geneva, Switzerland

Program Committee

Adrian Munteanu	Vrije University Brussel, Belgium
Ajay Divakaran	Sarnoff Corporation, USA
Akiyo Nadamoto	Konan University, Japan
Alan Smeaton	Dublin City University, Ireland
Alberto Messina	RAI Centre for Research and Technological Innovation, Italy
Alexander Loui	Kodak Research Laboratories, USA
Allan Hanbury	Vienna University of Technology, Austria
Andrea Cavallaro	Queen Mary University of London, UK
Andreas Henrich	University of Bamberg, Germany
Andreas Uhl	Salzburg University of Applied Sciences, Austria

Andrew Salway	Burton Bradstock Research Labs, UK
Benjamin Wah	University of Illinois, USA
Carlos Monzo	Universitat Oberta de Catalunya (UOC), Spain
Cathal Gurrin	Dublin City University, Ireland
Chabane Djeraba	University of Lille 1, France
Chia-Wen Lin	National Tsing Hua University, Taiwan
Chris Poppe	Ghent University, Belgium
Christian Beecks	RWTH Aachen University, Germany
Christian Timmerer	Klagenfurt University, Austria
Cuneyt Taskiran	Motorola Application Research Center, USA
Dalibor Mitrovic	Vienna University of Technology, Austria
Daniel Thalmann	EPFL, Switzerland
David Vallet	Universidad Autonoma de Madrid, Spain
Duy-Dinh Le	National Institute of Informatics, Japan
Enrique Costa-Montenegro	University of Vigo, Spain
Erik Mannens	Ghent University, Belgium
Ernesto Damiani	Università di Milano, Italy
Fabio Verdicchio	University of Aberdeen, UK
Felix Lee	Joanneum Research, Austria
Feng Wu	Microsoft Research Asia, China
Fernando Pereira	Instituto Superior Técnico, Portugal
Filippo Speranza	Communications Research Center, Canada
Florian Metze	Carnegie Mellon University, USA
Florian Stegmaier	University of Passau, Germany
Francesc Pinyol	Ramon Llull University (URL), Spain
Francesco De Natale	University of Trento, Italy
Francesco Robbiano	IMATI, Italy
Gene Cheung	National Institute of Informatics, Japan
Georg Thallinger	Joanneum Research, Austria
Georges Quenot	LIG/IMAG, France
Guojun Lu	Monash University, Australia
Guo-Jun Qi	University of Illinois at Urbana-Champaign, USA
Harald Kosch	University of Passau, Germany
Hari Sundaram	Arizona State University, USA
Harry Agius	Brunel University, UK
Hermann Hellwagner	Klagenfurt University, Austria
Hérve Jégou	INRIA, France
Hyowon Lee	Dublin City University, Ireland
Ichiro Die	Nagoya University, Japan
Ingo Kofler	Klagenfurt University, Austria
Isabel Trancoso	INESC ID, Portugal
Jean Martinet	University of Lille, France
Jean-Claude Moissinac	Telecom-ParisTech, France
Jenny Benois-Pineau	University Bordeaux 1, France
Jianping Fan	University of North Carolina, USA

Jiebo Luo	Kodak Research Laboratories, USA
Jinhui Tang	University of Science and Technology, China
Jinqiao Wang	Chinese Academy of Sciences, China
Jiro Katto	Waseda University, Japan
Joakim Söderberg	Ericsson Research, Sweden
Joao Magalhaes	Universidade Nova de Lisboa, Portugal
Joemon Jose	University of Glasgow, UK
Joo-Hwee Lim	Institute for Infocomm Research, Singapore
Jose Martinez	Universidad Autonoma de Madrid (UAM), Spain
Jun-Wei Hsieh	National Taiwan Ocean University, Taiwan
Keiji Yanai	University of Electro-Communications, Japan
Keith Mitchell	Lancaster University, UK
Kobus Barnard	University of Arizona, USA
Koichi Shinoda	Tokyo Institute of Technology, Japan
Konstantinos Chorianopoulos	Ionian University, Greece
Laszlo Böszörmenyi	Klagenfurt University, Austria
Laurent Amsaleg	CNRS-IRISA, France
Liang-Tien Chia	Nanyang Technological University, Singapore
Lin Yang	Google Inc., USA
Luca Celetto	ST Microelectronics, Italy
Luiz Fernando Gomes Soares	Catholic University of Rio de Janeiro, Brazil
Lyndon Kennedy	Yahoo! Research, USA
Lyndon Nixon	STI International, Austria
Maia Zaharieva	Vienna University of Technology, Austria
Manfred Del Fabro	Klagenfurt University, Austria
Marcel Worring	University of Amsterdam, The Netherlands
Marcin Detyniecki	Laboratoire dInformatique de Paris 6 LIP6, France
Marco Paleari	EURECOM, France
Mario Döller	University of Passau, Germany
Markus Koskela	Aalto University School of Science, Finland
Markus Schedl	Johannes Kepler University (JKU) Linz, Austria
Marta Mrak	BBC R&D, UK
Martha Larson	Delft University of Technology, The Netherlands
Martin Halvey	University of Glasgow, UK
Matthew Cooper	FX Palo Alto Laboratory, USA
Matthias Rauterberg	Eindhoven University of Technology, The Netherlands
Matthias Zeppelzauer	Vienna University of Technology, Austria
Mauro Barbieri	Philips Research, The Netherlands
Max Mühlhäuser	Technische Universität Darmstadt, Germany
Meng Wang	National University of Singapore, Singapore
Michael Granitzer	Know-Center, Austria

Michael Lew	Leiden University, The Netherlands
Michel Crucianu	CNAM, France
Milan Bjelica	University of Belgrade, Serbia
Mohamed Daoudi	Telecom Lille 1, France
Mohan Kankanhalli	National University of Singapore, Singapore
Mylene Farias	University of Brasilia, Brazil
Nadia Magnenat-Thalmann	University of Geneva (MIRALab), Switzerland
Naoko Nitta	Osaka University, Japan
Nicola Adami	Università degli Studi di Brescia, Italy
Noel O'Connor	Dublin City University, Ireland
Oge Marques	Florida Atlantic University (FAU), USA
Ognjen Arandjelovic	Trinity College Cambridge, UK
Pablo Cesar	CWI Amsterdam, The Netherlands
Paulo Villegas	Telefonica R&D, Spain
Phivos Mylonas	National Technical University of Athens, Greece
Pierangelo Migliorati	University of Brescia, Italy
Qi Tian	University of Texas at San Antonio, USA
Rahul Sukthankar	Intel Labs Pittsburgh, USA
Rainer Lienhart	University of Augsburg, Germany
Ralf Klamma	RWTH Aachen University, Germany
Raphael Troncy	EURECOM, France
Reede Ren	University of Glasgow, UK
Rene Kaiser	Joanneum Research, Austria
Richang Hong	HeFei University of Technology, China
Roger Zimmermann	National University of Singapore, Singapore
Rossana Damiano	Università di Torino, Italy
Selim Balcisoy	Sabanci University, Turkey
Shiguo Lian	France Telecom R&D Beijing, China
ShinIchi Satoh	National Institute of Informatics, Japan
Shingo Uchihashi	Carnegie Mellon University, USA
Shuicheng Yan	National University of Singapore, Singapore
Sid-Ahmed Berrani	Orange Labs France Telecom, France
Stefan Göbel	Technical University of Darmstadt, Germany
Stefano Bocconi	Vrije University Amsterdam, The Netherlands
Susanne Boll	University of Oldenburg, Germany
Suzanne Little	The Open University, UK
Tao Mei	Microsoft Research Asia, China
Tat-Seng Chua	National University of Singapore, Singapore
Tobias Bürger	Capgemini sd&m, Germany
Trista Chen	Gracenote, USA
Valerie Gouet-Brunet	CNAM, France
Vasileios Mezaris	Informatics and Telematics Institute / Centre for Research and Technology Hellas, Greece
Victor de Boer	Vrije University Amsterdam, The Netherlands
Vincent Charvillat	University of Toulouse, France

Vincent Oria	New Jersey Institute of Technology, USA
Vladimir Zlokolica	University of Novi Sad, Serbia
Wai-Tian Tan	Hewlett-Packard, USA
Wei-Guang Teng	National Cheng Kung University, Taiwan
Weisi Lin	Nanyang Technological University, Singapore
Wei-Tsang Ooi	National University of Singapore, Singapore
Wen-Hsiang Tsai	National Chiao Tung University, Taiwan
Werner Bailer	Joanneum Research, Austria
Wesley De Neve	Korea Advanced Institute of Science and Technology (KAIST), Korea
William Grosky	University of Michigan, USA
Winston Hsu	National Taiwan University, Taiwan
Wolfgang Effelsberg	University of Mannheim, Germany
Wolfgang Hürst	Utrecht University, The Netherlands
Xavier Anguera	Telefonica R&D, Spain
Xian-Sheng Hua	Microsoft Research Asia, China
Xiaoyi Jiang	University of Münster, Germany
Yannick Prie	University Claude Bernard Lyon 1, France
Yannis Avrithis	National Technical University of Athens, Greece
Yantao Zheng	Institute for Infocomm Research, Singapore
Yasuo Ariki	Kobe University, Japan
Yiannis Kompatsiaris	Informatics and Telematics Institute, Greece
Yi-Shin Chen	National Tsing Hua University, Taiwan
Yu-Gang Jiang	Columbia University, USA
Zheng-Jun Zha	National University of Singapore, Singapore
Zhiwei Li	Microsoft Research Asia, China
Zhongfei Zhang	State University of New York at Binghamton, USA
Zhu Liu	AT&T Laboratories, USA

External Reviewers

Carlos Eduardo Batista
Carsten Eickhoff
Jiashi Feng
Rudolf Granitzer
David Hauger
Stephan Kopf
Lu Liu
Vincenzo Lombardo
Marcio Moreno
Bingbing Ni
Zhongang Qi
Min-Hsuan Tsai
Raynor Vliegendhart
Kong Wah Wan
Fang Zheng

Sponsoring Institutions

FascinatE (EU FP7-248138)
Förderverein Technische Fakultät, Klagenfurt, Austria
Lakeside Labs, Klagenfurt, Austria
MediaEval

Table of Contents

Invited Talks

Regular Papers

Annotation I

Annotation II

Annotation and Interactive Multimedia Applications

Event and Activity I

Event and Activity II

Mining and Mobile Multimedia Applications

Search I

Search II

Summarization and Visualization

Visualization and Advanced Multimedia Systems

Poster Papers

Demo Session Papers

Video Browser Showdown

Special Session Papers

Interactive and Immersive Entertainment and Communication

Multimedia Preservation: How to Ensure Multimedia Access over Time?

Multi-modal and Cross-modal Search

Video Surveillance

A New Gap to Bridge: Where to Go Next in Social Media Retrieval?
(Extended Abstract)

Alan Hanjalic

Delft Multimedia Information Retrieval Lab
Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands
a.hanjalic@tudelft.nl

Abstract. Research in Multimedia Information Retrieval (MIR) aims at matching multimedia content and user needs and so at bringing image, audio and video content together with users. Users expectations regarding multimedia content access in terms of semantically rich and personalized relevance criteria have always been high and have imposed high demands on the level of sophistication of MIR solutions. The potential to develop MIR technology that meets such high demands has rapidly grown over the past twenty years by building on intensive international research effort. This growth accelerated, however, with the increasing contextualization of images, video and music in rapidly expanding social networks that link distributed content, diverse metadata and users of various profiles and interests. It is clear that user demands regarding the sophistication of MIR technology have further grown in the social network context in view of new ways of interacting with multimedia content and with other people via and about this content. However, this new context has also brought vast new opportunities for improving the quality of MIR solutions. These opportunities lie in synergetic integrations of multidisciplinary scientific contributions and rich information resources found there. Revisiting MIR from the viewpoint of the social network context, using the approaches that are often jointly referred to as social media retrieval, can help the field not only resolve the problems that impeded its development in the past, but also address the new emerging demands. I will show how contextualizing the MIR in online networked communities of users can help us achieve a fundamental shift in the MIR grand challenge, from bridging the research-oriented semantic gap to bridging the much more important, user-oriented utility gap, that explicitly addresses the overall usefulness of a MIR system output for the user. I will highlight some of the opportunities in pursuing this new, utility-oriented MIR grand challenge.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, p. 1, 2012.

Mining Multimedia Data for Meaning
(Extended Abstract)

John R. Smith

IBM T. J. Watson Research Center 19 Skyline Drive, Hawthorne, NY 10532
jsmith@us.ibm.com

Abstract. The explosion of images, video and multimedia is creating a valuable source for insights. It can tell us about things happening in the world, give clues about a persons preferences or experiences, indicate places of interest in a new town, and even capture a rolling log of our history. But, as a non-traditional source for data mining, there are numerous challenges to be overcome in order to handle the volume, velocity and variety of multimedia data in practice. In this talk, we review several application areas across Web, social media, mobile and safety/security and show how they benefit from mining of multimedia data. We review novel approaches for modeling semantics and automatically classifying visual contents and demonstrate examples in the context of IBM Multimedia Analysis and Retrieval System (IMARS).

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, p. 2, 2012.

Challenges in Serious Gaming as Emerging Multimedia Technology for Education, Training, Sports and Health
(Extended Abstract)

Ralf Steinmetz and Stefan Göbel

Multimedia Communications Lab (KOM) TU Darmstadt, FB 18, KOM,
Rundeturmstr. 10, 64283 Darmstadt, Germany
{Ralf.Steinmetz,Stefan.Goebel}@KOM.tu-darmstadt.de

Abstract. Digital computer games are very popular and successful, both as leisure activity and contemporary information and communication medium in the digital age, and as relevant economic factor and prospering market, not only in the creative industries. Games tackle a diversity of research aspects, e.g. Computer Graphics, AI, Storytelling, interfaces and sensors, authoring and production, usability and user experience or other ICT and multimedia technologies. Game technology and game techniques are broadly used by other application domains apart from pure entertainment as well. The rather new field of Serious Games, games with an additional purpose other than mere entertainment, offers a variety of new challenges and new fields of research. In our opinion, the term Serious Games comprises games for education (in terms of learning and practice), training, sports, and health. The core idea of Serious Games is to use the motivation inherited in games for other purposes like learning, sports, rehabilitation exercises, or even advertisement or opinion forming. Prominent examples in the field of Serious Games (games 'more than fun') are games for health, persuasive games, advergames or games for education and training, for instance in the form of multiplayer online games as tools to support collaborative learning settings. The combination of gaming technologies and gaming concepts with other research disciplines, technologies, methods and concepts results in a broad range of application do-mains. The resulting research areas are Authoring of Serious Games, Collaborative Learning using multiplayer Serious Games, Serious Games in Social Networks, and sensor technology for Serious Games for Sports & Health. In this talk, we will review the various aspects and application areas of Serious Games and point out some of the grand challenges in the field of Serious Gaming. Some of the core research topics of the Serious Games at the Technische Universitt Darmstadt and the httc will be reviewed. StoryTec, an authoring environment for the creation of Serious Games for non-programmers, will be illustrated, as well as 3D multiplayer Serious Games for collaborative learning and team (leader) training. Furthermore, Serious Games for sports & health, especially for fall prevention, rehabilitation, and management of obesity will be outlined.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, p. 3, 2012.

Building Semantic Hierarchies Faithful to Image Semantics

Hichem Bannour and Céline Hudelot

Applied Mathematics and Systems Department, Ecole Centrale Paris
92 295 CHÂTENAY-MALABRY, France
{Hichem.bannour,Celine.hudelot}@ecp.fr

Abstract. This paper proposes a new image-semantic measure, named "Semantico-Visual Relatedness of Concepts" (*SVRC*), to estimate the semantic similarity between concepts. The proposed measure incorporates visual, conceptual and contextual information to provide a measure which is more meaningful and more representative of image semantics. We also propose a new methodology to automatically build a semantic hierarchy suitable for the purpose of image annotation and/or classification. The building is based on the previously proposed measure *SVRC* and on a new heuristic, named *TRUST-ME*, to connect concepts with higher relatedness till the building of the final hierarchy. The built hierarchy explicitly encodes a general to specific concepts relationship and therefore provides a semantic structure to concepts which facilitates the semantic interpretation of images. Our experiments showed that the use of the constructed semantic hierarchies as a hierarchical classification framework provides a better image annotation.

1 Introduction

Achieving high level semantic interpretation of images is necessary to match user expectations in image retrieval systems. Effective tools are then required to allow a precise semantic description of images and allow at the same time a good interpretation of them. A wide number of approaches have been proposed for automatic image annotation, i.e. the textual description of images, to address the well-known *semantic gap* [23] problem. However in most of the proposed approaches the semantics is often limited to its perceptual manifestation, i.e. by the learning of high-level concepts from low-level features [3,14]. These approaches adequately describe the visual content of images but are unable to extract image semantics as humans can do. They are also faced with the scalability problem when dealing with broad content image databases [16]. The obtained performance varies significantly according to the concept number and the targeted data sets as well [13]. This variability may be explained by the huge intra-concept variability and wide inter-concept similarities on their visual properties that often lead to uncertain annotations and even contradictory. Thus, it is clear there is a lack of coincidence between the high-level semantic concepts and the low-level features, and that semantics is not always correlated

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 4–15, 2012.

with visual appearance. Therefore, the only use of machine learning seems to be insufficient to solve the problem of image annotation.

A new trend to overcome the aforementioned problems is to use semantic hierarchies [2]. Indeed, the use of explicit knowledge such as semantic hierarchies can help reduce, or even remove this uncertainty by supplying formal frameworks to argue about the coherence of extracted information from images. Semantic hierarchies have shown to be very useful to narrow the semantic gap [7]. Three types of hierarchies have been recently explored for image annotation and classification: 1) language-based hierarchies: based on textual information (ex. tags, surrounding context, WordNet, Wikipedia, etc.) [18,24,8], 2) visual hierarchies: based on low-level image features [22,4,26], 3) semantic hierarchies: based on both textual and visual features [15,9,25]. Although the two first approaches have received more attention, they showed a limited success in their general usage. Indeed, conceptual semantics is often not correlated with perceptual semantics, and is then insufficient to build a good hierarchy for image annotation. Whereas perceptual semantics cannot lead by itself to have a meaningful semantic hierarchy, as it is hard to interpret in higher levels of abstraction. Therefore, it seems mandatory to combine the both component of image semantics in order to build a semantic hierarchy faithful to image application purposes. The use of semantic hierarchies is then more convenient as they consider both, perceptual and conceptual semantics.

The rest of this paper is structured as follows: Section 2 reviews some related work. Section 3 introduces our proposal to build suitable semantic hierarchies for image annotation. Section 4 reports our experimental results on Pascal VOC dataset. The paper is concluded in Section 5.

2 Related Work

Several methods [15,9,18,24,22,4] have been proposed to build semantic hierarchies dedicated to image annotation. A semantic hierarchy classifier based on WordNet is proposed in [18]. Their hierarchy is built by extracting the relevant subgraph of WordNet that may link all concepts. ImageNet is proposed in [8], which is a large-scale ontology of images built upon the backbone of WordNet. LSCOM [19] aims to design a taxonomy with a coverage of around 1000 concepts for broadcast news video retrieval. An Ontology-enriched Semantic Space (OSS) was built in [24] to ensure globally consistent comparison of semantic similarities. The above approaches can be qualified as language-based hierarchies, as those hierarchies are built upon textual information. While these hierarchies are useful to provide a meaningful structure (organization) for concepts, they ignore visual information which is an important part of image semantics.

Other approaches are based on visual information [22,4,26]. An image parsing to text description (I2T) framework is proposed in [26], which generates text descriptions for images and videos. I2T is mainly based on an And-or Graph for visual knowledge representation. Sivic & al. propose to group visual objects using a multi-layer hierarchy tree that is based on common visual elements [22].

Bart & al. proposed a Bayesian method to organize a collection of images into a tree shaped hierarchy [4]. A method to automatically build classification taxonomy in order to increase classification rapidity is proposed in [12]. These hierarchies serve to provide a visual taxonomy, and a major problem with them is how they can be interpreted in higher levels of abstraction. Therefore, building meaningful semantic hierarchies should be done upon both semantic and visual information.

Among approaches for building semantic hierarchies, Li & al. [15] proposed a method based on visual features and tags to automatically build the "semantivisual" image hierarchy. A Semantic hierarchy based on contextual and visual similarity is proposed in [9]. Fan & al. [10] proposed an algorithm to integrate the visual similarity contexts between the images and the semantic similarity contexts between their tags for topic network generation. Flickr distance is proposed in [25], which is a novel measurement of the relationship between semantic concepts in visual domain. A visual concept network (VCNet) based on Flickr distance is also proposed [25]. Semantic hierarchies have great potential to improve image annotation, particularly through their explicit representation of concepts relationships that may help to understand image semantics.

2.1 Discussion

Many approaches for hierarchical image annotation use WordNet as a hierarchy of concepts [18,8]. However, WordNet is not very appropriate to model image semantics. Concepts organization in WordNet follows a psycholinguistic structure, which may be useful for reasoning about concepts and understand their meaning, but is limited and inefficient to reason about image context or its content. Indeed, distances between related concepts in WordNet do not necessarily reflect an appropriate semantic measure for reasoning about images, i.e. distances between concepts is not proportional to their semantic relatedness with respect to image domain. For example, according to the shortest path in WordNet the semantic relatedness of "shark" and "whale" is 11 (nodes), and of "man" and "whale" is 7. This is meant that concept "whale" is closer to "human" than to "shark". This is coherent from a biological point of view because "whale" and "human" are mammal while "shark" is not. However, in image domain it is more accurate to have higher similarity between "shark" and "whale" as they live in the same environment, share many visual features, and it is more common that they co-appear in a photo, unlike with humans. Then, an appropriate semantic hierarchy should represent this information or allow it to be deducted to help understand image semantics.

3 Building of the Hierarchy

Based on the previous discussion, we define the following assumptions underlying our approach: *A suitable semantic hierarchy for image annotation should: 1) model images context (as defined in the previous section), 2) allow grouping*

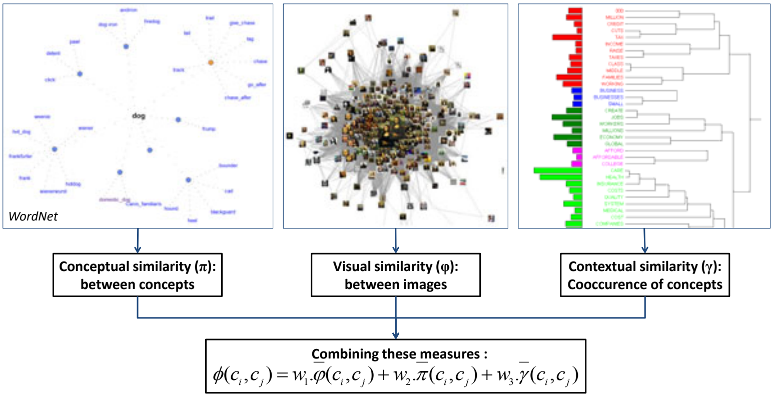

dog
WordNet
Conceptual similarity (π): between concepts
Visual similarity (φ): between images
Contextual similarity (γ): Cooccurence of concepts
Combining these measures :
$\phi(c_i,c_j)=w_1.\bar{\varphi}(c_i,c_j)+w_2.\bar{\pi}(c_i,c_j)+w_3.\bar{\gamma}(c_i,c_j)$

Fig. 1. The *SVRC* is based on visual, conceptual and contextual similarities

visually similar concepts in order to obtain better performance of classifiers, 3) reflect image semantics, i.e. the organization of concepts into the hierarchy and their semantic relatedness reflect image semantics.

Following the above assumptions, we propose in this paper a new method for building appropriate semantic hierarchies to images annotation. Our approach is based on a new measure to estimate the semantic relatedness between concepts, which is more faithful to image semantics since it is based on its different modalities. This measure, named *SVRC*, is based on 1) a visual similarity which represents the visual correspondence between concepts, 2) a conceptual similarity which defines a relatedness measure between target concepts, based on concepts definition in WordNet, and 3) a contextual similarity which measures the distributional similarity between each pair of concepts (cf. Fig.1). *SVRC* is then used in *TRUST-ME*, a set of heuristic rules that allow deciding the likelihood of the semantic relatedness between concepts, and help building the hierarchy.

Given a set of pairs image/annotation, where each annotation describes a set of concepts associated with an image, our approach allows to automatically build a semantic hierarchy suitable for image annotation. Formally, we consider $I =< i_1, i_2, \cdots, i_{\mathcal{L}} >$ all images of a considered database, and $C =< c_1, c_2, \cdots, c_{\mathcal{N}} >$ the annotation vocabulary of these images, i.e. the set of concepts associated with these images. The approach we propose consists in identifying $\mathcal{M}$ new concepts that link all the concepts of C in a hierarchical structure that best represents image semantics.

3.1 Visual Similarity

Let x_i^v be any visual representation of an image i (a visual features vector), we learn for each concept c_j a classifier that can associate this concept with its visual features. For this, we use $\mathcal{N}$ binary Support Vector Machines (SVM) [6] (one-versus-all) with a decision function $\mathcal{G}(x^v)$:

$$\mathcal{G}(x^v) = \sum_k \alpha_k y_k \mathbf{K}(x_k^v, x^v) + b \tag{1}$$

where $\mathbf{K}(x_i^v, x^v)$ is the value of a kernel function for the training sample x_i^v and the test sample x^v, $y_i \in \{1, -1\}$ the class label of x_i^v, α_i the learned weight of the training sample x_i^v, and b is a learned threshold parameter. Notice that the training samples x_i^v with weight $\alpha_i > 0$ are the *support vectors*.

After several tests on the training sample, we decided to use a radial basis function kernel:

$$\mathbf{K}(x, y) = exp\Big(\frac{\|x - y\|^2}{\sigma^2}\Big) \tag{2}$$

Now, given these $\mathcal{N}$ trained SVMs where inputs are images visual features and outputs are concepts (image classes), we want to define a centroid $\vartheta(c_i)$ for each concept class c_i that best represent it. These centroids should then minimize the sum of squares within each set S_i:

$$\underset{S}{\operatorname{argmin}} \sum_{i=1}^{\mathcal{N}} \sum_{x_j^v \in S_i} \|x_j^v - \mu_i\|^2 \tag{3}$$

where S_i is the set of *support vectors* of class c_i, $S = \{S_1, S_2, \cdots, S_{\mathcal{N}}\}$, and μ_i is the mean of points in S_i.

The objective being to estimate a distance between these classes in order to assess their visual similarities, we compute the centroid $\vartheta(c_i)$ of each visual concept c_i using:

$$\vartheta(c_i) = \frac{1}{|S_i|} \sum_{x_j \in S_i} x_j^v \tag{4}$$

The visual similarity between two concepts c_i and c_j, is then inversely proportional to the distance between their visual features $\vartheta(c_i)$ and $\vartheta(c_j)$:

$$\varphi(c_i, c_j) = \frac{1}{1 + d(\vartheta(c_i), \vartheta(c_j))} \tag{5}$$

where $d(\vartheta(c_i), \vartheta(c_j))$ is the Euclidean distance between $\vartheta(c_i)$ and $\vartheta(c_j)$.

3.2 Conceptual Similarity

Conceptual similarity reflects the semantic relatedness between two concepts from a linguistic and a taxonomic point of view. Several conceptual similarity measures have been proposed [5,21,1]. Most of them are based on a lexical resource, such as WordNet [11]. A first family of approaches is based on the structure of this external resource (often used as a semantic network or a directed graph), and the similarity between concepts is computed according to the distances of the paths connecting them in this structure [5]. However, as aforementioned, the structure of these resources does not necessarily reflect image semantics, and therefore such measures does not seem suited to our problem. An alternative approach to measure the semantic relatedness between concepts is to use their provided definition. In the WordNet case, these definitions are known as the glosses and are provided by the synsets associated to each concept. For example, Banerjee and Pedersen [1] proposed a measure of semantic relatedness between concepts that is based on the number of shared words (overlaps) in their definitions (glosses).

In this work we used the gloss vector relatedness measure proposed by [20], in which they suggest to exploit "second order" co-occurrence vector of glosses rather than matching words that co-occur in it. Specifically, in a first step a word space of size $\mathcal{P}$ is built by taking all the significant words used to define all synsets of WordNet. Thereby, each concept c_i is represented by a context vector $\overrightarrow{w}_{c_i}$ of size $\mathcal{P}$, where each n^{th} element of this vector represents the number of occurrences of n^{th} word in the word space in the gloss of c_i. The semantic relatedness of two concept c_i and c_j is therefore measured using the cosine similarity between $\overrightarrow{w}_{c_i}$ and $\overrightarrow{w}_{c_j}$:

$$\eta(c_i, c_j) = \frac{\overrightarrow{w}_{c_i} \cdot \overrightarrow{w}_{c_j}}{|\overrightarrow{w}_{c_i}||\overrightarrow{w}_{c_j}|} \tag{6}$$

Some concepts definitions in WordNet are very concise and thus make the measure unreliable. Consequently, [20] proposed extending the glosses of concepts with the glosses of adjacent concepts (located in their immediate neighborhood). Hence, for each concept c_i the set Ψ_{c_i} is defined as all the adjacent glosses connected to c_i (Ψ_{c_i}={gloss(c_i), gloss(hyponyms(c_i)), gloss(meronyms(c_i)), etc.}). Then each element x (gloss) of Ψ_{c_i} is represented by $\overrightarrow{w}_x$ as explained above. The similarity measure between two concepts c_i and c_j is then defined as the sum of the individual cosines of the corresponding gloss vectors:

$$\theta(c_i, c_j) = \frac{1}{|\Psi_{c_i}|} \sum_{x \in \Psi_{c_i}, y \in \Psi_{c_j}} \frac{\overrightarrow{w}_x \cdot \overrightarrow{w}_y}{|\overrightarrow{w}_x||\overrightarrow{w}_y|} \ , \quad where |\Psi_{c_i}| = |\Psi_{c_j}|. \tag{7}$$

Finally, each concept in WordNet may match several senses (synsets) that differ from each other in their position in the hierarchy and their definition. A disambiguation step is then necessary to identify the good synset. For example, the similarity between "Mouse" (Animal) and "Keyboard" (device) differs widely from the one of "Mouse" (device) and "Keyboard" (device). Therefore, we first compute the conceptual similarity between the different senses (synset) of c_i and c_j. The maximum value of similarity is then used to identify the most likely meaning of these two concepts, i.e. disambiguate c_i and c_j. Thus, the conceptual similarity is calculated as following:

$$\pi(c_i, c_j) = \underset{\delta_i \in s(c_i), \delta_j \in s(c_j)}{\text{argmax}} \ \theta(\delta_i, \delta_j) \tag{8}$$

where $s(c_x)$ is "all synsets that can be associated to the meanings of c_x".

3.3 Contextual Similarity

It is intuitively clear that if two concepts are similar or related, it is likely that their role in the world will be similar, and thus their context of occurrence will be equivalent (i.e. they tend to occur in similar contexts, for some definition of context). The information related to the context of appearance of concepts, called contextual, is used to connect concepts that often appear together in images although semantically distant from the taxonomic point of view. Moreover, this contextual information can also help to infer higher-level knowledge from images. For example, if a photo contains "Sea" and "Sand", it is likely that the scene depicted in this photo is the one of beach. It is therefore important to measure the contextual similarity between concepts. However, unlike the visual and the conceptual similarity, this one is a "corpus-dependent" measure, and more precisely depends on the distribution of concepts in the corpus.

In our approach, we define the contextual similarity between two concepts c_i and c_j as the Pointwise Mutual Information (PMI) $\rho(c_i, c_j)$:

$$\rho(c_i, c_j) = \log \frac{P(c_i, c_j)}{P(c_i)P(c_j)} \tag{9}$$

where: $P(c_i)$ is the probability of occurrence of c_i, and $P(c_i, c_j)$ is the joint probability of c_i and c_j. These probabilities are estimated by computing the frequency of occurrence and cooccurrence of concepts c_i and c_j in the database.

Given $\mathcal{N}$ the total number of concepts in the database, $\mathcal{L}$ the total number of images, n_i the number of images annotated by c_i (occurrence frequency of c_i) and n_{ij} the number of images co-annotated by c_i et c_j, the above probabilities can be estimated by: $\widehat{P(c_i)} = \frac{n_i}{\mathcal{L}}$, $\widehat{P(c_i, c_j)} = \frac{n_{ij}}{\mathcal{L}}$.

$$\Rightarrow \rho(c_i, c_j) = \log \frac{\mathcal{L} * n_{ij}}{n_i * n_j} \tag{10}$$

$\rho(c_i, c_j)$ quantifies the amount of information shared between the two concepts c_i and c_j. Thus, if c_i and c_j are independent concepts, then $P(c_i, c_j) = P(c_i) \cdot P(c_j)$ and therefore $\rho(c_i, c_j) = log\ 1 = 0$. $\rho(c_i, c_j)$ can be negative if si c_i et c_j are negatively correlated. Otherwise $\rho(c_i, c_j) > 0$ and quantifies the degree of dependence between these two concepts. In this work, we only want to measure the positive dependence between concepts and therefore we set negative values of $\rho(c_i, c_j)$ to 0. Finally, to normalize the contextual similarity between two concepts c_i and c_j into [0,1], we compute it in our approach by:

$$\gamma(c_i, c_j) = \frac{\rho(c_i, c_j)}{-\log[\max(P(c_i), P(c_j))]} \tag{11}$$

3.4 Semantico-Visual Relatedness of Concepts ($SVRC$)

For two given concepts c_i and c_j, their similarity measures: visual $\varphi(c_i, c_j)$, conceptual $\pi(c_i, c_j)$ and contextual $\gamma(c_i, c_j)$ are first normalized into the same interval using the Min-Max Normalization. Then, the Semantico-Visual Relatedness $\phi(c_i, c_j)$ of these concepts c_i and c_j is defined as:

$$\phi(c_i, c_j) = \omega_1 \cdot \overline{\varphi}(c_i, c_j) + \omega_2 \cdot \overline{\pi}(c_i, c_j) + \omega_3 \cdot \overline{\gamma}(c_i, c_j)\ ,\ \sum_{i=1}^{3} \omega_i = 1 \tag{12}$$

The choice of weights ω_i is very important. According to the target application, some would prefer to build a domain-specific hierarchy (that best represents a specific-domain or corpus), and can therefore assign a higher weight to the contextual similarity ($\omega_3 \nearrow$). Others would be conducted to build a generic hierarchy, and will therefore assign a higher weight to the conceptual similarity ($\omega_2 \nearrow$). However if the purpose of the hierarchy is rather to build a hierarchical framework to image classification, it may be advantageous to assign a higher weight to the visual similarity ($\omega_1 \nearrow$).

3.5 Heuristic Rules for Hierarchy Building

Once we have estimated the semantic relatedness between each pair of concepts, it is important to regroup them in a more comprehensive hierarchy despite the uncertainty introduced by semantic similarity measurements. In the following we propose a heuristic named *TRUST-ME*, that allows to infer Hypernym relationships between concepts, and to bring together these various concepts in a hierarchical structure.

Let us define the following functions to understand the reasoning rules we used for the building of our hierarchy:

- $Closest(c_i)$ returns the closest concept to c_i according to the $SVRC$ measure:

$$Closest(c_i) = \operatorname*{argmax}_{c_k \in \mathcal{C} \setminus \{c_i\}} \phi(c_i, c_k) \tag{13}$$

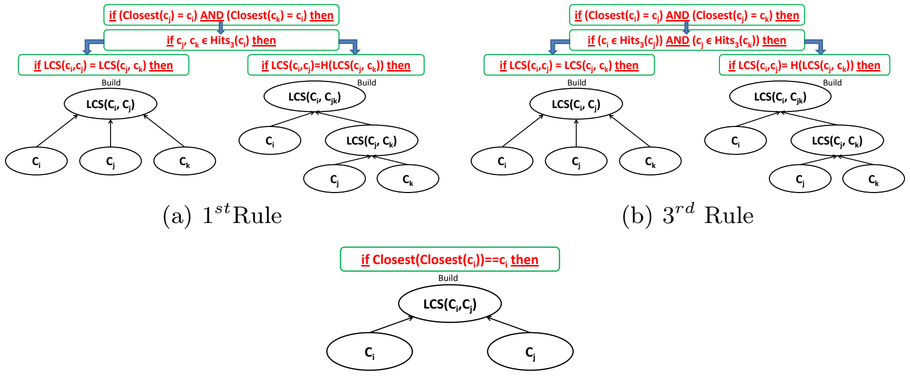
- $LCS(c_i, c_j)$ allows to find the *Least Common Subsumer* of c_i and c_j in WordNet:

$$LCS(c_i, c_j) = \operatorname*{argmin}_{c_l \in \{H(c_i) \cap H(c_j)\}} len(c_l, root) \tag{14}$$

  where $H(c_i)$ allows to find all of hypernyms of c_i in WordNet, *root* is the root node of WordNet and $len(c_x, root)$ returns the length of the shortest path in WordNet between c_x and *root*.
- $Hits_3(c_i)$ returns the 3 closest concepts to c_i within the meaning of $Closest(c_i)$.

Basically *TRUST-ME* consists of three rules which are based on the $SVRC$ measure and on reasoning about the Least Common Subsumer (LCS) to select concepts to be connected to each other. These rules are illustrated and executed in the order described in Fig.2. First rule checks whether a concept c_i is classified as the closest relative to more than one concept $((Closest(c_j) = c_i), \forall j \in \{1, 2, \cdots\})$. If so and if these concepts $\{c_j\}$ are reciprocal in $Hits_3(c_i)$, then according to their LCS they will be connected either directly to their LCS or in a tow level structure as illustrated in Fig.2(a). In the second, if $(Closest(c_i) = c_j)$ and $(Closest(c_j) = c_i)$ (can also be written as $Closest(Closest(c_i)) = c_i$) then c_i and c_j are actually related and are connected to their LCS. The third rule covers the case when $(Closest(c_i) = c_j)$ and $(Closest(c_j) = c_k)$ - cf. Fig.2(b).

The building of the hierarchy is bottom-up (starts from leaf concepts) and uses an iterative algorithm until it reaches the root node. Given a set of tags


if (Closest(c_j) = c_i) AND (Closest(c_k) = c_i) then
if c_j, c_k ϵ Hits_3(c_i) then
if LCS(c_i,c_j) = LCS(c_j, c_k) then
if LCS(c_i,c_j)=H(LCS(c_j, c_k)) then
Build
LCS(C_i, C_j)
C_i
C_j
C_k
Build
LCS(C_i, C_jk)
C_i
LCS(C_j, C_k)
C_j
C_k
if (Closest(c_i) = c_j) AND (Closest(c_j) = c_k) then
if (c_i ϵ Hits_3(c_j)) AND (c_j ϵ Hits_3(c_k)) then
if LCS(c_i,c_j) = LCS(c_j, c_k) then
if LCS(c_i,c_j)= H(LCS(c_j, c_k)) then
Build
LCS(C_i, C_j)
C_i
C_j
C_k
Build
LCS(C_i, C_jk)
C_i
LCS(C_j, C_k)
C_j
C_k
if Closest(Closest(c_i))==c_i then
Build
LCS(C_i,C_j)
C_i
C_j

(a) 1^{st}Rule

(b) 3^{rd} Rule

(c) 2^{nd} Rule

Fig. 2. Rules in *TRUST-Me* allowing to infer the relationship between the different concepts. Preconditions (in red) and actions (in black).

associated with images in a dataset, our method compute the *SVRC* $\phi(c_i, c_j)$ between all pairs of concepts, then links most related concepts to each other while respecting the defined rules in *TRUST-ME*. Thus, we obtain a new set of concepts in a higher level resulted by the linked concepts in the lower level. We iterate the process until all concepts are linked to a root node. Fig.3 illustrates the built hierarchy on Pascal VOC dataset.

4 Experimental Result

As part of this work, we evaluate our semantic hierarchy by comparing the performance of a flat image classification versus a hierarchical based one. Pascal VOC'2010 dataset (11 321 images, 20 concepts) is used for building the hierarchy and evaluating the classification.

4.1 Visual Representation of Images

To compute the visual similarity of concepts, we used in our approach the Bag-of-Features (BoF) model, also known as bag-of-visual words. The used BoF model is built as following: feature detection using Lowe's DoG Detector [17], feature description using SIFT descriptor [17] and codebook generation. The generated codebook is a set of features assumed to be representative of all images features. Given the collection of detected patches from the training images of all categories, we generate a codebook of size $D = 1000$ by performing k-means algorithm. Thus, each patch in an image is mapped to the most similar visual word in the codebook through a KD-Tree. Each image is then represented by a histogram of D visual words, where each bin in the histogram correspond to the occurrence number of a visual word in that image.

4.2 Weighting

As this paper aims to build a hierarchy suitable for image classification/annotation, we set the weighting factors in an experimental way as follows: $\omega_1 = 0.4, \omega_2 = 0.3$, and $\omega_3 = 0.3$. Our experimentations on the impact of weights (ω_i) showed also that the visual similarity is more representative of concepts similarity, as it will be illustrated with the produced hierarchies in Fig.3.

4.3 Evaluation

To evaluate our approach, we used 50% of VOC images for learning concepts and the others for testing. Each image may belong to one or more of the 20 existing classes. For the flat classification we used $\mathcal{N}$ SVM one-against-all, where the inputs are the BoF images representations and outputs are the desired SVM responses for each image (1 or -1) - for details cf. Section 3.1. However Pascal VOC dataset is unbalanced, i.e. many concepts are represented by few hundred of images among the 11321 images in the database (much more negative data than the positive ones for many concepts). To overcome this problem we used cross-validation, taking at each fold as many positive as negative images. Hierarchical classification is made by training a set of $(\mathcal{N}+\mathcal{M})$ hierarchical classifiers consistent with the structure of the hierarchy in Fig.3. $\mathcal{M}$ is the number of new concepts created during the building of the hierarchy. For training the classifier of each concept in the hierarchy, we took all images of son nodes (of a given

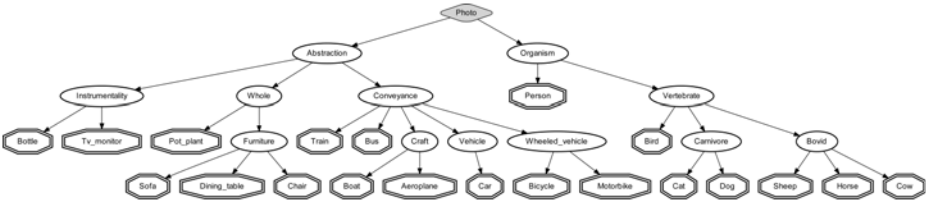


Fig. 3. The semantic hierarchy built on Pascal VOC'2010 dataset. Double octagon nodes are original concepts, and the diamond one is the root of the produced hierarchy.

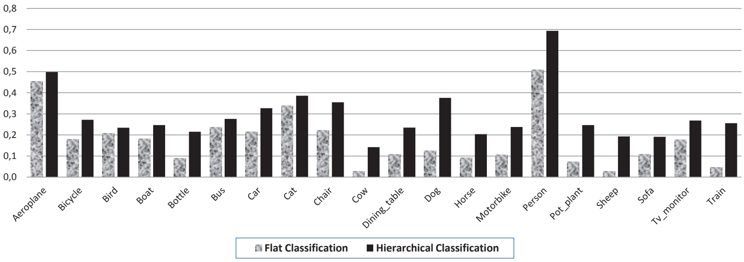
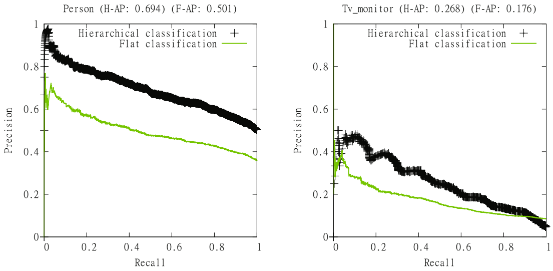


Fig. 4. Average precision of flat and hierarchical classification on Pascal VOC concepts



(a) Concept Person (b) Concept Tv_monitor

Fig. 5. Precision/recall curves for hierarchical and flat classification on concepts "Person" and "TV_Monitor"

concept) as positive and all images of son nodes of its immediate ancestor as negative. For example, to train a classifier for "Carnivore" all images of "Dog" and "Cat" are taken as positive while images of "Bird", "Sheep", "Horse" and "Cow" as negative. Thus, each classifier is trained to distinguish one class from others in the same category. For testing the hierarchical classification, a given image can take one (or more) path in the hierarchy based on classifiers responses, and starting from the root node until reaching a sheet node. Results are evaluated with the recall/precision curves and the average precision score.

Fig.4 compares the performance of our semantic hierarchic classifier with the performance of a flat classification. Our approach performs a better classification than the flat one, with a mean improvement of +8.4%. Using half of the training images from the VOC challenge (we have used the validation set for testing) and including the images marked as difficult, hierarchical classification achieves an average precision of 28.2% when the flat one achieves 19.8%. Fig.5 shows the recall/precision curves for concepts "Person" and "Tv_Monitor" using hierarchical and flat classification. This comparison shows that hierarchical classification has the best performance at all levels of recall.

5 Conclusion

This paper proposes a new approach to automatically build a suitable semantic hierarchy for image annotation. Our approach is based on a new measure of semantic relatedness, called *SVRC*, that takes into account the visual similarity, the conceptual and the contextual ones. *SVRC* allows estimating the semantico-visual relatedness of concepts. A new heuristic, *TRUST-ME*, is also proposed for reasoning about concepts relatedness, and to link together concepts that are semantically related in a semantic hierarchy. Our experiments showed that the built semantic hierarchy improves significantly the classification performance on Pascal VOC dataset. Our future research will concern the evaluation of our approach on larger datasets (MirFlicker and ImageNet), and the assessment of our hierarchy in terms of structure and contribution of knowledge.

References

1. Banerjee, S., Pedersen, T.: Extended gloss overlaps as a measure of semantic relatedness. In: IJCAI (2003)
2. Bannour, H., Hudelot, C.: Towards ontologies for image interpretation and annotation. In: CBMI (2011)
3. Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D.M., Jordan, M.I.: Matching words and pictures. JMLR 3, 1107–1135 (2003)
4. Bart, E., Porteous, I., Perona, P., Welling, M.: Unsupervised learning of visual taxonomies. In: CVPR (2008)
5. Budanitsky, A., Hirst, G.: Evaluating wordnet-based measures of lexical semantic relatedness. Comput. Linguist. 32, 13–47 (2006)
6. Cortes, C., Vapnik, V.: Support-vector networks. Machine Learning, 20 (1995)
7. Deng, J., Berg, A.C., Li, K., Fei-Fei, L.: What Does Classifying More Than 10,000 Image Categories Tell Us? In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6315, pp. 71–84. Springer, Heidelberg (2010)
8. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009)
9. Fan, J., Gao, Y., Luo, H.: Hierarchical classification for automatic image annotation. In: SIGIR (2007)
10. Fan, J., Luo, H., Shen, Y., Yang, C.: Integrating visual and semantic contexts for topic network generation and word sense disambiguation. In: CIVR (2009)
11. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press (1998)
12. Griffin, G., Perona, P.: Learning and using taxonomies for fast visual categorization. In: CVPR (2008)
13. Hauptmann, A., Yan, R., Lin, W.-H.: How many high-level concepts will fill the semantic gap in news video retrieval? In: CIVR (2007)
14. Lavrenko, V., Manmatha, R., Jeon, J.: A model for learning the semantics of pictures. In: NIPS. MIT Press (2003)
15. Li, L.-J., Wang, C., Lim, Y., Blei, D., Fei-Fei, L.: Building and using a semantivisual image hierarchy. In: CVPR (2010)
16. Liu, Y., Zhang, D., Lu, G., Ma, W.-Y.: A survey of content-based image retrieval with high-level semantics. Pattern Recognition 40(1), 262–282 (2007)
17. Lowe, D.G.: Object recognition from local scale-invariant features. In: ICCV (1999)
18. Marszalek, M., Schmid, C.: Semantic hierarchies for visual object recognition. In: CVPR (2007)
19. Naphade, M., Smith, J.R., Tesic, J., Hsu, W., Kennedy, L., Hauptmann, A., Curtis, J.: Large-scale concept ontology for multimedia. IEEE MultiMedia (2006)
20. Patwardhan, S., Pedersen, T.: Using wordnet-based context vectors to estimate the semantic relatedness of concepts. In: EACL (2006)
21. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: IJCAI (1995)
22. Sivic, J., Russell, B.C., Zisserman, A., Freeman, W.T., Efros, A.A.: Unsupervised discovery of visual object class hierarchies. In: CVPR (2008)
23. Smeulders, A.W.M., Member, S., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE PAMI, 22 (2000)
24. Wei, X.-Y., Ngo, C.-W.: Ontology-enriched semantic space for video search. In: MULTIMEDIA, pp. 981–990 (2007)
25. Wu, L., Hua, X.-S., Yu, N., Ma, W.-Y., Li, S.: Flickr distance. In: MM (2008)
26. Yao, B., Yang, X., Lin, L., Lee, M.W., Zhu, S.C.: I2t: Image parsing to text description. Proceedings of IEEE (2009)

Combining Image-Level and Segment-Level Models for Automatic Annotation

Daniel Kuettel, Matthieu Guillaumin, and Vittorio Ferrari

Computer Vision Laboratory, ETH Zurich, Switzerland

Abstract. For the task of assigning labels to an image to summarize its contents, many early attempts use segment-level information and try to determine which parts of the images correspond to which labels. Best performing methods use global image similarity and nearest neighbor techniques to transfer labels from training images to test images. However, global methods cannot localize the labels in the images, unlike segment-level methods. Also, they cannot take advantage of training images that are only locally similar to a test image. We propose several ways to combine recent image-level and segment-level techniques to predict both image and segment labels jointly. We cast our experimental study in an unified framework for both image-level and segment-level annotation tasks. On three challenging datasets, our joint prediction of image and segment labels outperforms either prediction alone on both tasks. This confirms that the two levels offer complementary information.

Keywords: image auto-annotation, image region labelling, keyword-based image retrieval.

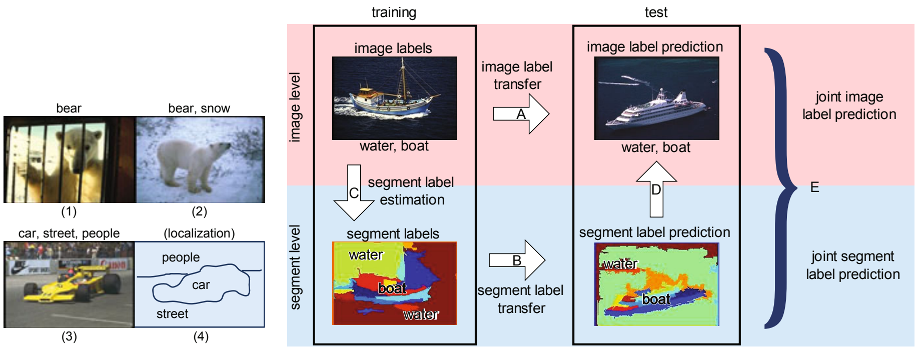
1 Introduction

In recent years, automatic image annotation has received increasing attention [11,13,17,18]. In its basic version, which we call *image-level annotation*, the task is to assign a few semantic labels to a test image, roughly describing its contents (fig. 1(a)). In its elaborate version, which we call *segment-level annotation*, the semantic labels are assigned to every segment in the image (fig. 1(a)d). The union over the segment labels is then proposed as image labels [2,4,7].

Segment-level annotation poses additional challenges compared to image-level annotation. First, labels for the segments in the training images are not given, and must be estimated from the image labels. As a consequence, segment-levels methods need to be robust to errors in this estimation. Second, appearance features extracted from segments are less distinctive than global image features, which incorporate contextual layout information. Finally, even with perfect segment labels, their union does not always match user-provided image labels, since the latter focus on the salient objects in the image. Overall, segment-level annotation is a much more difficult task, which explains why recent global methods outperform local ones for image-level annotation.

On the other hand, global methods cannot *localize* labels in the test images, but merely indicate their presence (fig. 1(a)3). This limits the interpretability of the different methods and reduces the spectrum of possible applications of the output predictions:

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 16–28, 2012.



(a) (b)

Fig. 1. Left (a): A test image (1) of a bear out of its typical context in the wild (2), highlighting the need for compositionality. On the other hand, context is a powerful force for recognizing cars in typical images such as (3). (4) shows a localization of the labels in (3). Right (b): **Summary of image annotation models.** For each arrow there are several applicable models. Alternatives are discussed in the respective sections. For E and F, we present novel methods to combine segment and image-level models.

image labels are restricted to classification and indexing purposes. With localized labels instead, it is possible to visualize the learned concepts and identify their spatial extent in the images. Therefore, segment labels can be used to train object detectors or compute class-specific features invariant to position and scale. Overall, they provide a deeper understanding of an image.

Our work builds on the observation that image-level and segment-level techniques have several complementary strengths. Segment-level methods explicitly attempt to determine which parts of the training images belong to each label. This is typically done by describing the local appearance of segments and then searching for recurrences over the training set with a probabilistic model [2,3,5,9,19]. Segment-level methods can recognize the presence of a class in a test image even if it appears in a context not observed during training (*e.g.* a bear in a cage while training images show bears in the wild, fig. 1(a)1+2). This *compositional* character is a strength of segment-level methods and endows them with great generalization potential. On the other hand, the global image layout is more characteristic than the appearance of individual segments, as it indicates certain combinations of labels (cars-roads in fig. 1(a)3). Recent image-level methods [17,25] employ global image similarities and predict labels for a test image based on the labels of its most similar training images. Those methods perform better on the image-level annotation task [1,11], as they better exploit the large number of available images annotated by keywords.

The observations above suggest that segment-level prediction is a task of its own, which should be evaluated on a per-pixel basis, and that combining segment-level and image-level predictions may help both tasks. The potential for interaction between the two levels is largely unexplored and very promising. Image labels help reduce the space of possible segment-level annotations. On the other hand, even imperfect segment labels carry valuable complementary information about image content.

In this paper we explore the combination of image and segment levels and make the following contributions: (i) we present a unified view of existing methods as processing stages in a generic scheme (sec. 2); (ii) we propose new alternative models to perform many of the stages (sec. 3 to 6); (iii) we propose novel joint models to combine the predictions from image and segment levels (sec. 7). In sec. 8 we present the datasets and features we used. Through extensive experiments, we demonstrate that our combined models perform better at both segment-level and image-level annotation than either component alone (sec. 9). We conclude and draw directions for future research in sec. 10.

Related works. Our work relates to the numerous segment-level and image-level methods discussed above, as we seek to combine the two strands.

Some earlier works tried to incorporate context in segment-level methods, *e.g.* by modeling co-occurence of labels [6] or their spatial relationships [23]. However, these methods typically do not use global image predictions. Most importantly, their training scenarios are radically different from ours, where ground-truth segment labels are available at training time. Therefore, they address a different task, known as *semantic segmentation* in the literature [14,20], which can be seen as the fully supervised version of segment-level annotation.

Note how several earlier methods proposed for image label prediction actually perform segment-level annotation. Early methods based on probabilistic models [2,5,19] describe the image as an orderless bag of segments. Non-parametric mixture models like multiple bernoulli relevance models [9] also rely on image regions.

2 Models Overview

Before investigating ways to combine segment-level and image-level information, we present a unified view which incorporates most previous works. Fig. 1(b) shows the two main existing ways to obtain predictions on a test image using image-level (arrow A) or segment-level methods (sequence of arrows B-C-D). Image-level methods [1,11,17,25] directly transfer labels from training images to test images using global image similarities (A). Segment-level methods [2,3,4,5,9,19] first estimate labels for the segments in the training images (B), then transfer them to the segments in the test image (C). Finally, they derive a prediction of image labels from these predicted segment labels (D).

In the following sections, we first present various alternatives for the components in fig. 1(b) (arrows), including new ones that we propose. We then present novel methods to combine segment and image-level models in sec. 7 (stages E and F) .

3 Image Label Transfer (A)

Transferring labels from training images to test images is the most direct way to predict image labels. This strategy has recently been shown to be very successful [1,11,17].

Formally, let $\mathcal{I}$ be the set of N training images I_i. The *dictionary* $\mathcal{D}$ is the set of unique labels in the annotations of the training images. There are V labels in $\mathcal{D}$ and we refer to them by their id $l \in \{1..V\}$. Each training image is annotated with labels from $\mathcal{D}$. We summarize the annotation as L_l, which is an indicator function for label l. If image I_i is annotated with label l, then $L_l(I_i) = 1$, and 0 otherwise.

Here, we focus on the recent, state-of-the-art TagProp [11]. which transfers labels using a weighted nearest neighbor approach, but other works fall in this category (A) [17,25].

3.1 TagProp

The label prediction $L_l(Y)$ for a test image Y is based on a weighted sum over the training images:

$$\text{tagprop}_l(Y) = p(L_l(Y)|\mathcal{I}) = \sum_{i=1}^{N} \pi_{yi} p(L_l(I_i)) \tag{1}$$

Where $p(L_l(I_i)) = 1 - \epsilon$ for $L_l(I_i) = 1$, ϵ otherwise. In [11] several variants for π_{yi} are presented. We summarize here the best performing variant, which produces state-of-the-art results. Specifically, the weights π_{yi} are

$$\pi_{yi} = \frac{\exp(-d_w(Y,I_i))}{\sum_j \exp(-d_w(Y,I_j))} \quad \text{with} \quad d_w(Y,i) = \mathbf{w}^T \mathbf{d}_{yi} \tag{2}$$

where $\mathbf{d}_{yi}$ is a vector of base distances between Y and I_i. A separate base distance is computed for each type of image feature and $\mathbf{w}$ is a vector of positive coefficients for combining these distances. This variant is called *ML*, for metric learning, because $\mathbf{w}$ is learned so as to maximize the log-likelihood $\mathcal{L}$ of the leave-one-out predictions on the training set

$$\mathcal{L} = \sum_{i,l} c_{il} \ln p(L_l(I_i)|\mathcal{I}\backslash I_i) \tag{3}$$

where $\mathcal{I}\backslash I_i$ is the set of training images without I_i, and c_{il} is a reweighting parameter for labels. It gives more weight to present labels than to absent ones since the absence of labels in the annotation is less reliable information [11]. As the log-likelihood (3) is concave, we maximize it using a projected-gradient algorithm. The first derivative of eq. (3) with respect to $\mathbf{w}$ is

$$\frac{\delta\mathcal{L}}{\delta\mathbf{w}} = \sum_{i,j} W_i(\pi_{ij} - \rho_{ij})d_{ij} \quad \text{with} \quad \rho_{ij} = \sum_l \frac{c_{iw}}{W_i} p(L_l(I_j)|L_l(I_i)) \tag{4}$$

This learning step was shown by [11] to outperform earlier, ad-hoc ways to transfer labels from image neighbors [17]. Note that, in order to keep learning efficient, the $\mathbf{d}_{yi}$ are only computed for the K nearest neighbors (typically 200) of Y in $\mathcal{I}$. We set $\pi_{yi} = 0$ for all others.

Weighted nearest neighbor models tend to have low recall, since rare labels are unlikely to appear in many neighbor images. Therefore, [11] further adds a word-specific logistic discriminant model to boost the probability for rare labels:

$$p(L_l(Y)|\mathcal{I}) = \sigma(\alpha_l x_{yl} + \beta_l) \quad \text{with} \quad \sigma(z) = (1 + \exp(-z))^{-1} \tag{5}$$

$$x_{yl} = \sum_{i}^{N} \pi_{yi} p(L_l(I_i)) \tag{6}$$

The parameters (α_l, β_l) and $\mathbf{w}$ are learned in alternating fashion to maximize eq. (3). See [11] for details.

4 Segment Label Estimation (B)

We discuss here models to estimate segment labels from image labels during training (fig. 1(b), arrow B). This stage is necessary since only ground-truth image labels are available for training. Estimating segment labels from image labels can be seen under different points of view: as a Multiple Instance Learning problem [12] where an image forms a bag of instances (segments); as a constrained clustering problem [7]; or the missing segment labels can be recovered by MRFs [21]. The same task is also referred to as the *Label-to-Region* problem by a few authors [16].

Formally, the task is to estimate the labels of every segment $s \in \mathcal{S}_i$ in every training image I_i, guided by the given image labels $L_l(I_i)$. This involves estimating the probability $p(L_l(s)|\{\mathcal{S}_i\},\mathcal{I})$ of $L_l(s) = 1$ for every label l and segment s in every image i. We present below three alternative approaches for this task (either one can be used).

4.1 Label Copy

As a straightforward approach, labels can be simply copied from an image to its segments. In this case, all segments in an image are assigned the same labels. We obtain the following expression for the segment labels

$$p(L_l(s)|\{\mathcal{S}_i\},\mathcal{I}) = L_l(I_i). \tag{7}$$

This is a conservative approach. It contains noise for the presence of a label, but almost none for the absence of a label. Some methods for segment label transfer (C) are very robust to label presence noise and perform surprisingly well with label copy.

4.2 Token Model

This model represents segments by visual words as in [7]. All N_s segments are collected in the set $\mathcal{S} = \cup_i \mathcal{S}_i$. We describe the appearance of each segment $s_j \in S$ with a feature vector f_j (sec. 9) and then apply k-means to all vectors to obtain Q cluster centers c_q. Each c_q is a *visual word* and $\mathcal{C} = \cup_q c_q$ is the *codebook* of visual words. We now assign each segment s_j to its closest cluster center c_q and denote the id q as the *token* $T(s_j)$ of s_j. The Token Model represents segments solely by their token. This turns the estimation of $p(L_l(s)|\{\mathcal{S}_i\},\mathcal{I})$ into

$$p(L_l(s)|\{\mathcal{S}_i\},\mathcal{I}) = p(L_l(T(s))|\{T(\mathcal{S}_j)\},\mathcal{I}). \tag{8}$$

Representing a segment as a token rather than a feature vector is beneficial because tokens are discrete and finite, whereas feature vectors live in a continuous and typically high-dimensional space. Therefore, estimating (8) is easier than estimating the distribution $p(L_l(s)|\{\mathcal{S}_i\},\mathcal{I})$ directly.

In the spirit of [7], we adopt a simple clustering approach, which assigns exactly one label z_{ij} to each segment s_{ij} of image I_i

$$L_l(s_{ij}) = \begin{cases} 1 \text{ if } l = z_{ij} \\ 0 \text{ otherwise.} \end{cases} \tag{9}$$

From a given segment-label assignment z we derive the empirical label-token distribution

$$p(L_l(t)|t,z) = Z \sum_{ij}^{T(s_{ij})=t} L_l(s_{ij}), \tag{10}$$

where Z is the normalization factor and t is a token.

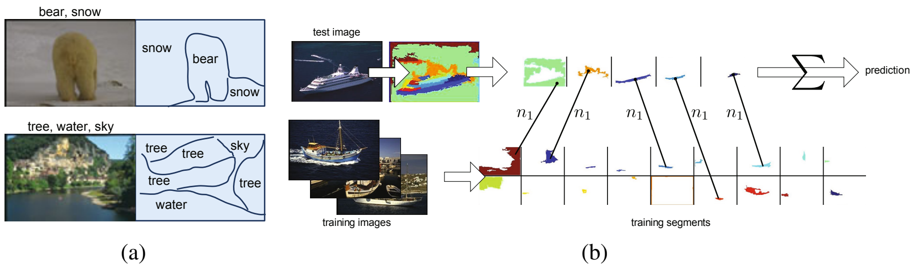


Fig. 2. Left (a): Example segment label estimations on two training images (ground-truth annotated only at the image level). Right (b): **The Global Segprop model.** The prediction for a test image (top) is a mixture over the nearest neighbors of the image's segments (center, shown with lines) in the training set (bottom). For clarity, only the first nearest neighbor n_1 of each segment is shown.

To learn the labeling we use an EM-like scheme. We initialize z_{ij} with a random label of image I_i. In the first step, the probability in eq. (10) is estimated using the last assignments z_{ij}. In the second step, z_{ij} are estimated using eq. (10) (keeping them restricted to the labels $L_l(I_i)$ of the ground-truth image labels). The steps are repeated until convergence.

4.3 Label-To-Region (LTR)

This is the approach described in the recent work of [16]. It consists of two stages. First, corresponding segments between image with common labels are found. Second, labels are assigned to segments based on these correspondences.

In the first stage, a segment s in an image I_i is approximated in the feature space as a sparse linear combination of segments $s' \in \mathcal{S}'$ in other images $\mathcal{I} \backslash I_i$ sharing at least one label. Then, labels are transferred to s from $\mathcal{S}'$ according to the sparse linear combination. This scheme is repeated for all segments until convergence. The initial labels for the segments are copied from the image, as in Label Copy (sec. 4.1). For each segment, this stage returns a probability vector over labels (multinomial distribution).

In the second stage, labels are assigned to segments. For each image, the probability vectors of the segments are clustered into as many clusters as there are labels for the image. The resulting clusters are then labeled with the most likely label according to the centroid. Finally, each segment is given the label of the its cluster.

5 Segment Label Transfer (C)

We present here two alternatives for transferring labels from training segments to segments in a test image Y. While this is not as direct as image-level predictions (A), it is more flexible as it can explain the test image as a combination of segments not observed during training. At this stage, segment labels on the training set have already been derived from ground-truth image labels (B). Throughout this section, $\mathcal{S}$ is the set of segments s_i in the training set.

5.1 Token Model

The Token Model trained in (B) is directly applicable to test images. We apply to each test image segment y the quantization procedure described in sec. 4.2 and obtain its token $t = T(y)$. Then, the multinomial distribution $p(L_l(t)|t)$ in (10) is used to predict the label of y

$$\text{tokenmodel}_l(y) = p(L_l(t)|t) \propto \sum_{s \in S}^{T(s)=t} L_l(s). \tag{11}$$

For any given token, this is the vector of frequencies of estimated segment labels in the training set.

5.2 SegProp

As a novel alternative to the Token Model, we propose here an approach analog to TagProp (sec. 3) to transfer labels from training segments to test segments. We refer to it as SegProp, for Segment-level Propagation. The output of SegProp for label l for a test image segment y is

$$\text{segprop}_l(y) = p(L_l(y)|\mathcal{S}) = \sum_{i=1}^{N_s} \pi_{yi} p(L_l(s_i)), \tag{12}$$

where $p_k(L_l(s)) = 1 - \epsilon$ for $L_l(s) = 1$, ϵ otherwise. Therefore, the label prediction of a segment is a weighted sum over the training segments s_i. As in sec. 3, we restrict ourselves to the K nearest neighbors, set $\pi_{yi} = 0$ for all others, and use the same projected-gradient method to learn this model. Note that, for a test segment y, SegProp outputs a vector of probabilities with one entry per label (*e.g.* $[p(L_1(y)) \dots p(L_V(y))]$).

6 Image Labels from Segment Predictions (D)

The last stage of predicting image labels using segments is to transfer labels to the image from the predicted labels of its segments. When each segment label is predicted as a multinomial or multiple Bernoulli distributions, it is natural to combine them, for instance using a mixture model. We detail two alternatives below. Let Y denote a test image and $\{y_r\}$ the set of its segments.

6.1 Maximum Prediction

In this approach, we combine segment-level predictions into an image-level one by keeping, for each label, the largest prediction over the segments. This procedure takes advantage of the compositionality of segments. If two regions are predicted to have different labels, it indeed transfers both labels to the image. Formally, we define:

$$p(L_l(Y)|\{y_r\}) = \max_r p(L_l(y_r)). \tag{13}$$

6.2 Global SegProp

Instead of considering each segment to have the same importance in the final prediction, an alternative is to use a mixture over the segments. This is the base of our new Global SegProp model. Specifically, Global SegProp outputs an image-level prediction as a mixture of the labels of the training neighbors of its R largest segments $\{y_r\}$ (largest area relative to image):

$$p(L_l(Y)|\{y_r\}) = \sum_{i=1}^{N_s} \pi_{yi} p(L_l(s_i)) \tag{14}$$

Where $p(L_l(s)) = 1 - \epsilon$ for $L_l(s) = 1$, ϵ otherwise. The components for $\mathbf{d}_{yi}$ (see eq. (2)) are the feature space distances for segment s_i to the R largest segments $\{y_r\}$. As before, we compute the K nearest neighbors for every of the R largest segments, take the union set, and set $\pi_{yi} = 0$ for segments not in this set.

Importantly, the weights are now optimized for image-label prediction during training, whereas SegProp optimizes them for segment-label prediction. Hence, this model perform stages (C) and (D) jointly (fig. 2(b)).

7 Joint Label Prediction

In this section we propose several models for combining the image and segment levels for predicting labels of a test image Y. This is desirable as the information that the two levels offer is orthogonal. The global, image-level models are more distinctive because they capture context. The local, segment-level models are more flexible thanks to compositionality. Moreover, they can annotate the test image at the segment level. By doing the prediction jointly, we can hope to bring some contextual information into the segment-level predictions as well as improving image annotation by exploiting compositionality.

We devise three alternatives to combine TagProp (A) with segment-level predictions (C), for achieving both segment-level prediction (E) and image-level prediction (F). The first two are rather simple and based on multiplying the output probabilities (sec. 7.1 and 7.2). Last, we propose a more complex one, based on combining neighborhoods of image-level and segment-level models (sec. 7.3).

7.1 Joint Segment-Level Prediction by Product (E)

In this joint model, the image-level prediction acts as a prior to guide the segment-level prediction. To include the prediction for image Y to predict its segment y_i, we compute $p(L_l(y_i)|Y)$ as:

$$p(L_l(y_i)|Y) = p(L_l(Y))p(L_l(y_i)), \tag{15}$$

where $p(L_l(Y))$ is the output of any image-level method (A), and $p(L_l(y_i))$ of any segment-level prediction (C).

For (A), we have only considered TagProp, so $p(L_l(Y)) = \text{tagprop}_l(Y)$. For (C), $p(L_l(y_i))$ can be set to either $\text{tokenmodel}_l(y_i)$ or $\text{segprop}_l(y_i)$ (sec. 5), leading to combinations that we refer to as "TagProp×Token" and "TagProp×SegProp".

7.2 Joint Image-Level Prediction by Product (F)

In order to achieve the effect of improving image-level prediction using segment-level prediction, we propose to combine the output of any image-level method (A) with the image-level prediction (D) corresponding to a segment-level method (C):

$$p(L_l(Y)|\{y_i\}, Y) = p(L_l(Y)|Y)p(L_l(Y)|\{y_r\}). \tag{16}$$

Table 1. Summary of pixel annotation results on the MSRC-21 dataset

Name (Parameters)	A	B	C	E	Overall acc.
Token Model ($Q = 2300$)	-	Token	Token	-	24.4%
SegProp ($Q = 2300, K = 50$)	-	Token	SegProp	-	25.6%
SegProp ($K = 50$)	-	LTR	SegProp	-	29.6%
SegProp ($K = 50$)	-	Copy	SegProp	-	31.4%
TagProp+Token	TagProp	Token	Token	Prod.	27.8%
TagProp+SegProp	TagProp	Copy	SegProp	Prod.	**33.8%**

Again, TagProp will be used for $p(L_l(Y)|Y)$, while $p(L_l(Y)|\{y_r\})$ can be obtained by Maximum Prediction (D) from any segment-level method, or by using Global SegProp (sec. 6.2). As in the previous section, we refer to these as "TagProp×Token" and "TagProp×SegProp".

7.3 Tagprop + Global SegProp (F)

We propose a novel and more elaborate technique to predict image labels by combining image-level and segment-level information. We include both segment neighbors (as in Global Segprop) and image neighbors (as in Tagprop)

$$p(L_l(Y)|\{y_r\}, \mathcal{I}) = \sum_i^{N_s} \pi^s_{yi} p(L_l(s_i)) + \sum_i^{N} \pi^I_{yi} p(L_l(I_i)) \qquad (17)$$

Note that there are two sets of weights, π^S for segment neighbors, and π^I for image neighbors. By fixing one set of weights, we can maximize the log-likelihood over the other set as done for eq. (3). So, we learn both sets in alternation. As done in sec. 3.1, for efficient learning we only consider the K nearest neighbors of Y for image neighbors. For segment neighbors, we include the T nearest neighbors for each of the R top largest segments in Y. In total, there are $K + RT$ neighbors. We set to 0 the π weights for training images/segments not in this set.

8 Data Sets and Features

In this section, we describe the datasets we experiment on, and the image/segment features we use. Note that to properly evaluate our approaches on segment-level annotation from image labels, datasets with ground-truth pixel annotation are required (MSRC-21, SIFT-Flow).

The MSRC-21[1] dataset contains 591 images of 23 object classes, annotated at the pixel level. We adopt the evaluation protocol of [21] and keep the 21 most frequent classes and *void*, leaving *horses* and *mountain* out. As in [21,24], we use a random selection of 531 images for training and the other 60 for testing.

The SIFT-Flow[2] dataset [15] contains 2688 images with a total of 33 objects and background classes annotated at the pixel level (*sky*, *sea*, etc.). We use the training and test subsets defined in [15], with 200 images for testing and the rest for training.

[1] http://research.microsoft.com/en-us/projects/objectclassrecognition/

[2] http://people.csail.mit.edu/celiu/CVPR2009/

The Corel 5k[3] dataset [7] is commonly used for image auto-annotation. It comes with pre-defined training and test images that have been manually labeled with at most 5 keywords out of a vocabulary of 260. The training set consists of 4500 images while the test set has 499 images, which we use to evaluate image-level prediction. There is no pixel-level annotation for this dataset.

To describe images globally, we adopt the features of [11]. They consist of GIST, color histograms (RGB, LAB, HSV) with 16 bins per channel, and bag-of-features histograms. For the latter, SIFT and Hue [22] descriptors are computed on a multiscale grid of points and at Harris interest points. These descriptors are quantized using K-means with 1000 centroids for SIFT and 100 for Hue. Additionally, histograms over three horizontal regions are also computed for all descriptors except for GIST. This results in 15 different descriptors. For the base distances, we use $L2$ for GIST, $L1$ for color, χ^2 for bag-of-features.

For segments, we adapt the descriptors described above. First, color histograms are computed with only 12 bins per channel to reduce the dimensionality. Quantized local descriptors are accumulated in individual histograms of segments based on the location of the interest points. In total, there are 7 descriptors. The base distances are computed analog to the image-level case. For the Token Model, we have reimplemented the segment features of [2]: relative size and position in the image, average and standard deviation of pixel RGB and LAB, and shape features such as ratio of area to perimeter, eccentricity and ratio of area to convex hull. Here, $L2$ is used as a distance measure. Our segments are computed using [8].

9 Experimental Evaluation

We present here the experimental protocols and our results for both segment and image label prediction tasks.

Segment-level prediction. Segment-level prediction is evaluated using a standard measure for semantic segmentation [15,20,21]: the percentage of correctly predicted pixels over all pixels (*overall pixel accuracy*).

In tab. 1, we summarize the different methods that we compare for segment-level annotation on the MSRC-21 dataset. The Token Model achieves an overall accuracy of 24.4%. Our proposed SegProp model performs considerably better, reaching 29.6% in conjunction with LTR for stage B, and 31.4% with the simple label copy mechanism for stage B. As SegProp is very robust to the presence of label noise, it performs well in conjuction with label copy.

More importantly, when combining the segment-level predictions with image-level predictions from TagProp, we obtain significant improvements: +2.2% for SegProp and +3.4% for the Token Model. The larger improvement for the Token Model can be explained by the higher complementarity of the methods and features, compared to SegProp. Our TagProp+SegProp combination achieves the best overall accuracy of 33.8%.

In tab. 2, we give the accuracy on the SIFT-Flow dataset. The same conclusions can be drawn: SegProp is superior to the Token Model for segment-level annotation, and

[3] http://kobus.ca/research/data/eccv_2002/

Table 2. Pixel annotation results on the SIFT-Flow dataset

Name (Parameters)	Overall acc.
Token Model ($Q = 2300$)	18.5%
SegProp ($K = 50$)	34.2%
TagProp+Token	31.1%
TagProp+SegProp	**35.9%**

Table 3. Image annotation results on the Corel5k dataset. TagProp is abbreviated as TP and SegProp as SP.

Name (Parameters)	A	B	C	D	E	BEP
Token Model ($Q = 2300$)	-	Token	Token	Max.	-	8.2%
SP ($Q = 2300, K = 50$)	-	Token	SP	Max.	-	11.2%
SP ($K = 50$)	-	Copy	SP	Max.	-	14.9%
Global SP ($R = 10, K = 5$)	-	Copy	Global SP		-	19.8%
TP ($K = 200$)	TP	-	-	-	-	36.2%
TP+Token	TP	Token	Token	Max.	Prod.	22.2%
TP+SP	TP	Copy	SP	Max.	Prod.	27.9%
TP+Global SP ($K = 200, R = 10, T = 5$)	-	Copy	-	-	TP+G SP	**37.0%**

the combination with TagProp improves both models. In fig. 3, we illustrate the benefit of using image-level prediction to guide segment-level prediction.

Note that several works [15,20] report higher scores than ours for both datasets. However, they operate in the *fully supervised* scenario, i.e. using ground-truth pixel labels for training, whereas we use *only image labels*. Those methods are able to train strong appearance classifiers, and can leverage position and smoothness priors.

Image-level prediction. Following previous works [10,11], we measure the Break-Even Point score (BEP). To compute the BEP, first the images are ordered by the predicted probability for a label l. This list is truncated to the length of the true number of relevant images (using ground-truth). The BEP measures the percentage of relevant images in this truncated list, averaged over all labels $l = 1 \ldots V$. Some works [7,11,17] additionally measure precision/recall after assigning the 5 highest-scoring labels to each test image. However, as many test images have fewer than 5 ground-truth labels, the algorithm performance is incorrectly penalized. As a result, the maximum achievable precision is not 100%. We report BEP scores and agree with [10,11] that they are more meaningful.

Tab. 3 summarizes the performance of the methods we compare on the Corel5k dataset. The Token Model achieves a low performance of 8.2%, in line with the published results of a similar model [2]. As in the segment-level evaluation, our SegProp model improves over the Token Model for stage C and reaches 11.2%. Moreover, the gain is higher when using label copy in stage B: 14.9%. Further improvement is obtained by fusing the C and D stages in our newly proposed Global SegProp model: 19.8%.

As the 'TagProp' row shows, consistent with previous observations [11,17], directly predicting image labels using a global similarity outperforms segment-level methods on this task. Note that our result of 36.2% using TagProp with $K = 200$ closely matches the best variant of TagProp reported in [11] (36.3%).

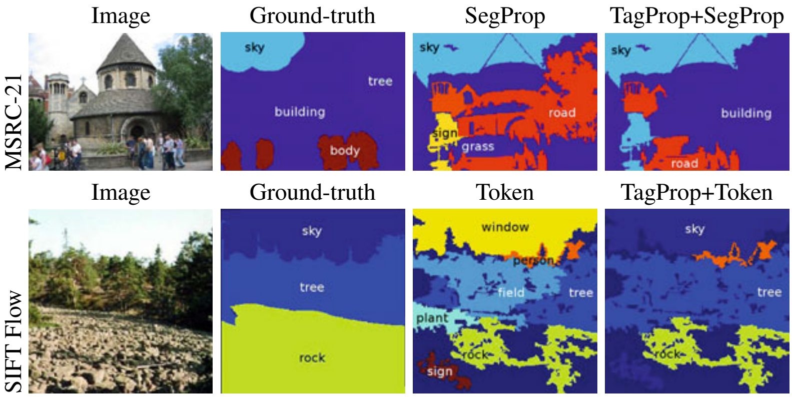


Fig. 3. Example images from the MSRC-21 (top row) and SIFT Flow (bottom row) data set. The first column shows a test image for each. The ground-truth segmentations with their labels are shown in the second column. The last two columns highlight the benefits of using image-level predictions to help segment level prediction. Label predictions using SegProp and TagProp+SegProp (top row), Token and TagProp+Token respectively (bottom row), are shown. In both cases, the combined method improves over the segment-level one.

Our integrated TagProp+Global SegProp method brings a large improvement over Global SegProp (+17.2%). Importantly, it also improves over state-of-the-art TagProp alone. Therefore, our method also improves over other works such as [9,13], which were outperformed by TagProp (see scores for MBRM or TGLM within [11]).

10 Conclusion

We have presented a unified view on image-level and segment-level methods, where existing works can be casted in a common framework. We have proposed new models for some of the stages and, importantly, novel models to perform joint prediction on both levels.

We have conducted extensive experiments on two challeging data sets for pixel-level annotation and on a third one for image-level annotation. Our evaluation confirms that combining image-level and segment-level models brings better results than either model alone, on both tasks. The improvement is particularly strong for the segment labeling task. This shows that both levels have complementary strengths. Finally, note that our combined method TagProp+SegProp performs *both tasks at the same time*. It labels both the pixels and the whole image, unlike TagProp and image-level methods in general, which only deliver image labels.

References

1. Babenko, B., Branso, S., Belongie, S.: Similarity metrics for categorization: from monolithic to category specific. In: ICCV (2009)
2. Barnard, K., Duygulu, P., de Freitas, N., Forsyth, D., Blei, D., Jordan, M.: Matching words and pictures. JMLR (2003)
3. Barnard, K., Fa, Q., Swaminatha, R., Hoog, A., Collin, R., Rondo, P., Kaufhold, J.: Evaluation of localized semantics: data, methodology, and experiments. IJCV (2007)

4. Barnard, K., Forsyth, D.A.: Learning the semantics of words and pictures. In: ICCV (2001)
5. Blei, D., Jordan, M.: Modeling annotated data. In: Proceedings of the ACM SIGIR Conference (2003)
6. Choi, M., Lim, J., Torralba, A., Willsky, A.: Exploiting hierarchical context on a large database of object categories. In: CVPR (2010)
7. Duygulu, P., Barnard, K., de Freitas, J.F.G., Forsyth, D.A.: Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part IV. LNCS, vol. 2353, pp. 97–112. Springer, Heidelberg (2002)
8. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. IJCV 59(2) (2004)
9. Feng, S., Manmatha, R., Lavrenko, V.: Multiple Bernoulli relevance models for image and video annotation. In: CVPR (2004)
10. Grangier, D., Bengio, S.: A discriminative kernel-based model to rank images from text queries. PAMI 30(8), 1371–1384 (2008)
11. Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: TagProp: discriminative metric learning in nearest neighbor models for image auto-annotation. In: ICCV (2009)
12. Jin, R., Wang, S., Zhou, Z.H.: Learning a distance metric from multi-instance multi-label data. In: CVPR (2009)
13. Li, J., Li, M., Liu, Q., Lu, H., Ma, S.: Image annotation via graph learning. Pattern Recognition 42(2), 218–228 (2009)
14. Lim, Y., Jung, K., Kohli, P.: Energy Minimization under Constraints on Label Counts. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6312, pp. 535–551. Springer, Heidelberg (2010)
15. Liu, C., Yuen, J., Torralba, A.: Nonparametric scene parsing: Label transfer via dense scene alignment. In: CVPR (2009)
16. Liu, X., Cheng, B., Yan, S., Tang, J., Chua, T., Jin, H.: Label to region by bi-layer sparsity priors. In: ACM Multimedia (2009)
17. Makadia, A., Pavlovic, V., Kumar, S.: A New Baseline for Image Annotation. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 316–329. Springer, Heidelberg (2008)
18. Me, T., Wan, Y., Hu, X., Gon, S., Li, S.: Coherent image annotation by learning semantic distance. In: CVPR (2008)
19. Monay, F., Gatica-Perez, D.: PLSA-based image auto-annotation: constraining the latent space. In: ACM Multimedia, pp. 348–351. ACM (2004)
20. Shotton, J., Winn, J.M., Rother, C., Criminisi, A.: *TextonBoost*: Joint Appearance, Shape and Context Modeling for Multi-class Object Recognition and Segmentation. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 1–15. Springer, Heidelberg (2006)
21. Verbeek, J., Triggs, B.: Region classification with Markov field aspect models. In: CVPR (2007)
22. van de Weijer, J., Schmid, C.: Coloring Local Feature Extraction. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part II. LNCS, vol. 3952, pp. 334–348. Springer, Heidelberg (2006)
23. Yuan, J., Li, J., Zhang, B.: Exploiting spatial context constraints for automatic image region annotation. In: ACM Multimedia (2007)
24. Zha, Z., Hua, X., Mei, T., Wang, J., Qi, G., Wang, Z.: Joint multi-label multi-instance learning for image classification. In: CVPR (2008)
25. Zhang, H., Berg, A., Maire, M., Malik, J.: SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In: CVPR (2006)

Multi-layer Local Graph Words for Object Recognition

Svebor Karaman[1], Jenny Benois-Pineau[1], Rémi Mégret[2], and Aurélie Bugeau[1]

[1] LaBRI - University of Bordeaux, 351, Cours de la Libération,
33405 Talence Cedex, France
{Svebor.Karaman,Jenny.Benois-Pineau,Aurelie.Bugeau}@labri.fr
[2] IMS - University of Bordeaux, 351, Cours de la Libération
33405 Talence Cedex, France
Remi.Megret@ims-bordeaux.fr

Abstract. In this paper, we propose a new multi-layer structural approach for the task of object based image retrieval. In our work we tackle the problem of structural organization of local features. The structural features we propose are nested multi-layered local graphs built upon sets of SURF feature points with Delaunay triangulation. A Bag-of-Visual-Words (BoVW) framework is applied on these graphs, giving birth to a Bag-of-Graph-Words representation. The multi-layer nature of the descriptors consists in scaling from trivial Delaunay graphs - isolated feature points - by increasing the number of nodes layer by layer up to graphs with maximal number of nodes. For each layer of graphs its own visual dictionary is built. The experiments conducted on the SIVAL and Caltech-101 data sets reveal that the graph features at different layers exhibit complementary performances on the same content. The combination of all layers, yields significant improvement of the object recognition performance.

Keywords: Feature representation, Structural features, Bag-of-Visual-Words, Graph Words, Delaunay triangulation, Context Dependent Kernel.

1 Introduction

Visual object retrieval in images and videos is one of the most active fields of research. One of the most popular techniques relies on the use of local features, e.g. using for instance SIFT (Scale Invariant Feature Transform) of Lowe [1] or SURF (Speed-Up Robust Features) of Bay [3]. SIFT and SURF key points descriptors are robust and discriminative local features. In the trending approach of Bag-of-Visual-Words [2], the features are quantized in visual dictionaries by clustering and images are modeled by a distribution of the visual words within them. The Bag-of-Visual-Words approach is an adaptation of the text retrieval approach Bag-of-Words (BoW) to images. The BoVW operates on local visual features such as key points when the BoW operates on words. The semantic power of a word is much higher than which of a local key point, a visual word is also much more ambiguous than a text word. Moreover, the BoVW approach discards all spatial information about the relations between key points. Having a similar local distribution of key points in two images indicates a stronger similarity of content than sparse isolated key points.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 29–39, 2012.

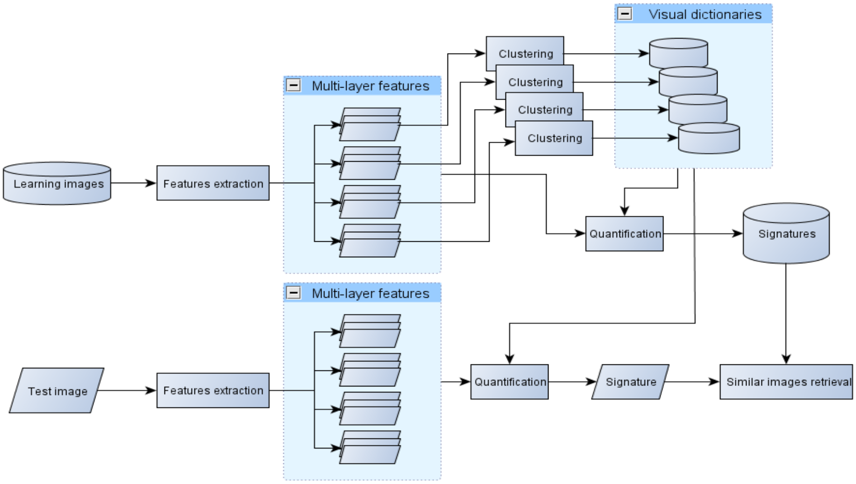


Fig. 1. Flowchart of the Bag-of-Words framework applied to our multi-layer features

To overcome this limitation of the BoVW, some approaches have been developed in the past few years. The spatial pyramid matching proposed in [4] compares the distributions on several areas generated by splitting the image spatially. However, such an approach is not invariant to affine transformations loosing the most important characteristic of points invariant local features. In [5] an approach called "Visual Phrases" is introduced to group visual words according to their proximity in the image plane as a sequence of features. The visual phrases are represented by a histogram containing the distribution of the visual words in the phrase. In these works, the common idea is to build local signature according to a visual dictionary from an arbitrary splitting for the spatial pyramid matching or on a set built by a proximity criterion for visual phrases. Compared to these works, our approach consists in introducing the local topological information within the visual features.

In this paper we propose a spatial embedding of features with local Delaunay graphs. Thanks to the invariance of Delaunay triangulation with regard to affine transformations of image plane: rotation, translation and scale, the graphs inherit the invariance of key point features such as SURF. We propose to combine the structural information injected by the Delaunay graph with the robustness brought by the BoVW approach. We therefore consider multiple local Delaunay graphs as visual words, and plunge them into a Bag-of-Visual-Words framework, by building visual dictionaries obtained by clustering the sets of local graphs. Then state-of-the-art visual signatures are used for object retrieval. Increasing the number of nodes of the local graphs yields a layered approach where each layer induces a stronger spatial embedding within graph features. We call this approach "nested", as each local graph is obtained by adding nodes to a local graph from the previous layer. It combines visual signatures of all graphs from trivial graphs which are isolated SURF points to larger graphs that contain about ten nodes. The proposed framework is summarized in the flowchart presented in Figure 1.

The paper is organized as follows, in section 2 we discuss the process of building these graphs and introduce their nested construction. In section 3, we introduce the dissimilarity measure used to compare graphs and built visual dictionaries by clustering. The latter are presented in section 4. Experiments with these new features are presented in section 5. Conclusions and perspectives are given in section 6.

2 Graph Feature Construction

Let us consider a graph $\boldsymbol{G=(X,E)}$ with $\boldsymbol{X}$ a set of nodes corresponding to some feature points $\boldsymbol{x_{k,k=1,.,K}}$, in image plane and $\boldsymbol{E=\{e_{kl}\}_{,k=1,.,K,l=1,.,K}}$, where $\boldsymbol{e_{kl}=(x_k,x_l)}$, a set of edges connecting these points. We call such a graph a "graph feature". We will build these features upon sets of neighboring feature points in image plane. Hence we propose a spatial embedding of local features with graphs. To build such graphs two questions have to be addressed: i) the choice of feature points sets $\boldsymbol{X}$ and ii) the design of connectivity as edges $\boldsymbol{E}$.

To define the feature point sets $\boldsymbol{X}$ upon which graphs will be built we are looking for a set of feature points that we call the "seeds". Around them, other feature points will be selected to build each graph feature. Selected seeds have to form a set of SURF points which are more likely to be detected in various instances of the same object. SURF points are detected where local maxima of the response of the approximated Hessian determinant are reached [3]. The amplitude of this criterion is a good choice for selecting the seeds, as SURF points with higher response correspond to more salient visual structures and are therefore more likely to be more repeatable. Hence, the seeds considered for building the graphs will be the SURF points with highest responses. Considering a fixed number of seeds $\boldsymbol{N_{Seeds}}$, we can define the set of seeds $\boldsymbol{S}$:

$$S = \{s_1, \dots, s_{N_{seeds}}\} \tag{1}$$

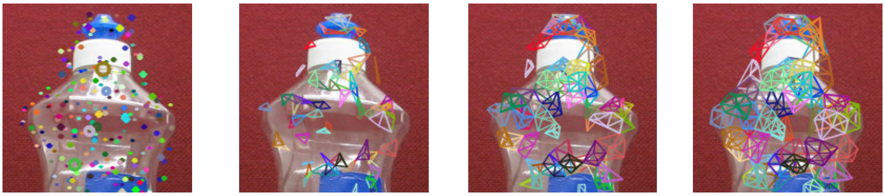
Given $\boldsymbol{S}$, our aim is to add partial structural information of the object while keeping the discriminative power of SURF key points. We will therefore define graphs over the seeds and their neighboring SURF points.

Finding the $\boldsymbol{k}$ spatial nearest SURF neighbors of each seed $\boldsymbol{s_i}$ gives the set of neighbors $\boldsymbol{P_i}$:

$$P_i = \{p_1, \dots, p_k\} \tag{2}$$

Hence the set of nodes for each graph upon a seed point is built. For the edges we use the Delaunay triangulation which is invariant with regard to affine transformations of image plane preserving angles: translation, rotation and scaling. Furthermore, regarding the future extensions of this work to video, the choice of Delaunay triangulation is also profitable for its good properties in tracking of structures [6]. The set of all vertices used for building the graph $\boldsymbol{G_i}$ is $\boldsymbol{X^{Gi}}$, the union of the seed and its neighborhood:

$$X^{Gi} = \left\{x_1^{Gi}, \dots, x_k^{Gi}\right\} = P_i \cup \{s_i\} \tag{3}$$



(a) SURF features (b) 3-nearest neighbors graphs (c)6-nearest neighbors graphs (d) 9-nearest neighbors graphs

Fig. 2. SURF and graph features on the object ajaxorange from SIVAL database

A Delaunay triangulation is computed on the points of $\boldsymbol{X}^{Gi}$, building triangles according to the Delaunay constraint. An edge $\boldsymbol{e_{ij}=(x_i^{Gi},x_j^{Gi})}$ is defined between two vertices of the graph $\boldsymbol{G_i}$ if an edge of a triangle connects these two vertices.

Introducing a layered approach, where each layer adds more structural information we define graphs of increasing size while moving from one layer to the upper one. Each layer has his own set of neighbors around each seed $\boldsymbol{s_i}$ and the triangulation is run separately on each layer. One layer will always contain the points of all the lower layers, hence we call this approach "nested" and illustrate it in Figure 3. To avoid a large number of layers, the number of nodes added at each layer should induce a significant change of structural information. To build a Delaunay triangulation, at least two points have to be added to the seed at the second layer. Adding three nodes may yield three triangles instead of just one, resulting in a more complete local pattern. Therefore, the number of nodes added from one layer to the upper one is fixed to three. We define four layers, the bottom one containing only the seed, and the top one containing a graph built upon the seed and its 9 nearest neighbors, see examples in Figure 2.

3 Graph Comparison

In order to integrate these new graph features in a Bag-of-Visual-Words framework a dissimilarity measure and a clustering method have to be defined. In this section, we define the dissimilarity measure. We are dealing with attributed graphs, where nodes can be compared with respect to their visual appearance. We aim to take into account both similarities of node features and graph topology information for defining a dissimilarity measure between local graphs. To achieve this we will investigate the use of the Context Dependent Kernel (CDK) presented in [7]. The definition of the CDK relies on two matrices: $\boldsymbol{D}$ which contains the distances between node features, and $\boldsymbol{T}$ which contains the topology of the graphs being compared.

Considering two graphs $\boldsymbol{A}$ and $\boldsymbol{B}$ with respective number of nodes $\boldsymbol{m}$ and $\boldsymbol{n}$, let us denote $\boldsymbol{C}$ the union of the two graphs, see (5). The feature correspondence square matrix $\boldsymbol{D}$ of size $(m+n)\text{x}(m+n)$ contains the "entrywise" L_2-norm of the difference between SURF features:

$$\begin{gathered} \mathrm{D} = (\mathrm{d_{ij}})_{\mathrm{ij}} \\ \text{where } \mathrm{d_{ij}} = \left\|\mathrm{x_i^C} - \mathrm{x_j^C}\right\|_2 \end{gathered} \tag{4}$$

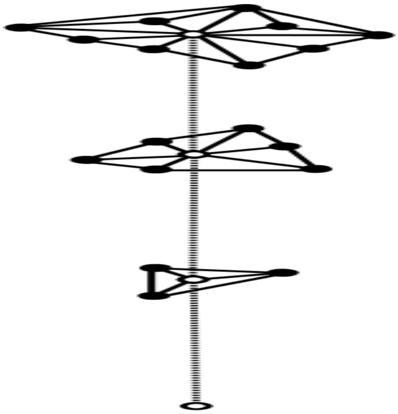


Fig. 3. The nested approach. Bottom to top: SURF seed depicted as the white node, 3 neighbors graph where neighbours are in black, 6 neighbors graph and 9 neighbors graph at the top level.

$$C = A \oplus B$$
$$\text{with} \begin{cases} x_i^C = x_i^A & \text{for} \quad i \in [1..m] = I_A \\ x_i^C = x_{i-m}^B & \text{for} \quad i \in [m+1..m+n] = I_B \end{cases} \tag{5}$$

The square topology matrix $\boldsymbol{T}$ of size $(m+n)\text{x}(m+n)$ defines the connectivity between two vertices $\boldsymbol{x_i}^{\boldsymbol{C}}$ and $\boldsymbol{x_j}^{\boldsymbol{C}}$. In this work we define a crisp connectivity as we set $\boldsymbol{T_{ij}}$ to one if an edge connects the vertices $\boldsymbol{x_i}^{\boldsymbol{C}}$ and $\boldsymbol{x_j}^{\boldsymbol{C}}$ and 0 otherwise. Hence, only sub matrices where both lines and columns in $\boldsymbol{I_A}$ or $\boldsymbol{I_B}$ are not entirely null. More precisely, we can define sub matrices $\boldsymbol{T_{AA}}$ and $\boldsymbol{T_{BB}}$ corresponding to the topology of each graph $\boldsymbol{A}$ and $\boldsymbol{B}$ respectively, while sub matrices $\boldsymbol{T_{AB}}$ and $\boldsymbol{T_{BA}}$ are entirely null, vertices of graphs $\boldsymbol{A}$ and $\boldsymbol{B}$ are not connected.

$$T = (T_{ij})_{ij}$$
$$\text{where } T_{ij} = \begin{cases} 1 \text{ if edge } (x_i^C, x_j^C) \text{ belongs to } \boldsymbol{A} \text{ or } \boldsymbol{B} \\ \quad 0 \text{ otherwise} \end{cases} \tag{6}$$

The CDK denoted $\boldsymbol{K}$ is computed by an iterative process consisting of the propagation of the similarity in the description space according to the topology matrix.

$$K^{(0)} = \frac{\exp\left(-\frac{D}{\beta}\right)}{\left\|\exp\left(-\frac{D}{\beta}\right)\right\|_1} \quad , \quad K^{(t)} = \frac{G(K^{(t-1)})}{\left\|G(K^{(t-1)})\right\|_1}$$
$$G(K) = \exp\left(-\frac{D}{\beta} + \frac{\alpha}{\beta} T K^{(t-1)} T\right) \tag{7}$$

Where *exp* represents the coefficient-wise exponential and $\|\boldsymbol{M}\|_1 = \Sigma_{ij}|\boldsymbol{M_{ij}}|$ represents the L1 matrix norm. The two parameters $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ can be seen respectively as weights for features distance and topology propagation. Similarly to the definition of sub matrices in topology matrix $\boldsymbol{T}$ we can define sub matrices in the kernel matrix $\boldsymbol{K}$. The sub matrix $\boldsymbol{K_{AB}}^{(t)}$ represents the strength of the inter-graph links between graphs $\boldsymbol{A}$ and

B once the topology has been taken into account. We can therefore define the dissimilarity measure that will be used for clustering:

$$
\begin{aligned}
s(A,B) &= \textstyle\sum_{\{i\in I_A, j\in I_B\}} K_{ij}^{(t)} \in [0,1] \\
\rho(A,B) &= s(A,A) + s(B,B) - 2s(A,B) \in [0,1]
\end{aligned}
\tag{8}
$$

This dissimilarity measure will be applied separately on each layer. However, for the bottom layer, since there is no topology to take into account for isolated points we will use directly the "entrywise" L_2-norm of the difference between SURF features.

4 Visual Dictionaries

The state-of-the-art approach for computing the visual dictionary of a set of features is the use of the K-means clustering algorithm [2] with a large number of clusters, often several thousands. The code-word is either the center of a cluster or a non-parametric representation like a K-Nearest Neighbors (K-NN) voting approach.

Both of these approaches are not suitable for the graph-features as using the K-means clustering algorithm implies iteratively moving the cluster centers with interpolation whereas defining a mean graph is a difficult task; and a fast K-NN requires an indexing structure which is not available in our graph feature space since it is not a vector space. Therefore, we present in the following section the selected method which is a two pass agglomerative hierarchical clustering. The model of a cluster is chosen to be the median instead of the mean.

4.1 Clustering Method

In order to quantize a very large database, it can be interesting to use a two pass clustering approach as proposed in [8], as it enables a gain in terms of computational cost. Here, the first pass of the agglomerative hierarchical clustering will be run on all the features extracted from training images of one object. The second pass is applied on clusters generated by the first pass on all objects of the database. To represent a cluster, we use the following definition of the median:

$$median = \mathrm{argmin}_{G\in V} \textstyle\sum_{i=1}^{m} \|v_i - G\| \tag{9}$$

With V – a cluster and $\boldsymbol{v}_i$ – members of a cluster, G the candidate median and $\|\cdot\|$ is a distance or dissimilarity measure in our case.

For the first pass, the dissimilarities between all the features, of the same layer, extracted on all the images of an object are computed. For the second pass, only the dissimilarities between all the medians of all object clusters are computed. Each layer being processed independently, we obtain a visual dictionary for each layer of graphs with 1, 3, .., N_{max} nodes.

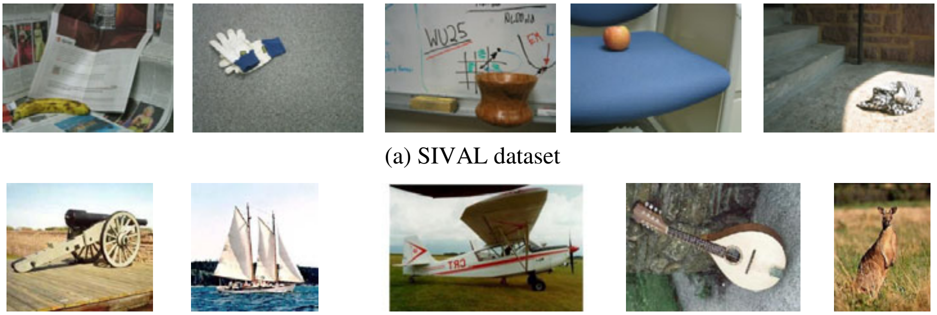
4.2 Visual Signatures

The usual representation of an image in a BoVW approach is to compute a histogram of all the visual words of the dictionary within the image. Each feature extracted from an image is assigned to the closest visual word of the dictionary. We use this representation without rejection, a feature is always assigned to a word in the dictionary. The signatures are then normalized to sum to one by dividing each value by the number of features extracted from the image. Once the visual signatures of images have been computed, one can define the distance between two images as the distance between their visual signatures. In preliminary experiments we have compared results using Hamming distance, Euclidean distance and L_1 distance for this task. The L_1 distance giving better results, final results are presented using this measure only.

5 Experiments

The experiments are conducted on two publicly available datasets. The first one, the SIVAL (Spatially Independent, Variable Area, and Lighting) data set [9] includes 25 objects, each of them being present in 60 images taken in 10 various environment and different poses yielding a total of 1500 images. This data set is quite challenging as the objects are depicted in various lighting conditions and poses. It has also been chosen as the longer term perspective of this work is the recognition of objects of the daily living that may appear in different places of a house, for example a hoover that may be moved in all the rooms in one's house. The second one is the well known Caltech-101 [10] dataset, composed of 101 object categories. The categories are different types of animals, plants or objects. See a snippet of both datasets in Figure 4a and b.

We separate learning and testing images by a random selection. On each dataset, 30 images of each category are selected as learning images for building the visual dictionaries and for the retrieval task. Some categories of Caltech-101 have several hundred of images when others have only a few more than 30. The testing images are therefore a random selection of the remaining images up to 50. We only take into account the content of a bounding box of each object as the focus of this paper is only object recognition and not yet localization. SURF key points of 64 dimensions are extracted within the bounding box, the numbers of seeds for the graphs building process is fixed to 300. The second layer corresponds to graphs built upon the seeds and their 3 nearest neighbors, the third layer with the 6 nearest neighbors and the



(a) SIVAL dataset

(b) Caltech dataset

Fig. 4. Excerpts from image datasets

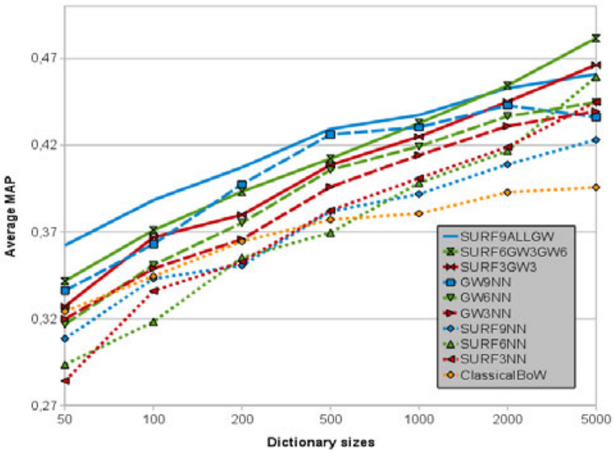


Fig. 5. Average MAP on the whole SIVAL data set. Isolated SURF features are the dotted curves, single layer Graphs Words are drawn as dashed curves and the multi-layer approach in solid curves.

fourth and last layer with the 9 nearest neighbors. For the CDK, α is set to 0.0001, β to 0.1 (ensuring **K** is a proper kernel) and the number of iterations is fixed to 2, as H. Sahbi [7] has shown that the convergence of the CDK is fast. The first pass clustering compute 500 clusters for each object. The final dictionary size varies in the range 50-5000. Each layer will yield its own dictionary. We compare our method with standard BoVW approach. For that purpose, we use all the SURF features available on all images of the learning database to build the BoVW dictionary. The visual words are obtained by performing k-means clustering on the set of all these descriptors. Each visual word is characterized by the center of a cluster.

The graph features are not built using all available SURF points, therefore to analyze the influence of this selection, signatures are computed for the set of SURF which have been selected to build the different layers of graphs. These configurations will be referred to as SURF3NN, SURF6NN and SURF9NN corresponding respectively to all the points upon which graphs with 3, 6 and 9 nearest neighbors have been defined. In this case the dictionaries are built with our two pass clustering approach as for graphs.

For each query image and each database image, the signatures are computed for isolated SURF and the different layers of graphs. We have investigated the combination of isolated SURF and the different layers of graphs by an early fusion of signatures i.e. concatenating the signatures. For SIVAL that concatenation has been done with the signature from the selected SURF corresponding to the highest level whereas for Caltech-101we use the classical BoW SURF signature. Finally, the L1-distance between histograms is computed to compare two images. The performance is evaluated by the Mean Average Precision (MAP) measure. For each test images, all images in the learning set are ranked from the closest (in terms of L_1 distance between visual signatures) to the furthest. The average precision is evaluated for each test image of an object, and the MAP is the mean of these values for all the images of an

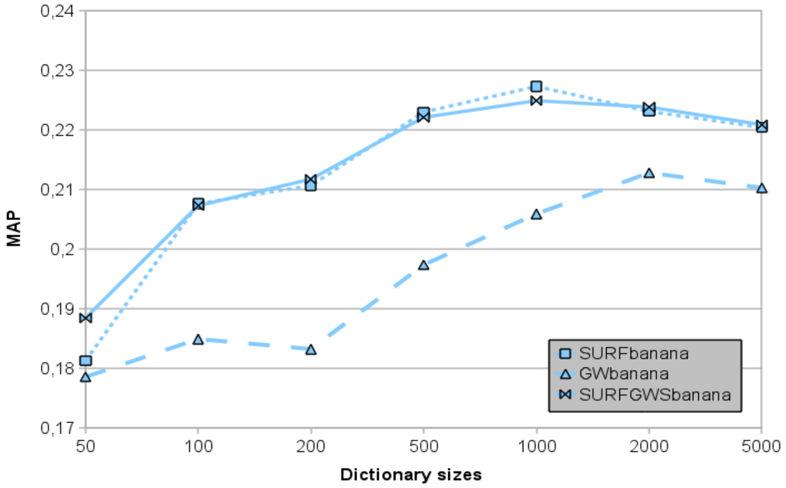


Fig. 6. MAP for the object "banana" from SIVAL where isolated SURF features (dotted curves) outperforms graphs (dashed curves). The multi-layer approach is the solid curve.

object in the test set. For the whole database we measure the performance by the average value of the MAP i.e. we do not weight the MAP per class by the number of query as this would induce more consideration to categories with more testing.

5.1 SURF Based BoW vs Graphs Words

First of all, it is interesting to analyze if the graph words approach where each layer is taken into consideration separately obtains similar performances compared to the classical BoVW approach using only SURF features. This is depicted in Figure 5 where isolated SURF points are depicted as dotted lines, single layer of graph words are dashed lines and the combination of SURF and different graphs layers are plotted as continuous lines. At first glance, we can see that for SIVAL isolated SURF features perform the poorest, separated layers of graphs performs better and the combination of different layers of graphs and the SURF features upon which the highest layer have been computed obtain the best performances. Our clustering approach seems to give worst results for very small size of dictionaries but better results for dictionaries bigger than 500 visual words, which are the commonly used configurations in BoVW approaches. Each layer of graph words performs much better than the SURF upon which they are built. The introduction of the topology in our features have a significant impact on the recognition performance using the same set of SURF features.

The average performance hides however some differences in the performance of each feature on some specific objects. To illustrate this we select two object categories where graph features and SURF give different performances in Figure 6 and 7. For the object "banana" from SIVAL, the isolated SURF features outperform the graph approach, see Figure 6, but for the "Faces" category from Caltech-101 the graphs features perform better, see Figure 7. This unequal discriminative power of each layer leads to the use of the combination of all layers in a single visual signature.

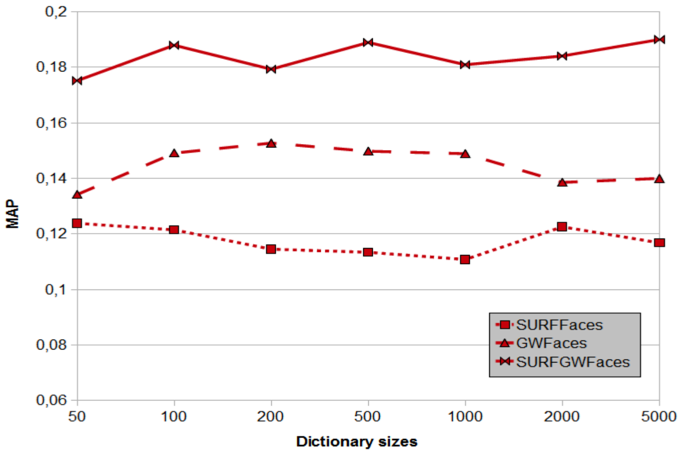


Fig. 7. MAP for category “Faces” from Caltech-101 where graphs (dashed curves) outperforms isolated SURF features (dotted curves). The multi-layer approach is the solid curves.

5.2 The Multi-layer Approach

The combination of graphs and SURF features upon which the graphs have been built is done by the concatenation of the signatures of each layer. The three curves in solid lines in Figure 5 correspond to the multi-layer approach using only the two bottom layers (SURF + 3 nearest neighbors graphs) in red, the three bottom layers (SURF + 3 nearest neighbors graphs + 6 nearest neighbors) in green and all the layers in blue. The improvement in the average MAP is clear, and each addition of layer improves the results. The average performance of the combination always outperforms the performance of each layer taken separately. For Caltech-101, the average MAP values of all methods are much lower which is not surprising as there are much more categories and images. Single layer of graphs gives results in the range 0.050-0.061 while the classical BoVW framework on SURF features performances are within 0.057-0.073 of average MAP values. The combination of all layers outperforms here again SURF or graphs used separately with average MAP values in the range of 0.061-0.077. The detailed results presented in Figure 6 and 7 show that the combination of the visual signatures computed on each layer separately performs better or at least as well as the best isolated feature.

6 Conclusion and Perspectives

In this paper, we have presented new graph features built upon SURF points as nodes and expressing spatial relations between local key points. The multi-layer approach using growing neighborhoods in several layers enables to capture the most discriminative visual information for different types of objects. Using growing spatial neighborhood clearly improves the results while each layer taken separately yields smaller improvements. Moreover, this approach introduces spatial information within the features and is therefore complementary and compatible with other recent improvements of the BoW framework for tacking geometry into account, such as the Spatial Pyramid Matching.

The future of this work is the application of the method to the recognition of objects in videos. The approach could be enhanced by refining some steps of the graphs construction and comparison. For instance, the selection of seeds could be performed by an adaptive method and the topology matrix be defined with a soft connectivity. In order to be efficient when processing a large amount of images, i.e. in videos, a graph embedding procedure could be applied to use an indexing structure that would speed up the recognition process.

Acknowledgments. This work is partly supported by a grant from the ANR (Agence Nationale de la Recherche) with reference ANR-09-BLAN-0165-02, within the IMMED project.

References

[1] Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)

[2] Sivic, J., Zisserman, A.: Video google: a text retrieval approach to object matching in videos. In: ICCV 2003, vol. 2, pp. 1470–1477 (2003)

[3] Bay, H., Ess, A., Tuytelaars, T., Gool, L.V.: Surf:Speeded up robust features. Computer Vision and Image Understanding 110, 346–359 (2008)

[4] Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Proc. CVPR (2006)

[5] Albatal, R., Mulhem, P., Chiaramella, Y.: Visual Phrases for automatic images annotation. In: CBMI 2010, Grenoble, France (2010)

[6] Mahboubi, A., Benois-Pineau, J., Barba, D.: Joint tracking of polygonal and triangulated meshes of objects in moving sequences with time varying content. In: IEEE International Conference on Image Processing, vol. 2, pp. 403–406 (2001)

[7] Sahbi, H., Audibert, J.-Y., Rabarisoa, J., Keriven, R.: Robust matching and recognition using context-dependent kernels. In: Proceedings of the 25th International Conference on Machine Learning, pp. 856–863 (2008)

[8] Gosselin, P.H., Cord, M., Philipp-Foliguet, S.: Combining visual dictionary, kernel-based similarity and learning strategy for image category retrieval. Computer Vision and Image Understanding 100(3) (June 2008)

[9] SIVAL Data set, `http://accio.cse.wustl.edu/sg-accio/SIVAL.html`

[10] Fei-Fei, L., Fergus, R., Perona, P.: One-Shot learning of object categories. IEEE Trans. Pattern Recognition and Machine Intelligence

Multimodal Video Concept Detection via Bag of Auditory Words and Multiple Kernel Learning

Markus Mühling, Ralph Ewerth, Jun Zhou, and Bernd Freisleben

Department of Mathematics & Computer Science, University of Marburg
Hans-Meerwein-Str. 3, D-35032 Marburg, Germany
{muehling,ewerth,zhouj,freisleb}@informatik.uni-marburg.de

Abstract. State-of-the-art systems for video concept detection mainly rely on visual features. Some previous approaches have also included audio features, either using low-level features such as mel-frequency cepstral coefficients (MFCC) or exploiting the detection of specific audio concepts. In this paper, we investigate a bag of auditory words (BoAW) approach that models MFCC features in an auditory vocabulary. The resulting BoAW features are combined with state-of-the-art visual features via multiple kernel learning (MKL). Experiments on a large set of 101 video concepts from the MediaMill Challenge show the effectiveness of using BoAW features: The system using BoAW features and a support vector machine with a χ^2-kernel is superior to a state-of-the-art audio approach relying on probabilistic latent semantic indexing. Furthermore, it is shown that an early fusion approach degrades detection performance, whereas the combination of auditory and visual bag of words features via MKL yields a relative performance improvement of 9%.

Keywords: Visual concept detection, video retrieval, bag of words, bag of auditory words, audio codebook, multiple kernel learning.

1 Introduction

The detection of audiovisual concepts in video shots is an essential prerequisite for semantic video retrieval, navigation and browsing. State-of-the-art systems concentrate on high-level features serving as intermediate descriptions to bridge the "semantic gap" between data representation and human interpretation. Hauptmann et al. [6] stated that less than 5000 concepts, detected with a minimum accuracy of 10% mean average precision, are sufficient to provide search results comparable to text retrieval in the World Wide Web.

Current approaches mainly focus on visual features based on local keypoints and scale-invariant feature transform (SIFT) descriptors [15] that currently achieve top performance in visual recognition tasks. Such descriptors are clustered to create a visual vocabulary (codebook), where the cluster centers are regarded as "visual words". Similar to the representation of documents in the field of text retrieval, an image or a video shot can then be represented as a bag

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 40–50, 2012.

of visual words (BoVW) by mapping local descriptors to the visual vocabulary. In some previous approaches, audio features are used for visual concept detection, either by using low-level features such as mel-frequency cepstral coefficients (MFCCs) or by using detection results of specific audio events such as silence, speech, music and noise as mid-level features for subsequent training of video concept classifiers.

In this paper, we leverage the bag of words approach for audio features to enhance video concept detection and propose multiple kernel learning (MKL) as the appropriate fusion scheme for these bag of auditory words (BoAW) and state-of-the-art BoVW features. First, MFCC audio features are extracted from each video shot. Then, an auditory vocabulary is created via k-means clustering. This vocabulary or codebook, respectively, is then exploited to describe and represent a shot via a histogram (bag) of auditory words. These histograms are used to train audio models for video concepts using support vector machines (SVM) and to finally classify video shots based on these models. Experimental results show that a χ^2-kernel is more appropriate for BoAW features than a radial basis function (RBF) kernel, and the proposed system relying on the auditory vocabulary significantly outperforms a state-of-the-art approach that uses probabilistic latent semantic indexing (pLSA). In addition, BoAW features are combined with state-of-the-art visual features (visual vocabulary based on dense sampled SIFT descriptors) via MKL. In contrast to an early fusion approach, the system relying on MKL for fusing auditory and visual features clearly improves a state-of-the-art concept detection system.

The paper is organized as follows. Section 2 discusses related work. Section 3 describes the construction of the auditory vocabulary and the multimodal concept detection system. Experimental results are presented in section 4. Section 5 concludes the paper and outlines areas for future research.

2 Related Work

In recent years, researchers have shifted their attention to generic video concept detection systems, since the development of specialized detectors for hundreds or thousands of concepts seems to be infeasible. Continuous progress has been reported in the field of visual concept detection using bag of (visual) words approaches (BoW). The top 5 official runs at the TRECVid 2010 semantic indexing task rely on BoW representations [20].

In addition to the visual modality, the audio signal of videos carries important information that can help to improve the performance of generic video concept detection systems. Most of the approaches that incorporate audio information directly use additional low-level features such as MFCCs, ΔMFCCs, pitch, zero-crossing rate, energy, or log-power to classify semantic concepts [1][5][14]. For example, Bredin et al. [1] have extracted low-level features including MFCCs and their derivatives to build Gaussian mixture models (GMM) for each of the semantic concepts.

In other approaches, the results of audio event detectors are used as additional mid-level features. Besides acoustic events such as speech, non-speech, background and gender, Snoek et al. [21] detected the occurrence of 16 additional audio events such as “child-laughter”, “baby-crying”, “airplane-propeller”, “sirens”, “traffic-noise”, “car-engine”, “dog-barking”, or “applause” and used the results as additional inputs for concept classifiers like SVMs. Inspired by classical text document analysis, Lu and Hanjalic [16] try to automatically determine these audio elements by regarding them as natural clusters of the audio data. Between 2 and 20 elements are discovered per audio document using an iterative spectral clustering method.

The audio concept classification framework used by Feki et al. [4] first removes segments of silence and then separates the audio signal into speech, music and environmental sound. The environmental sound segments are further classified using a time-frequency analysis based on MFCC features. For video concept detection, visual features and the previously described audio classification results are fed into a fuzzy reasoning system to fuse the different modalities [3].

Jiang et al. [9] have introduced a novel representation called short-term audio-visual atoms. Audio features based on a matching pursuit representation [17] of the audio signal and region-based color, texture, edge, and motion features are combined and a joint audio-visual codebook is built using multiple instance learning.

Inoue et al. [7] have used a statistical framework to combine visual and audio features for video concept detection. The distribution of SIFT descriptors for each shot are described by GMMs, and a SVM with a GMM-kernel that compares GMMs was used for training and classification. In addition, hidden Markov models (HMM) were built for each concept based on audio features, including MFCCs, log-power and the correponding derivatives. The final classification result is a weighted combination of log likelihood ratios from the audio models and from the SIFT GMMs. Using additional audio features, the results for 20 semantic concepts on documentary films could be improved from 15% mean average precision to 16.4%. At the TRECVid [20] evaluation in 2010, the GMM kernel was also applied for MFCC features [8] resulting in a noticeable relative performance improvement for several concepts like “singing”, “dancing”, “cheering” or “animal”.

Peng et al. [18] have proposed a method that performs an audio-only analysis of the video data and investigates the use of an audio pLSA model for video concept detection. An audio vocabulary based on MFCC features from acoustically homogenous segments is built and the latent audio topics are discovered using pLSA. Each shot is then described by the probabilities of the discovered latent topics and classified by a SVM. Results are reported on 85 hours of news videos for 10 concepts from the MediaMill Challenge. Diou et al. [2] have combined BoW audio features based on MFCCs with visual features in an early fusion scheme. However, for the 30 evaluated concepts of the TRECVid 2010 semantic indexing task, the additional use of BoW audio features clearly decreased the performance from 4.5% to 3.5% mean inferred average precision.

The bag of auditory words approach has recently been successfully applied in the fields of music information retrieval and multimedia event detection. Riley et al. [19] have represented songs as a bag of auditory words showing robust results for a variety of signal distortions and Jiang et al. [12] combined bag of words representations for audio and visual features using a late fusion scheme to detect events like "making a cake" or "assembling a shelter".

3 Concept Detection System

In this section, our approach for multimodal video concept detection is presented. We describe the BoAW approach and the MKL framework that is proposed as an appropriate fusion scheme for the combination of BoAW and BoVW features. The application of the BoW representation to auditory features is presented in Section 3.1. SVMs have proven to be powerful for visual concept detection [20] and they are used to build audio models and to classify video shots based on these models. Besides the linear and the RBF-kernel, the χ^2-kernel is applied due to the representation of features as histograms. The used kernel functions for the SVMs are described in Section 3.2. The state-of-the-art visual features and the proposed MKL framework to combine the feature representations of both modalities are presented in Section 3.3.

3.1 Bag of Auditory Words

Since the BoW representation based on local SIFT descriptors achieves superior performance in the field of visual concept detection [20], we leverage the BoW paradigm for audio features. Using a time-frequency analysis of the audio signal, 12-order MFCCs (Mel-Frequency Cepstral Coefficients) are extracted from audio frames of 20 ms length with an overlap of 50%. Thus a video shot is represented as a set of 12-dimensional MFCC vectors. Based on these MFCC vectors, an auditory vocabulary is generated using the k-means clustering algorithm, and the final cluster centers can be interpreted as "auditory words". Similar to the representation of documents in the field of text retrieval, a video shot can then be represented as a bag of auditory words that are the results of a vector quantization process using the generated vocabulary or codebook, respectively. Finally a shot is described as a histogram, counting the occurences of auditory words. To diminish the quantization loss during histogram generation, a soft-weighting scheme similar to the one proposed by Jiang et al. [10] is used. Instead of mapping a MFCC vector only to its nearest neighbor, the top K nearest auditory words are selected. Using an auditory vocabulary of N auditory words, the importance of an auditory word t in a shot is represented by the weights of the resulting histogram bins $w = [w_1, \ldots, w_t, \ldots, w_N]$ with

$$w_t = \sum_{i=1}^{K} \sum_{j=1}^{M_i} sim(j, t), \tag{1}$$

where M_i is the number of MFCC vectors whose i-th nearest neighbor is the auditory word t.

3.2 Kernel Choice

Since SVMs are used to train audio models and to finally classify video shots, a kernel function needs to be specified. A kernel function intuitively measures the similarity between two data instances. Commonly used kernels are the linear and the radial basis function (RBF) kernel:

$$k_{linear}(x, y) = x^T y, \tag{2}$$

$$k_{RBF}(x, y) = e^{-\gamma \sum_i (x_i - y_i)^2}. \tag{3}$$

Since histogram representations are used in the proposed approach, we also apply the χ^2-kernel. It is based on the corresponding histogram distance:

$$k_{\chi^2}(x, y) = e^{-\gamma \chi^2(x,y)} \quad with \quad \chi^2(x, y) = \sum_i \frac{(x_i - y_i)^2}{x_i + y_i}. \tag{4}$$

Jiang et al. [11] have used the χ^2-kernel successfully for BoVW features in the context of visual concept detection. In their study, the χ^2-kernel has outperformed both the linear and the RBF-kernel.

3.3 Multimodal Fusion

In a multimodal fusion setting, BoAW features are combined with state-of-the-art visual features. For visual features we use the BoVW representation and extract densely sampled local SIFT descriptors from the keyframes using the implementation of the Vision Lab Features Library (VLFEAT) [24]. Color information is integrated using RGB-SIFT, where the SIFT descriptors are computed independently for the three channels of the RGB color model (red, green, blue). Thus, the final feature vector is the concatenation of the individual descriptors. Based on these local descriptors, a global visual vocabulary is generated using the k-means algorithm. Each keyframe or shot, respectively, is described as a histogram indicating the presence of each “visual word”. Again, the previously described soft-weighting scheme is applied to consider the similarities of the local descriptors to the codebook entries.

The easiest way to combine BoAW and BoVW features is the early fusion scheme. Using this method, visual and audio features are simply concatenated and directly fed into a SVM. A more sophisticated approach to combine the capabilities of different modalities is MKL. It is applied to find an optimal kernel weighting

$$k_{multimodal} = \alpha \cdot k_{audio} + \beta \cdot k_{visual} \quad with\ \alpha \geq 0,\ \beta \geq 0 \tag{5}$$

where the kernel functions k_{audio} and k_{visual} take both feature modalities into account. We use the l_2-norm to control the sparsity of the weights α and β for audio and visual features, respectively. Throughout our experiments, we use the MKL framework provided by the Shogun library [23] in combination with the SVM implementation of Joachims [13], called SVM^{light}.

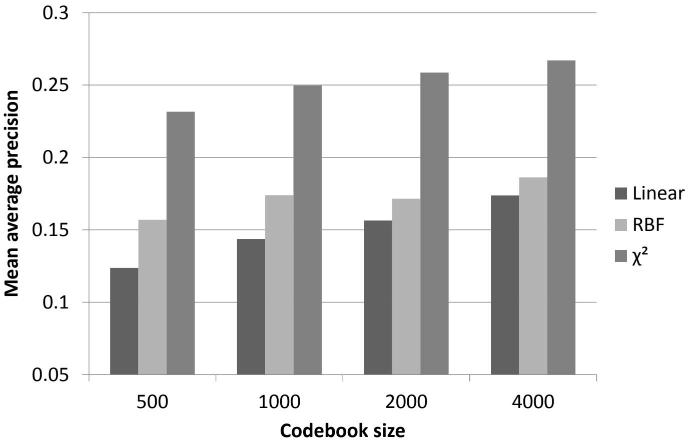


Fig. 1. Performance evaluation of different kernel functions and codebook sizes using BoAW features

4 Experimental Results

In this section, the performance impact of BoAW features in the field of video concept detection is investigated. For this purpose, the MediaMill Challenge [22] is used. It offers a dataset based on the TRECVid 2005 [20] training set with an extensive set of 101 annotated concepts, including objects, scenes, events and personalities. It consists of 86 hours of news videos containing 43,907 completely annotated video shots. These shots are divided into a training set of 30,993 shots and a test set of 12,914 shots.

4.1 Evaluation Criteria

To evaluate the concept retrieval results, the measure of average precision (AP) is used. For each concept, the implemented system returns a list of ranked shots, which is used to calculate the average precision as follows:

$$AP(\rho) = \frac{1}{|R|} \sum_{k=1}^{N} \frac{\left|R \cap \rho^k\right|}{k} \psi(i_k) \tag{6}$$

where $\rho^k = i_1, i_2, \ldots, i_k$ is the ranked shot list up to rank k, N is the length of the ranked shot list, R ist the set of relevant shots and $\left|R \cap \rho^k\right|$ is the number of relevant shots in the top k of ρ. The function $\psi(i_k) = 1$ if $i_k \in R$ and 0 otherwise. To evaluate the overall performance, the mean AP score is calculated by taking the mean value of the average precisions for the individual concepts. Furthermore, the official partial randomization test in the TRECVid evaluation [20] is used to determine whether our system is significantly better than a reference system, or whether the difference is only due to chance.

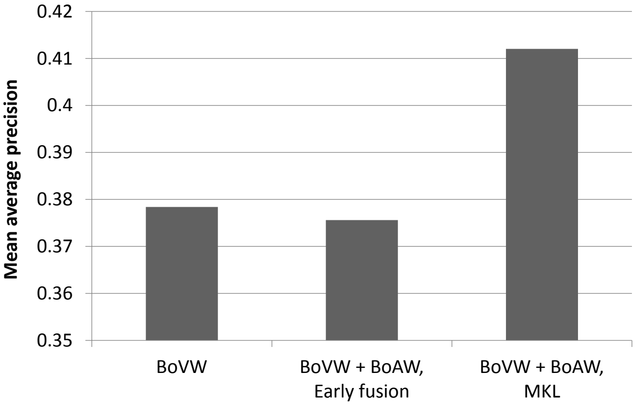


Fig. 2. Performance evaluation of BoAW features in a multimodel setting using early fusion and MKL

4.2 Results

We have performed several experiments to investigate the performance impact of BoAW features both alone and in combination with visual features.

In a first experiment based on an audio-only analysis of the data, different auditory vocabulary sizes and kernel methods have been taken into account. We have compared the linear, RBF and χ^2-kernel in combination with codebook sizes between 500 and 4000 auditory words. The experimental results are presented in Figure 1. The χ^2-kernel significantly outperformes the linear as well as the RBF-kernel. Using 4000 auditory words, the χ^2-kernel yields 43.3% improvement compared to the RBF-kernel. A larger vocabulary also has a positive impact on the overall performance. In combination with the χ^2-kernel, a vocabulary size of 4000 auditory words achieves a mean AP of 26.7% compared to 23.2% for 500 words. Based on these results, the χ^2-kernel and a vocabulary size of 4000 auditory words are used exclusively in the experimental evaluations below.

In a second experiment, we have investigated the impact of BoAW features in a multimodal concept detection system. The state-of-the-art baseline system performs a visual-only analysis of the data using dense sampled RGB-SIFT descriptors with a vocabulary of 4000 "visual words". Both modalities, visual and audio features, are combined using MKL and by using a simple early fusion scheme. In order to save computation time, we have trained the models using a reduced number of negative training samples per concept. The results of the two different fusion strategies are presented in Figure 2. While the early fusion strategy causes a slight performance decrease, the fusion of visual and audio features via MKL achieves a relative performance improvement of 8.9% compared to the baseline system. In total, 31 concepts yield a relative performance improvement of more than 10%. In particular, the concepts representing personalities profit

Table 1. Performance comparison between the visual only baseline system and the multimodal system using MKL, showing average precision values of concepts with relative performance improvements of at least 18%

AP [%]	BoVW	BoVW+BoAW
Motorbike	0.3	4.1
Cycling	13.7	91.7
Racing	11.4	52.3
Bicycle	17.6	80.0
Baseball	0.7	1.6
Natural disaster	8.7	18.0
Boat	18.3	34.1
Golf	36.4	51.3
Waterbody	36.9	49.4
Aircraft	16.6	21.8
Football	54.6	70.6
River	69.9	89.8
Entertainment	55.0	70.2
Sports	49.3	62.2
Table	10.9	13.7
Food	52.6	64.2
Basketball	54.6	65.7
Soccer	72.4	85.6

from the additional audio features, increasing the mean AP for this group of concepts from 9.2% to 11.1%. Further concepts with relative improvements of at least 18% are shown in Table 1.

4.3 Discussion

The experiments indicate that the kernel choice is a critical decision for the performance of the BoAW approach. While the RBF-kernel concentrates on the largest histogram differences due to the quadratic exponential decay, the χ^2-kernel considers the bins more equally. This seems to be beneficial regarding the large intra-class variations of audio signals. Keeping in mind that the ground truth annotation of the 101 semantic concepts is based upon a visual inspection of the video shots, the BoAW approach achieves an impressive performance of 26.7% mean AP on the MediaMill Challenge. The performance is even significantly better than the baseline system provided by the MediaMill Challenge with 21.6% mean AP, which uses local as well as global texture information. The state-of-the-art approach of Peng et al. [18] relying on audio pLSA attained a mean AP of approximately 20.7% on a subset of 10 concepts from the MediaMill Challenge. On the same subset, we achieve a superior performance of 26.8% mean AP using BoAW features, yielding a relative improvement of approximately 30%. Besides the mean AP, Peng et al. displayed AP scores for half of the ten concepts. For these concepts, performance comparisons between the BoAW method and the audio pLSA approach are shown in Figure 3.

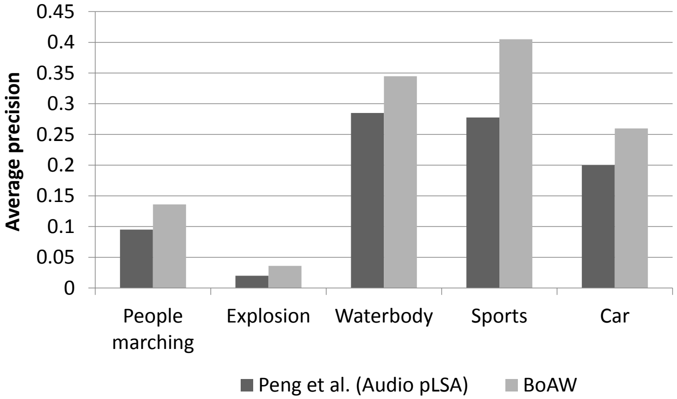


Fig. 3. Performance comparison between the BoAW method and the audio pLSA approach of Peng et al. [18]

Via MKL, additional BoAW features clearly improve the performance of a state-of-the-art video concept detection system that relies on visual features only. The weak performance of the early fusion strategy confirms the results of Diou et al. [2] at the TRECVid 2010 challenge, where the additional use of BoW audio features in an early fusion scheme clearly decreased the performance. This is not surprising, since audio information is more or less important depending on the semantic concept. While "racing" or "motorbike", for example, are characterized by engine noise, there is no discriminative audio information for concepts such as "house" or "gras". In this case, audio features can be even misleading for the classification process. MKL instead of early fusion learns optimized kernel weights that provide information about the relevance of both modalities for the discrimination of semantic concept classes. Hence, audio features are more or less considered depending on the corresponding concept.

5 Conclusions

In this paper, we have presented a bag of auditory words approach for video concept detection that models MFCC features in an auditory vocabulary. This vocabulary is used to describe video shots via histograms of auditory words. SVMs are employed to build the audio models and to finally classify the video shots. Experimental results on a large set of 101 semantic video concepts have shown the effectiveness of the proposed approach. Using BoAW features in combination with the χ^2-kernel yields almost 45% improvement compared to the RBF-kernel.

The proposed system relying on BoAW features outperforms a state-of-the-art audio approach that uses pLSA [18] and is even significantly better than the baseline system provided by the MediaMill Challenge, which used local as well as global texture features.

Furthermore, the resulting BoAW features are combined with visual features via MKL. Using MKL instead of an early fusion scheme significantly improves the results of a state-of-the-art video concept detection system that relies on visual features only.

Areas for future work are the integration of temporal information beyond the scope of audio frames and the investigation of features based on the matching pursuit method instead of MFCCs.

Acknowledgements. This work is supported by the German Ministry of Education and Research (BMBF, D-Grid) and by the German Research Foundation (DFG, PAK 509).

References

1. Bredin, H., Koenig, L., Farinas, J.: IRIT @ TRECVid 2010: Hidden Markov Models for Context-aware Late Fusion of Multiple Audio Classifiers. In: TREC Video Retrieval Evaluation Workshop, TRECVid 2010 (2010)
2. Diou, C., Stephanopoulos, G., Delopoulos, A.: The Multimedia Understanding Group at TRECVID-2010. In: TREC Video Retrieval Evaluation Workshop, TRECVid 2010 (2010)
3. Elleuch, N., Zarka, M., Feki, I., Ammar, A.B.E.N., Alimi, A.M.: REGIMVID at TRECVID 2010: Semantic Indexing. In: TREC Video Retrieval Evaluation Workshop, TRECVid 2010 (2010)
4. Feki, I., Ammar, A.B., Alimi, A.M.: Audio Stream Analysis for Environmental Sound Classification. In: International Conference on Multimedia Computing and Systems (2011)
5. Gorisse, D., Precioso, F., Gosselin, P., Granjon, L., Pellerin, D., Rombaut, M., Bredin, H., Koenig, L., Lachambre, H., Khoury, E.E., Vieux, R., Mansencal, B., Zhou, Y., Benois-Pineau, J., Jégou, H., Ayache, S., Safadi, B., Quénot, G., Benoît, A., Lambert, P.: IRIM at TRECVID 2010: Semantic Indexing and Instance Search. In: TREC Video Retrieval Evaluation Workshop, TRECVid 2010 (2010)
6. Hauptmann, A., Yan, R., Lin, W.H.: How Many High-Level Concepts Will Fill the Semantic Gap in News Video Retrieval? In: International Conference on Image and Video Retrieval, pp. 627–634. ACM, New York (2007)
7. Inoue, N., Saito, T., Shinoda, K., Furui, S.: High-Level Feature Extraction Using SIFT GMMs and Audio Models. In: 20th International Conference on Pattern Recognition, pp. 3220–3223. IEEE (2010)
8. Inoue, N., Wada, T., Kamishima, Y., Shinoda, K., Kim, I., Byun, B., Lee, C.H.: TT+GT at TRECVID 2010 Workshop. In: TREC Video Retrieval Evaluation Workshop, TRECVid 2010 (2010)
9. Jiang, W., Cotton, C., Chang, S.F., Ellis, D., Loui, A.: Short-Term Audio-Visual Atoms for Generic Video Concept Classification. In: 17th ACM International Conference on Multimedia, pp. 5–14. ACM Press, New York (2009)
10. Jiang, Y.G., Ngo, C.W., Yang, J.: Towards Optimal Bag-of-Features for Object Categorization and Semantic Video Retrieval. In: International Conference on Image and Video Retrieval, pp. 494–501. ACM, New York (2007)
11. Jiang, Y.G., Yang, J., Ngo, C.W., Hauptmann, A.G.: Representations of Keypoint-Based Semantic Concept Detection: A Comprehensive Study. IEEE Transactions on Multimedia 12, 42–53 (2010)

12. Jiang, Y.G., Zeng, X., Ye, G., Bhattacharya, S., Ellis, D., Shah, M., Chang, S.F.: Columbia-UCF TRECVID 2010 Multimedia Event Detection: Combining Multiple Modalities, Contextual Concepts, and Temporal Matching. In: TREC Video Retrieval Evaluation Workshop, TRECVid 2010 (2010)
13. Joachims, T.: Text Categorization With Support Vector Machines: Learning With Many Relevant Features. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 137–142. Springer, Heidelberg (1998)
14. Li, H., Bao, L., Gao, Z., Overwijk, A., Liu, W., Zhang, L.F., Shoou-I, Y., Chen, M.Y., Florian, M., Hauptmann, A.: Informedia @ TRECVID 2010. In: TREC Video Retrieval Evaluation Workshop, TRECVid 2010 (2010)
15. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
16. Lu, L., Hanjalic, A.: Audio Keywords Discovery for Text-Like Audio Content Analysis and Retrieval. IEEE Transactions on Multimedia 10(1), 74–85 (2008)
17. Mallat, S., Zhang, Z.: Matching Pursuits With Time-Frequency Dictionaries. IEEE Transactions on Signal Processing 41(12), 3397–3415 (1993)
18. Peng, Y., Lu, Z., Xiao, J.: Semantic Concept Annotation Based on Audio PLSA Model. In: 17th ACM International Conference on Multimedia (MM 2009), pp. 841–844. ACM Press, New York (2009)
19. Riley, M., Heinen, E., Ghosh, J.: A Text Retrieval Approach to Content-based Audio Retrieval. In: 9th International Conference of Music Information Retrieval, pp. 295–300 (2008)
20. Smeaton, A.F., Over, P., Kraaij, W.: Evaluation Campaigns and TRECVid. In: 8th ACM International Workshop on Multimedia Information Retrieval, pp. 321–330. ACM Press, New York (2006)
21. Snoek, C.G.M., van de Sande, K.E.A., Rooij, O.D., Huurnink, B., Uijlings, J.R.R., Liempt, M.V., Bugalho, M., Trancoso, I., Yan, F., Tahir, M.A., Mikolajczyk, K., Kittler, J., de Rijke, M., Geusebroek, J.M., Gevers, T., Worring, M., Smeulders, A.W.M., Koelma, D.C.: The MediaMill TRECVID 2009 Semantic Video Search Engine. In: TREC Video Retrieval Evaluation Workshop, TRECVid 2009 (2009)
22. Snoek, C.G.M., Worring, M., van Gemert, J.C., Geusebroek, J.M., Smeulders, A.W.M.: The Challenge Problem for Automated Detection of 101 Semantic Concepts in Multimedia. In: ACM International Conference on Multimedia, pp. 421–430. ACM, New York (2006)
23. Sonnenburg, S., Rätsch, G., Henschel, S., Widmer, C., Behr, J., Zien, A., Bona, F., Binder, A., Gehl, C., Franc, V.: The SHOGUN Machine Learning Toolbox. Journal of Machine Learning Research 99, 1799–1802 (2010)
24. Vedaldi, A., Fulkerson, B.: VLFeat: An Open and Portable Library of Computer Vision Algorithms (2008), http://www.vlfeat.org/

Content-Based Video Description for Automatic Video Genre Categorization

Bogdan Ionescu[1,3], Klaus Seyerlehner[2], Christoph Rasche[1],
Constantin Vertan[1], and Patrick Lambert[3]

[1] LAPI, University "Politehnica" of Bucharest 061071 Bucharest, Romania
{bionescu,rasche,cvertan}@alpha.imag.pub.ro
[2] DCP, Johannes Kepler University, A-4040 Linz, Austria
klaus.seyerlehner@jku.at
[3] LISTIC, University of Savoie, BP 80439, 74944 Annecy-le-Vieux Cedex, France
patrick.lambert@univ-savoie.fr

Abstract. In this paper, we propose an audio-visual approach to video genre categorization. Audio information is extracted at block-level, which has the advantage of capturing local temporal information. At temporal structural level, we asses action contents with respect to human perception. Further, color perception is quantified with statistics of color distribution, elementary hues, color properties and relationship of color. The last category of descriptors determines statistics of contour geometry. An extensive evaluation of this multi-modal approach based on on more than 91 hours of video footage is presented. We obtain average precision and recall ratios within [87% − 100%] and [77% − 100%], respectively, while average correct classification is up to 97%. Additionally, movies displayed according to feature-based coordinates in a virtual 3D browsing environment tend to regroup with respect to genre, which has potential application with real content-based browsing systems.

Keywords: video genre classification, block-level audio features, action segmentation, color perception, contour geometry, video indexing.

1 Introduction

The automatic labeling of video footage according to genre is a common requirement when dealing with indexing of large and heterogenous collection of video materials. This task may be addressed, either *globally*, or *locally*. Global-level approaches aim at classifying videos into one of several main genres, e.g. cartoons, music, news, sports, documentaries, etc.; or even more fine-grained into sub-genres, e.g. identifying specific types of sports (football, hockey, etc.), movies (drama, thriller, etc.), and so on. On the other hand, with local-level approaches video segments are labeled according to specific human-like concepts, e.g. outdoor vs. indoor scenes, action segments, violence scenes, etc. (see TRECVid campaign [1]). In this paper we focus on the global classification task only and video genre classification is consequently interpreted as a typical machine learning problem that involves two fundamental steps: *feature extraction* and *data*

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 51–62, 2012.

classification. Especially the choice of a suitable task specific feature set is critical for the success of such a classification approach and an ideal feature set should contain as many genre specific cues as possible. In the literature so far, various sources of information have been exploited [2]. Some sources of information may provide more informative cues than others, like for instance visual elements compared to text or even some audio descriptors. The most reliable approaches (which also target the wider range of genres) are however *multi-modal*, i.e. multi-source.

In the following we shall highlight the performance of several approaches we consider relevant for the present work. A simple, single modal approach is the one proposed in [3]. It addresses the genre classification task using only video dynamics. Motion information is extracted at two levels: background camera motion and foreground or object motion. A single feature vector is constituted in the DCT transformed space. This is to assure low-pass filtering, orthogonality and a reduced feature dimension. A Gaussian Mixture Model (GMM) based classifier is then used to identify 3 common genres: sports, cartoons and news. Despite the limited content information used, when applied to a reduced number of genres, it is able to achieve detection errors below 6%.

A much more complex approach which uses spatio-temporal information is proposed in [4]. At temporal level, video contents is described using average shot length, cut percentage, average color difference and camera motion (4 cases are detected: still, pan, zoom, and other movements). Spatial features include face frames ratio, average brightness and color entropy. The genre classification task is addressed at different levels, according to a hierarchical ontology of video genres. Several classification schemes (decision trees and several SVM approaches) are used to classify video footage into main genres: movie, commercial, news, music and sports; and into sub-genres, movies into action, comedy, horror and cartoon, and finally sports into baseball, football, volleyball, tennis, basketball and soccer. The highest precision for video footage categorization is around 88.6%, while for sub-genres, sports categorization achieve 97% and movies up to 81.3%.

A truly multi-modal approach, which combines several categories of content descriptors, is proposed in [5]. Features are extracted from four informative sources, which include visual-perceptual information (color, texture and motion), structural information (shot length, shot distribution, shot rhythm, shot clusters duration and saturation), cognitive information (face properties, such as number, positions and dimensions) and aural information (transcribed text, sound characteristics). These features are used for training a parallel Neural Network system and achieve an accuracy rate up to 95% in distinguish between seven video genres, namely: football, cartoons, music, weather forecast, newscast, talk shows and commercials.

In our approach, we exploit for genre classification both audio and visual modalities. The proposed set of *audio features* are block-level based, which compared to classic approaches, e.g. Mel-Frequency Cepstral Coefficients - MFCC [6], have the advantage of capturing local temporal information by analyzing sequences of consecutive frames in a time-frequency representation. On the other hand, *visual information* is described with temporal information, color and contour geometry. Temporal descriptors are first derived using a classic confirmed approach, i.e. analyzing the frequency of shot changes [4]. However, the novelty is in the way we measure action content, which is based on the assessment of action perception. Color information is extracted globally. Compared to most of the existing approaches which use mainly local or low-level descriptors, e.g. predominant color, color variance, color entropy, frame based histograms [2], the novelty of our approach is in the analysis of color perception. Using a color naming system, color perception is quantified in terms of statistics of color distribution, elementary hues distribution, color properties (e.g. amount of light colors, cold colors, saturated colors, etc.) and relationship of color. The final visual descriptors are related to contour information, which was hardly exploited with genre classification [2]. Instead of describing closed region shapes, as most of the existing approaches do, e.g. MPEG-7 visual descriptors [7], we broke contours into segments and describe curve contour geometry, individually and in relation with neighbor contours.

The main contribution of our work is however the combination of the proposed descriptors, which together form a highly descriptive feature set that is especially well-suited for video genre classification. The remainder of the paper is organized as follows: Section 2, Section 3, Section 4 and Section 5 deal with feature extraction: audio, temporal, color and contour, respectively. Experimental results are presented in Section 6 while Section 7 presents the conclusions and discuses future work.

2 Audio Descriptors

Audio information is an important cue when addressing automatic genre classification. Most of the common video genres have very specific audio signatures, e.g. music clips contain music, in news there are a lot of monologues/dialogues, documentaries have a mixture of natural sounds, speech and ambience music, in sports there is the specific crowd noise, etc.

To address this specificity we propose audio descriptors which are related to rhythm, timbre, onset strength, noisiness and vocal aspects [8]. The proposed set of audio descriptors, called block-level audio features, have the key advantage of capturing also local temporal information from the audio track. Temporal integration is realized by analyzing sequences of consecutive frames *called blocks*, in a time-frequency representation, instead of using single frames only. Blocks are of variable length and can be overlapping (e.g. by 50% of their frames). After converting the video soundtrack into a $22kHz$ mono signal, we compute short-time Fourier transform and perform a mapping of the frequency axis according to the logarithmic cent-scale to account for the logarithmic human frequency perception. Then, the following complex audio features are derived:

Spectral Pattern (SP)**:** characterize the soundtrack's timbre via modeling those frequency components that are simultaneously active. Dynamic aspect of the signal are kept by sorting each frequency band of the block along the time axis. The block width varies depending on the extracted patterns, which allows to capture temporal information over different time spans.

Delta Spectral Pattern (DSP)**:** captures the strength of onsets. To emphasize onsets, first the difference between the original spectrum and a copy of the original spectrum delayed by 3 frames is computed. Then, each frequency band is sorted along the time axis similar to the spectral pattern.

Variance Delta Spectral Pattern ($VDSP$)**:** is basically an extension of the delta spectral pattern and captures the variation of the onset strength over time.

Logarithmic Fluctuation Pattern (LFP)**:** captures the rhythmic aspects of the audio signal. In order to extract the amplitude modulations out of the temporal envelope in each band, periodicities are detected by computing the FFT along each frequency band of a block.

Spectral Contrast Pattern (SCP)**:** roughly estimates the *"tone-ness"* of an audio track. For each frame, within a block, the difference between spectral peaks and valleys in 20 sub-bands is computed and the resulting spectral contrast values are sorted along the time axis in each frequency band.

Correlation Pattern (CP)**:** To capture the temporal relation of loudness changes over different frequency bands, the correlation coefficient among all possible pairs of frequency bands within a block is used. The resulting correlation matrix forms the so-called correlation pattern.

These audio features in combination with a Support Vector Machine (SVM) classifier constitute a highly efficient automatic music classification system. During the last run of the Music Information Retrieval Evaluation eXchange, this approach ranked first with respect to the task of automatic music genre classification [8]. However, the proposed approach has not yet been applied to automatic video genre classification. Existing approaches are limited to use standard audio features, e.g. a common approach is to use Mel-Frequency Cepstral Coefficients (MFCC) or to compute time domain features, e.g. Root Mean Square of signal energy (RMS), or Zero-Crossing Rate (ZCR) [2] (preliminary tests proved the superiority of the block-based representation over classic MFCC features).

3 Temporal Structure Descriptors

As stated in the Introduction, temporal descriptors are derived using a classic confirmed approach, i.e. analyzing the frequency of shot changes [4]. Compared to existing approaches, we determine the action content based on human perception. Temporal based information is strongly related to movie genre, e.g. commercials and music clips tend to have a high visual tempo, commercials use a lot of gradual transitions, documentaries have a reduced action content, etc.

One of the main success factors of temporal descriptions is an accurate preceding temporal segmentation. To this end we detect both cuts and also gradual transitions. Cuts are detected using an adaptation of the histogram-based approach proposed in [9]. Fades and dissolves are detected using a pixel-level statistical approach [10] and the analysis of fading-in and fading-out pixels (adaptation of [11]), respectively. Then, the temporal descriptors are computed, thus:

Rhythm. To capture the movie's visual changing tempo, first we compute the relative number of shot changes occurring within a time interval $T = 5s$, denoted ζ_T. Then, the rhythm is defined as the movie average shot change ratio, $E\{\zeta_T\}$.

Action. We aim at highlighting two opposite situations: video segments with a high action content (denoted hot action, e.g. fast changes, fast motion, visual effects, etc.) with $\zeta_T > 3.1$, and video segments with low action content (i.e. containing mainly static scenes) with $\zeta_T < 0.6$. Thresholds were determined experimentally. Several persons were asked to manually label video segments into the previous two categories. Based on this ground truth, we determined the average ζ_T intervals for each type of action content. Further, we quantify the action content with two parameters, hot-action ratio (HA) and low-action ratio (LA): $HA = T_{HA}/T_{total}$, $LA = T_{LA}/T_{total}$, where T_{HA} and T_{LA} represent the total length of hot and low action segments, respectively, and T_{total} is the movie total length.

Gradual Transition Ratio. High amounts of gradual transitions are in general related to a specific video contents, therefore we compute: $GT = (T_{dissolves} + T_{fade-in} + T_{fade-out})/T_{total}$, where T_X represents the total duration of all the gradual transitions of type X. This provides information about editing techniques which are specific to certain genres, like movies or artistic animated movies.

4 Color Descriptors

Color information is an important source to describe visual content. Most of the existing color-based genre classification approaches are limited to use intensity-based parameters or generic low-level color features, e.g. average color differences, average brightness, average color entropy, variance of pixel intensity, standard deviation of gray level histograms, percentage of pixels having saturation above a certain threshold, lighting key (measures how well light is distributed), object color and texture, etc. [2].

We propose a more elaborated strategy which addresses the perception of the color content [12]. One simple and efficient way to accomplish this is with the help of color names; associating names with colors allows everyone to create a mental image of a given color or color mixture. We project colors on to a color naming system and colors properties are described using: statistics of color distribution, elementary hue distribution, color visual properties (e.g. amount of light colors, warm colors, saturated colors, etc.) and relationship of color (adjacency and complementarity).

Our strategy is motivated by the fact that different genres have different global color signatures, e.g. animated movies have specific color palettes and color contrasts (light-dark, cold-warm), music videos and movies tend to have darker colors (mainly due to the use of special effects), sports usually show a predominant hue (e.g. green for soccer, white for ice hockey), and so on.

Prior to parameter extraction, we use an error diffusion scheme to project colors into a more manageable color palette, i.e. the non-dithering 216 color Webmaster palette (which provides an efficient color naming system). Further, the proposed color parameters are computed as follows:

Global Weighted Color Histogram is computed as the weighted sum of each shot color histogram, thus: $h_{GW}(c) = \sum_{i=0}^{M} \left[\frac{1}{N_i} \sum_{j=0}^{N_i} h_{shot_i}^{j}(c) \right] \cdot \frac{T_{shot_i}}{T_{total}}$, where M is the total number of video shots, N_i is the total number of the retained frames for the shot i (we use temporal sub-sampling), $h_{shot_i}^{j}$ is the color histogram of the frame j from the shot i, c is a color index from the Webmaster palette (we use color reduction) and T_{shot_i} is the length of the shot i. The longer the shot, the more important the contribution of its histogram to the movie's global histogram.

Elementary Color Histogram. The next feature is the distribution of elementary hues in the sequence, thus: $h_E(c_e) = \sum_{c=0}^{215} h_{GW}(c)|_{Name(c_e) \subset Name(c)}$, where c_e is an elementary color from the Webmaster color dictionary (colors are named according to color hue, saturation and intensity) and $Name()$ returns a color's name from the palette dictionary.

Color Properties. With this feature set we aim at describing color properties. We define several color ratios. For instance, light color ratio, P_{light}, reflects the amount of bright colors in the movie, thus: $P_{light} = \sum h_{GW}(c)|_{W_{light} \subset Name(c)}$, where c is a color with the property that its name contains one of the words defining brightness, i.e. $W_{light} \in \{$"light", "pale", "white"$\}$. Using the same reasoning and keywords specific to each property, we define dark color ratio (P_{dark}), hard saturated color ratio (P_{hard}), weak saturated color ratio (P_{weak}), warm color ratio (P_{warm}) and cold color ratio (P_{cold}). Additionally, we capture movie color wealth with two parameters: color variation, P_{var}, which accounts for the amount of significant different colors and color diversity, P_{div}, defined as the amount of significant different color hues.

Color Relationship. Finally, we compute P_{adj}, the amount of similar perceptual colors in the movie and P_{compl}, the amount of opposite perceptual color pairs.

5 Contour Descriptors

The last category of descriptors provide information based on visual structures, that is on contours and their relations. So far, contour information was only limitedly exploited within genre classification. For instance, some approaches use MPEG-7 inspired contour descriptors [7], e.g. use of texture orientation histograms, edge direction histograms, edge direction coherence, which are highly low-level edge pixel statistics.

Our approach in contrast, proposes a novel method which uses curve partitioning and curve description [13]. The contour description is based on a characterization of geometric attributes for each individual contour, e.g. degree of curvature, angularity, "wiggliness", and so on. These attributes are taken as parameters in a high-dimensional image vector and have been exploited in a (statistical) classification task with good success. For instance, the system has achieved the benchmark in the photo-annotation task of the ImageCLEF competition 2010 where this approach ranks in the upper third of all performances.

Contour Characterization. Contour processing starts with edge detection, which is performed with the Canny edge detection algorithm. For each contour, a type of curvature space is created. This space is then abstracted into spectra-like functions, from which in turn a number of geometric attributes are derived, such as the degree of curvature, angularity, circularity, symmetry, "wiggliness" and so on. In addition to those geometric parameters, a number of "appearance" parameters are extracted. They consist of simple statistics obtained from the luminance values extracted along the contour, such as the contrast (mean, standard deviation; abbreviated c_m, c_s respectively) and the "fuzziness", obtained from the convolution of the image with a blob filter (f_m, f_s, respectively).

Pair Relations. In addition to the attributes for individual contours, we also obtain attributes for pairs of contours which are selected based on spatial proximity (i.e. either their contour endpoints are proximal or in the proximity of the other segment). For each selected pair, a number of geometric attributes are determined such as the angular direction of the pair, denoted γ_p; distance between the proximal contour end points, denoted d_c; distance between the distal contour end points, denoted d_o; distance between the center (middle) point of each segment, denoted d_m; average segment length, denoted l; symmetry of the two segments, denoted y; degree of bendness of each segment, denoted b_1 and b_2; structural biases, abbreviated with $\hat{s}$, that express to what degree the pair alignment is a L feature ($\hat{s}_L$), T feature ($\hat{s}_T$) or a "closed" feature (two curved segments facing each other as '( )', $\hat{s}_{()}$).

The structural information is extracted only from a summary of the movie. In this case, we retain around 1% of each shot frames (uniformly distributed). For each image, contour properties are captured with histograms. To address the temporal dimension, at sequence level, resulting feature vectors are averaged forming so the structure signature of the movie.

6 Experimental Results

To evaluate the descriptive power of the proposed audio-visual content descriptors we have selected seven of the most common video genres, namely: *animated movies*, *commercials*, *documentaries*, *movies*, *music videos*, *news broadcast* and *sports*. The data set consists of 30 sequences for each genre, summing up more then 91 hours of video footage. Video materials were retrieved from several TV programmes, thus: 20h30min of animated movies (long, short clips and series, sources: Folimage - France, Disney, Pixar and DreamWorks animation companies); 15min of commercials (source 1980th TV commercials and David Lynch clips); 22h of documentaries (wildlife, ocean, cities and history, source BBC, IMAX, Discovery Channel); 21h57min of movies (long, episodes and sitcom, e.g. Friends, X-Files, Sex and the City series); 2h30min of music (pop, rock and dance video clips, source MTV Channel); 22h of news broadcast (source TVR Romanian National Television Channel); 1h55min of sports (various clips from the Internet). Prior to analysis, a basic normalization is adopted by converting all sequences to a reference video format.

For our classification experiments we have selected three binary classifiers, namely: K-Nearest Neighbors (KNN, with k=1, cosine distance and majority rule), Support Vector Machines (SVM, with a linear kernel) and Linear Discriminant Analysis (use PCA to reduce dimensionality). Method parameters were tuned based on preliminary experimentations. All evaluations are conducted using a cross-validation approach, i.e. generating all possible combinations between training and test data. Additionally, we vary the amount of training data (from 10% to 70%) and test different combination of descriptors.

To assess performance, at genre level we evaluate average precision (P) and recall (R) ratios, thus: $P = \overline{TP}/(\overline{TP} + \overline{FP})$, $R = \overline{TP}/(\overline{TP} + \overline{FN})$, where $\overline{TP}$, $\overline{FP}$ and $\overline{FN}$ represent the *average* number of true positives, false positives and false negatives, respectively, computed over all experimentations for a given amount of training data. As a global measure of performance we compute F_{score} ratio and average correct classification ($\overline{CD}$), thus: $F_{score} = 2 \cdot P \cdot R/(P + R)$, $\overline{CD} = \overline{N_{GD}}/N_{total}$, were $\overline{N_{GD}}$ is the average number of good classifications (in both classes, target and others) and N_{total} is the number of test sequences. Experimental results are presented in the following.

6.1 One Genre at a Time Classification

In Figure 1 we present average precision against recall for different amounts of training data and different descriptor combinations, as well as the overall correct detection $\overline{CD}$ (descriptors are combined based on early fusion). We obtain very promising results considering the content similarity of some of the genres and also compared to the literature (see Section 1). We obtain $P \in [87.5\%; 100\%]$ (from which $P > 95\%$ for music, news, commercials and sports), and $R \in [77.6\%; 100\%]$ (excluding animated movies and commercials, we achieve $R > 95\%$). At global level, the overall correct classification ratio ranges from 92.2% to 97.2% while the highest F_{score} is up to 90.6%. One may observe that the overall performance is high, even for a reduced amount of training data, thus $\overline{CD} > 92\%$ with only 10% of data as training data (i.e. from 189 sequences, in average 174 were correctly assigned to one of the two classes, target genre and others).

The most interesting result is however that each descriptor set highlights relatively different properties of the video contents, as the most efficient approach (both in terms of overall classification performance and genre precision and recall) is the combination of all audio-visual descriptors (i.e. audio-contour-color-action, see SVM results depicted with the red line in Figure 1). Table 1 summarizes the precision and recall in this case (these results are encircled in Figure 1).

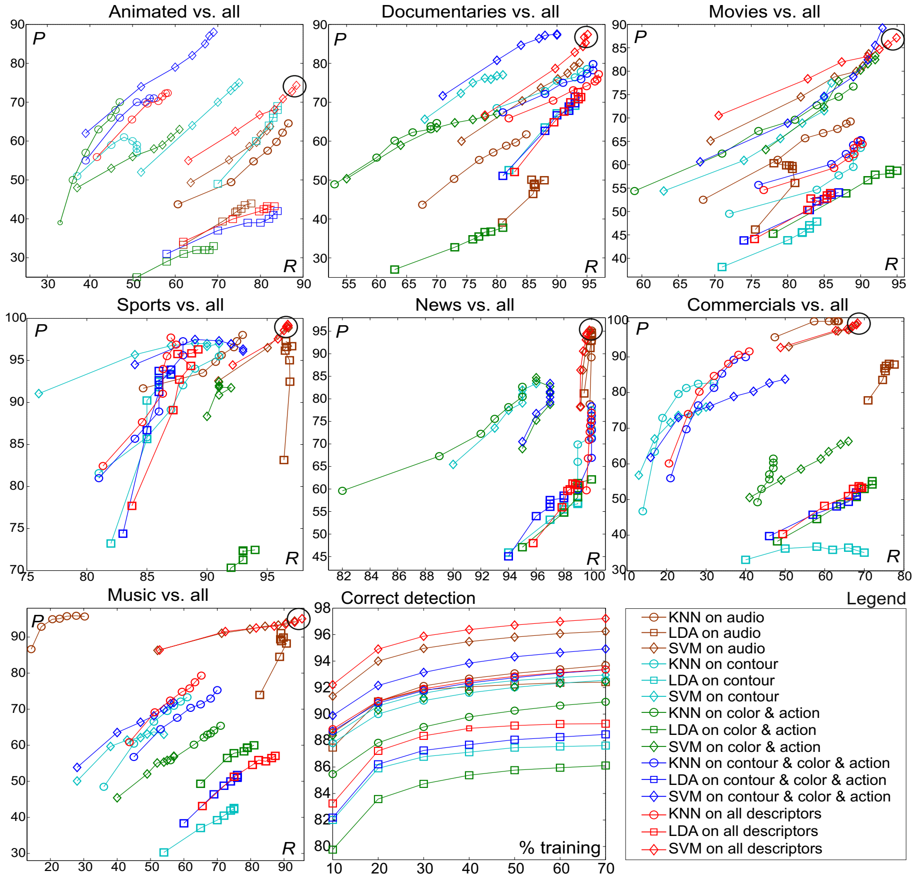


Fig. 1. Precision (P) against recall (R) for different runs and amounts of training data (increases along the curves from 10% to 70%) and overall correct classification ($\overline{CD}$)

Table 1. SVM vs. KNN and LDA (using all audio-visual descriptors)

genre	Precision (P)			Recall (R)		
	SVM	KNN	LDA	SVM	KNN	LDA
animated	**74.3%**	72.3%	43.2%	**88.4%**	58.2%	83.3%
documentaries	**87.4%**	77.2%	72.6%	**95.1%**	96.3%	93.5%
movies	**87.1%**	65%	53.9%	**94.9%**	89.6%	85.8%
music	**95.1%**	79.3%	57%	**95.4%**	65.2%	87.3%
sports	**99.3%**	97.7%	96.3%	**96.7%**	86.9%	89.2%
news	**95.2%**	76.9%	60.8%	**99.8%**	99.9%	99.1%
commercials	**99.5%**	91.5%	53.3%	**68.3%**	40.9%	69.4%

Globally, the lowest accuracy is obtained for animated movies and commercials, which is mainly due to their heterogenous contents and resemblance with other genres, e.g. many commercials include animation, music clips are similar to commercials, movies may contain commercial-like contents, etc. On the other hand, the best performance (as anticipated) is obtained for genres with a certain repetitiveness in content structure, i.e. news and sports (average precision or recall up to 100%).

In what concerns the informational sources, compared to visual information, audio information proves to be highly efficient to this task, alone leading to very good classification ratios (depicted with Maroon in Figure 1). At genre level, audio features are more accurate for classifying music, sports, news and commercials, which have specific audio patterns. On the other hand, contour and color-action information used alone, prove to be less efficient. Contour parameters, compared to color-action parameters, provide better performance for documentaries, sports and news, which have specific signatures, e.g. skyline contours, people silhouettes, etc. (depicted with Cyan in Figure 1). Compared to contours, color-action features perform better for music, commercials, movies and news (which can be assigned to the specific rhythm and color diversity, depicted with Green in Figure 1). Compared to audio, visual descriptors together are more discriminative for animated, movies and documentaries (depicted with Blue in Figure 1). As stated before, the best performance in classifying each individual genre is however achieved when using all audio-visual information.

6.2 Descriptor-Based Visualization

In our final experiment we try to find out whether the proposed features are discriminative enough to provide genre-based separation for real browsing applications. Movies were displayed on a 3D spherical coordinate system according to the first three principal components of the audio-visual descriptors, thus: inclination (θ) - 1st component (normalized in $[0;\pi]$), azimuth (φ) - 2nd component (normalized in $[0;2\pi]$) and radius (r) - 3rd component (normalized in $[0;1]$). Several screenshots taken from different angles are presented in Figure 2 (a demo is available at `http://imag.pub.ro/~bionescu/index_files/MovieGlobe.avi`).

Although, we use only the first three principal components (which account for up to 94% of the initial data variance), one may observe that certain genres are visibly grouping together, which is quite an interesting result. Due to the similarity of the content and structure, the mostly regrouped are the news (see view C) and sports (see view D). Other genres tend to be more "interleaved" (e.g. documentaries, see view B), which is at some point expectable, considering the fact that even for human observer is difficult to draw a sharp delimitation between genres. Nevertheless, sequences with similar contents tend to regroup around a basis partition (see in view A).

Enhanced by genre labeling provided by the SVM classification, this might be a powerful genre-based browsing tool. Even though this experiment proves the potential of our descriptors with real browsing applications, these are however preliminary results and more elaborated tests are to be conducted.

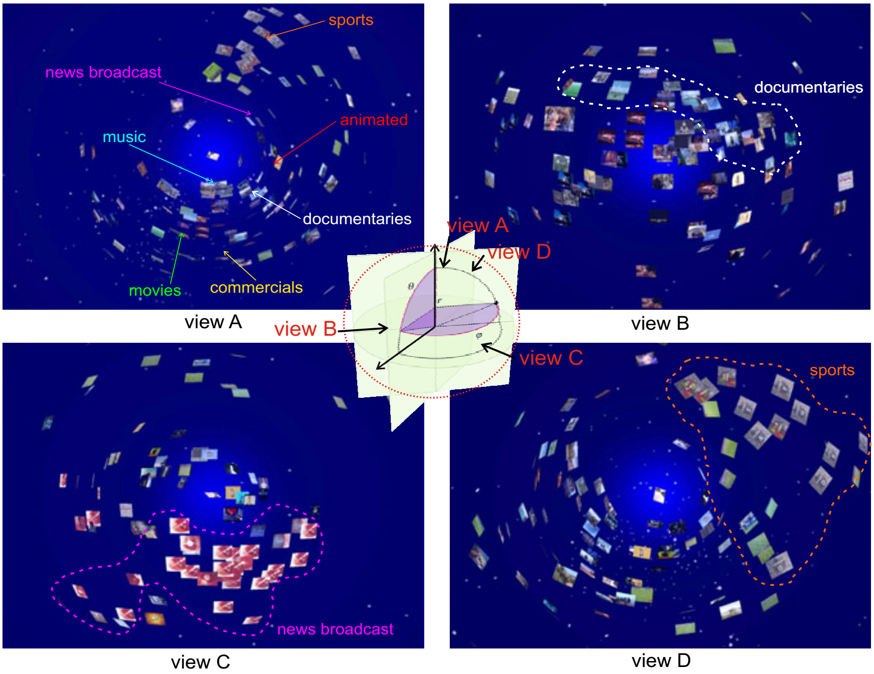


Fig. 2. Feature-based 3D movie representation (each movie is represented with one image). View A to D are screenshots made from different perspectives (the used points of view are synthesized with the system diagram presented in the center).

7 Conclusions

In this paper we addressed the issue of automatic video genre categorization and we have proposed four categories of content descriptors: block-level audio features, temporal structure-based action descriptors, perceptual color descriptors and contour statistics.

Although these sources of information have already been exploited in the literature, the main contribution of our work is the way we compute the content descriptors and the high descriptive power of the combination of these descriptors. An extensive evaluation was performed based on 91 hours of video footage. We achieve average precision and recall ratios within [87%−100%] and [77%−100%], respectively, while average correct classification is up to 97%.

Additionally, preliminary experiments based on a prototypical video browsing system demonstrate the prospective application potential of our approach. Future work will mainly focus on more detailed sub-genre classification and on extending the scope of our work towards web media platforms (e.g. blip.tv, see MediaEval campaign).

Acknowledgments. This work was supported by the Romanian Sectoral Operational Programme Human Resources Development 2007-2013 through the Financial Agreement POSDRU/89/1.5/S/62557 and by the Austrian Science Fund (FWF): L511-N15.

References

1. Smeaton, A.F., Over, P., Kraaij, W.: High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements. In: Multimedia Content Analysis, Theory and Applications, pp. 151–174. Springer, Berlin (2009) ISBN 978-0-387-76567-9
2. Brezeale, D., Cook, D.J.: Automatic Video Classification: A Survey of the Literature. IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews 38(3), 416–430 (2008)
3. Roach, M.J., Mason, J.S.D.: Video Genre Classification using Dynamics. In: IEEE Int. Conf. on Acoustics, Speech and Signal Processing, USA, pp. 1557–1560 (2001)
4. Yuan, X., Lai, W., Mei, T., Hua, X.-S., Wu, X.-Q., Li, S.: Automatic Video Genre Categorization using Hierarchical SVM. In: IEEE Int. Conf. on Image Processing, pp. 2905–2908 (2006)
5. Montagnuolo, M., Messina, A.: Parallel Neural Networks for Multimodal Video Genre Classification. Multim. Tools and Applications 41(1), 125–159 (2009)
6. Wang, H., Divakaran, A., Vetro, A., Chang, S.-F., Sun, H.: Survey of Compressed-Domain Features used in Audio-Visual Indexing and Analysis. Journal of Visual Communication and Image Representation 14(2), 150–183 (2003)
7. Sikora, T.: The MPEG-7 Visual Standard for Content Description - An Overview. IEEE Trans. on Circ. and Systems for Video Technology 11(6), 696–702 (2001)
8. Seyerlehner, K., Schedl, M., Pohle, T., Knees, P.: Using Block-Level Features for Genre Classification, Tag Classification and Music Similarity Estimation. In: 6th Annual Music Information Retrieval Evaluation eXchange (MIREX 2010), Utrecht, Netherlands, August 9-13 (2010)
9. Ionescu, B., Buzuloiu, V., Lambert, P., Coquin, D.: Improved Cut Detection for the Segmentation of Animation Movies. In: IEEE Int. Conf. on Acoustic, Speech and Signal Processing, Toulouse, France (2006)
10. Fernando, W.A.C., Canagarajah, C.N., Bull, D.R.: Fade and Dissolve Detection in Uncompressed and Compressed Video Sequence. In: IEEE Int. Conf. on Image Processing, Kobe, Japan, pp. 299–303 (1999)
11. Ionescu, B., Buzuloiu, V., Lambert, P., Coquin, D.: Dissolve Detection in Abstract Video Contents. In: IEEE Int. Conf. on Acoustic, Speech and Signal Processing, Prague, Czech Republic (2011)
12. Ionescu, B., Coquin, D., Lambert, P., Buzuloiu, V.: A Fuzzy Color-Based Approach for Understanding Animated Movies Content in the Indexing Task. Eurasip Journal on Image and Video Processing (2008), doi:10.1155/2008/849625
13. Rasche, C.: An Approach to the Parameterization of Structure for Fast Categorization. Int. Journal of Computer Vision 87(3), 337–356 (2010)

Towards Category-Based Aesthetic Models of Photographs

Pere Obrador[1], Michele A. Saad[2], Poonam Suryanarayan[3], and Nuria Oliver[1]

[1] Telefonica Research, Barcelona, Spain
pere@tid.es
[2] University of Texas at Austin, Austin, TX 78712
[3] The Pennsylvania State University, PA 16802

Abstract. We present a novel data-driven category-based approach to automatically assess the aesthetic appeal of photographs. In order to tackle this problem, a novel set of image segmentation methods based on *feature contrast* are introduced, such that *luminance*, *sharpness*, *saliency*, *color chroma*, and a measure of *region relevance* are computed to generate different image partitions. Image aesthetic features are computed on these regions (*e.g. sharpness*, *colorfulness*, and a novel set of *light exposure* features). In addition, *color harmony*, image *simplicity*, and a novel set of *image composition* features are measured on the overall image. Support Vector Regression models are generated for each of 7 popular image categories: *animals*, *architecture*, *cityscape*, *floral*, *landscape*, *portraiture* and *seascapes*. These models are analyzed to understand which features have greater influence in each of those categories, and how they perform with respect to a generic state of the art model.

Keywords: Image Analysis, Image Aesthetics, Regression.

1 Introduction

We live in a multimedia-rich world, where the ubiquity of camera-phones and digital cameras, combined with increasingly popular photo-sharing websites and online social networks result in billions of consumer photographs available on the Web, as well as in personal photo repositories. In this scenario, computational approaches to automatically assess the aesthetic value of photographs are becoming increasingly important to, *e.g.* enable novel automatic photo story-telling applications [18], and aesthetics based image re-ranking [14,17].

In the world of photography, the term *aesthetics* refers to the concept of appreciation and judgement of beauty and taste in photographic images, which is generally a subjective measure, highly dependent on image content and personal preferences. There is no universally agreed upon objective measure of aesthetics. However, despite this major challenge, photographic theory [7] proposes a set of rules, regarding composition, exposure, contrast, and color harmony, etc., which seem to generate appealing images for humans in general.

Philosophers have tried for a long time to unify the aesthetic judgements across different categories of objects [11], – *i.e.*, a house vs. a sunset vs. a horse.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 63–76, 2012.

Similarly, different photographic categories, may have common criteria that apply across categories, but each photo category may also have its own intrinsic aesthetic criteria. Consequently, is not a surprise that new aesthetic photo categories are introduced regularly in the aesthetics community in order to understand the criteria that work best for each new category [1].

The main contributions of this paper are four-fold, namely:

(1) Two novel aesthetically meaningful low-level features: (a) a set of exposure features to better represent the luminance histogram, which is one of the critical tools to render a highly aesthetic photograph; and (b) an image edge map based composition feature set;

(2) A new approach to measure low-level features on the image's contrasting regions that are generated using *sharpness*, *chroma*, saliency, *luminance*, and a measure of *region relevance*. In our experiments, we show that this representation increases the discriminative power of the low-level features;

(3) A publicly available image dataset composed of seven image categories, each of them with 300 images rated by at least 5 people on DPChallenge.com;

(4) Seven category-dependent image aesthetics models, one for each category.

The remainder of the paper is organized as follows. In Section 2 we present the related work in the literature. Section 3 contains a description of the image corpus used to evaluate our approach. The features used for aesthetic appeal prediction are presented in Section 4. In Section 5 we describe the machine learning methodology followed to build our models that are presented in Sections 5.1 and 5.2. Results are described in Section 6, and finally we conclude and highlight our lines of future research in Section 7.

2 Prior Art

The field of image aesthetics assessment has recently gained increased attention as a result of the ubiquity of digital visual information and related applications. Datta *et al.* in [4] propose an algorithm for classifying images into one of two aesthetics categories – *high* versus *low*, with an accuracy of 70.12% on a set of images collected from *photo.net*, which contains image aesthetics ratings in the range 1 to 7.

Wong *et al.* in [23] proposed a saliency region extraction method to classify images into 2 classes; professional photos versus snapshots. The approach emphasizes the features extracted from the salient regions of the image. The method achieved a 78.8% classification accuracy –based on an SVM classifier. The experiments were conducted on the same *photo.net* set as in [4].

Li *et al.* in [13] focused on predicting aesthetics quality scores for consumer photographs with faces. The dataset of images with ratings was obtained after conducting a subjective study. The algorithm extracts image features from the face region of an image. The algorithm classifies an image into one of 5 classes of aesthetic ratings using an SVM classification model with 68% accuracy. Additionally, the authors carried out a linear and SVM regression on the collected ground truth, achieving only up to a 25% improvement over random score prediction. Recently, Bhattacharya et *al.* [2], presented a system for photo enhancement, based on a two category image composition aesthetic analysis, *i.e.*, one category is related to outdoor photographic compositions with a single foreground object, whereas the other category relates to landscapes and seascapes that lack a dominant object.

Finally, each of the approaches described above has its shortcomings. [23] only classifies into 2 classes and does not predict a continuous score that correlates with the human judgement of aesthetics quality. [4] presents both results for classification into 2 classes and a polynomial regression, but does not take categories into account. [13] proposed a method only tailored towards images with faces, and hence the approach is not directly generalizable across all photo categories. In addition, the two categories presented in [2] are not very diverse.

In view of all previous work, we turn our attention to understanding the importance of image content in the characterization of image aesthetic appeal. We propose and experimentally validate a novel category-based approach to image aesthetic appeal prediction. In addition, for each category we study in detail the role that different image features play in defining the aesthetic appeal of the images.

3 The Image Corpus

In order to train and evaluate computational models of image aesthetics, a set of labeled images is needed as ground truth [4,14,20,23]. The DPChallenge.com image contest website was chosen as our source of data for the following reasons: 1) It has image categories labeled by the photographers; 2) It provides an image count within each category. It was hence possible to determine what image categories had the largest number of photographs in order to determine which were the most dominant; 3) It has subjective photo score ratings ranging from 1 to 10, where 10 is the highest aesthetics score; 4) Each photo has metadata about how many people voted for it, which can be used as a confidence measure –in our case, we consider images which received at least 5 votes; and 5) it has already been used by other researchers in the literature of image aesthetics [14]. After avoiding visually incoherent categories, the seven categories of interest were, namely, *architecture, animals, cityscape, floral, landscape, portraiture,* and *seascapes.* After visual inspection only 300 images from the *seascapes* category were selected as semantically relevant, and therefore we decided to use 300 images per category in order to train our models[1]. See Fig. 3 for a few examples of each image category.

4 Aesthetic Feature Extraction

The features extracted from each image include: (1) *simplicity* features; (2) *global features* that are computed on the whole image; and (3) a novel approach to

[1] The full 7 categories datasets, along with their ratings are available at $http://mm2.tid.es/categoryBasedAesthetics$

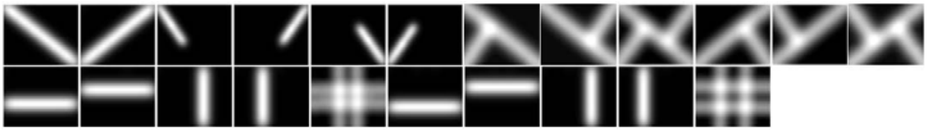


Fig. 1. The composition templates used [20]. The naming convention is T_i, with i starting at 1 top-left, and incrementing left to right, and top to bottom, down to T_{22}.

measure *low-level features in contrasting regions of the image* in order to increase their discriminative power. Note that we avoid using high-level semantic features, such as face pose or expression [13], and also features that require the comparison of the image at hand with the rest of the images in a certain dataset – *i.e.*, the familiarity measure in [4] – and focus on more traditional photographic features [7] like *exposure*, *colorfulness*, *color harmony*, *composition*, and *clarity* – *i.e.*, *contrast* and *sharpness*.

4.1 Simplicity Features

An important rule in photography is *simplicity*. Simplicity is attained by avoiding distracting objects and clutter that could divert the attention of the observer away from the main subject. In highly aesthetic photos, the main subject tends to be isolated from the rest of the image so that it can be easily segmented out from the background [7]. For instance, in low depth of field images, the main subject is in good focus, *i.e.*, sharp, whereas the rest of the image is out of focus, *i.e.*, blurred; professional photographers also accomplish simplicity by placing their subjects in front of monochromatic backgrounds, etc. We use four measures of simplicity, M_1 to M_4: M_1 is the overall number of regions generated by the efficient graph-based image segmentation algorithm in [6]; M_2 and M_3 are the overall number of segmented regions whose sizes are larger than 5% and 10% of the image size, respectively – intuitively, an image that is segmented into many distinct regions cannot be a simple image; finally, M_4 is given by the background's homogeneity. M_4 is calculated using the approach presented in [19], where a well isolated subject from the background yields $M_4 << 1$.

4.2 Global Features

A total of 38 global low-level features are extracted from each image:

(1) Three luminance features: the average ($\overline{L}$), the minimum (L^m), and the maximum luminance (L^M) measures;

(2) The image root mean square contrast (N_1) as in [21];

(3) Five measures of colorfulness, namely, the distance to the neutral axis (μ_{ab}, a.k.a. color *chroma*) of the centroid of the pixel cloud in the color plane of CIELab color space (C_1); $\sigma_{ab} = \sqrt{\sigma_a^2 + \sigma_b^2}$, with σ_a and σ_b being the standard deviation of that cloud on each of the color axes (C_2); the D parameter as defined

in [8] (C_3); and $\widehat{M}^{(1)}$ as defined in [10] (C_4), where both (C_3) and (C_4) are linear combinations of μ_{ab} and σ_{ab}; And a novel colorfulness measure (C_5) set to 1 for $\sigma_{ab} < 10$, and set to 2 for $\sigma_{ab} > 20$, and linear with σ_{ab} for $10 \leq \sigma_{ab} \leq 20$;

(4) Seven color harmony features, which have been found to be pleasing to the eye, and are described by their relative positions around the color wheel (Hue coordinate, in HSV color space). We compute these features as in [16], where the normalized Hue-histogram of the image is convolved with each of 7 templates. We do however keep the 7 results, one for each template, in order to enrich our model. Our feature H_1 corresponds to template i, H_2 to V, H_3 to L, H_4 to I, H_5 to T, H_6 to Y, and H_7 to X, where the template definitions are the same as the ones proposed in [3].

(5) Image composition features that are extracted using the templates presented in [20], which include the *rule of thirds*, the *golden mean* and the *golden triangles*, along with each of its individual segments, see Fig. 1. Contrary to [20], instead of detecting the centroid of the image regions on those templates, we intersect the image edge map – calculated as in [15] – with each of the templates and extract the percentage of edge energy that they capture – $T_1 - T_{22}$. In early experiments, we found that these edge-based composition features provided better discriminative power than the region centroids.

4.3 Contrasting Region Features

Another important factor that contributes to creating an interesting and aesthetic photograph is the contrast or tension between features in different regions of the image [7]. For instance, the *Chiaroscuro* [5] photographic style is characterized by strong contrasts between bright and dark. We hypothesize that low-level features calculated on each of these regions may help discriminate aesthetics in a different way, depending on which of the regions they were calculated on – *i.e.*, high levels of sharpness in the bright region of the image may be more discriminative than high levels of sharpness in the dark region of the image.

Luo et *al.* [14] made a first attempt at capturing features in contrasting sharpness regions. They captured the ratio of luminance and the ratio of clarity between the sharp region – *i.e.*, the subject – and the regions that were not sharp. In [23], Wong et *al.* calculate a set of global features, some of which (exposure, saturation, hue, blurriness, and texture details) are also computed for both the salient – *i.e.*, foreground – and background regions. They use global features in their approach, as well as the squared differences between the features measured in the background and in the foreground. In our models, instead, we keep the feature values for each of the contrasting regions. We now proceed to describe how the *contrasting regions* are calculated, followed by a description of the low-level features that are extracted from each *contrasting region.*

Contrasting Regions. We analyze five different types of features that can generate contrasting regions in a photograph: 1) sharpness or focus (F) – sharp vs. non-sharp region; 2) luminance (L) – bright vs. dark region; 3) chroma (C)

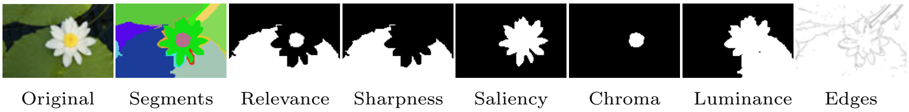


Fig. 2. *Spring lilly pads*, by jseyerle; its segmentation map; its 5 different contrasting regions binary maps –each with two regions, with white being the regions above the threshold, and black the regions below the threshold – and its edge map

– colorful vs. non-colorful region; 4) relevance (R) – relevant vs. non-relevant region; and 5) saliency (S) – salient vs. non-salient image region.

First, a thumbnail version of the image – whose largest side is 128 pixels – is segmented using the algorithm in [6]. After segmentation, each segmented region is assigned a value for each of the contrasting features:

1. *Sharpness* F: The maximum sharpness value within the segmented region, as defined in [17];
2. *Luminance* L: The average luminance, in CIELab color space, over the segmented region;
3. *Chroma* C: μ_{ab}, as described in Section 4.2, calculated in the segmented region;
4. *Relevance* R: The average relevance of the segmented region, similar to the *region appeal* measure defined in [17]. This is a sharpness dependent, linear combination of the maximum sharpness in the segmented region, its contrast and its μ_{ab}.
5. *Saliency* S: The maximum saliency value, which is obtained by extracting the saliency map of the image, as defined in [9]; then performing a thinning operation with a circle structuring element of diameter 21 pixels; finally the maximum of the resulting map in each segmented region is selected.

Once these segmented region-based maps are generated, they are thresholded to yield 5 binary maps – see Fig. 2, composed of a total of 10 regions, *i.e.*, 5 above and 5 below the threshold. The threshold is set to one half the maximum level of each specific contrasting feature.

The contrasting region identifiers, FH, LH, CH, RH, and SH, for the contrasting regions that are above their corresponding threshold, and FB, LB, CB, RB, and SB, for the contrasting regions that are below the threshold, will be used as subscripts of each low-level feature to be calculated on those regions, as explained next.

Features on the Contrasting Regions. A set of low-level features is computed on each of the ten contrasting regions.

1. *Sharpness*, f: Weighted average of maximum sharpness.
2. *Exposure*, $\bar{l}$, l^{σ}, l^{Q0}, l^{Q1}, l^{Q2}, l^{Q3}, l^{Q4}: Photographic theory explains that the real scene brightness should be rendered in a realistic manner in the final photograph. This means that, for certain categories, a specific luminance

distribution in one of the contrasting regions may be due to a good rendition resulting in a highly aesthetic photo, or vice-versa – *i.e.*, due to over or under exposure situations. We therefore represent the luminance distribution in each region by a set of 7 statistical values: mean ($\bar{l}$), standard deviation (l^{σ}, a measure of contrast), minimum (l^{Q0}), 1^{st} (l^{Q1}), 2^{nd} (l^{Q2}) and 3^{rd} (l^{Q3}), quartiles, and finally, maximum (l^{Q4}).

3. *Chroma*, c: Weighted average of μ_{ab}.
4. *Saliency*, s: Weighted average of maximum saliency.

where the averages of features f, c and s are weighted by the sizes of the segmented regions that conform each of the contrasting region.

In the following, we shall refer to each low-level feature with the subscript of the contrasting region where it has been computed. For instance, the low-level feature *sharpness* (f) calculated in the *luminance* (L) contrasting region above (H) the threshold will be denoted as f_{LH}.

After this process, a total of 100 contrasting region-based features are obtained. However, note that $L^M = l^{Q4}_{LH}$ –max luminance– and $L^m = l^{Q0}_{LB}$ –min luminance, yielding a total of 98 contrasting region-based features. Overall, we compute 140 features on each image: 4 simplicity features, 38 global features and 98 contrasting region-based features. The most discriminative features of this pool of 140 will be automatically selected both for the generic and the category-dependent aesthetics models, as it is described in the following section.

5 Aesthetics Models

We proceed to extract the 140 features from each of the images in our image corpus. The feature extraction is followed by feature selection in order to build the category-based aesthetic models, as well as the *generic* model.

Each of the experiments reported in this paper was done using a regression Support Vector Machine, and each result was averaged over 50 cross-validation runs. The cost (ν, where $\nu \in (0, 1)$) and the tube width or insensitivity (ϵ) parameter were optimized in our experiments; and, finally, the `ksvm()` [12] function that we use selects a near optimal γ hyper-parameter.

We use a filter and wrapper-based approach for feature selection, similar to [4]. For each of the models and for each individual feature, we obtain first the 5-fold cross validation mean squared error (MSE). We keep the top 50% performing individual features – *i.e.*, 70 features – discarding the rest. We then pick the top performing feature out of those 70, followed by the feature that predicts scores the best in conjunction with the feature that was picked in the iteration before it, and so on. All of these experiments show a similar MSE pattern, with a global minimum between 9 and 34 features, see Table 1. The set of features corresponding to that global minimum, for each category, is the one that is finally selected to build each model.

The models that result from these experiments are described next, with a brief discussion of the most discriminative features for each category.

Table 1. Performance of our aesthetics models on each categories: variance of the ratings on the entire category dataset; 5-way cross validation MSE ± standard deviation over the 50 runs on the category dataset; % reduction of the 5CV MSE over the variance for the category dataset; improvement over the performance of the generic model –*i.e.*, 8%; optimal ν and ϵ ksvm parameters; number of selected features that reach the 5CV MSE minimum; number of features to reach 80% reduction on the 5CV MSE over what the first selected feature – *i.e.*, the most discriminative one – obtains

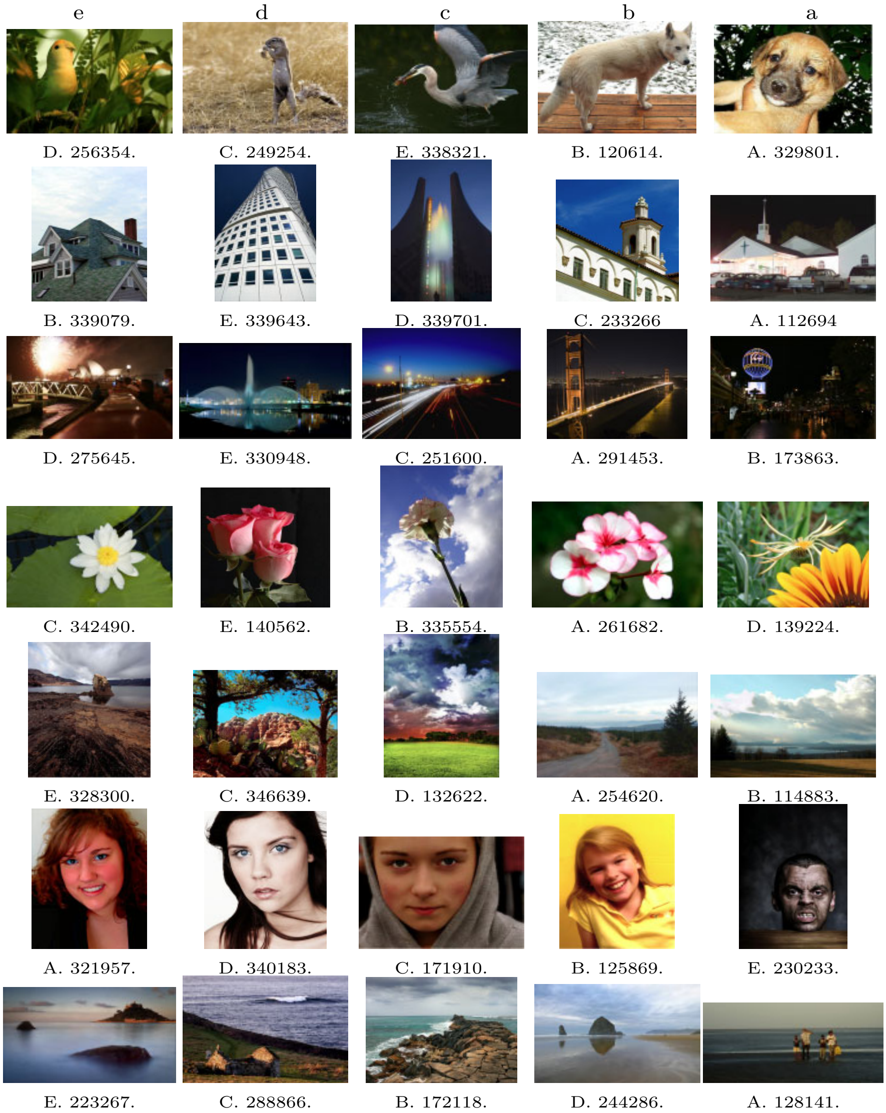
Category	σ^2	5CV	% Red.	Improv.	ν	ϵ	#Feat.	#F. 80%
Animals	0.50	0.42 ± 2.2%	16.2%	102%	0.4	0.1	22	11
Architecture	0.38	0.32 ± 2.1%	14.9%	86%	0.6	0.7	24	13
Cityscape	0.50	0.39 ± 2.2%	22.0%	175%	0.5	0.5	14	7
Floral	0.38	0.34 ± 2.0%	10.5%	31%	0.3	0.1	6	6
Landscape	0.50	0.38 ± 2.3%	24.4%	205%	0.3	0.5	28	19
Portraiture	0.58	0.51 ± 1.8%	12.0%	50%	0.3	0.7	9	7
Seascapes	0.55	0.42 ± 2.4%	24.0%	200%	0.7	0.3	17	9
Generic	0.49	0.45 ± 0.6%	8.0%	-	0.4	0.3	34	14

5.1 Category-Based Aesthetic Models

The models described in this section have been generated with all the images from each data set – *i.e.*, 300 images for each category, and 7×300 for the generic model. In this discussion, we will use the term *clarity* features to include both sharpness (f) and contrast (N and l^{σ}) features, since it helps to abstract the *clarity* high-level feature that humans perceive [7]. In the following sections we list the features that accomplish an 80% reduction over the MSE obtained by the most discriminative feature alone for each of the specific models –see Table 1. Fig. 3 shows a ranking example with the presented models for a test set, with respect to their ground truth ranking.

Animals Category. The *animals* category-based model is composed of 22 features, which reduces the ratings' variance by 16.2%. The top 11 features in the animals category, ordered by importance, are: C_5, c_{LB}, c_{RB}, l^{Q0}_{CB}, c_{CH}, $\bar{l}_{SB}$, N_1, l^{Q0}_{CH}, $\bar{l}_{LH}$, T_{13}, l^{Q4}_{LB}. Notice the strong influence that *chroma* features have in this model (top three features, and a total of 4 out of the 11, *i.e.*, $\frac{4}{11}$), both globally and in contrasting regions – dark region and non-relevant region are usually background regions. Out of the 5 selected luminance features, two of them are on the contrasting chroma regions (average and minimum luminance). One clarity feature and one composition feature are also selected. Finally, eight out of the 11 features are calculated on the contrasting regions, being the chroma and luminance the most discriminative contrasting regions.

Architecture Category. The *architecture* category-based model is composed of 24 features, which reduces the ratings' variance by 14.9%. The top 13 features in this category are, ordered by importance: f_{SH}, M_4, l^{Q4}_{SH}, T_5, l^{σ}_{CB}, l^{Q0}_{FB}, M_3, f_{FH}, f_{CB}, l^{σ}_{CH}, s_{RH}, C_1, and l^{Q0}_{CB}. In this category, we find a large dominance



D. 256354. C. 249254. E. 338321. B. 120614. A. 329801.

B. 339079. E. 339643. D. 339701. C. 233266 A. 112694

D. 275645. E. 330948. C. 251600. A. 291453. B. 173863.

C. 342490. E. 140562. B. 335554. A. 261682. D. 139224.

E. 328300. C. 346639. D. 132622. A. 254620. B. 114883.

A. 321957. D. 340183. C. 171910. B. 125869. E. 230233.

E. 223267. C. 288866. B. 172118. D. 244286. A. 128141.

Fig. 3. Categories in rows ordered as in Table 1. Images ranked by our models from left (e –highest) to right (a –lowest aesthetics). Images sampled at each of the quartiles of aesthetic ground truth, out of a test set of 40 images –created using stratified sampling. Each image shows its ground truth aesthetic ranking (E –highest), and their DPChallenge ID, useful to access them online at *http* : *//www.dpchallenge.com/image.php?IMAGE_ID* = 123456, for an ID=123456. The photographers are, in order: notesinstones, suemack, Saker, camelotnorth, sandeepsalwan45, NathanWert, birkin, DemonLlama, Germaine, rayg544, jaysvette, Marjo, Femme du monde, floydrowe, jfaulkner, jseyerle, rscorp, tfarrell23, rRishinicolai, Terramar, arngrimur, Artifacts, nico_blue, barka, camelotnorth, tfarrell23, LalliSig, bobdaveant, Bkerr, heida, Falc, mexico, Bran-O-Rama, janruss, saffronism.

of clarity features ($\frac{6}{13}$) – the first feature is the sharpness inside the salient region, which is usually the photographed building itself. Simplicity features are also important ($\frac{2}{13}$) – the second feature is the background homogeneity which points out the importance of having the building well isolated from the background; followed by luminance features ($\frac{3}{13}$) – the third feature is the maximum luminance inside the salient region. One composition and one chroma feature are also picked. Nine out of 13 features are calculated on the contrasting regions, being the saliency contrasting regions the most discriminative, followed by the chroma and sharpness contrasting regions.

Cityscape Category. The *cityscape* category-based model is composed of 14 features, which reduces the ratings' variance by 22%. The top 7 features are, ordered by importance: l^{Q2}_{SB}, f_{LH}, L^m, T_1, l^{Q3}_{SH}, f_{FH}, and l^{Q2}_{SB}. Luminance features are predominant in this category ($\frac{4}{7}$) – the first feature is the median of the luminance in the low saliency region, *i.e.*, the background of the cityscape itself. Next in importance are *clarity* features ($\frac{2}{7}$) – the second feature is the sharpness in the high luminance region, which in most images it turns out to be areas of the cityscape itself, since a large percentage of the photos are taken at twilight, or against a dark blue sky. One composition feature is also picked. Five out of 7 features are calculated on the contrasting regions, being the saliency contrasting region the most discriminative.

Floral Category. The *floral* category-based model is composed of 6 features, which reduces the ratings' variance by 10.5%. The 6 features, ordered by importance in the model are: T_{16}, l^{Q0}_{LH}, M_2, T_6, l^{Q4}_{CB} and s_{FB}. In this category, edge-based composition features are very important ($\frac{2}{6}$) – the first feature is the left vertical segment template for the *golden mean*, which implies a framing preference for positioning of flowers or stems; followed by luminance ($\frac{2}{6}$) – the second feature is the minimum luminance in the bright region, which, most of the times, it is the flower itself; and finally simplicity ($\frac{1}{6}$) – the third feature is the number of regions larger than 5% of the image size. One saliency feature is also selected. Three out of 6 features are calculated on the contrasting regions, being the luminance and chroma regions the most discriminative.

Landscape Category. The *landscape* category-based model is composed of 28 features, which reduces the ratings' variance by 24.4%. The top 19 features in the landscape category are, ordered by importance: c_{LB}, L^m, l^{Q4}_{CB}, T_{14}, l^{Q0}_{CB}, T_2, l^{σ}_{CB}, T_8, T_4, $\bar{l}_{RB}$, T_7, M_2, l^{σ}_{LH}, l^{Q2}_{CH}, M_3, l^{Q0}_{SH}, M_1, l^{Q0}_{FB}, f_{RB}. The most discriminative feature is the chroma in the dark region, *i.e.*, the non-sky regions usually with grass, rocks or trees. Luminance-based features dominate ($\frac{7}{19}$) – the second feature is the global minimum luminance level, and, actually, the minimum level of luminance has been selected 3 more times for different contrasting regions; the third feature is the maximum luminance level in the non chromatic region. The next features in importance are the composition features ($\frac{5}{19}$) – the fourth feature is the top horizontal segment of the golden mean, which indicates one of the preferred positions for the horizon in landscape photography. Three clarity and three simplicity features are also selected. Ten out of 19 features are calculated on the contrasting regions, being the chroma contrasting regions the most discriminative, followed by relevance.

Portraiture Category. The *portraiture* category-based model is composed of 9 features, which reduces the ratings' variance by 12%. The top 7 features in this category are, ordered by importance: H_1, T_2, l^{Q0}_{FH}, C_1, $\bar{l}_{CH}$, H_4, and l^{Q1}_{SH}. Color harmony turns out to be important in portraits $(\frac{2}{7})$ – the first feature is the analogous color harmony feature, which means that very close hues should be the norm in the portrait. The second feature is one of the diagonals (bottom-left to top-right) of the composition templates, which favors a certain portrait pose. Finally, luminance-related features are the most important $(\frac{3}{7})$ – the third feature is the minimum luminance level in the sharp region, which is usually either the face and hair, or the eyes and teeth in softer focus portraits. Three out of the 7 features were calculated on contrasting regions, being focus the most discriminative.

Seascapes Category. The *seascapes* category-based model is composed of 17 features, which reduces the ratings' variance by 24%. The top 9 features in this category are, ordered by importance: l^{Q1}_{FH}, L^m, N_1, l^{Q0}_{CH}, H_3, l^{σ}_{CH}, s_{RB}, L^M, C_5. Luminance is the dominant feature $(\frac{4}{9})$, being the minimum and first quartile the most discriminative luminance features – the first feature is the first quartile of luminance in the sharp region, which is usually the coast in the image; lower l^{Q1} provides better aesthetic appeal, which points at the fact that the images with sharp white surf are less favored than images with more tranquil waters; the second feature is the global minimum level of luminance. Next in importance are clarity features $(\frac{2}{9})$ with the third feature being the global contrast of the image. Note that the *clash* color scheme H_3 is also selected, which accounts for the blue-green color contrast in tropical beaches. Four out of the 9 features were calculated on contrasting regions, being sharpness and luminance the most discriminative.

5.2 Generic Aesthetics Model

The *generic* model is composed of 34 features, which reduces the ratings' variance by 8%. We list the 14 most discriminative features in order of importance: $\bar{l}_{FH}$, l^{Q2}_{LH}, T_{10}, f_{LB}, L^m, T_{22}, T_{17}, l^{Q3}_{LH}, $\bar{l}_{RH}$, f_{RH}, C_5, l^{Q4}_{FH}, f_{SH}, and T_8. We observe that luminance is the most important feature for the generic model $(\frac{6}{14})$: the most discriminative feature is the average luminance in the sharp region, which in most of the cases is the subject of interest; the second feature is the median luminance in the bright region. Next in importance are the composition features $(\frac{4}{14})$, with the third feature being one of the *golden triangles*, T_{10}, with the intersection of the two segments – a.k.a. *power point* – on the right. Note that the other *golden triangle* template with the *power point* on the right, T_8, is also selected. It is hypothesized that the direction of writing could bias observers towards a certain region of the image, rendering certain *power points* more powerful than others, *i.e.*, it would be culture dependent [22]. This might explain the preference for templates T_{10} and T_8 in this generic model[2]. The other composition templates that are selected are T_{22} and T_{17}, which are the traditional *golden mean*, and *rule of thirds*, with no orientation preference. Finally, sharpness ($\frac{3}{14}$) and chroma ($\frac{1}{14}$) features are also selected. Eight out of the 14 features are calculated on contrasting regions, with the luminance contrasting region being the most discriminative, followed by sharpness.

Since this is the combination of all categories, we expected the selected features to be generic and make good photographic sense, as it turned out to be, *i.e.*, good average exposure on the subject, good sharpness, and generic composition rules. In the following section we compare how well the individual models perform with respect to this generic model in estimating the aesthetic value of the photographs in our dataset.

6 Experimental Results

In order to see whether our generic feature set is competitive with the state-of-the-art, we trained a model using the same approach described in Section 5 on the data set presented in [4]. After performing our feature selection, and optimizing $\nu = 0.45$ and $\epsilon = 0.9$, we ended up with a generic model consisting of 31 features yielding a 5-way cross validation MSE of 0.55. This is comparable to what Datta et *al.* obtained (MSE=0.50) in [4] by using 5 polynomial terms for each of the 56 low-level features they proposed. As an exercise, we trained a generic model on a subset of our entire DPChallenge dataset – 16777 images – that generated consistent results with the generic model trained on the combination of the 7 categories datasets, *i.e.*, var=0.52, 5CV MSE = 0.46.

When we compare the 5-way cross-validation MSE for each of the category-based models with respect to the original variance of the ratings in each data set – the same regression performance measure used in [4] – we see that the category-based models yield significantly better performance than the one obtained by the generic model, with an average of 121% (min=31%, max=205%) improvement over the generic model. See Table 1. This confirms our hypothesis that category-dependent aesthetic models improve the prediction of the aesthetic appeal of a photograph.

In particular *seascapes*, *landscape* and *cityscape* models perform better than the other categories, which might be due to the fact that the background is more predictable –*i.e.*, usually a large patch of sky– and also the variability of main subjects is less profound than for all other categories.

The *floral* category obtains the lowest improvement over the generic model, and we hypothesize that it may be due to the similarity of the features selected in both the floral and generic models, *i.e.*, big focus on composition and exposure features. The *portraiture* category also yields low improvement over the generic

[2] For cultures where writing is from left to right, the eye enters the picture frame through the left side of the picture frame, travels to the center being intersected by the diagonal, which guides the eye to the *power point*.

model. We hypothesize that this poor performance is due to the importance of the emotional message conveyed by the facial expressions in portraits, combined with artistic resources that are hard to capture in our small data set. Moreover, as Li *et al.* presented in [13], specific face features, such as pose, eyes closed/open, etc., would be needed in order to better capture the aesthetics of the images in this category.

7 Conclusions and Future Work

In this paper we have presented a novel category-based approach to automatically estimate image aesthetic appeal. We have created a new image dataset composed of seven categories, each of them with 300 images that have been semantically selected, and rated by at least 5 people on the DPChallenge.com web site. We have made this dataset available to the research community. A new approach to measuring low-level features on contrasting regions of the image, using *sharpness*, *chroma*, *saliency*, *luminance*, and a measure of *region relevance*, has been introduced. We have shown that this contrasting regions approach increases the discriminative power of the low-level features. We have also introduced a new set of low-level features to better represent the luminance histogram, which is one of the critical tools to render a highly aesthetic photograph, and an image edge map based composition framework. In experiments with real images from DPChallenge.com, we have shown how the category-based models improve over the performance of a generic model. Future work would include closing the loop by building a complete system that automatically labels the image into one of the categories – either by analyzing a text query, or by an image recognition algorithm – and performs the aesthetic appeal prediction with the right category-based model. We are aware that the size of each category dataset is relatively small (300), and therefore future work will focus on increasing the size of these datasets. Also, more image category-based datasets would be welcome, some of which might be a combination of the basic categories.

References

1. Benzaquen, S.: Postcolonial aesthetic experiences: thinking aesthetic categories in the face of catastrophe at the beginning of the twenty-first century. In: European Congress of Aesthetics (2010)
2. Bhattacharya, S., Sukthankar, R., Shah, M.: A framework for photo-quality assessment and enhancement based on visual aesthetics. In: Proc. of ACM Multimedia, pp. 271–280 (2010)
3. Cohen-Or, D., Sorkine, O., Gal, R., Leyvand, T., Xu, Y.-Q.: Color harmonization. ACM Transactions on Graphics 25(3), 624–630 (2006)
4. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Studying Aesthetics in Photographic Images Using a Computational Approach. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part III. LNCS, vol. 3953, pp. 288–301. Springer, Heidelberg (2006)
5. Dyer, A.P.: A study of photographic chiaroscuro, M.A. dissertation. University of Northern Colorado (2005)

6. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. International Journal of Computer Vision 59(2), 167–181 (2004)
7. Freeman, M.: The image. revised edition. William Collins Sons & Co Ltd., (1990)
8. Gasparini, F., Schettini, R.: Color balancing of digital photos using simple image statistics. Pattern Recognition 37(6), 1201–1217 (2004)
9. Harel, J., Koch, C., Perona, P.: Graph-based visual saliency. In: NIPS (2006)
10. Hasler, D., Susstrunk, S.: Measuring colourfulness in natural images. SPIE/IS&T Hum. Vis. Elec. Img. 5007, 87–95 (2003)
11. Kant, I.: The critique of judgement. Forgotten Books, forgottenbooks.org (2008)
12. Karatzoglou, A., Smola, A., Hornik, K., Zeileis, A.: Kernlab – an S4 package for kernel methods in R. Journal of Statistical Software 11(9), 1–20 (2004)
13. Li, C., et al.: Aesthetics quality assessment of consumer photos with faces. In: Proceedings of IEEE ICIP, pp. 3221–3224 (2010)
14. Luo, Y., Tang, X.: Photo and Video Quality Evaluation: Focusing on the Subject. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 386–399. Springer, Heidelberg (2008)
15. Meer, P., Georgescu, B.: Edge detection with embedded confidence. Transaction in Pattern Analysis and Machine Intelligence 12(23), 1351–1365 (2001)
16. Moorthy, A.K., Obrador, P., Oliver, N.: Towards Computational Models of the Visual Aesthetic Appeal of Consumer Videos. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 1–14. Springer, Heidelberg (2010)
17. Obrador, P., Anguera, X., de Oliveira, R., Oliver, N.: The role of tags and image aesthetics in social image search. In: Proc. of the SIGMM WSM, pp. 65–72 (2009)
18. Obrador, P., de Oliveira, R., Oliver, N.: Supporting personal photo storytelling for social albums. In: Proc. of ACM Multimedia, pp. 561–570 (2010)
19. Obrador, P., Moroney, N.: Low-level features for image appeal measurement. In: Proceedings of the SPIE, vol. 7242 (2009)
20. Obrador, P., Schmidt-Hackenberg, L., Oliver, N.: The role of image composition in image aesthetics. In: Proc. of IEEE ICIP, pp. 3185–3188 (2010)
21. Peli, E.: Contrast in complex images. Journal of the Optical Society of America 7(10), 2032–2040 (1990)
22. Rice, P.: Professional Techniques for Black & White Digital Photography. Amherst Media, Inc. (2005)
23. Wong, L.K., Low, K.L.: Saliency-enhanced image aesthetics class prediction. In: Proceedings of IEEE ICIP, pp. 997–1000 (2009)

Scene Signatures for Unconstrained News Video Stories

Ehsan Younessian and Deepu Rajan

School of Computer Engineering,
Nanyang Technological University, Singapore
{ehsa0001,asdrajan}@ntu.edu.sg

Abstract. We propose a novel video signature called scene signature which is defined as a collection of SIFT descriptors. A scene signature represents the visual cues from a video scene in a compact and comprehensive manner. We detect Near Duplicate Keyframe clusters within a news story and then for each of them we generate an initial scene signature including most informative mutual and distinctive visual cues. Compared to conventional keypoint-trajectory-based signatures, we take the co-occurrence of SIFT keypoints into account. Moreover, we utilize keypoints describing novel visual clues in the scene. Next, through three steps of refinements on the initial scene signature we shorten the semantic gap to obtain more compact and semantically meaningful scene signatures. The experimental results confirm the efficiency, robustness and uniqueness of our proposed scene signature compared to other global and local video signatures.

Keywords: Scene signature, Near-Duplicate Keyframe, News retrieval.

1 Introduction

The problem of an effective representation of a video sequence is an important one since it has a direct bearing on the performance of several tasks like content-based video retrieval, near duplicate video detection, topic detection and threading and etc. In this paper, we propose a robust and compact video signature to describe news videos. The proposed signature is generated at the scene level as opposed to earlier approaches that were generated at the frame-level (i.e., using individual frames or keyframes) [6,7,8] or at the shot-level [12]. The scene-level video signature would enable much of the semantics to be encoded in the signature, as we shall illustrate.

In this paper, we focus on the news video domain where similar stories share duplicate/near-duplicate or partially near-duplicate visual clues. Unlike datasets used in [6,7,8], the similarity between videos is more challenging in the news video domain. For instance, Figure 1 shows keyframes from the similar news stories broadcast in ABC, NDTV, and CNN channels. We notice a significant difference in the visual cues, even though they address the same topic. In addition the number of keyframes as well as their temporal order is different. They also

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 77–88, 2012.

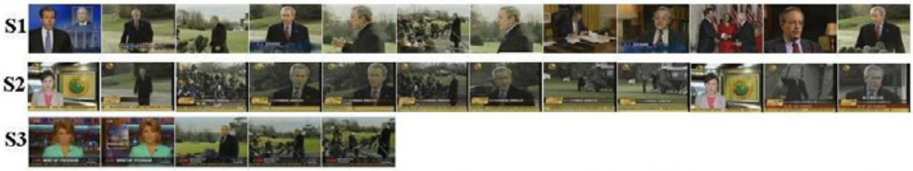


Fig. 1. An example of similar news stories

include dissimilar visual contents irrelevant to the main topic, like anchorwoman and reporter. Such unconstrained news videos also contain significant variations in lighting conditions, object displacements, viewpoints and etc, causing tasks like near-duplicate video detection to fail.

The proposed video signature aims to represent a news story in a compact as well as in a semantically meaningful way. Since the signature is generated at the scene-level, we call it a *scene signature*, abbreviated as SS. Given the keyframes of a news story, the first step in generating a scene signature is to group the keyframes using near duplicate keyframe (NDK) clustering. The SIFT keypoints in each cluster are categorized into connected keypoints and isolated keypoints (i.e., they do not have any matching keypoints in other frames within the cluster). These two categories are analyzed to generate an initial scene signature for each cluster. Through three refinement steps we merge some of the clusters which, in turn, are represented by more compact and discriminative scene signatures.

The rest of this paper is organized as follows. In Section 2 we indicate related work and point out their usability and drawbacks in the context of news story retrieval. In Section 3 we explain our proposed approach to generate the scene signature. In Section 4 we conduct extensive experiments to show the effectiveness, robustness and compactness of the proposed scene signature compared to other video signatures.

2 Related Work

In the video retrieval literature, wide range of approaches have been proposed for duplicate/near-duplicate [4,6,7]/partially near duplicate [5] video detection tasks with the main focus on the accuracy or/and the efficiency of the proposed approach. Two main groups of these methods are signature matching based and sequence matching based approaches.

Signature matching based approaches can be subdivided into the global and local signatures. Typical example of global signature are the use of global color histogram [2] or employing color information to average frames in video as a tiny fingerprint [6]. In [3] authors proposed random histogram to project low-level features and embed them into a high dimensional space using locality sensitive hashing. Although global features are straightforward to extract and known to perform well for copies with low level of variations in global features, but their accuracy is typically low because they change dramatically under intense photometric variations such as brightness and contrast changes. Moreover, they take the whole frame as input which leads to poor distinguishing power when we deal with keyframes that have a fixed logo, banner or captions which often exist in news videos.

In local signature matching based methods [6,12,7,8,11] a bag of keyframe local features is considered as the representation of the whole video. Using a keyframe to represent a shot is a well-known method in the area of video retrieval [6]. Although these methods have been most robust methods against wide range of transformations such as lighting changes, object occlusion, cropping, view point changes and etc, but as mentioned in [12] bottlenecks of local-feature-based approaches are: First, the stability of local interest points may be unsatisfactory when working with unconstrained video content since the detection rate of near-duplicates in keyframe-based approaches is dependent on the result of keyframe selection to certain extent. Second, the number of local interest points that needs to be determined for a video frame/keyframe is typically high, which results in an expensive computational cost when a large video database is in use. To cope with these challenges, authors in [12] proposed shot-based interest points for effective and efficient near-duplicate video retrieval. They assume that the variation of the keypoints in a shot frame is diverse. They use the reference extraction (RE) approach to find the local descriptors with higher occurrence frequencies.

As another group of local signature matching based methods, trajectory-based approaches track keypoints along the video sequence to enrich keypoint features with spatio-temporal information. In [8] the whole shot is represented using a bag of trajectories where each trajectory in turn is described as temporal patterns of discontinuities.

In the sequence matching approaches the temporal structure of videos plays an important role in the near-duplicity detection. While generally due to using the global features, these methods facilitate fast retrieval of duplicate videos, the localization of duplicate video segments are often being carried out in a heuristic manner. In [5] authors address the issue of partial near-duplicate detection and localization. The connections across videos are established through partially aligning video content. They convert partial alignment problem to a network flow problem. In [4] authors introduce a compact yet effective video signature called video distance trajectory (VDT). This method suffers from severe information loss. Since generally speaking similar news stories have short shots and different length and even temporal order (e.g. Figure 1), sequence matching based approaches do not work well in this domain.

3 Framework to Generate Scene Signature

The proposed framework to generate a scene signature for a given news story is illustrated in Figure 2. The first step is to extract keyframes by sampling the video frames at a constant interval. These keyframes are clustered using the NDK

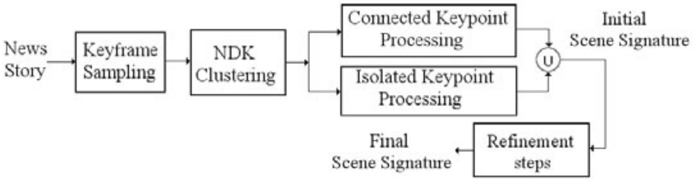


Fig. 2. Our proposed framework for generation of scene signature

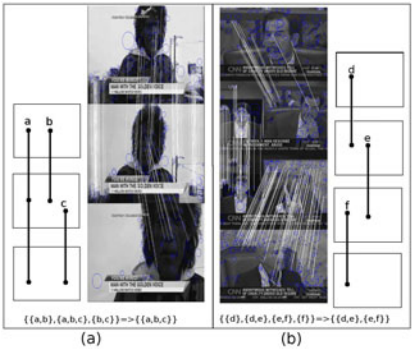
clustering algorithm of [10] where the SIFT descriptors are used to find matching keypoints across keyframe pairs. The connected keypoints and isolated keypoints are analyzed and an initial scene signature is generated for each cluster. These clusters are then refined so that some of them are merged and the final scene signature which is more compact and discriminative is created for each cluster. We explain each of the blocks of the framework in the following.

3.1 Keyframe Sampling and NDK Clustering

News videos often contain styles like picture-in-picture or complex scene transitions like fade in, fade-out or dissolve, causing detection of shot boundaries to be erroneous. As a result keyframe extraction using shot boundary detection becomes inadequate. Hence, we employ uniform sampling at a rate of one frame-per-second. Although this leads to more keyframes and subsequently higher computational costs for the scene signature generation. Note that the scene signature can be generated through an off-line process for each new story, which allows the on-line video retrieval to be significantly more efficient and effective using scene signature rather than sampled keyframes as we will explain in Section 4.1.

Next, we extract SIFT keypoints and corresponding descriptors from each sampled keyframe. Keypoints located on the logo or banner which some channels watermark on their own published news stories, contribute to keypoint matches across the frames. Similarly, keypoints extracted from textual regions (i.e. close captions, subtitles or inserted static texts in the keyframe) also contribute to the matches. These irrelevant matches may contribute to erroneous NDK detection. Therefor the keypoints, located in these regions, are removed by the process described in [9].

Near duplicate keyframes within a news story are detected using the NDK detection algorithm of [10], which is based on matching of SIFT keypoints. Instead of brute-force pairwise comparisons across all extracted keyframes, we look for NDK pairs within a predefined temporal sliding window since two keyframes that are temporally close to each other are more likely to be NDKs. This allows us to reduce the computational expense dramatically from $O(n^2)$ to $O(n)$ for the window size of one where n indicates the number of keyframe within the news story. The NDK clusters are formed by grouping together those keyframes which detected as NDKs.


a
b
c
d
e
f
{{a,b},{a,b,c},{b,c}}=>{{a,b,c}}
{{d},{d,e},{e,f},{f}}=>{{d,e},{e,f}}
(a)
(b)

```
$C_k$: Candidate keypoint-set of size $k$
$L_k$: frequent keypoint-set of size $k$
$L_1 = \{$ connected keypoints $\}$;
for ($k = 1; L_k \neq \emptyset; k++$) do begin
    $C_{k+1}$ = candidates generated from $L_k$;
    for each transaction $t$ in database do increment
        the count of all candidates in $C_{k+1}$ that are contained in $t$
    $L_{k+1}$ = candidates in $C_{k+1}$ with min-support (i.e. 1)
    end
return $\cup_k L_k$;
```

(c)

Fig. 3. Connected keypoint analysis

3.2 Processing of SIFT Keypoints

The first step in generating the scene signature is to categorize all keypoints within the keyframes in a cluster into two categories - connected and isolated. The former refers to matching keypoints that contribute to the NDK detection and therefore, they address the mutual visual cues in the NDK cluster. The latter refers to the rest of the keypoints that did not find any matches, yet they can convey novel visual cues for the keyframes in the cluster. In the following, we analyze these two groups of keypoints individually to generate a compact and representative scene signature for each cluster.

Connected Keypoint Analysis. We represent each set of connected keypoints, detected along the NDK cluster, by the keypoint with the largest scale and determine its degree as the number of connected keypoints minus one. Then we study co-occurrence of the connected keypoints in the NDK cluster. An NDK cluster will typically have a main object as shown in Figure 3(a). Line segments a, b and c in Figure3(a) denote sets of matching keypoints between the frames where the line starts and it ends, e.g., a represents matching keypoints between every pair of keyframes. However, in an NDK cluster that contains keyframes with split screen (which often exists in news videos) or with large camera or object motion, there may not be matching keypoints between every pair of keyframes. For instance, in Figure 3(b) the visual cues carried by keypoint sets d and f are irrelevant to each other. Thus, we can divide a scene containing a large number of keyframes into sub-scenes with the coherent visual content and take into account the co-occurrence of keypoints to generate a more precise scene signature.

To study co-occurrence of the connected keypoints in the NDK cluster, we employ the concept of frequent itemset pattern identification. For each cluster, we determine a transaction database where its items are all connected keypoints and its itemsets are keyframes within the cluster. We find the maximal pattern for each cluster transaction database considering the minimum support of one using *Apriori* algorithm [1] and use them to consider the co-occurrence of connected keypoints as shown in Figure 3(c). Returning to Figure 3, the maximal

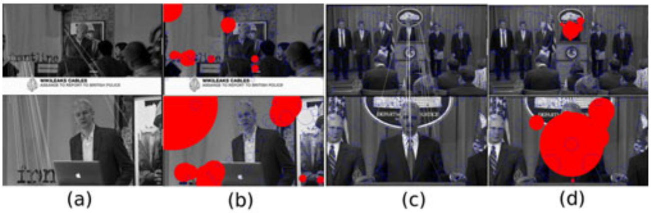


Fig. 4. Isolated keypoints selection.(a) and (c) connected and isolated keypoints, (b) and (d) connected keypoints neighborhoods.

pattern set for Figure 3(a) and (b) are $\{a, b, c\}$ and $\{\{d, e\}, \{e, f\}\}$, respectively. Thus, we obtain a set of keypoints that represents a particular visual content in the corresponding scene better. Note that if we simply consider all connected keypoints together to generate the scene signature, we ignore the fact that some of them did not appear together. In Section 3.3, we utilize this co-occurrence information inherent with max-patterns for each NDK cluster to determine similarity between scene signatures and between a scene signature and a keyframe.

Isolated Keypoint Analysis. As stated earlier, isolated keypoints refer to keypoints that have not been matched through NDK clustering process and we assign zero degree to them. Isolated keypoints arise either because there is indeed no matching keypoint in other keyframes, e.g., Figure 4(c) or the NDK detection algorithm fails to identify other matching keypoints due to lighting changes or partial occlusion, e.g., Figure 4(a). In either case, isolated keypoints possess valuable visual information which could even be the most informative visual content, e.g., in Figure 4(b), the isolated keypoints marked in blue circles are semantically more meaningful compared to the connected keypoints on the background shown in Figure 4(a). In such cases ignoring the isolated keypoints and solely considering the connected keypoints can lead to an ineffective scene signature since the desired scene signature must cover all visual contents in the scene.

Since roughly more than 80% of the keypoints are isolated keypoints, it is important to select the most discriminative among them. They should be spatially as far away as possible from the keypoints that have been matched. In other words, isolated keypoints near a bunch of matching keypoints does not carry sufficient discriminative information. Hence, we need to identify a neighborhood of connected keypoints from which the isolated keypoints must be removed. Figure 4(a) shows a pair of keyframes that belongs to a particular news story and their matching keypoints. Each matching keypoint is depicted as a circle in Figure 4(b) whose radius is proportional to the scale. We remove isolated keypoints from within the circle, which serves as the neighborhood. The union of the remaining isolated keypoints in each keyframe in the NDK cluster gives the final set of isolated keypoints. Figure 4(c) shows another pair of NDK frames whose neighborhood of matched keypoints are shown in circles in Figure 4(d).

3.3 Scene Signature Generation and Similarity

An initial scene signature is generated by first defining a fixed budget for the connected keypoints (i.e. $N_c = 800$) and determining a dynamic budget for isolated keypoints as $N_i = min(max(0, 400 - 0.4 \times n_c), n_i)$, where n_c and n_i are the number of connected keypoint representatives and isolated keypoints, respectively. The importance of the i^{th} keypoint is determined by $S_i = d_i + scale_i/max(scale_i), i = 1, 2, .., n_c \; or \; n_i$, where d_i and $scale_i$ refer to the degree and the scale of the keypoint, respectively. d_i is equal to zero for isolated keypoints as mentioned earlier. We simply rank connected keypoints representatives and isolated keypoints based on their scores and pick the best N_c and N_i ones respectively to form the initial scene signature.

To calculate scene signature (SS_1 and SS_2) similarity, we take co-occurrence information into account and first determine the co-occurrence matrix $C = [c_{ij}]_{n_{all} \times n_{mp}}$ for each generated scene signature where n_{all} refers to all keypoints within the NDK cluster and n_{mp} indicates the number of detected max-patterns and

$$c_{ij} = \begin{cases} 1 \;\; \text{if } (KP_i \in MP_j) \; or \; (\exists KP_m \in MP_j | KP_m \in KF_i), \\ 0 \qquad\qquad\qquad\qquad\qquad Otherwise \end{cases} \tag{1}$$

where KP_i indicates the i^{th} keypoint in the NDK cluster and MP_j refers to the j^{th} max pattern and KF_i indicates keyframe containing KP_i. For instance, co-occurrence matrix for Figure 3(a) is a column of one while co-occurrence matrix for 3(b) has two columns marking the first and the second max-pattern sets of keypoints (i.e. *d,e* and *e,f*) and the corresponding isolated keypoints.

Next, we determine affinity matrix $S = [s_{ij}]_{n1 \times n2}$ where n_1 and n_2 are the numbers of keypoints in SS_1 and SS_2, respectively. s_{ij} is equal to 1 if the i^{th} keypoint from first scene signature is matched with the j^{th} keypoint from the second scene signature. The final similarity score between two scene signatures is determined as:

$$Sim(SS_1, SS_2) = 1 - e^{-\frac{max(C_1^T.S.C_2)}{\tau}} \tag{2}$$

where C_1 and C_2 are $n_1 \times np_1$ and $n_2 \times np_2$ co-occurrence matrices of SS_1 and SS_2 where np_1 and np_2 are numbers of max-patterns of SS_1 and SS_2, respectively. τ is set to 17. To determine scene signature-keyframe similarity we can use the same formula by considering a unit vector $(n_2 \times 1)$ as C_2 where n_2 is the number of keypoints in the keyframe of interest.

3.4 Refinement of Initial Scene Signature

In some news stories, there will be NDKs that are temporally far from each other and which would not be clustered together through the proposed NDK clustering scheme, since we limited the exploring area for each keyframe using a sliding window. Considering the fact that the scene signature is basically bag-of-SIFT, we can use the same NDK clustering approach stated in the Section 3.1 for scene signature clustering as well. Therefore, we utilize the same keypoint matching method to match keypoints across scene signatures. After clustering the near-duplicate scene signature (i.e. which are scene signatures sharing adequate matching keypoints), we can generate the second generation of scene signatures similar to the first generation but with the different configuration.

A further refinement of the scene signatures is done by re-clustering all keyframes again using the second generation of scene signatures as cluster centroids and soft-assigning each keyframe to the clusters whose centroid is similar to based on the equation 2. From our observations, this step results in a more enriched and semantic cluster since some similar keyframes could not be clustered together in the earlier stages. This semantic improvement is obtained by the integration of relevant visual clues obtained in the second generation of scene signatures.

The final refinement step involves the transitivity property of keyframes within a cluster, i.e., we assume that clusters sharing the same keyframes are associated to each other and their scene signatures can be merged accordingly to come up with the more semantic and compact scene signature. We re-use the first step of the refinement procedure, explained in the first paragraph of this Section to merge corresponding scene signatures.

4 Experimental Results

In this Section, first we evaluate the distinguishing power of our proposed scene signature against other global and local signatures. We also illustrate the efficiency of the proposed scene signature and explain the role of keypoint budgeting to reach a unique and compact representation of the news story. Finally, we evaluate our proposed scene signature performance through the retrieval task.

The dataset consists of 100 news stories from different channels downloaded from YouTube in Feb. 2011. They have a wide range of lengths from 2 to 5 minutes covering worldwide events.

4.1 Discriminative and Compactness Analyses of the Scene Signature

In this part, we try to detect the NDK clusters in each news video story using different video signatures. We compare the clustering performance of our proposed scene signature against other global and local video signatures listed in Table 1. As global signature we use gray-scale histogram(GH), color histogram(CH) and edge orientation histogram(EOH) while as local signature we utilize SIFT descriptor in different basis of keyframes (BOSK), NDK cluster (BOSC) where we aggregate all keypoints within the NDK cluster, average of connected keypoints per NDK cluster(ACKP), connected keypoints representative per NDK cluster(CKPR), and our proposed scene signature (MSS). As the dissimilarity measure we utilize Bhattacharya coefficients and Cosine similarity for the global signature while we measure SIFT-based signature dissimilarity based on number of matching keypoints through an exponential function.

Table 1. Global and local video signature and their description and similarity measures

Symbol	Signature	Description	Dissimilarity measure
GH	Gray-scale histogram	normalized 32-dimensional intensity histogram	$d(H_1, H_2) = 1 - \sqrt{1 - \sum_I \frac{\sqrt{H_1(I).H_2(I)}}{\sqrt{\sum_I H_1(I).\sum_I H_2(I)}}}$
CH	Color (RGB) histogram	normalized 96-dimensional color histogram	$d(H_1, H_2) = 1 - \frac{\sum_I H_1(I).H_2(I)}{\sqrt{\sum_I H_1^2(I).\sum_I H_2^2(I)}}$
EOH	Edge Orientation histogram	(16 × 8)-dimensional edge orientation histogram extracted from 4 × 4 blocks	
BOSK	Bag-of-SIFT per Keyframe	n keypoints described by 128-dimensional SIFT descriptor	$d(B_1, B_2) = e^{-\frac{n}{12}}$ where n is the number of matching keypoints
BOSC	Bag-of-SIFT per NDK cluster		
ACKP	Average of Connected keypoints per NDK cluster		
CKPR	Connected keypoints Representative per NDK cluster		
MSS	merged scene signatures per NDK cluster		$d(SS_1, SS_2) = 1 - Sim(SS_1, SS_2)$ (2)

We compute the mean and variance of within-cluster sum of squares (WSS) and between-cluster sum of squares (BSS) as

$$WSS = \frac{1}{n} \sum_{i=1}^{n_C} \frac{1}{|C_i|} \sum_{x,y \in C_i} d^2(x, y), \tag{3}$$

$$BSS = \frac{1}{n} \sum_{i=1}^{n_C} \frac{1}{n - |C_i|} \sum_{x \in C_i, y \notin C_i} d^2(x, y), \tag{4}$$

where x and y refer to data points. $|C_i|$ is the size of the i^{th} cluster. $d(,)$ is determined for each signature based on Table 1. n and n_C refer to the number of data points and number of clusters, respectively. As shown in Table 2, the lowest WSS-to-BSS ratio belongs to MSS while the highest BSS belongs to BOSK. Although the lowest WSS belongs to BOSC, it is not discriminative and its corresponding BSS is relatively high. Among the global signatures, EOH performs reasonably well and GH signature outperform CH both in terms of BSS and WSS-to-BSS ratio.

Table 2. WSS, BSS, and WSS-to-BSS ratio for different video signatures

Signature	GH	CH	EOH	BOSK [10]	BOSC	ACKP [12]	CKPR	Our
WSS	0.17 ± 0.16	0.17 ± 0.18	0.16 ± 0.15	0.22 ± 0.33	**0.12 ± 0.29**	0.21 ± 0.33	0.17 ± 0.34	0.16 ± 0.33
BSS	0.56 ± 0.17	0.50 ± 0.57	0.64 ± 0.36	**0.92 ± 0.08**	0.63 ± 0.25	0.85 ± 0.13	0.86 ± 0.14	0.85 ± 0.13
WSS/BSS	0.30	0.34	0.28	0.24	0.21	0.24	0.20	**0.18**

To study discriminative characteristic of local signatures BOSK, BOSC, ACKP, CKPR and MSS and their difference in more detail, we determine their performance (in terms of BSS and WSS) for NDK clusters with different numbers of keyframes. As shown in Figure 5(a), WSS decreases with increasing number of keyframes since (except for BOSK) other local signatures will have more keypoints describing the identical scene. It ends up with more matching keypoints between keyframes within the NDK cluster.

To study the compactness of the proposed scene signature, we show the distribution of keypoints with different degree within the scene signature extracted from NDK clusters with different number of keyframes in Figure 6(a). For all NDK clusters, most of the information is contained in the fist-degree keypoints. In Figure 6(b) we show the same plot after imposing the budget determined in Section 3.3. Depending on the number of keyframes within NDK cluster, we

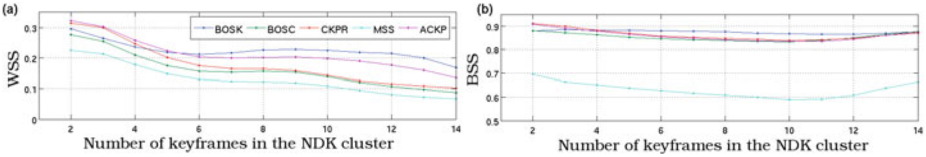


Fig. 5. WSS and BSS values of different local signatures for NDK clusters with different number of keyframes

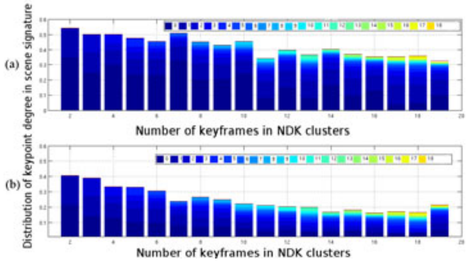


Fig. 6. Distribution of keypoint degrees in the scene signatures normalized with the number of keypoints in the NDK cluster with different number of keyframes

could compact the visual information by about $17\% - 40\%$ in terms of number of keypoints. This compactness in indexing of visual clues accelerate the news story retrieval process dramatically (roughly by 50 times), since we deal with quadratic comparison of signatures of two stories.

4.2 Retrieval Using Scene Signature

After within-story analysis of our proposed scene signature, we aim to study the uniqueness of the proposed scene signature through the retrieval task. For each story we group extracted keyframes into two sub-stories of odd and even keyframes based on their extraction time stamp as shown in Figure 7(a) and (b). Basically, for every shot longer than 4 sec we expect to have equivalent NDK cluster and consequently similar video signature in each sub-story like cluster 1 and 3 in Figure 7(a) and cluster 1 and 2 in Figure 7(b). Note that they will not be the same due to possible slight to intense variations across successive keyframes.

We use 2070 manually labeled NDK clusters extracted from the mentioned dataset. We consider each of them as the query and measure the similarity between the query and all reference clusters using different local signatures indicated in Table 1 and then rank them, accordingly. In Figure 8, we show *top-k* NDK cluster retrieval results. The retrieval performance is quantified by the probability of retrieving the corresponding NDK cluster in the top-k position of the ranked list given as $P(K) = Z_c/Z$, where Z_c is the number of queries that rank their corresponding NDK cluster within the top-k position and Z is the total number of queries(2070).

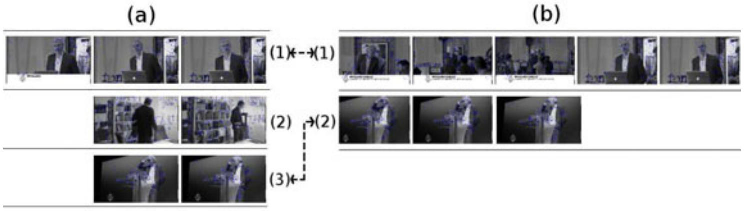


Fig. 7. An example of two sub-stories including (a)odd and (b)even keyframes extracted from original news story and their equivalent NDK clusters

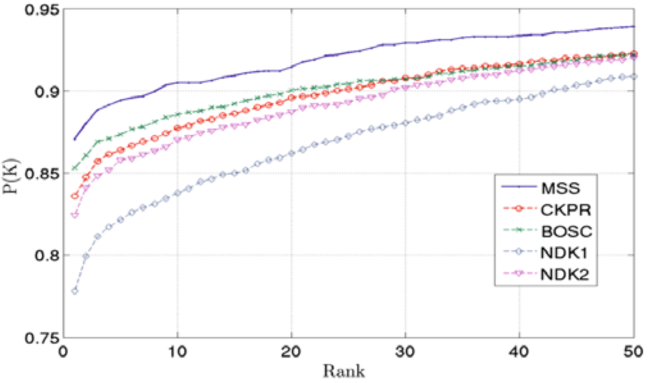


Fig. 8. Rank P(K) NDK cluster retrieval using different local signatures

In Figure 8, we also illustrate NDK cluster retrieval result based on the keyframe-level similarity across their keyframes. Through NDK1 and NDK2, we use the least and most detected similar keyframes across NDK cluster to determine the cluster similarity, respectively. The results show that even if we do not consider the effect of keyframe selection, in the best case (i.e. NDK2) we will not obtain the retrieval performance as good as the scene signature approach. Note that performance of keyframe-level approach can vary between NDK1 and NDK2 depending on the selected keyframes from NDK clusters. It should be mentioned that in addition to the better retrieval performance gained through our approach compared to NDK1 and NDK2, the former is much faster (up to 50 times) due to the compact structure of the generated scene signature explained in Section 4.1.

We also compare the proposed MSS with the keyframe-level bag-of-word (BOW) [11] approach for the news video retrieval task. To do so, after finding similar scenes /keyframes, we determine between-story similarity as $|S_i \cap S_j|.(1/|S_i|+1/|S_j|)$, where S_i refers to the set of scene/keyframe signatures contained in S_i. We use one month TRECVID 2006 dataset including more than 800 news stories out of which there are 132 similar news videos with the similar visual clues(e.g. Figure 1). We could improve average top-3 retrieval results from

38.7% to 43.3% using merged scene signature. While both BOW and MSS are based on SIFT, but integration of the informative visual clues through merged scene signature results in the better retrieval performance.

5 Conclusion

In this paper we proposed a novel video signature called scene signature which can be applicable for variety of tasks in the unconstrained news video domain. Since scene signature is solely based on the visual cues presented in the video scene context and does not consider the temporal information, it is also robust against variations in the temporal order, story/shot lengths and additional/redundant visual information which is common in news videos. The experimental results show the efficiency as well as robustness and uniqueness of our proposed scene signature compared to other global and local video signatures.

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: VLDB 1994, pp. 487–499. Morgan Kaufmann (1994)
2. Cheung, S., Zakhor, A.: Efficient video similarity measurement with video signature. In: ICIP 2002, vol. 1, pp. 59–74 (January 2002)
3. Dong, W., Wang, Z., Charikar, M., Li, K.: Efficiently matching sets of features with random histograms. In: ACM MM 2008, pp. 179–188. ACM, NY (2008)
4. Huang, Z., Shen, H.T., Shao, J., Cui, B., Member, S., Zhou, X.: Practical Online Near-Duplicate Subsequence Detection for Continuous Video Streams. IEEE Transactions on Multimedia 12(5), 386–398 (2010)
5. Tan, H., Ngo, C., Hong, R., Chua, T.: Scalable detection of partial near-duplicate videos by visual-temporal consistency. In: ACM MM 2009, pp. 145–154. ACM (2009)
6. Wu, X., Hauptmann, A., Ngo, C.: Practical elimination of near-duplicates from web video search. In: ACM MM 2007, pp. 218–227. ACM (2007)
7. Wu, X., Ngo, C.-W., Hauptmann, A.G., Tan, H.-K.: Real-time near-duplicate elimination for web video search with content and context. Trans. Multi. 11, 196–207 (2009)
8. Wu, X., Takimoto, M., Satoh, S., Adachi, J.: Scene duplicate detection based on the pattern of discontinuities in feature point trajectories. In: ACM MM 2008, page 51 (2008)
9. Younessian, E., Adamek, X., Oliver, N.: Telefonica Research at TRECVID 2010 Content-Based Copy Detection (2010), http://www-nlpir.nist.gov
10. Younessian, E., Rajan, D., Siong, C.E.: Improved keypoint matching method for near-duplicate keyframe retrieval. In: ISM 2009, pp. 298–303 (2009)
11. Zhao, W.-L., Ngo, C.-W.: Scale-rotation invariant pattern entropy for keypoint-based near-duplicate detection. Trans. Img. Proc. 18, 412–423 (2009)
12. Zhou, X., Zhou, X., Chen, L., Bouguettaya, A., Xiao, N., Taylor, J.: An efficient near-duplicate video shot detection method using shot-based interest points. IEEE Transactions on Multimedia 11(5), 879–891 (2009)

Visual Vocabulary Optimization with Spatial Context for Image Annotation and Classification

Zhiguo Yang, Yuxin Peng*, and Jianguo Xiao

Institute of Computer Science and Technology, Peking University,
Beijing 100871, China
{yangzhiguo,pengyuxin,jgxiao}@ pku.edu.cn

Abstract. In this paper, we propose a new approach of visual vocabulary optimization with spatial context, which contains important spatial information that has not been fully exploited. The novelty of our method mainly lies in two aspects: when spatial information is considered, and how spatial information is used. For the first aspect, the existing methods generally consider spatial information after the visual vocabulary is built, while we employ the spatial information in the construction of visual vocabulary, to produce more accurate visual vocabulary. For the second aspect, different from existing methods which use spatial information to re-rank the original retrieval results, to generate the local keypoint groups such as visual phrases, or in spatial pyramid matching kernel, etc, we propose a novel method that employs spatial information as side information to constrain the construction of visual vocabulary. Instead of simply assigning keypoints to the nearest cluster centers, we also take the spatial context of keypoints into consideration in the clustering process. With the proposed approach, more accurate visual vocabulary can be generated, and the evaluation results can be improved in both image annotation and classification tasks. Experiments on widely-used 15-scenes dataset demonstrate the effectiveness of the proposed approach.

Keywords: Bag-of-Visual-Words, Visual Vocabulary Optimization, Spatial Context, Constrained Clustering.

1 Introduction

Bag-of-visual-words (BoVW) model [1], which derives from the bag-of-words (BoW) representation of text documents, has become a popular representation of images. The BoVW method models an image as a bag of its local image patches, which are usually detected or densely-sampled keypoints in state-of-the-art approaches [1,2,3,4,8,11]. In this way, an image in BoVW model corresponds to a text document in BoW model, while a local image patch corresponds to a single text word occurrence. In the BoVW model, the vocabulary (or dictionary) is usually generated by clustering the local image patches, and the cluster centers are used as

* Corresponding author.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 89–102, 2012.

visual words. In the quantization step, a local image patch is usually assigned to the nearest visual word in feature space. The BoVW model has been applied in image applications such as image annotation, image classification and image retrieval.

Despite the popularity of BoVW model, it still has the following two problems: (1) Visual vocabulary, especially the visual vocabulary with small number of visual words, is not accurate and discriminative enough to describe the complex content of image. (2) Useful information other than the visual statistics of local image patches, such as the spatial information between keypoints and semantic information of images, has not been fully exploited. Recent approaches try to improve the BoVW model from the following aspects: (1) increase the efficiency of vocabulary construction to generate large visual vocabulary; (2) employ spatial information to improve performance; and (3) make better use of semantic information.

With regard to the first aspect, several approaches have been proposed to improve the flat vocabulary [1] generated by k-means algorithm, which is difficult to scale to large vocabularies. For example, Nister et al. propose the vocabulary tree [2] as an integrated quantization and indexing scheme. They use a hierarchical tree with 1 million leaf nodes as visual vocabulary, which improves both effectiveness and efficiency. Yeh *et al.* [6] propose to build vocabulary forest that consists of multiple vocabulary trees. The multiple vocabulary trees are combined to reduce the quantization effects near the boundaries of nodes in individual trees. Philbin *et al.* propose approximate k-means [3][4], which is based on approximate nearest neighbor methods. A forest of randomized k-d trees [5] is built over the cluster centers, and the computation of nearest neighbors between data points and cluster centers are replaced by the computation of approximate nearest neighbors based on the random forest. With the approximate k-means approach, large-scale flat visual vocabulary could be generated efficiently. In the above approaches, only visual statistics of local image patches is used in the generation of visual vocabulary.

With regard to the second aspect, many methods have been proposed to exploit the spatial information in applications based on BoVW model. These methods can be divided into the following categories:

(1) Use spatial information in the post-processing steps: For example, Sivic *et al.* [1] first conduct video frame retrieval without spatial information, and then re-rank the result list based on the spatial consistency between matching keypoint pairs. Philbin *et al.* [3] use RANSAC algorithm [7] to generate transformation hypotheses between query and test images, and re-rank the top-ranked images based on how well its keypoint locations are predicted by the estimated transformation.

(2) Use spatial information in local groups of keypoints: Zhang *et al.* [8] propose to use descriptive visual words and visual phrases as the visual correspondences to text words and phrases. Visual phrases refer to the frequently co-occurring visual word pairs, and they are used to improve the performances in image retrieval, image re-ranking, and object recognition. Zheng *et al.* [10] construct visual phrases from frequently co-occurring visual word-sets with similar spatial context, and further cluster visual phrases into visual synsets based on class probability distribution.

(3) Use spatial information in kernels: For example, Lazebnik *et al.* [11] propose "spatial pyramid matching kernel" based on the spatial pyramid representation of image, which partitions an image into increasingly fine sub-regions and computes the histograms of keypoints found inside each sub-region. Lu *et al.* [25] propose spatial mismatch kernels, which is a new class of 2D string kernels, to capture the spatial dependencies across visual keywords within the image. The images are firstly represented as 2D sequences of visual keywords, and then decomposed into row-wise and column-wise 1D sequences. After that, the proposed spatial mismatch kernels are used to measure image similarity based on shared occurrences of 1D subsequence.

(4) Other methods: Perdoch *et al.* [12] propose to learn a highly memory-efficient representation of local geometry associated with visual words, based on minimization of average re-projection error in the space of ellipses. Qin *et al.* [29] propose contextual visual words for image classification, which introduce contextual information from the coarser scale and neighborhood regions into densely-sampled keypoints.

All the above approaches consider spatial information after the visual vocabulary is constructed. Without the use of spatial information in the construction of visual vocabulary, inaccurate visual vocabulary may be generated and the inaccuracy may propagate to the following steps. The above approaches could partially eliminate the influence of inaccurate visual vocabulary, but do not solve the problem at root.

With regard to third aspect, several approaches have been proposed to exploit the semantic information (class labels) of images. For example, Ji *et al.* [23] present a semantic embedding framework to integrate semantic information from *Flickr* labels for supervised vocabulary construction. Cai *et al.* [30] present a codebook learning approach, which learn a weighted similarity metric to satisfy that the similarity between images with the same label is larger than the similarity between images with different labels with largest margin.

In the above three aspects, we focus on the second aspect in this paper. Different from existing methods that use spatial information after the construction of visual vocabulary, we propose to use spatial information at an earlier stage with a novel method, to optimize the visual vocabulary. We propose a modified version of k-means algorithm that takes the spatial relationship between keypoints into consideration, to generate more accurate visual words by iterative expectation-maximization (**EM**) steps: In the **E** step, we propose to modify the distance between a keypoint and a visual word according to the keypoint's spatial context in image. That is to say, the assignment of a keypoint is dependent on the assignemt (in last iteration) of its contextual keypoints. In the **M** step, the cluster centers (visual words) are updated according to the new assignment of keypoints.

The proposed method is similar to the *COP-kmeans* algorithm [20], which is applied to the problem of lane finding in GPS data, in that both methods use side-information other than the data points to constrain the clustering process. Our proposed method differs from *COP-kmeans* in the following ways: (1) The *COP-kmeans* algorithm uses two types of pair-wise constraints, namely must-link and cannot-link, which respectively specify that two data points should be in the same or different clusters. In our method, the constraints are in the form of contextual keypoints in the same image. (2) In the *COP-kmeans* algorithm, all the constraints have to be obeyed, otherwise the clustering process fails. In our proposed method, we use the spatial constraints to modify the distance between data point and cluster centers, which are more robust to the noise in constraints.

In summarization, the contribution of this paper can be summarized as follows: we propose a novel method of utilizing spatial information in the optimization of visual vocabulary. The novelty of our method mainly lies in two aspects: when spatial information is considered (at an earlier stage, in the optimization of visual vocabulary), and how spatial information is used (with a modified k-means algorithm constrained by spatial context). With the proposed approach, more accurate visual vocabulary can be generated, quantization errors can be reduced, and thus performance improvements can be achieved.

The rest of this paper is organized as follows: We will present the motivation and the detailed algorithm of the proposed approach in Section 2 and Section 3 respectively. In Section 4, experiment results on fifteen scene categories (15-scenes) dataset are reported. Finally, we conclude this paper in Section 5.

2 Motivation

As stated in Section 1, the existing methods generally use spatial information after the visual vocabulary is constructed. Without the use of spatial information in the construction of visual vocabulary, inaccurate visual vocabulary may be generated.

We have observed the following phenomenon from the experiments: in the quantization (keypoint assignment) step, some keypoints are simultaneously close to several visual words in the feature space, and the difference between the distance to the nearest visual word and the distance to the second nearest visual word is not significant. These keypoints are more likely to be influenced by the inaccuracy of visual vocabulary, and are prone to errors in the quantization step. Since these problematic keypoints lie in the border area between visual words, we name this phenomenon as the "*border keypoint*" problem.

Under this situation, the usual weighting methods such as TF and TF-IDF would suffer from the inaccuracy of visual vocabulary. Soft-weighting [9][13] method can partially eliminates the influence of inaccurate visual vocabulary by assigning one keypoint to several nearest visual words with different weights, but does not solve the problem at root.

We propose to consider the spatial context in the quantization of keypoint, which is illustrated in Fig. 1. The top part (in dotted rectangle) shows four sample images, while the bottom part shows the relative position of the keypoints (empty circles) and visual words (black solid circles) in feature space. The red keypoints in the first three images are assigned to visual word w_1, and the blue keypoints belong to w_3. Since w1 and w_3 co-occur frequently in the first three images (and also in many other images not shown in Fig. 1), they form a visual phrase (w_1,w_3). In the fourth image, the purple keypoint belongs to w_1, since w_1 is much closer to the purple keypoint than the second nearest word w_2.

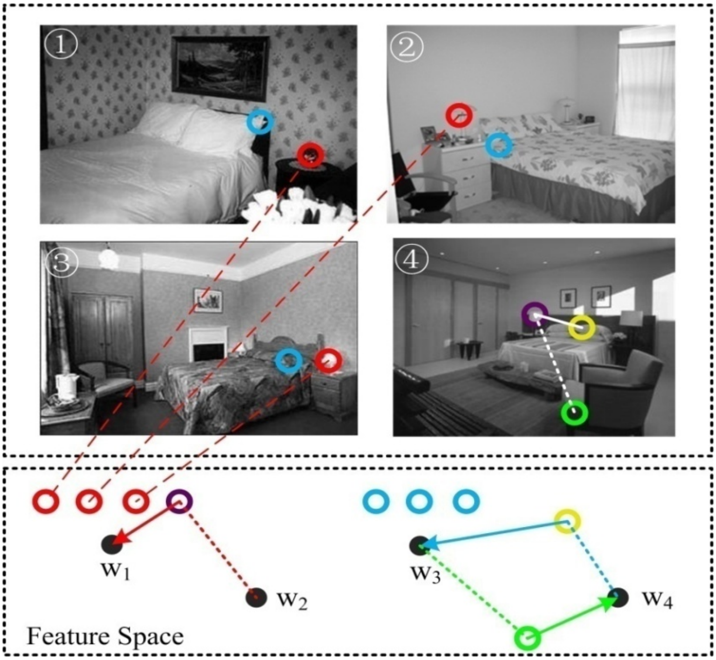


Fig. 1. Keypoint assignment considering spatial context (empty cirles represent keypoints, while black solid circles represent visual words). For clarity and simplicity, irrelevant keypoints are not plotted, and the angle and scale information are not demonstrated in the denotation of keypoint. Please notice that this is a contrived demo for clear illustration.

The yellow keypoint is a "*border keypoint*", since it is almost the same distance from visual word w_3 and visual word w_4. In this case, the keypiont-word distance alone is not dependable enough for accurate quantization and we propose to take the yellow keypoint's spatial context into consideration. The purple keypoint which is close to the yellow keypoint in the fourth image, lies in the spatial context of the yellow keypoint. Since the purple keypoint belongs to w_1, it "*supports*" or "*suggest*" that we should assign the yellow keypoint to word w_3, which would result in a new instance of visual phrase (w_1,w_3). This "*support*" or "*suggestion*" could be considered as kind of indirect soft constraint, which can be used in constrained clustering methods.

The scope of spatial context depends on the scale of keypoint. For example, the green keypoint in the fourth image, which is also a "*border keypoint*", is not influenced by the purple keypoint, since they are too far away from each other (the scale information is not shown in Fig. 1). Generally speaking, keypoints with greater scale values would have bigger scope of spatial context. However, spatial context will not be used for keypoints with too large scale values, due to heavy noises.

3 Our Approach

Motivated by the above observation, and inspired by the *COP-kmeans* algorithm [20] in constrained learning, we propose a modified k-means algorithm that takes the spatial context of keypoints as side information, to constrain the clustering process, and to generate more accurate visual vocabulary.

The context of keypoint p_i is defined as the set of keypoints that are located within a certain range in the same image:

$$N_i = \{\, p_j \in \mathrm{P} \mid \mathrm{I}_i = \mathrm{I}_j \wedge \|x_i - x_j\| < R_c \cdot scale_i \,\} \tag{1}$$

where $\mathrm{P} = \{p_1, p_2, \dots, p_{\mathrm{PN}}\}$ denotes the set of keypoints, and PN is the number of keypoints in all images; I_i, x_i, and $scale_i$ respectively denotes the id of image containing keypoint p_i, the position of keypoint p_i in image, and the scale of keypoint p_i ; R_c is a parameter that controls the range of spatial context. A large R_c is necessary to make full use of the spatial information between keypoints, but a large R_c is more prone to noise and also increases computational cost. According to Zhang *et al.* [8], we set $R_c = 6$, which shows a good tradeoff between effectiveness and efficiency.

The proposed vocabulary optimization algorithm consists of the following steps:

(1) Generate the initial visual vocabulary:
Cluster the set of keypoints into WN groups, where WN is the pre-specified cluster number. Use the cluster centers as the original visual words, denoted as

$$\mathrm{W}^{(0)} = \{w^{(0)}{}_1, w^{(0)}{}_2, \dots, w^{(0)}{}_{\mathrm{WN}}\} \tag{2}$$

where $\mathrm{w}^{(0)}{}_{\mathrm{i}}$ is the *i*th visual word. The number 0 in bracket on the top right corner means this is the 0th iteration step, that is, the initialization step.

(2) Optimize the visual vocabulary in iterative EM steps as follows:

a) **Keypoint assignment without spatial information:**
Assign each keypoint to the visual word with the shortest distance in the feature space, and generate the quantization vector:

$$\mathrm{Q}^{(\mathrm{T})} = < \mathrm{q}^{(\mathrm{T})}{}_1, \mathrm{q}^{(\mathrm{T})}{}_2, \dots, \mathrm{q}^{(\mathrm{T})}{}_{\mathrm{PN}} > \tag{3}$$

where

$$\mathrm{q}^{(\mathrm{T})}{}_i = \mathrm{argmin}_{1\le \mathrm{j} \le \mathrm{WN}}\, \mathrm{D}\left(p_i, \mathrm{w}^{(\mathrm{T})}{}_{\mathrm{j}}\right) \tag{4}$$

is the index of visual word (in current vocabulary) that is closest to the *i*th keypoint p_i in the feature space.

b) **Generate visual phrases according to keyword assignment result:**
Adopt rotation-invariant spatial histogram [22] to count the frequency of co-occurrence of visual word pairs, which is similar to the work by

Zhang *et al.* [8]. Then the top FN pairs of visual words with the highest frequency are selected as visual phrases, which is denoted as:

$$\mathrm{F}^{(\mathrm{T})} = \{\mathrm{f}^{(\mathrm{T})}{}_1, \mathrm{f}^{(\mathrm{T})}{}_2, \dots, \mathrm{f}^{(\mathrm{T})}{}_{\mathrm{FN}}\} \tag{5}$$

where $\mathrm{f}^{(\mathrm{T})}{}_i = (i_1, i_2)$ represents that visual words $\mathrm{w}^{(\mathrm{T})}{}_{i1}$ and $\mathrm{w}^{(\mathrm{T})}{}_{i2}$ form a visual phrase.

c) **Keypoint assignment with spatial context:**
Taking the spatial context of keypoint into consideration, we modify the distance function between keypoint p_i and visual word $\mathrm{w}^{(\mathrm{T})}{}_{\mathrm{j}}$ as:

$$\mathrm{DS}(p_i, \mathrm{w}^{(\mathrm{T})}{}_{\mathrm{j}}) = \mathrm{D}\left(p_i, \mathrm{w}^{(\mathrm{T})}{}_{\mathrm{j}}\right) * \alpha^{\left|\mathrm{C}(p_i, \mathrm{w}^{(\mathrm{T})}{}_{\mathrm{j}})\right|} \tag{6}$$

where $0 < \alpha < 1$ is a pre-specified parameter, and $|\mathrm{C}(p_i, \mathrm{w}^{(\mathrm{T})}{}_{\mathrm{j}})|$ is the number of keypoints in p_i's spatial context that "supports" or "implies" that keypoint p_i belongs to visual word $\mathrm{w}^{(\mathrm{T})}{}_{\mathrm{j}}$, that is,

$$\mathrm{C}(p_i, \mathrm{w}^{(\mathrm{T})}{}_j) = \{p_k | i \neq k \wedge \mathrm{I}_i = \mathrm{I}_k \wedge \left(j, \mathrm{q}^{(\mathrm{T})}{}_k\right) \in \mathrm{F}^{(\mathrm{T})}\} \tag{7}$$

where $\mathrm{I_i} = \mathrm{I_k}$ means p_i and p_k are in the same image, and $\left(\mathrm{j}, \mathrm{q}^{(\mathrm{T})}{}_{\mathrm{k}}\right) \in \mathrm{F}^{(\mathrm{T})}$ means keypoints p_i and p_k will form an instance of visual phrase if p_i belongs to $\mathrm{w}^{(\mathrm{T})}{}_{\mathrm{j}}$.
Assigning each keypoint to the visual word with the minimum value of the modified distance function, we get the new quantization vector:

$$\mathrm{QS}^{(\mathrm{T})} = < \mathrm{qs}^{(\mathrm{T})}{}_1, \mathrm{qs}^{(\mathrm{T})}{}_2, \dots, \mathrm{qs}^{(\mathrm{T})}{}_{\mathrm{PN}} > \tag{8}$$

where

$$\mathrm{qs}^{(\mathrm{T})}{}_i = \mathrm{argmin}_{1 \le j \le \mathrm{WN}} \mathrm{DS}\left(p_i, \mathrm{w}^{(\mathrm{T})}{}_j\right) \tag{9}$$

d) **Update visual vocabulary:**
Calculate the mean value of the keypoints that belong to each visual word according to $\mathrm{QS}^{(\mathrm{T})}$, and generate the new visual vocabulary:

$$\mathrm{V}^{(\mathrm{T+1})} = \{\mathrm{w}^{(\mathrm{T+1})}{}_1, \mathrm{w}^{(\mathrm{T+1})}{}_2, \dots, \mathrm{w}^{(\mathrm{T+1})}{}_{\mathrm{WN}}\} \tag{10}$$

where

$$\mathrm{w}^{(\mathrm{T+1})}{}_j = \frac{\sum_{p_k \in \mathrm{S}_j} p_k}{|\mathrm{S}_j|} \tag{11}$$

$$\mathrm{S}_j = \{p_k | \mathrm{qs}^{(\mathrm{T})}{}_k = j\} \tag{12}$$

(3) The iteration stops when the visual vocabulary does not change between two iterations, or the pre-specified number of iteration is reached.

The proposed algorithm can be summarized as in Fig. 2.

```
Algorithm: Optimizing Visual Vocabulary with Spatial Information
Input:
    Keypoints detected from images: $P$
    Visual word number: $WN$
    Visual phrase number: $FN$
    Maximum number of iteration: $N_T$
Initialization:
    $\{P\} \rightarrow \{V^{(0)}\}$                    Visual vocabulary initialization               (1)
Iterative EM Steps: while{ $V^{(T+1)} \neq V^{(T)}$ and $T \leq N_T$ } do
    E Step:
        $\{P, V^{(T)}, D\} \rightarrow \{Q^{(T)}\}$           Assignment without spatial information    (2).a
        $\{P, Q^{(T)}\} \rightarrow \{F^{(T)}\}$              Visual phrases selection                  (2).b
        $\{P, V^{(T)}, DS\} \rightarrow \{QS^{(T+1)}\}$       Assignment considering spatial context    (2).c
    M Step:
        $\{P, QS^{(T+1)}\} \rightarrow \{V^{(T+1)}\}$         Update visual vocabulary                  (2).d
Output:
    Optimized visual vocabulary $V^*$
```

Fig. 2. The proposed algorithm to optimize visual vocabulary with spatial context

4 Experiments

In our experiments, the tasks of image annotation and image classification are carried out to evaluate the effectiveness of the proposed visual vocabulary optimization algorithm. Generally speaking, the image annotation task annotates whether an image can be described by a label, while the image classification task decide which one of pre-specified categories an image belongs to. In our experiments, we firstly carry out image annotation task using the image category information as concept labels, and then carry out the image classification task based on the predicted concept probability in image annotation task. Adopting the one-against-all strategy, each time images in one category are used as positive samples while the rest categories are considered as negatives samples, based on which classifiers are trained. For image annotation task, the models are used to predict the probability of existence of the current concept (category) on the test images. For image classification task, the test images are classified into the category with the highest predicted probability.

Mean average precision (MAP [15]) is used as the overall evaluation metric for the image annotation task, where the average precision of each category reflects the quality of the ordered list sorted by the prediction score. Mean accuracy (MAC) is used to evaluate the performance of the image classification task. The test images are assigned to the category with the highest prediction score, based on which a confusion matrix is constructed. MAC is the mean value of the diagonal elements of the confusion matrix. Higher MAP and MAC results mean better performance in image annotation and image classification respectively.

We adopt the classic BoVW approach [1] with soft-weighting as our baseline: Keypoints are detected by detectors such as Difference-of-Gaussian (DoG) [16], Harris Laplace [17] and Dense Sampling [11], and described by SIFT [16] feature. Then the keypoints' feature vectors are clustered into visual words, which form the visual vocabulary. In the quantization step we adopt the soft-weighting method [9][13], and assign each keypoint to 4 nearest visual words simultaneously with different weights according to the keypoint-word distance. A histogram of visual words is generated for each image, forming the BoVW representation. In the baseline method, no spatial information is used.

We conduct the comparison experiments on fifteen scene categories (*15-scenes*) dataset [11]. The *15-scenes* dataset contains 15 categories and 4485 images in total. The major sources of the images include the COREL collection, personal photographs, and Google image search. Several researchers contribute to this dataset: Oliva and Torralba [18] collected eight categories, then Fei-Fei and Perona [19] added five more categories, and finally Lazebnik *et al.* [11] contributed the rest two categories. The number of images belonging to each category ranges from 200 to 400, and the average image size is about 300 × 250 pixels. Fig. 4 shows some sample images of the 15-scenes dataset, which are resized due to space limitation. The *15-scenes* dataset is adopted extensively in the research area of image classification.

As described above, two tasks are evaluated on *15-scenes* dataset for comprehensive comparison: image annotation and image categorization. The experimental results are shown in Table 1. Following the same experiment procedure of [11] and [21], we randomly select 100 images per class for training and use the rest for testing. The experiments are carried out on 10 different random splits, and the mean value and standard deviation of MAP and MAC are given, which can evaluate objectively the performance of our approach.

We adopt SVM as classifier, with LibSVM implementation, histogram intersection kernel and default parameters. The number of visual words WN, the number of visual phrases FN and the number of iterations $\mathrm{N_T}$ are all important parameters that would influence the performance of the proposed approach. We adopt the following parameter settings heuristically: $\mathrm{FN} = 100$, and $\mathrm{N_T} = 10$, while optimal parameters could be selected by cross-validation.

For more comprehensive comparison, the same experiments are done for three different keypoint detectors separately: Difference-of-Gaussian (DoG) [16], Harris Laplace [17] and Dense Sampling [11]. From Table 1, we can see that the proposed approach improves the results of both image annotation and image classification tasks, and for all the three keypoint detectors.

We further check the robustness of the proposed approach by varying the size of visual vocabulary. The experimental results for DoG and Harris Laplace detectors are shown in Tables 2 and 3 respectively. We can see that the proposed approach keeps improving the performance consistently with different visual vocabulary sizes, for both DoG and Harris Laplace detectors. We do not carry out this experiment for Dense Sampling keypoint detector, due to its heavy computation cost.

In Fig. 3, we take DoG detector for example, and further illustrate the reason why the proposed approach can improve the performance of image annotation and image

Table 1. Comparison between baseline and our approach for different keypoint detectors (WN=1000)

	Annotation(MAP)		Classification(MAC)	
	Baseline	Our	Baseline	Our
DoG	0.739±0.006	**0.753**±0.006	0.720±0.005	**0.733**±0.004
Harris Laplace	0.769±0.004	**0.789**±0.004	0.743±0.006	**0.755**±0.006
Dense Sampling	0.805±0.004	**0.817**±0.006	0.788±0.005	**0.801**±0.006

classification. The horizontal ordinate represents the number of iterations, while the vertical ordinate represents the ratio of "*border keypoints*" whose r21 value is lower than a specified threshold (*T=1.05*), where the *r21* of a keypoint is defined as the ratio of its distance to the second-nearest visual word versus its distance to the nearest visual word. In Fig. 3, we can see that the percentage of "*border keypoints*" decreases quickly as the iteration continues, which keeps true for different sizes of visual vocabulary. This shows that the proposed approach can effectively optimize the visual vocabulary, by reducing the percentage of "*border keypoints*" and achieving more accurate keypoint assignment in the quantization step of the clustering process.

Table 2. Comparison between baseline and our approach for different visual vocabulary sizes (with DoG detector)

WN	Annotation(MAP)		Classification(MAC)	
	Baseline	Our	Baseline	Our
500	0.722±0.006	**0.736**±0.007	0.703±0.009	**0.715**±0.005
1000	0.739±0.006	**0.753**±0.006	0.720±0.005	**0.733**±0.004
1500	0.745±0.006	**0.755**±0.006	0.724±0.004	**0.736**±0.006
2000	0.749±0.006	**0.766**±0.007	0.729±0.008	**0.742**±0.005
2500	0.754±0.006	**0.766**±0.006	0.735±0.007	**0.743**±0.006
3000	0.758±0.007	**0.770**±0.006	0.739±0.005	**0.751**±0.006

Table 3. Comparison between baseline and our approach for different visual vocabulary sizes (with Harris Laplace detector)

WN	Annotation(MAP)		Classification(MAC)	
	Baseline	Our	Baseline	Our
500	0.748±0.005	**0.764**±0.004	0.723±0.006	**0.736**±0.006
1000	0.769±0.004	**0.789**±0.004	0.743±0.006	**0.755**±0.006
1500	0.776±0.004	**0.793**±0.005	0.750±0.006	**0.762**±0.006
2000	0.784±0.005	**0.797**±0.005	0.757±0.007	**0.770**±0.005
2500	0.788±0.005	**0.803**±0.005	0.760±0.006	**0.773**±0.007
3000	0.790±0.005	**0.802**±0.005	0.763±0.006	**0.776**±0.008

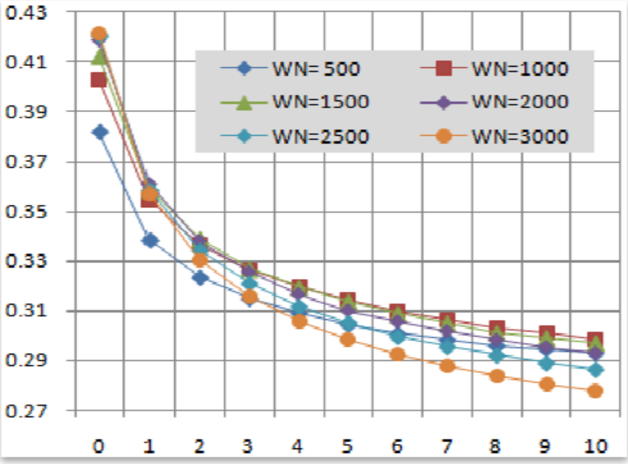


Fig. 3. The reduction of "*border keypoints*" during iterations (DoG keypoints). Vertical ordinate represents the percentage of "*border keypoints*".

Since the proposed approach utilizes spatial information in an earlier stage and with different methods from existing approaches, it should be complementary with existing approaches. That is, the combination of our proposed method with existing approaches that utilizes spatial information shall further enhance the performance. In Table 4, we adopt our proposed approach on the basis of spatial pyramid matching (SPM) [11] method, which utilizes spatial information by concatenating histograms generated from image regions at different pyramid levels. Experimental results show that our proposed approach can further improve the performance of SPM method by providing more accurate visual vocabulary.

Table 4. Combination with SPM method (with WN=1000)

		Baseline	**Baseline+SPM**	**Baseline+SPM+Our**
DoG	MAP	0.739±0.006	0.768±0.006	**0.780**±0.006
	MAC	0.720±0.005	0.748±0.006	**0.758**±0.007
Harris Laplace	MAP	0.769±0.004	0.784±0.006	**0.799**±0.006
	MAC	0.743±0.006	0.771±0.005	**0.785**±0.004
Dense Sampling	MAP	0.805±0.004	0.822±0.006	**0.834**±0.006
	MAC	0.788±0.005	0.814±0.005	**0.821**±0.007

Table 5. Comparison with some state-of-the-art approaches on *15-Scenes* Dataset

Approach	**MAC**
SPM [11], CVPR 2006	0.814±0.005
KC [27], ECCV 2008	0.767±0.004
ScSPM [28], CVPR 2009	0.803±0.009
Our Approach	**0.821**±0.007

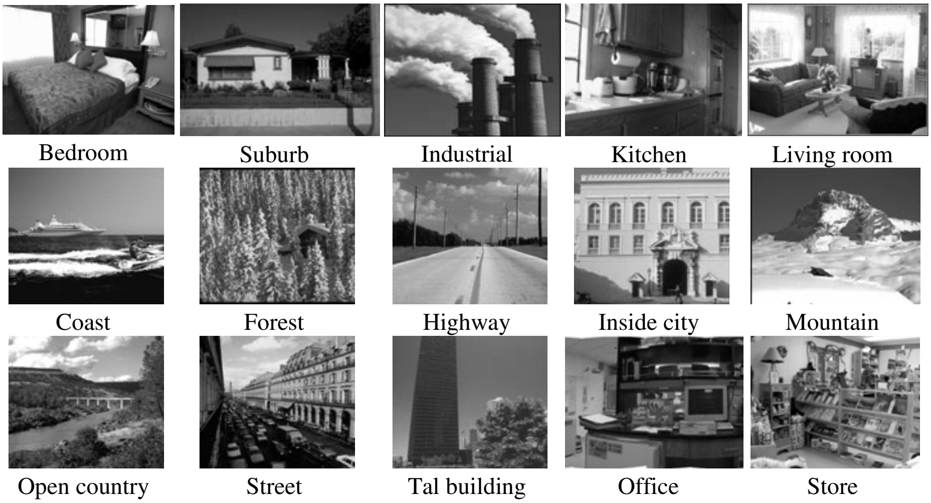


Fig. 4. Sample images of *15-Scenes* dataset (resized due to space limit)

In Table 5, we further compare the performance of our proposed approach with some state-of-the-art methods, where we can see our proposed approach achieves comparable result. For the purpose of fair comparison, in our approach we do not fuse the results with different keypoint detectors, although this is a trivial way to further improve performance.

From the above experiments, we can see that our proposed approach can improve the performance both in independent usage, and in combination with existing methods that utilizes spatial information such as the SPM method, which shows the effectiveness of our approach and its complementation with existing methods. This also supports our argument that spatial information, which has already been considered in several related works, has not been fully exploited yet. Our proposed approach, which uses spatial information at an earlier stage, fills the gap.

5 Conclusion

In this paper, we propose a new visual vocabulary optimization method based on spatial context for image annotation and classification. Different from existing methods that use spatial information after the construction of visual vocabulary, we propose to use spatial information to optimize the visual vocabulary. The novelty of our method mainly lies in two aspects: when spatial information is considered (at an earlier stage, in the optimization of visual vocabulary), and how spatial information is used (with a modified k-means algorithm constrained by spatial context). With the proposed approach, more accurate visual vocabulary can be generated, quantization errors can be reduced, and thus performance improvements can be achieved.

Future work will be carried out focusing on the following aspects: (1) More effective visual vocabulary initialization methods such as Vocabulary Tree [5] and

Approximate K-means [3][4] will be used, generating larger visual vocabulary with more visual words, which could be further optimized by the proposed method. (2) We will combine the proposed approach with other existing methods such as spatial re-ranking to achieve better performance.

Acknowledgements. The work described in this paper was fully supported by the National Natural Science Foundation of China under Grant Nos. 60873154 and 61073084, and the National Development and Reform Commission High-tech Program of China under Grant No. [2010]3044.

References

1. Sivic, J., Zisserman, A.: Video Google: A Text Retrieval Approach to Object Matching in Videos. In: ICCV (2003)
2. Nister, D., Stewenius, H.: Scalable Recognition with a Vocabulary Tree. In: CVPR (2006)
3. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object Retrieval with Large Vocabulary and Fast Spatial Matching. In: CVPR (2007)
4. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in Quantization: Improving Particular Object Retrieval in Large Scale Image Databases. In: CVPR (2008)
5. Lepetit, V., Lagger, P., Fua, P.: Randomized Trees for Real-time Keypoint Recognition. In: CVPR (2005)
6. Yeh, T., Lee, J., Darrell, T.: Adaptive Vocabulary Forests for Dynamic Indexing and Category Learning. In: ICCV (2007)
7. Fischler, M.A., Bolles, R.C.: Random Sample Consensus. Comm. ACM 24(6), 381–395 (1981)
8. Zhang, S., Tian, Q., Hua, G., Huang, Q., Li, S.: Descriptive Visual Words and Visual Phrases for Image Applications. ACM Multimedia (2009)
9. Jiang, Y.G., Ngo, C.W., Yang, J.: Towards Optimal Bag-of-Features for Object Categorization and Semantic Video Retrieval. In: CIVR (2007)
10. Zheng, Y.-T., Neo, S.-Y., Chua, T.-S., Tian, Q.: Visual Synset: a Higher-level Visual Representation for Object-based Image Retrieval. The Visual Computer (2009)
11. Lazebnik, S., Schmid, C., Ponce, J.: Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In: CVPR (2006)
12. Perdoch, M., Chum, O., Matas, J.: Efficient Representation of Local Geometry for Large Scale Object Retrieval. In: CVPR (2009)
13. Jiang, Y.G., Yang, J., Ngo, C.W., Hauptmann, A.G.: Representations of Keypoint-Based Semantic Concept Detection: A Comprehensive Study. In: TMM (2010)
14. Grauman, K., Darrell, T.: Approximate Correspondences in High Dimensions. In: NIPS (2007)
15. Yilmaz, E., Aslam, J.A.: Estimating Average Precision with Incomplete and Imperfect Judgments. In: CIKM (2006)
16. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. IJCV (2004)
17. Mikolajczyk, K., Schmid, C.: A Performance Evaluation of Local Descriptors. TPAMI (2005)
18. Oliva, A., Torraba, A.: Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelop. IJCV (2001)

19. Fei-Fei, L., Perona, P.: A Bayesian Hierarchical Model for Learning Natural Scene Categories. In: CVPR (2005)
20. Wagstaff, K., Cardie, C., Rogers, S., Schroedl, S.: Constrained K-means Clustering with Background Knowledge. In: ICML (2001)
21. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear Spatial Pyramid Matching using Sparse Coding for Image Classification. In: CVPR (2009)
22. Liu, D., Hua, G., Viola, P., Chen, T.: Integrated Feature Selection and Higher-Order Spatial Feature Extraction for Object Categorization. In: CVPR (2008)
23. Ji, R., Yao, H., Sun, X.: Towards Semantic Embedding in Visual Vocabulary. In: CVPR (2010)
24. Ji, R., Xie, X., Yao, H., Ma, W.-Y.: Vocabulary Hierarchy Optimization for Effective and Transferable Retrieval. In: CVPR (2009)
25. Lu, Z., Ip, H.H.S.: Image Categorization with Spatial Mismatch Kernels. In: CVPR (2009)
26. Grauman, K., Darrell, T.: The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features. In: ICCV (2005)
27. van Gemert, J.C., Geusebroek, J.-M., Veenman, C.J., Smeulders, A.W.M.: Kernel Codebooks for Scene Categorization. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 696–709. Springer, Heidelberg (2008)
28. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: CVPR (2009)
29. Qin, J., Yung, N.H.C.: Scene categorization via contextual visual words. Pattern Recognition (2010)
30. Cai, H., Yan, F., Mikolajczyk, K.: Learning Weights for Codebook in Image Classification and Retrieval. In: CVPR (2010)

Colorization Using Quaternion Algebra with Automatic Scribble Generation

Xiaowei Ding*, Yi Xu, Lei Deng, and Xiaokang Yang

Institute of Image Communication and Information Processing
ShangHai Jiao Tong University
ShangHai, China
{dingxiaowei,xuyi, dl0729,xkyang}@sjtu.edu.cn

Abstract. In current colorization techniques, major user intervention is required in the form of tedious, time-consuming scribble drawing. Moreover, color leakage usually occurs across contours and object boundaries. In this paper, we focus on automatic scribble generation and structure-preservation mechanism, which are still open issues of colorization. Firstly, we generate scribbles automatically along points where the spatial distribution entropy achieves locally extreme value. Given the color scribbles, we compute quaternion wavelet phases to conduct colorization along equal-phase lines. These lines across scribbles and monochrome patches locate textures with similar pattern distribution. Contour 'strength' model is also established in scale space to direct color propagation among similar edge structures. Finally, we reconstruct color image patches as vector elements using polar representation in quaternion algebra, well-preserving interrelationship between color channels. The experimental results demonstrate that the proposed colorization method can achieve natural color transitions between different objects with automatically generated scribbles.

Keywords: colorization, automatic scribbles generation, structure-preservation mechanism, quaternion algebra.

1 Introduction

Colorization is an image processing technique which allows us to add colors to monochrome images with the help of manual work. The basic idea of colorization is to set up a mapping from scalar-valued intensity image to vector-valued color image using a few color scribbles provided by users. Therefore, color scribbles should be selected carefully to provide sufficient color reference for monochrome regions. Correct and rapid color propagation should also be triggered from these color scribbles using a color optimization process. In addition, scribbling can be tedious for images with complex details, and requires some skill to obtain natural-looking results.

In scribble-based colorization techniques, most researches are motivated by Levin *et al.* 's work [1], taking the simple premise that nearby pixels in space-time

* Corresponding author.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 103–114, 2012.

that have similar gray levels should also have similar colors. Color leakage tends to occur across contours and object boundaries, lacking structure preservation mechanism. To achieve more natural appearance in colorization results, Drew makes the assumption that the contrast in the colorized image should match the gradient perceived in the original grayscale image [9]. To produce more reliable edge-preserving colorization results, Kim *et al.* enforce color consistency in the areas bounded by the edges [11]. Lezoray's work requires the user to provide more complex scribble guidance to achieve encouraging results, where colorization is considered as a graph regularization problem for a function mapping vertices to chrominance [12]. Heu *et al.* establishes priority propagation mechanism for colorization, ensuring the structure related area with the highest priority can be most reliably colorized [13]. Two basis issues are important in these works, that is, reliable scribble guidance and structure preservation.

Recently, some authors present example-based colorization techniques using one grayscale target image and multiple color reference images. Welsh *et al.* [2] transfer color from reference color image to target monochrome image by matching luminance and texture information between the images. Rather than relying on a series of independent pixel-level decisions, Irony *et al.* [3] develop a new strategy that attempts to account for the higher-level context of each pixel using spatial consistency constraint. X.P. Liu *et al.* focus on tackling illumination differences between grayscale target and color reference images by separating an image into a reflectance (albedo) component and an illumination (shading) component [14]. Sykora *et al.* [10] design a colorization scheme especially suitable for cartoon, which can exploit good image segmentation results prior to colorization. In these methods, the quality of colorized image greatly relies on the similarity between reference image and target image. Moreover, the reference image dataset might be difficult to achieve. In this paper, we are still interested in scribble-based colorization work with attempts to provide automatic scribble generation technique and well preserve image structures during color optimization process. Figure 1 shows the main steps of our colorization algorithm with comparison between our colorization result and the original color image as a rough illustration of our algorithm. The proposed algorithm automatically generates reliable and precise scribbles on input grayscale images based on spatial distribution entropy (SDE). As a result, users only need to decide the color schemes of the scribbles. Given the color scribbles, we compute quaternion wavelet phases to conduct colorization along equal-phase lines. These lines across scribbles and monochrome patches locate textures with similar pattern distribution. A structural priority mechanism based on contour 'strength' model is also established in scale space to direct color propagation among edge structures with different priority. Finally, we reconstruct color image patches as vector elements using polar representation in quaternion algebra, thus, avoid color distortion resulted from isolation of each color channel during colorization process.

The rest of this paper is organized as follows. In Section 2, automatic scribble generation technique is presented in detail. Section 3 proposes a new color optimization process, in which magnitude and phases are separately recovered

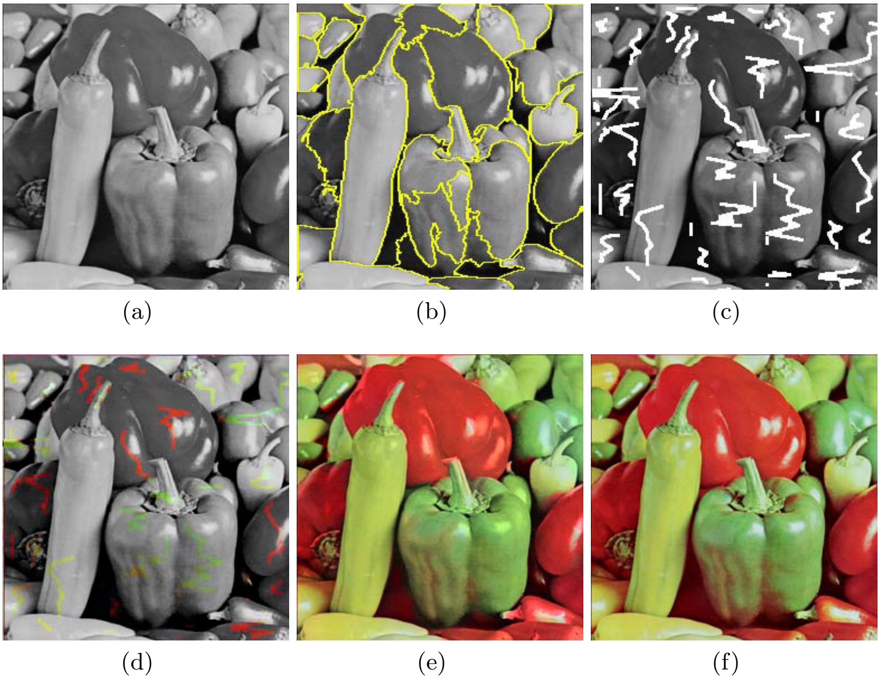


Fig. 1. Example of colorization algorithm in this paper: (a) the input grayscale image: peppers; (b) rough image segmentation for scribble generation; (c) generate scribbles automatically; (d) choose color for scribbles; (e) colorization result; (f) the original color image

under a quaternion color representation framework. Equal-phase lines and contour 'strength' model are computed to direct color propagation among similar image structures. This is followed by the experimental results in Section 4, where a comparison with the state-of-art methods is provided. Finally, conclusion remarks are drawn in Section 5.

2 Automatic Scribble Generation

Scribble plays a significant role in colorization. It contains all the color information that can be used in colorization process. However, images with complex structure need a large amount of careful scribbles by an experienced user, and also, hand-made scribbles will not necessarily trigger the most effective color propagation in consideration of extensive image contents. In this section, we propose an automatic scribble generation algorithm based on spatial distribution entropy, placing scribbles within the regions of high information density. As a result, the requested color information of each homogeneous segment is dominantly contained in the neighborhood of these scribbles.

2.1 The Spatial Distribution Entropy Concept

To measure the spatial distribution of information density in one area, we define the spatial distribution entropy(SDE) at one point in this section. Annular distribution model [5] which performs in a similar way to human visual system is adopted to define the scope of statistics. (see Figure 2 (b)).

Denote:

1. $[p_{xy}]_{M\times N}$ be the image matrix, where p_{xy} is the grayscale value at point (x, y).
2. $U_A = \{(x, y)|(x, y) \in A\}$ be the set of all the pixels in area A.
3. Q be the number of possible values for a single pixel ($Q = 256$ for a typical grayscale image). And the values are represented as $V_1, V_2, ..., V_Q$
4. $S_l = \{(x, y)|(x, y) \in A, p_{x,y} = V_l\}$ be the set of all the pixels in U_A that have value V_l.
5. M be the number of annulars in one scope.
6. $C_l = (x_l, y_l)$ be the center of area S_l where $x_l = \frac{1}{|S_l|} \times \sum_{(x,y)\in S_l} x$, $y_l = \frac{1}{|S_l|} \times \sum_{(x,y)\in S_l} y$.
7. Operator $|\cdot|$ counts the number of elements in a set.

We define the spatial distribution entropy at point p as:

$$E_p = -\sum_{m=1}^{M}\sum_{v=V_1}^{V_Q} \frac{|s_{vm}|}{|U_m|} \times log_2 \frac{|s_{vm}|}{|U_m|} \tag{1}$$

According to the entropy concept in information theory, signals with higher entropy is comparatively harder to predict. SDE measures efficiently this attribute which varies with position parameter in a 2D image signal.

2.2 Scribble Searching Strategy

To avoid one single scribble striding across different homogeneous segments, the image is first oversegmented into closed areas using Graph-Based Image Segmentation algorithm [4]. Then, we perform a search in each segment for the points, around which high information density is accessible. Spatial distribution entropy is adopted to compute information density. In the end, these points are interpolated to form smooth scribbles.

To ensure that there is enough color information for colorization process in a limited number of scribbles, we locate the scribbles at the places possessing high spatial distribution entropy(SDE). Figure 2 intuitively illustrates the details of this procedure. The search starts with the point C_l (the starting point) defined in Section 2.1. Implementation details of adaptive scope radius and selection of scopes in 8 directions are given in caption of Figure 2. Our search strategy allows backtracking of arbitrary number of steps to avoid a bad path brought by any unwise step in this procedure. In addition, it is possible that points are densely selected in a small area due to high entropy. We treat this situation as re-entry and avoid it by measuring the concentration of selected points in

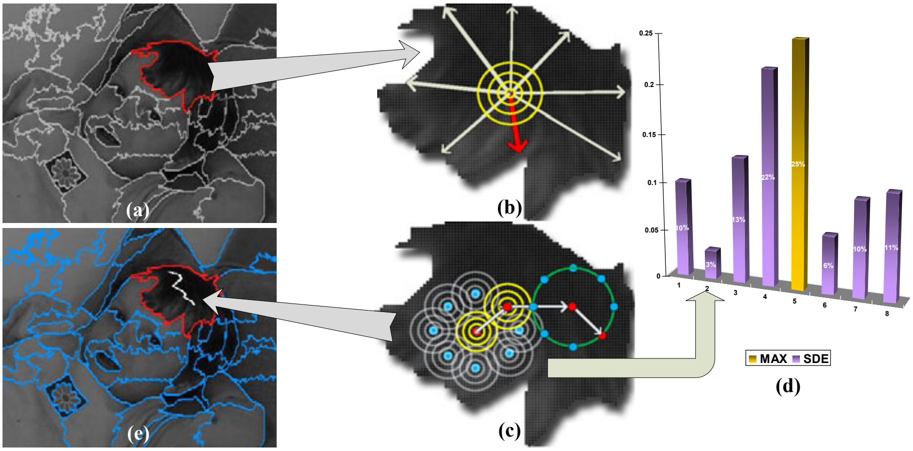


Fig. 2. Implementation details of our scribble generation algorithm (a) Searching procedure are conducted inside one segmented region (red) for each scribble line. (b) Adaptive scope radius is linearly related to the minimum distance between center of scope and region edge (red arrow). (c) Candidate scopes in 8 directions partially overlap with the current(central) scope and the searching path possesses 8 degrees of freedom. (d) Histogram of SDE in 8 directions of the current scope. (e) Fitting all the points on searching path with smooth curve, then one scribble line is generated (the white line in red region).

current scope. To direct the scribble drawing in a homogenous area, we concern about those candidate scopes whose correlation coefficients to current scope are higher than a threshold. Correlations between two scopes are measured by their correlation coefficient of grayscale histograms. Once all the possible scopes in 8 directions fail to satisfy this constraint, we need to backtrack to the previous step to redirect the searching path. Iterate this process until backtracking path is exhausted. We show the flow diagram of our searching strategy in one segment in Figure 3.

3 Colorization Process Based on Quaternion Algebra

This section will first introduce how we characterize image pattern using hyperanalytic signal representation and structural priority in scale space. Then we will focus on color optimization process to propagate color across similar image patches using polar quaternion color formulation.

3.1 Image Structure Analysis Using Quaternion Gabor Phases

Colors often differ in different structural parts in image. Luminance distribution does carry some of the image structure information; however, it cannot act as

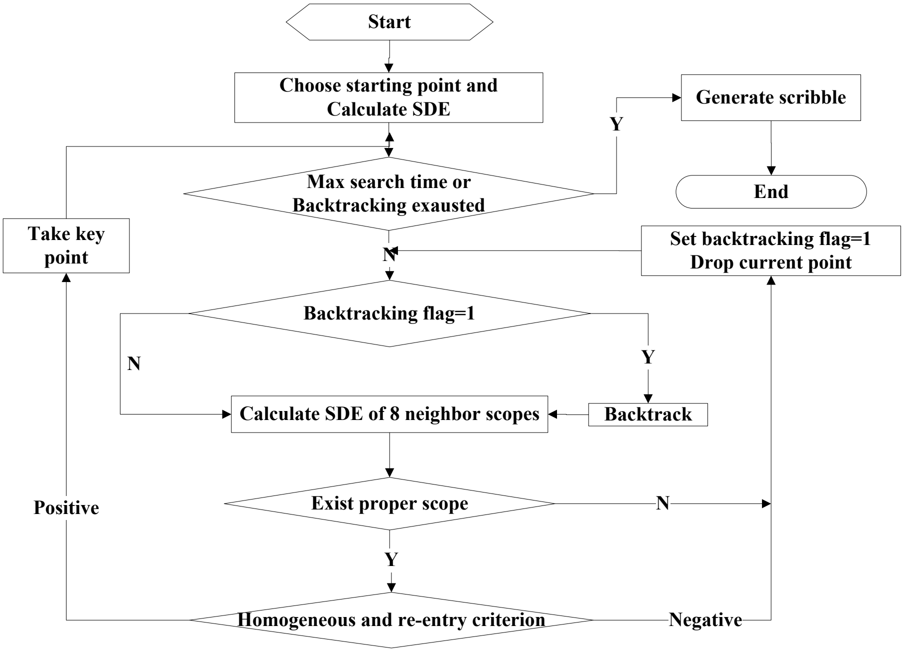

Start
Choose starting point and Calculate SDE
Generate scribble
Y
Max search time or Backtracking exausted
End
Take key point
Set backtracking flag=1 Drop current point
N
Backtracking flag=1
N
Y
Calculate SDE of 8 neighbor scopes
Backtrack
Positive
Exist proper scope
N
Y
Homogeneous and re-entry criterion
Negative

Fig. 3. Scribble searching strategy flow diagram

a useful tool for image analysis under irregular illumination variations. Meanwhile, complex Gabor phases have been extensively exploited to represent image pattern due to the robustness to illumination changes and scale disturbance. In recent work, quaternion Gabor phase is found to provide a better representation of image pattern since it can realize the analysis of intrinsically 2D features (corner-like)[8]. In contrast, complex Gabor phase can only provides a powerful tool for intrinsically 1D features (edge-like) analysis [6]. Considering that (quaternion) Gabor is a windowed (quaternion) Fourier transformation, we can use the Fourier shift theorem to explain it,

$$\begin{aligned}\mathbf{F}\{f(x-x_0,y-y_0)\} &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x-x_0,y-y_0)e^{-i2\pi ux}e^{-i2\pi vy}dxdy\\ &= \mathbf{F}\{f(x,y)\}e^{-i2\pi(ux_0+vy_0)} = \mathbf{F}\{f(x,y)\}e^{-i\Delta\Phi}\end{aligned} \tag{2}$$

$$\begin{aligned}\mathbf{F}^q\{f(x-x_0,y-y_0)\} &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x-x_0,y-y_0)e^{-i2\pi ux}e^{-j2\pi vy}dxdy\\ &= \mathbf{F}^q\{f(x,y)\}e^{-i2\pi ux_0}e^{-j2\pi vy_0} = \mathbf{F}^q\{f(x,y)\}e^{-i\Delta\varphi}e^{-j\Delta\theta}\end{aligned} \tag{3}$$

where $\mathbf{F}\{\cdot\}$ and $\mathbf{F}^q\{\cdot\}$ denote complex Fourier transform and quaternion Fourier transform, separately. It is noted that quaternion Gabor can encode the relative location shifts x_0 and y_0 using two separate phases $\Delta\varphi$ and $\Delta\theta$, while complex Gabor can provide only one phase $\Delta\phi$ and merges this important information.

In this section, we use quaternion Gabor phases to measure structural similarity between two neighborhoods. Firstly, we establish octave-band quaternion Gabors,

$$G^q_{\sigma\alpha}(\mathbf{x}, \mathbf{u}) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}} e^{-i2\pi ux'} e^{-j2\pi vy'} \tag{4}$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix} \tag{5}$$

where $G^q_{\sigma\alpha}$ represents the quaternion Gabor kernel with scale σ and orientation α. Then we get three quaternion Gabor phases φ, ψ, θ following the quaternion algebra rule in [6] strictly,

$$\Phi_{\sigma\alpha}\{\varphi, \psi, \theta\} = arg(I * G^q_{\sigma\alpha}) \tag{6}$$

where $\Phi_{\sigma\alpha}$ denotes quaternion phase vector and I represent the monochrome image.

Now we define a metric to estimate the structural homogeneity between two neighboring pixels $\mathbf{p_a}$, $\mathbf{p_b}$ under scale σ and orientation α,

$$\begin{aligned} H_{\sigma\alpha}(\mathbf{p_a}, \mathbf{p_b}) = & \frac{1}{4}(|\rho(\mathbf{p_a})\varphi^q_{\sigma\alpha}(\mathbf{p_a}) - \rho(\mathbf{p_b})\varphi^q_{\sigma\alpha}(\mathbf{p_b})|_{2\pi}) \\ & + \frac{1}{2}(|\rho(\mathbf{p_a})\theta^q_{\sigma\alpha}(\mathbf{p_a}) - \rho(\mathbf{p_b})\theta^q_{\sigma\alpha}(\mathbf{p_b})|_{\pi}) \\ & + (|\rho(\mathbf{p_a})\psi^q_{\sigma\alpha}(\mathbf{p_a}) - \rho(\mathbf{p_b})\psi^q_{\sigma\alpha}(\mathbf{p_b})|_{\frac{\pi}{2}}) \end{aligned} \tag{7}$$

where ρ is the amplitude of polar quaternion representation, the scalars $\frac{1}{4}$ and $\frac{1}{2}$ are used to keep the same value range of three quaternion phases. It should be noted that mod operator $|\cdot|_\beta(\beta = 2\pi, \pi, \frac{\pi}{2})$ is used to deal with phase wrapping problem when we conduct subtraction operation between two phases. In section 5, we will demonstrate in experiments that quaternion Gabor phase surpass complex Gabor phase in colorization application.

3.2 Scale-Space Contour 'Strength'

In this section, we will introduce a structural priority framework based on scale-space contour 'strength' in order to measure the structural importance of global contours and local contours. In our algorithm, contours are extracted to characterize the structures of objects in an image and act as a compensation of quaternion Gabor phase pattern. Furthermore, global contours and local ones would be differently treated since contours in larger scale scope stand for more critical structures.

We define the importance of a contour as strength and propose an algorithm to calculate strength value. Our algorithm is based on multi-scale canny edge representation [7], where scale factor is denoted as σ_i and the successive scales ranging from $\sigma_1(\sigma_{min})$ to $\sigma_n(\sigma_{max})$ satisfy $\sigma_i = \sigma_1 + (i-1)\Delta\sigma$. As a result, the contour image $C(\sigma_i)$ is expected to contain coarse structures and less fine

Table 1. Algorithm for calculation of Contour 'Strength' of an image

```
Calculate 'Strength' based on Multi-Scale Canny Edge Detection
Initialize: $Strength = C(\sigma_1)$
for $i = 2$ to $n$
  dilate $C(\sigma_i)$ with radius $i - 1$ obtaining a dilated image $d_i$
  for each edge pixel $p(x, y)$ in $d_i$
    if $p(x, y)$ is also the edge pixel of $C(\sigma_{i-1})$
      $Strength(x, y) = Strength(x, y) + 1$
    end if
  end for
  dilate $d_i$ with radius $i - 1$ obtaining another dilated image $d'_i$
  for each non-edge pixel $p(x, y)$ in $d'_i$
    if $p(x, y)$ is the edge pixel of $C(\sigma_i)$
      $Strength(x, y) = Strength(x, y) + i$
    end if
  end for
end for
```

details. Further, a contour should be assigned a large strength value if it has a long lifetime in scale space since these contours always represent the most significant structures of an image. This means that it should have higher priority in colorization process. Since multi-scale contours might not exactly match at the same location throughout the scale space, we use morphology algorithm dilating to achieve more reasonable strength estimation. The proposed algorithm is illustrated in table 1.

3.3 Color Diffusion

The final step in colorization process is color diffusion based on quaternion phase pattern and contour strength designed above. We implement the color diffusion process by minimizing a cost function in which variables are quaternion color phases. In quaternion color space, each pixel is represented by a pure quaternion $q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$ where $a = 0$ and b, c, d are RGB values. In polar coordinates, the quaternion can be formed as amplitude and phases which can be calculated as described in [6].

Vector operation in color space allows different color channels to be treated as a unity rather than channel being independently manipulated. Illumination variation has little impact on quaternion phases of color image. Hence, it is more reasonable to perform colorization by reconstructing quaternion phases vector in the minimization process.

Inspired by Levin's work [1], We build the cost function to be minimized as follows.

$$\Phi^{opt} = Arg \ \min_{\Phi} \sum_{p} \left| \Phi(p) - \frac{\sum\limits_{q \in N(p)} W_{pq} \Phi(q)}{\sum\limits_{q \in N(p)} W_{pq}} \right|, \quad \Phi = \{\varphi, \psi, \theta\} \tag{8}$$

As a matter of fact, the optimization problem is treated as three independent minimization problems about three phase components ϕ, ψ, and γ.

$$W_{pq} = W_{pq}^{Strength} \times W_{pq}^{QGabor} \tag{9}$$

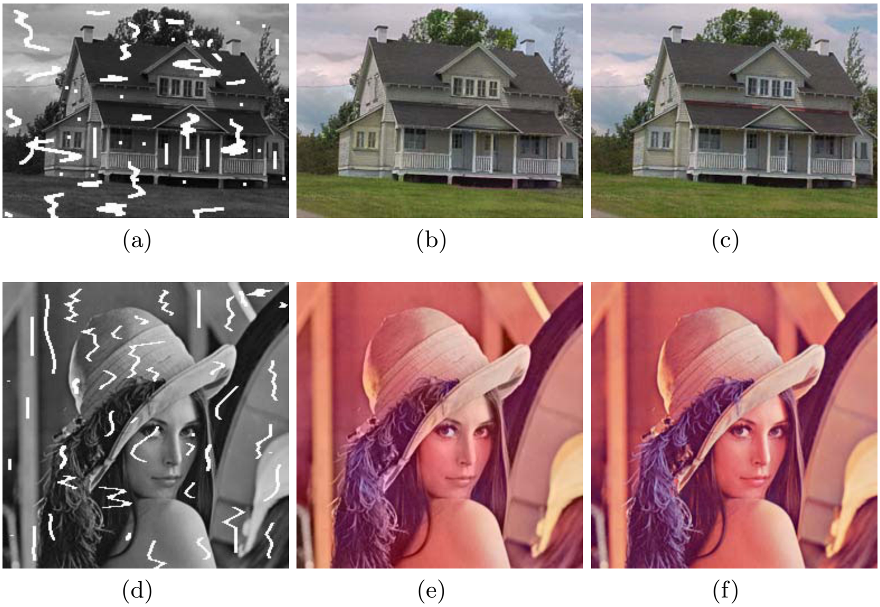
$$W_{pq}^{Strength} = \exp(-|strength(\mathbf{p}) - strength(\mathbf{q})|) \tag{10}$$

$$W_{pq}^{QGabor} = \exp(-\sum_{\sigma}\sum_{\alpha} H_{\sigma\alpha}(\mathbf{p}, \mathbf{q})) \tag{11}$$

Where $N(\mathbf{p})$ is the 3 by 3 neighborhood of pixel $\mathbf{p}$. W_{pq}^{QGabor} is designed to restrict the neighboring pixels in one homogeneous area to have similar color. Value of $W_{pq}^{Strength}$ tend to emphasize the colorization of contours with high strength. It keeps constant in non-contour area as defined in Section 3.2. This weight leads to promising color propagation along the edge. We use Matlab's built- in least squares solver for sparse linear systems.

4 Experimental Results

Here we show our colorization results, where scribbles are generated automatically. Performance comparison with other colorization algorithms is also provided in this section. It should be noted that our algorithm only needs a rough over-segmentation of the input image in automatic scribble generation process. Hence, we simply set the area of the smallest segment part as $M \times N \times 5\%$,



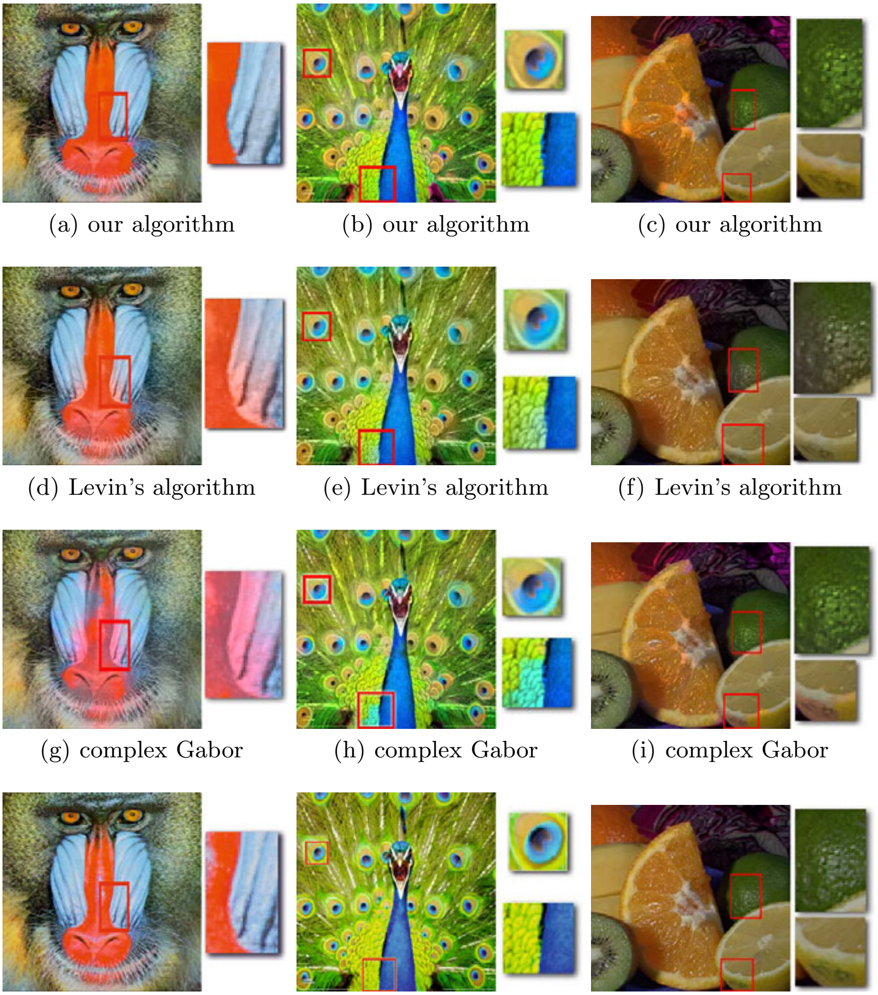
(a) (b) (c)

(d) (e) (f)

Fig. 4. Colorization results compared with the original color images: (a) automatically-produced scribbles (house); (b) colorization result (house); (c) original color image (house); (d) automatically-produced scribbles (Lena); (e) our colorization result (Lena); (f) original color image (Lena)

where $M \times N$ stands for the image resolution. In our simulation, the correlation threshold (see Section 2.2) for the M^{th} scope on the searching path has the value $T_{M+1} = \frac{0.6}{M}\sum_{i=2}^{M} r_i$, where r_i represents the correlation between the i^{th} scope and the $(i-1)^{th}$ one.

Figure 4 gives two colorization results on house image and Lena image, where automatically-produced scribbles are marked using white curves. As shown in Figure 4, scribbles are automatically located in those regions which are supposed to contain critical color structures. No scribble strides across different colors.



(a) our algorithm (b) our algorithm (c) our algorithm

(d) Levin's algorithm (e) Levin's algorithm (f) Levin's algorithm

(g) complex Gabor (h) complex Gabor (i) complex Gabor

(j) Original image (k) Original image (l) Original image

Fig. 5. Comparison with other algorithms; (a), (b), (c): Our algorithm; (d), (e), (f): Levin's algorithm; (g), (h), (i): Algorithm using complex Gabor; (j), (k), (l): Original color image

The original images in the third column are listed to provide visual evaluation of colorization performance. We can see that our algorithm reconstructs the missing color information very well.

In figure 5, we compare our algorithm with those methods depending on intensity similarity or complex Gabor phases, proving that quaternion Gabor phases and scale-space contour strength model can represent fundamental image structures. The quality of the colorization results in figure 5 are measured by PSNR. PSNR of our results: baboon(a) 23.81dB, peacock(b) 20.88dB, fruits(c) 21.73dB. PSNR of Levin's results: baboon(d) 23.58dB, peacock(e) 21.59dB, fruits(f) 21.94dB. PSNR of results using complex Gabor phases: baboon(g) 22.78dB, peacock(h) 19.85dB, fruits(i) 20.80dB. Levin's results and our results have very small differences in PSNR. Results by algorithm using complex Gabor phases are about 1dB lower than ours and Levin's, which leads to unsatisfying visual effects. Comparing with Levin's algorithm in PSNR, our algorithm performs better on baboon image(a), while a little bit worse on peacock image(b) and fruits image(c). However, our colorization results promisingly recover the complex color texture in baboon's villus, peacock plumage and citrus peel. We also achieve almost zero color leakage around object contours, which cannot be achieved by other two algorithms. From the close-up image in Figure 5, we can see that our results have better visual effects than algorithm based on intensity similarity(Levin's algorithm) under similar PSNR. Human visual perception is highly adapted for extracting structural information from a scene [15], which explains why our structure-preserving colorization results are more visually satisfying.

It takes about 11s in average to generate high quality scribbles automatically for each of these input images on PC. Manual scribbles usually take at least several minutes to sketch with professional guidance. Automating this time-consuming process in colorization is user friendly and greatly shortens the overall time of colorization.

5 Conclusion

The evaluation criterion for an outstanding colorization algorithm is whether it can achieve prospective effect requiring as little manual input as possible. Our work achieves this goal by proposing a colorization algorithm based on quaternion algebra with automatic scribble generation. Users are expected to enjoy the convenience of choosing colors for scribbles that have already been created. At the same time, colorization effects are improved in complex color texture regions under quaternion color representation and hyperanalytic signal analysis. Concept of structure priority based on scale-space contour strength together with quaternion phases forms a reliable structure-preserving colorization foundation.

Acknowledgments. This work was supported in part by Research Fund for the Doctoral Program of Higher Education of China (200802481006), NSFC-60902073, NSFC-60932006.

References

1. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. ACM Trans. Graph. 3, 689–694 (2004)
2. Welsh, T., Ashikhmin, M., Mueller, K.: Transferring color to grayscale images. ACM Transaction on Graphics 21(3), 277–280 (2002)
3. Irony, R., Cohen-Or, D., Lischinski, D.: Colorization by example. In: Eurographics Symposium on Rendering, pp. 201–210 (2005)
4. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient Graph-Based Image Segmentation. International Journal of Computer Vision, 167–181 (2004)
5. Rao, A., Srihari, R.K., Zhang, Z.: Spatial Color Histograms for Content-Based Image Retrieval. In: ICTAI, pp. 183–186 (1999)
6. Bayro-Corrochano, E.: The Theory and Use of the Quaternion Wavelet Transform. Journal of Mathematical Imaging and Vision 24, 19–35 (2006)
7. Canny, J.: A Computational Approach To Edge Detection. IEEE Trans. Pattern Analysis and Machine Intelligence 8(6), 679–698 (1986)
8. Xu, Y., Song, L., Yang, X., Traversoni, L., Lu, W.: QWT: retrospective and new Applications. Geometric Algebra Computing for Engineering and Computer Science. Springer, London (2010)
9. Drew, M.S., Finlayson, G.D.: Realistic colorization via the structure tensor. In: ICIP, pp. 457–460 (2008)
10. Sykora, D., Burinek, J., Zra, J.: Unsupervised colorization of black-and-white cartoons. In: 3rd Int. Symp. NPAR (2004)
11. Kim, T.H., Lee, K.M., Lee, S.U.: Edge-preserving colorization using data-driven random walks with restart. In: ICIP, pp. 1641–1644 (2009)
12. Lezoray, O., Ta, V., Elmoataz, A.: Nonlocal graph regularization for image colorization. In: ICPR, pp. 1–4 (2008)
13. Heu, J., Hyun, D., Kim, C., Lee, S.: Image and video colorization based on prioritized source propagation. In: ICIP, pp. 465–468 (2009)
14. Liu, X., Wan, L., Qu, Y., Wong, T., Lin, S., Leung, C., Heng, P.: Intrinsic colorization. ACM Trans. Graph. 27(5), 152:1–152:9 (2008)
15. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image Quality Assessment: From Error Visibility to Structural Similarity. IEEE Trans. Image Processing. 13(4), 600–612 (2004)

Do-It-Yourself Eye Tracker: Low-Cost Pupil-Based Eye Tracker for Computer Graphics Applications

Radosław Mantiuk, Michał Kowalik, Adam Nowosielski, and Bartosz Bazyluk

West Pomeranian University of Technology in Szczecin,
Faculty of Computer Science,
Żołnierska 49, 71-210, Szczecin, Poland
rmantiuk@wi.zut.edu.pl,
http://rmantiuk.strony.wi.ps.pl

Abstract. Eye tracking technologies offer sophisticated methods for capturing humans' gaze direction but their popularity in multimedia and computer graphics systems is still low. One of the main reasons for this are the high cost of commercial eye trackers that comes to 25,000 euros. Interestingly, this price seems to stem from the costs incurred in research rather than the value of used hardware components. In this work we show that an eye tracker of a satisfactory precision can be built in the budget of 30 euros. In the paper detailed instruction on how to construct a low cost pupil-based eye tracker and utilise open source software to control its behaviour is presented. We test the accuracy of our eye tracker and reveal that its precision is comparable to commercial video-based devices.

Keywords: eye tracking, human computer interfaces, eye tracker accuracy, computer graphics.

1 Introduction

Eye tracking is a technique of gathering real-time data concerning *gaze direction* of human eyes. In particular, position of the point, called *point-of-regard*, that a person is looking at is captured. Interestingly, eye tracking is not popular in imaging and computer visualisation applications despite its undeniable potential. A human is able to see details only by the *fovea* - a part of the eye located in the middle of the macula on the retina. Fovea covers about 2° of the human viewing angle, therefore, information about gaze direction can be very useful in many multimedia applications.

In the last decade much work has been done in the study of using eye tracking as a user interface to multimedia systems [5]. Capturing of the visual attention was considered in the context of supporting multimedia learning [6], understanding web page viewing behaviour [7], and many others [9]. Eye trackers are used in real time graphics systems, e.g. in serious games to support activity rehabilitation [8], to reduce computation time (e.g. render with non-uniform pixel

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 115–125, 2012.

distribution [1], simplified scene geometry [2]) or support 3D rendering (e.g. to locate the accommodation plane during depth-of-field rendering [25]). Our goal is to popularise the eye tracking technology in multimedia systems, especially in applications that use the 3D visualisation techniques.

The main issue of the contemporary eye trackers is their high price. A precision below 0.5° of the visual angle (roughly 10 pixels on a 17" display observed from 60 cm distance) is possible to achieve only with the use of very expensive intruisive eye trackers. This type of devices requires that the observer would place her head on the chin rest or use the bite bar that further hinders the practical use of the device. Even less accurate devices with precision of about 1° cost over 20,000 euros [3]. The high cost of the commercial eye trackers seems to stem from the costs incurred in research rather than the price of the hardware components.

In this work we argue that eye tracker with sufficient accuracy can be built in the budget of 30 euros. As a proof of concept, we have designed and built a low-cost head-mounted eye tracker, called *Do-It-Yourself* (DIY) eye tracker. Its construction is based on a web camera and the 3rd party ITU Gaze Tracker software [4]. To estimate accuracy of DIY eye tracker we conduct subjective experiments measuring its precision for a number of observers. As a case of study we test DIY eye tracker in our virtual environment software in which the depth-of-field rendering is controlled by captured gaze direction (see Section 4.5).

The paper starts with a survey of eye tracking techniques (Section 2). In Section 3 we present the DIY eye tracker and describe details of its construction. Section 4 describes the conducted experiment and depicts its results. We conclude the paper in Section 5.

2 Eye Tracking Technologies

Tracking of humans' gaze direction is acquired in numerous ways encompassing intruisive and remote techniques.

Intruisive eye trackers require some equipment to be put in physical contact with the user. In early works a coil embedded into a contact lens was used [10]. The eye gaze was estimated from measuring the voltage induced in the coil by an external electro-magnetic field. In another electro-oculogram technique (EOG) [11] electrodes are placed around the eye. Eye movement is estimated by measuring small differences in the skin potential. In general, intruisive techniques are very accurate and often used in scientific experiments (accuracy reaches 0.08 *deg* of human visual angle) but due to intruisive nature are rather useless in the most computer graphics and imaging applications.

More suitable for vision systems are remote techniques that use cameras to capture image of the eye. Even if they require some intrusive head mounted devices [12, Sect. 6], they are still acceptable for many applications, e.g. for virtual environments and augmented reality.

The most common remote eye trackers apply the *corneal reflection* (CR) method. The eyes are exposed to direct invisible infra-red (IR) light, which results in appearance of Purkinje image with reflection in the cornea (see Fig. 1, left). The reflection is accompanied by image of the pupil. Captured by a video camera sensitive to the infra-red spectrum, the relative movement of both pupil and corneal reflections are measured, which enables the estimation of observer's gaze point. It is reported that commercial eye trackers can achieve the accuracy below 0.5° [3]. The CR eye trackers require calibration to establish a mapping between the reflection-pupil vector and the actual screen-space target point.

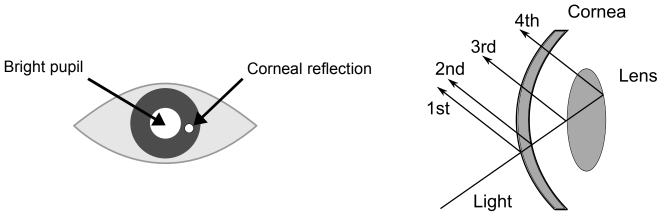


Fig. 1. Left: the corneal reflection in the infra-red light, relative location of the pupil and the corneal reflection are used to estimate observer's gaze point. Right: Purkinje images.

There are eye trackers that simultaneously process more than one corneal reflection. The first Purkinje image (used in CR eye tracking) corresponds to the reflection from the external surface of the cornea. The three remaining images are created by reflections from internal surface of the cornea, and both surfaces of the lens (see Fig. 1, right). In literature various eye tracking systems based on 1st and 4th Purkinje images [13], as well as 3rd and 4th [14] were proposed. The most popular are DPI (dual Purkinje image) eye trackers that estimate gaze point with very high accuracy of about 1 min of arc. Their drawback is the need of using a chin rest and/or a bite bar for head stabilisation [12, Sec. 5.4].

The sufficient eye tracking accuracy can be achieved detecting pupil's centre. In our project, we built the pupil-based eye tracker suitable for many computer graphics tasks including free-walking in virtual environments (if combined with the head tracker system). The detailed description of our eye tracker is presented in Section 3.

A similar low-cost head-mounted eye tracker was constructed by Li et al. [15] (openEyes project). The authors report that accuracy of this CR-based eye tracker is close to 0.6° (for the 4th generation of the device). EyeSecret project (continuation of openEye) presents auto-calibration eye tracker of accuracy about 1° [16]. The mentioned projects were inspired by Pelz et al. [17] work, in which analog camera and mini-DVD camcorder were used to record user's eye. Then, analysis of the video was performed off-line to capture points of regard. In contemporary solutions analog camera and a camcorder can be replaced with a digital camera and wireless data transfer techniques to allow remote connection between an eye tracker and a computer. Another low-cost solution was presented by Augustin et al. in [19]. The authors tested performance of target acquisition and eye typing of the developed webcam-based eye tracker. They assessed the ability of using the eye tracker as a user interface rather than measured its geometric accuracy. Their eye tracker must be held with observer's teeth what seems to be inconvenient for users.

Detailed reviews of eye tracking techniques are presented in [18,20] and [12].

3 Do-It-Yourself Eye Tracker

We designed and constructed a prototype eye tracker called Do-It-Yourself eye tracker (DIY). The main goal of this work was to develop an eye tracker suitable for computer graphics applications. We assumed that this device should base on remote gaze tracking technique, it should be cheap and possible to build with components available at consumer market.

We constructed the eye tracker which can be used for free-walking tasks in virtual environments. However, it would require the head tracker to capture the head position.

The DIY eye tracker operation is based on the detection of centre of the pupil. In our system, the accompanying ITU Gaze Tracker software (see Section 3.2) analyses an infrared image of the eye and locates position of the pupil. Coefficients gathered during the calibration phase are used to compute the gaze position in screen coordinates.

3.1 DIY Hardware

The DIY eye tracker consists of two main components: a modified safety goggles that act as a frame and a capture module attached to the goggles (see Figure 3, right).

The capture module is based on a typical web camera (we used Microsoft Lifecam VX-1000, working in 640x480 pixels resolution). This camera is placed in 5 cm distance from the left eye. The camera should be as small as possible to avoid occluding the observer's field of view. The original VX-1000 camera was modified by removing the chassis and replacing the infrared light blocking filter with the visible light blocking filter. For this purpose we used a fragment of the overexposed analog camera film which filters light in a similar way as infrared filter does, but this solution is much cheaper. In Figure 2 differences between images taken with various filters are presented.

The capture module is equipped with infrared photodiodes to additionally illuminate the eye in the infrared spectrum. Position of photodiodes was carefully chosen to assure correct illumination of the eye and avoid strong corneal reflection which could influence results of the software pupil detection algorithm. We found that three photodiodes (45 mW/sr) spread in the triangle topology around the camera lens give satisfactory results (see Figure 3 left).

The capture module is mounted on the safety goggles. A flexible connection based on aluminium rod allows to adjust position of the camera in relation to the eye. The plastic glass of goggles was removed to avoid image deformation

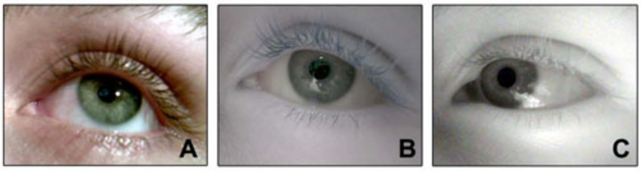


Fig. 2. Image of the human eye, from left: image captured by a regular web-camera, without the infrared light blocking filter, and with visible light blocking filter (with a fragment of the burned analog camera film). Notice that the dark pupil is very clearly visible in the rightmost image.

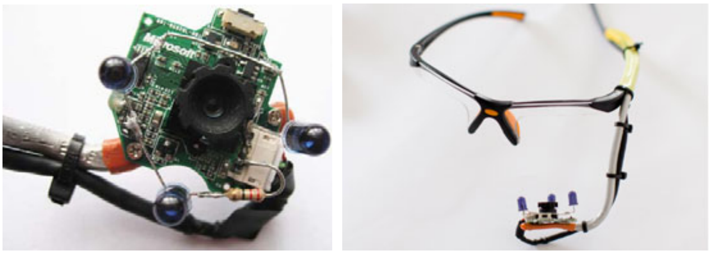


Fig. 3. Left: topology of the photodiodes used to illuminate the eye. Right: DIY eye tracker hardware.

and unwanted reflections. The capture module is connected to computer via the USB cable which acts also as a power source for the camera and photodiodes. Detailed description of the DIY construction is available on the project web site[1].

The total cost of all components needed for building the DIY eye tracker is under 30 euros. The eye tracker can be assembled by a student in a few hours. After installation of a typical USB driver for the camera module, DIY eye tracker is automatically detected by the ITU Gaze Tracker software (see Section 3.2) and there is no need for additional configuration of its software.

3.2 ITU Gaze Tracker Software

We use the ITU Gaze Tracker software [4] to control the communication between a PC computer and the DIY eye tracker and to execute the eye tracking functionalities. The ITU Gaze Tracker software is developed at the IT University of Copenhagen. It is delivered as a C# open source package under the GPLv3 license.

The ITU Gaze Tracker front-end allows to calibrate eye tracker and then computes a current position of the gaze point. The software captures images taken by the DIY camera module. The images are analysed to find the pupil

[1] http://rmantiuk.strony.wi.ps.pl/projects/diy/index.html

centre. Detection of pupil position is supported by the OpenCV package and the algorithm parameters can be adjusted with the ITU Gaze Tracker interface. Pupil detection implemented in ITU is based on image thresholding and points extraction in the contour between the pupil and iris. The points are then fitted to an ellipse using RANSAC technique [21].

Each eye tracking session starts with the calibration procedure. Observer is asked to watch at the target points that appear in different positions on the screen. The target points are displayed one by one in random order. After calibration, a current gaze position in the screen coordinates is computed and transfer using UDP protocol to an external application.

4 Evaluation of DIY Eye Tracker Accuracy

The main goal of the tests was to measure the accuracy of DIY eye tracker. We present detailed description of the measurement procedure and the way in which the achieved data were analysed.

4.1 Participants

Nine observers with an age from 21 to 27 participated in our experiment with an average of 22.8 years, standard deviation 2.044, all male. Eight participants had normal vision, one of them had corrected vision with contact lenses. We asked each participant to repeat the experiment twice. We have performed 18 measurement sessions in total. No session took longer than 4 minutes for one participant to avoid fatigue. Participants were aware that accuracy of the eye tracker is tested, however they were not informed about the details of the experiment.

4.2 Hardware and Software Setup

Our experimental setup is presented in Figure 4. It consists of DIY eye tracker controlled by the ITU Gaze Tracker software (version 2.0 beta) and PC with 2.8 GHz Intel i7 930 CPU equipped with NVIDIA GeForce 480 GTI 512MB graphics card and 8 GB of RAM (Windows 7 OS). The experiments were run on a 22” Dell E2210 LCD display with the screen dimensions of 47.5x30 cm, and native resolution of 1680x1050 pixels (60Hz). The second monitor was used to control the eye tracker through the ITU Gaze Tracker software. Observers sit in the front of the display in 63 cm distance and were asked to use the chin-rest (adopted from the ophthalmic slit lamp). The illumination in the laboratory was subdued by black curtains to minimise the effect of display glare and to focus observers’ attention on experiment tasks.

We developed a software which implements the validation procedure. This software is responsible for communication with external applications (in our case with ITU Gaze Tracker). It collects eye tracking data using the UDP protocol interface, renders graphics, supports user interactions required during experiment, and stores experiment results. The software was implemented in C++ and as Matlab scripts using Psychtoolbox.

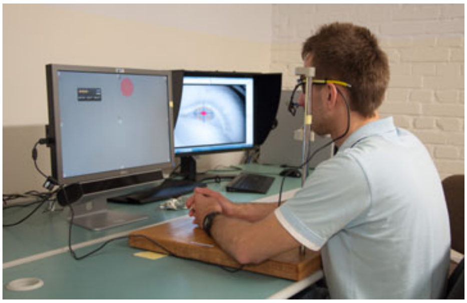


Fig. 4. Hardware setup used for the experiments

4.3 Stimuli and Experimental Procedure

Following [23] recommendation, the experiment started with a training session in which observers could familiarise themselves with the task, interface, chin rest, and how to wear DIY eye tracker. After that session, they could ask questions or start the main experiment.

The experiment started with DIY eye tracker calibration controlled by the ITU Gaze Tracker software. This procedure took about 20 seconds and consisted of observation of the markers displayed in different screen areas. In the next step, the actual validation of eye tracker accuracy was performed. During this procedure controlled by our validation software, participants were asked to look at a set of 25 target points that acted as known and imposed fixation points. As observers used the chin-rest, we knew estimated geometrical position of these points in relation to participants' eyes. The target points were displayed in random order for 2 seconds each. The location of the points on the screen is depicted in Figure 5 (yellow dots).

4.4 Results

Captured positions of gaze points together with positions of corresponding target points were transformed from screen coordinates (pixels) to degrees of the visual angle. We used geometrical dimensions of the hardware setup to compute the transformation, assuming perpendicular view direction at a half of the screen in the horizontal direction and 1/3rd from top in the vertical direction. The gaze direction error angle was computed as a difference between direction towards a target point and towards gaze point captured during observer's fixation on this target point.

The results of the experiment for the individual target points are depicted in Figure 5. Average error for all target points amounts to 1°. Before computation of the average error, we removed 10% of gaze point outliers for every target point. ANOVA analysis did not reveal dependence of the mean results on positions of target points (p=0.0632). However we noticed that the accuracy error is

Table 1. Average error angle in degrees of the visual angle (GP-gaze points)

	DIY eye tracker			RED250 eye tracker		
observer	no. of GP	mean error [°]	std [°]	no. of GP	mean error [°]	std [°]
observer A	5507	0.8125	0.4174	1478	1.2784	0.6363
observer B	5035	1.0866	0.5320	5229	1.2282	0.6177
observer C	3363	1.1619	0.4956	5438	1.0800	0.5968
observer D	5281	1.1492	0.4436	5357	1.2180	0.6147
observer E	5175	1.1365	0.5717	2728	1.4723	0.6794
observer F	4466	1.3932	0.6152	5590	0.9771	0.5242
observer G	4995	1.2167	0.5851	5303	1.2469	0.6350
observer H	5669	0.9424	0.4415	3302	1.4808	0.7697
observer I	5754	0.7510	0.3988	4718	1.2998	0.6247
all observers	45245	1.0557	0.5371	39143	1.2218	0.6425

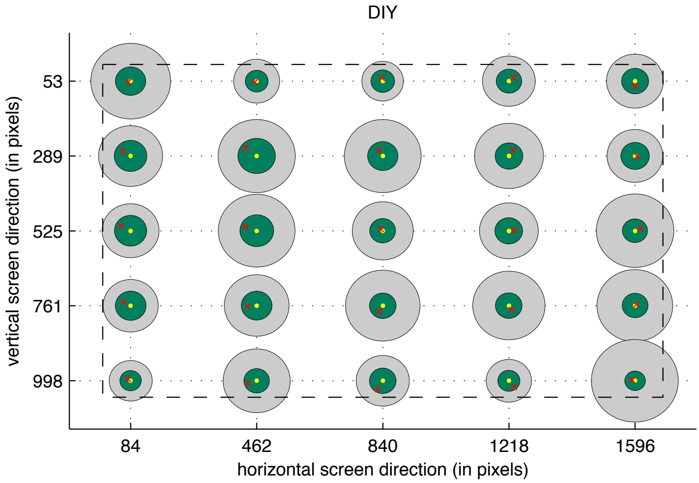


Fig. 5. Gaze direction error measured for 25 target points. The yellow dots denote positions of the target points, the red crosses - positions of median of gaze points (captured individually for every target point), the green circles - median of gaze direction error, and the grey circles - maximum value of gaze direction error (after outliers filtering).

observer dependent. Figure 6 depicts means of the error angle for every observer and their significant statistical difference. In Table 1 number of samples (captured gaze points), average error angles and their standard deviations for every individual observer are presented.

We conducted the same experiment for commercial RED250 eye tracker and achieved average error amounts to about 1.2° which favours our device (see Table 1).

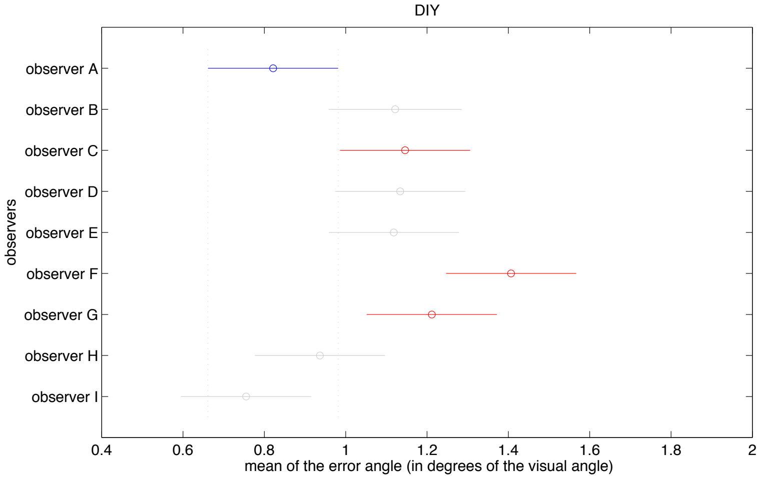


Fig. 6. The average error angle for each observer. The red circle denotes value of the average error angle and the horizontal line is the 95% confidence interval. Note that observers' average gaze directions are significantly different (red lines for observers C,F, and G denote observations significantly different than results for observer A).

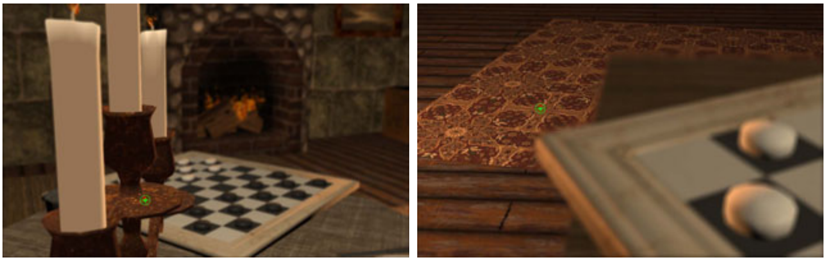


Fig. 7. Example screenshots from the virtual reality renderer

4.5 Application

We tested whether DIY eye tracker can be used to control the depth of field effect rendering in a computer-generated virtual environment. The gaze direction read from the eye tracking system's output can be used to estimate an exact 3D point in the displayed scene. By reading its virtual camera-space depth from the z-buffer, a physical model can be used to calculate the blurriness of different parts of the screen simulating an image viewed through a real optical lens [24]. The user can have a more realistic impression of the scene's depth (see Figure 7).

To measure the actual users' impressions, we have conducted a perceptual experiment (details are presented in [25]). The results show that the gaze-dependent simulation of a depth-of-field phenomenon affects the observer's immersion and has a significant advantage over the non-interactive version of this visual effect. During the experiment we noticed however that the gaze data's accuracy offered by DIY is still inadequate to provide a completely comfortable and realistic simulation comparable with the expected image. The methods for filtering data has to be improved for this use, so as the actual eye tracker's accuracy.

5 Conclusions and Future Work

A price of eye tracking devices inevitably determines the universality of this technology. In this work we describe how to build eye tracker within a very limited budget. We evaluate our eye tracker's accuracy conducting subjective experiments measuring the accuracy of DIY eye tracker for a number of observers. The resulting accuracy (1° of the visual angle) is acceptable for many applications and comparable with similar devices.

The main drawback of DIY eye tracker is the necessity of using a chin rest. We plan to reconstruct our device so that it also supported the head tracking. Interesting solution was proposed in [22] where four infrared photodiodes are located in the corners of a monitor screen. Infrared camera captures image of the eye and reflections of the photodiodes' light are detected in the image. The solution does not require calibration and combines the eye tracking with the head tracking. However, reported accuracy of about one degree of the visual angle could be increased.

Acknowledgements. This work was supported by the Polish Ministry of Science and Higher Education through the grants no. N N516 193537 and no. N N516 508539.

References

1. Peli, E., Yang, J., Goldstein, R.B.: Image invariance with changes in size: the role of peripheral contrast thresholds. JOSA A 8(11), 1762–1774 (1991)
2. Ohshima, T., Yamamoto, H., Tamura, H.: Gaze-directed Adaptive Rendering for Interacting with Virtual Space. In: Proceedings of the 1996 Virtual Reality Annual International Symposium (VRAIS 1996), p. 103 (1996)
3. RED250 Technical Specification. SensoMotoric Instruments GmbH (2009)
4. ITU Gaze Tracker software, IT University of Copenhagen, ITU GazeGroup, http://www.gazegroup.org/home
5. Jacob, R.J.K., Karn, K.S.: Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. The Minds Eye: Cognitive and Applied Aspects of Eye Movement Research (2003)
6. Gog, T., Scheiter, K.: Eye tracking as a tool to study and enhance multimedia learning. Learning and Instructions 20(2), 95–99 (2010)
7. Pan, B., Hembrooke, H.A., Gay, G.K., Granka, L.A., Feusner, M.K., Newman, J.K.: The determinants of web page viewing behavior: an eye-tracking study. In: Proc. of the 2004 Symposium on Eye Tracking Research & Applications (ETRA 2004), pp. 147–154 (2004)

8. Lin, C.S., Huan, C.C., Chan, C.N., Yeh, M.S., Chiu, C.: Design of a computer game using an eye-tracking device for eye's activity rehabilitation. Optics and Lasers in Engineering 42(1), 91–108 (2004)
9. Duchowski, A.T.: A breadth-first survey of eye-tracking applications. Behavior Research Methods 34(4), 455–470 (2003)
10. Robinson, D.A.: A method of measuring eye movements using a scleral search coil in a magnetic field. IEEE Trans. Biomed. Eng. 10, 137–145 (1963)
11. Kaufman, A., Bandopadhay, A., Shaviv, B.: An eye tracking computer user interface. In: Proc. of the Research Frontier in Virtual Reality Workshop, pp. 78–84. IEEE Computer Society Press (1993)
12. Duchowski, A.T.: Eye Tracking Methodology: Theory and Practice, 2nd edn. Springer, London (2007)
13. Cornsweet, T., Crane, H.: Accurate two-dimensional eye tracker using first and fourth Purkinje images. J. Opt. Soc. Am. 63(8), 921–928 (1973)
14. Crane, H., Steele, C.: Accurate three-dimensional eye tracker. J. Opt. Soc. Am. 17(5), 691–705 (1978)
15. Li., D., Babcock, J., Parkhurst, D.J.: OpenEyes: a low-cost head-mounted eye-tracking solution. In: Proceedings of the 2006 Symposium on Eye Tracking Research & Applications, ETRA 2006, pp. 95–100 (2006)
16. Yun, Z., Xin-Bo, Z., Rong-Chun, Z., Yuan, Z., Xiao-Chun, Z.: EyeSecret: an inexpensive but high performance auto-calibration eye tracker. In: Proc. of ETRA 2008, pp. 103–106 (2008)
17. Pelz, J., Canosa, R., Babcock, J., Kucharczyk, D., Silver, A., Konno, D.: Portable eyetracking: A study of natural eye movements. In: Proc. of the SPIE, Human Vision and Electronic Imaging, pp. 566–582 (2000)
18. Morimoto, C.H., Mimica, M.: Eye gaze tracking techniques for interactive applications. Computer Vision and Image Understanding 98(1), 4–24 (2005)
19. San Agustin, J., Skovsgaard, H., Hansen, J.P., Hansen, D.W.: Low-cost gaze interaction: ready to deliver the promises. In: Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems, pp. 4453–4458 (2009)
20. Morimoto, C., Koons, D., Amir, A., Flickner, M., Zhai, S.: Keeping an Eye for HCI. In: Proc. of the XII Symposium on Computer Graphics and Image Processing, pp. 171–176 (1999)
21. Agustin, J.S., Mollenbach, E., Barret, M.: Evaluation of a Low-Cost Open-Source Gaze Tracker. In: Proc. of ETRA 2010, Austin, TX, March 22-24, pp. 77–80 (2010)
22. Yoo, D.H., Chung, M.J., Ju, D.B., Choi, I.H.: Non-intrusive Eye Gaze Estimation using a Projective Invariant under Head Movement. In: Proc. of the Internat. Conf. on Automatic Face and Gesture Recognition, Washington, DC, pp. 94–99 (2002)
23. ITU-R.REC.BT.500-11: Methodology for the subjective assessment of the quality for television pictures (2002)
24. Riguer, G., Tatarchuk, N., Isidoro, J.: Real-time depth of field simulation. ShaderX2: Shader Programming Tips and Tricks with DirectX 9.0, 529–579 (2002)
25. Mantiuk, R., Bazyluk, B., Tomaszewska, A.: Gaze-Dependent Depth-of-Field Effect Rendering in Virtual Environments. In: Ma, M. (ed.) SGDA 2011. LNCS, vol. 6944, pp. 1–12. Springer, Heidelberg (2011)

Symbiotic Black-Box Tracker

Longfei Zhang[1,2], Yue Gao[2,3], Alexander G. Hauptmann[2], Rongrong Ji[4],
Gangyi Ding[1], and Boaz Super[5]

[1] School of Software, Beijing Institute of Technology, Beijing, China
[2] School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA
[3] Department of Automation, Tsinghua University, Beijing, China
[4] Department of Electronic Engineering, Columbia University, New York, NY, USA
[5] Motorola Senior Research, Schaumburg, IL, USA
{longfeizhang,alex}@cs.cmu.edu, kevin.gaoy@gmail.com

Abstract. Many trackers have been proposed for tracking objects individually in previous research. However, it is still difficult to trust any single tracker over a variety of circumstances. Therefore, it is important to estimate how well each tracker performs and fusion the tracking results. In this paper, we propose a symbiotic black-box tracker (SBB) that learns only from the output of individual trackers, which run in parallel, without any detailed information about these trackers and selects the best one to generate the tracking result. All trackers are considered as black-boxes and SBB learns the best combination scheme for all existing tracking results. SBB estimates confidence scores of these trackers. The confidence score is estimated based on the tracking performance of each tracker and the consistency performance among different trackers. SBB is employed to select the best tracker with the maximum confidence score. Experiments and comparisons conducted on the "Caremedia" dataset and the "Caviar" dataset demonstrate the effectiveness of the proposed method.

Keywords: Object tracking, information propagation, data association, multi-tracker.

1 Introduction

Effective and efficient object tracking is a challenging task in computer vision applications [1] and real time multimedia application. Extensive research efforts have been dedicated to object tracking in recent years. However, it is still difficult to track objects accurately in real situations.

There have been many trackers proposed by different researchers [4,5,6,7,8] that employ different types of visual information, e.g., optical flow, color information, etc. Extended Kalman filtering is widely used in tracker fusion[14,15,16,17]. For object tracking, tracker fusion is a hot topic about how to improve the tracking performance. Generally, tracker fusion can be divided into two categories: multi-sensor (camera) fusion [2] and multi- tracker fusion [3], which is also referred to as multi-sensor fusion or soft multi-sensor fusion. Multi-sensor fusion utilizes different views/videos from multiple cameras (sensors) to locate the target object positions. Multi-tracker fusion aims to optimally generate the object track information using multiple single tracker results.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 126–137, 2012.

Generally, traditional fusion methods get tracking results by learning weights of different trackers and weights of their tracking results. However, these traditional methods need to create the pixel level analysis, which process has high computational cost and therefore limit their real time applications.

In this work, we propose an efficient spatial-temporal multi-tracker fusion algorithm for object tracking without requiring detailed information of individual trackers. Different from existing methods, our proposed symbiotic black-box tracking (SBB) is formulated as a two-stage information propagation procedure. One stage is the frame-to-frame (F2F) intra-tracker prediction, and the other stage is the tracker-to-tracker (T2T) propagation procedure. Our objective is to estimate confidence scores, which are used to select the best tracking results. Since the proposed method only requires tracking results from other trackers for tracking fusion, it is efficient and suitable for real time tracking applications.

The rest of this paper is organized as follows. In Section 2, we briefly review related work. Section 3 introduces the proposed spatial-temporal multi-tracker fusion algorithm. Experimental results on the “Caremedia” dataset and the “Caviar” dataset are provided in Section 4. Section 5 concludes the paper.

2 Related Work

Multi-tracker fusion aims to optimally generate object track information using multiple single tracker tracking results (e.g. [3]). Many works track multiple objects at the same time [2].

Ensemble tracking [3] and Multiple Instance Learning [10] employ Ada-boost or SVM to train models from image pixels. The pixel features inside the bounding box are used as positive samples and pixel features outside the bounding box are used as negative samples. In [3], the best target bounding box is selected using mean-shift. However, this selection is problematic because the mean-shift clustering method can potentially lose the original object information if there is no consistent template. Du [8] propose a Linked Hidden Markov Model that combines particle filters with different cues and belief propagation in a Hidden Markov Model, while the correlation of different trackers is ignored. Grabner et al. [9] employ online boosting and evaluate a large pool of 250 weak classifiers. Zhong et al. [7] use a graphic model, which infers the true bounding box at time T and an estimate of “image difficulty”, as well as tracker accuracy using an Expectation-Maximization algorithm based on the output bounding boxes from all trackers only. However, their online learning is computationally expensive and hard to perform in real time.

Rasmussen et al. [4] combine multiple trackers using tracker confidence based on distinctiveness and occlusion probability with a probabilistic data association filter, which is based on Kalman filtering. Stenger et al. [6] propose a method to select fusion methods for multiple different tracker results using training data about the trackers gathered earlier The problem with this method is it is an off-line tracking method. Multiple features including color information are employed in [5] to evaluate several trackers in parallel and subsequently select one by switching between them.

In contrast to these methods, our proposed method is computational effective. SBB focuses on how to enable real time tracking with multiple symbiotic trackers running

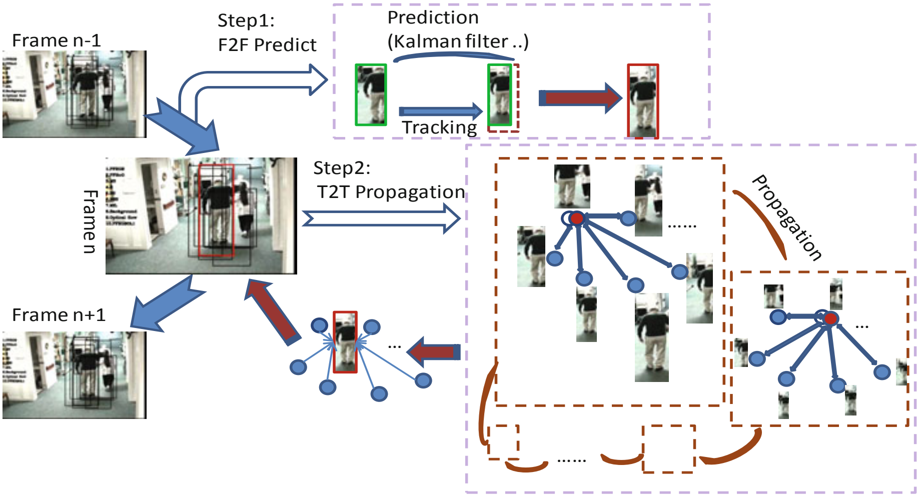


Fig. 1. The framework of the proposed SBB algorithm

in parallel. Our approach employs the same trackers throughout. Our intuition is to estimate the confidence of each tracker based on the tracking performance of the tracker (F2F confidence) and the consistency of tracking performance among different trackers (T2T consistency). We propagate the outputs of all trackers using the confidence of each tracker and the correlation between each tracker pair to evaluate the trackers in parallel. Without processing any detailed features, such as pixel information, and without any online learning, the computational cost is linear in the number of trackers. Therefore, the system is able to run in real time.

3 Symbiotic Black-Box Tracking

3.1 The Framework

In this section, we introduce the proposed SBB tracking algorithm. First, we define the tracker features that employed in the proposed algorithm, and then describe the temporal correlation analysis by F2F intra-tracker prediction and the spatial correlation analysis by T2T propagation. Finally, the Maximum Likelihood tracker selection using confidence and consistency of all trackers is presented.

The architecture of SBB is shown in Figure 1. In the first step, green rectangle is tracking status of the tracking result in frame $n-1, n$, and $n+1$ by individual tracker $T_i(i = 1, \ldots, M)$. The red dashed boxes and red solid boxes are predicted result by F2F prediction. In the second step,the blue circles are the status of individual trackers, and these circles are alternated by the blue solid circles, which are tracker status initialed by step 1 and updated by the T2T propagation processing. The red solid circles with blue edges are the consensus bounding boxes generated by neighbor trackers' status. Brown arrows are propagation procedures. Brown arrows with blue edges are the consensus generation procedures.

3.2 Features for Trackers

Here we introduce the definition of “features” for trackers. Assume there are M trackers, and each tracker has its own approach, which influences the correctness of the results at each frame, we define following “features” for the i-th tracker.

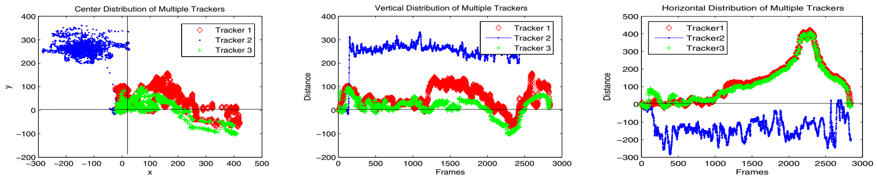
- Bounding box Center c_i^n. Bounding box center of the i-th tracker in frame n obeys a corresponding distribution $c_i^n\sim p(c_i^n)$. The distribution of correct object center is formulated as a Gaussian model: $c_i\sim p(c|\mu_i^n,\sigma_i^n)$, which shows the probability of the true bounding box center's consensus with the detected bounding box center for the ith tracker.
- Vertical/horizontal information for bounding box v_i^n/h_i^n .Vertical/horizontal information is represented with a corresponding distribution $v_i^n\sim p(v_i^n)$, $h_i^n\sim p(h_i^n)$. The vertical/horizontal distributions reflect all possible correct object bounding box distributions for each tracker's results. The distribution is formulated by Gaussian model:$v_i^n\sim p(v|\mu_i^n,\sigma_i)$ and $h_i^n\sim p(h|\mu_i^n,\sigma_i)$.
- F1 score. The F1 score ($F_1(.)$) combines Precision ($P(.)$) and Recall ($R(.)$), and it scales between 0 and 1. A higher F1 score (close to 1) indicates better tracking performance. Denote b_i as the bounding box of the to-be-evaluated tracker, and b_j as the reference bounding box. Then F1 score is defined by:

$$F_1\left(b_i, b_j\right) = \frac{2 \times P\left(b_i, b_j\right) \cdot R\left(b_i, b_j\right)}{P\left(b_i, b_j\right) + R\left(b_i, b_j\right)}, \tag{1}$$

$$P\left(b_i, b_j\right) = \frac{A\left(b_i \cap b_j\right)}{A\left(b_j\right)}, R\left(b_i, b_j\right) = \frac{A\left(b_i \cap b_j\right)}{A\left(b_i\right)}, \tag{2}$$

where $A(.)$ is the area operator.

In our work, following kernels are employed to measure the transition influence values between different nodes of our tracking graph. The nodes are the confidence scores of each tracker that are shown in Figure 2. Using different kernels, distance measurement differs.



(a) (b) (c)

Fig. 2. Feature distribution of three trackers. Tracker 1, in red, is the particle filter using RGB feature. Tracker 2, in green, is the Mean-shift, and tracker 3, in blue, is the particle filter using HOG feature. (a) Center error bounding box distribution. X and Y axes are the Euclidian distance between tracking results (bounding box) and groundtruth. (b) Vertical error bounding box distribution. X axis is the error between the trackers and groundtruth, and Y axis is the frame number. (c) Horizontal error distribution of bounding boxes. X and Y axes are defined the same as (b).

3.3 F2F Tracker Prediction

The features of each tracker show the distribution of the correct bounding box information generated by the tracking results. An illustration is shown in Figure 2.

To exploit the temporal correlation within each tracker, F2F tracker prediction employs the previous tracking results to predict the trackers' confidence score for the current frame. The framework of the F2F tracker prediction for the i-th tracker is shown in Figure 2 and introduced in Section 3.1.

Here H previous frames are selected to estimate the current object position by a Kalman filter tracker like dynamic model [13]. Denotes S_i^n as the confidence score of the the i-th tracker in the n-th frame. Denotes the dynamic model built by H historical bounding boxes of the i-th tracker as $b_i^n(x_i^n, y_i^n, v_i^n, h_i^n)$. A trace $b(x,y,v,h) = f(t)$ is generated to fit these consensus of bounding boxes, and the virtual bounding box $b'(x', y', v', h')$ for the current frame will be predicted using the curve function.

Given the i-th tracker's tracking result, the normalized distance between $b_i^{n-1}$and b_i^n will be used to estimate the relationship between previous tracking results and the current tracking result of the i-th tracker in nth frame. Here the F2F temporal transition value tff_i^n employs the F1 score to model this relationship:

$$tff_i^n = F_1(b_i^n, b_i^m)(m \in Z). \tag{3}$$

This transition method guarantees that only nearby detected results will be used to enhance tracker confidence.

Based on these transition values and previous confidence scores of different trackers, the confidence scores for different trackers in the n-th frame S_i^n are calculated by:

$$S_i^n = \omega_i S_i^{(H)} + (1 - \omega_i)\, tff_i^n S_i^{n-1} \tag{4}$$

where ω_i is the weight for the i-th tracker and $w \in [0, 1]$.

3.4 T2T Propagation

To address the spatial correlation of different trackers, T2T propagation takes the relationship among different trackers within the current frame into consideration. The T2T spatial propagation procedure is shown in Figure 2. Given the different tracking results in one frame, we first define a T2T transition value between the ith tracker and the jth tracker as ttt_{ji}^n. Suppose these confidences of trackers' (S_i^n) are from some underlying manifold. We expect that each bounding box of trackers and its neighbors to lie on or close to a locally linear patch of the manifold too. Therefore, we characterize the local geometry of these patches by linear coefficients that reconstruct each bounding boxes from their neighbors. Reconstruction errors are measured by the cost function:

$$\varepsilon(ttt_i^n) = \sum_i \left| S_i^n - \sum\nolimits_j ttt_{ji}^n S_j^n \right|^2, \tag{5}$$

This function adds up the squared distances among all the bounding boxes and their reconstructions. The wight ttt_i^n summarizes the contribution of the j-th reconstruction. To compute the weight ttt_{ji}^n, we minimize the cost function subject to two constrains:

First,each bounding box's confidence S_i^n is reconstructed exclusively from its neighbors, enforcing $ttt_{ji}^n = 0$, if S_i^n is not belong to the set of neighbors, in another word, the bounding box is too far away from the other bounding boxes. Second, the sum of rows of the weight matrix equals to one: $\sum\limits_{j}^{N} ttt_{ji}^n = 1$.

In our work, $K(c_i^n,c_j^n)$ is calculated by:

$$ttt_i^n = \begin{cases} \exp\left(-\frac{d^2\left(c',c_i^n\right)}{\sigma_{tt}^2}\right) & if\ d\left(c',c_i^n\right) < T_1 \\ 0 & otherwise \end{cases} . \tag{6}$$

The confidence score of each tracker is computed by:

$$S_i^{n^k} = \alpha_1 S_i^{n^{k-1}} + \left(1-\alpha_i\right)\frac{1}{M-1}\sum_{j\neq 1}\overline{t_{tt}^{ji}}S_j^{n^k} \tag{7}$$

where k is the propagation iteration times, S_i^n equals to $S_i^{n^k}$ after converge, $c_i^n\left(x_i^n,y_i^n\right)$ and $c_j^n\left(x_j^n,y_j^n\right)$ are the detected object centers for the i-th tracker and the j-th tracker, respectively.

The transition value ttt_{ij}^n is determined by the distance between detected object results, i.e., the two trackers with closer object centers have larger transition values. This transition method guarantees that only nearby detected results will be used to enhance related trackers. Transition values will be normalized as $\overline{ttt}_i^n = \frac{ttt_i^n}{\sum_j ttt_j^n}$ and $\sum\limits_{j}^{N}\overline{ttt}_i^n = 1$. Based on these T2T transition values, the confidence scores for different trackers are updated with the equation 7.

We employ the output confidence scores S_1^n, S_2^n, ..., S_M^n to generate the final SBB tracking result.

3.5 The Tracking Fusing Strategy

The final SBB result is obtained by selecting the tracker with the maximum confidence with the consensus of the other trackers. The size of the consensus is generated by the expectation of the above distributions. This result is the default result of SBB.

4 Experiments

4.1 Experiment Setting

We evaluate the effectiveness of our proposed approach on a subset of the Caremedia surveillance dataset [20] and the "Caviar" dataset [21].

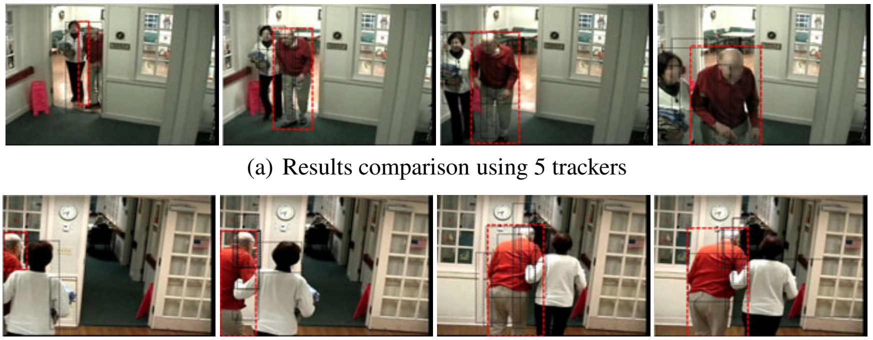
The "Caremedia" dataset is obtained from 23 cameras recording activities in a local nursing home over the course of multiple days. In our evaluation dataset, two residents are tracked from a stationary initial position until they disappear from the view. In some video, only parts of the subject's body appears. There are 13 clips of surveillance video, with the resolution of 720×480. The targets' appearances vary significantly. Frame

Table 1. Properties of the Caremedia Dataset

Video ID	Number of frames	Half tracker lost (%)	Average Lost(frames)	Average F1 score	Best F1score
c102	538	75.0929	103.2	0.4083	0.7311
c102g	199	0.0000	20	0.7156	0.9198
c102m	896	29.9107	180.8	0.3896	0.8334
c102r	2842	74.7713	1012.8	0.3033	0.81
c102w	89	0.0000	15.4	0.6198	0.7856
c106	483	81.7805	197	0.3058	0.5498
c122	187	35.2941	45.2	0.2694	0.4085
c131	348	0.0000	60.6	0.5289	0.7492
c197	404	22.0297	85.6	0.2856	0.4176
c198	190	90.5263	55.8	0.4687	0.7476
c206	1585	97.1609	559.8	0.3788	0.9422
c211	437	76.4302	147.2	0.3238	0.5768
c216	412	88.5922	143.8	0.2936	0.6497
Average	**662.30**	**51.66**	**202.09**	**0.40**	**0.71**

Table 2. Tracker list

No.	Name	Description	No.	Name	Description
1	PFRGB	Particle filter based on RGB feature[19]	6	BSS	Beyond semi-supervised [12]
2	PFHoG	Particle filter based on HOG feature	7	MIL	Multiple instance learning tracker[10]
3	MS	Mean-shift[18]	8	BG	Background tracker
4	B	Online Boosting tracker[9]	9	OF	Optical flow tracker
5	SS	Semi-supervised tracker[11]	10	PFRGBOL	Online update with last frame



(a) Results comparison using 5 trackers

(b) Results comparison using 10 trackers

Fig. 3. Comparison of tracking results using two different tracker sets on the "Caremedia" dataset. The red rectangle shows SBB's result, and the black rectangle shows the individual tracker's result.

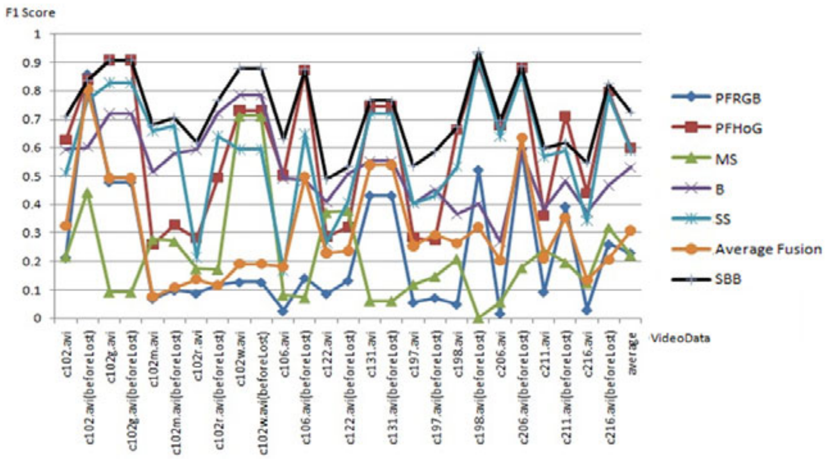


Fig. 4. SBB results using the first set of 5 trackers. In the graph, the black solid line is the performance of our approach.

numbers in these videos range from 89 frames to 2842 frames. Table 1 shows the properties of videos from "Caremedia" dataset.

The "Carviar" dataset focuses on the city center surveillance, and wide angle lens was widely used in this dataset. There are 419 person tracking tasks in 79 videos from "Carviar" dataset.

In our experiments, we used 10 trackers (see Table 2). In order to evaluate the fusion tracking result, we set 2 tracker sets. Tracker set 1 has tracker No.6 to No.10, tracker set 2 has tracker No.1 to No.10. Table 2 lists the 10 trackers that been used in our tracking fusion task.

Figure 3 shows the comparison of tracking results using two different tracker sets on the "Caremedia" dataset.

To get a clear idea of which tracker has the best result. We test the five trackers in tracker set 1 and SBB tracker on the "Caremedia" dataset. Figure 4 shows the results. It can be seen that SBB has the best result, except for 2 video clips, where it has the second best result.

4.2 Parameter Tuning

We further explore the influence of different parameters on the proposed SBB framework. In our experiment, we use different weights in the F2F and the T2T procedures to measure influence of different parameters.

The results of calculating t^i_{ff} and t^{ij}_{tt} using the F1 score and the results distribution of different parameters is shown in Figure 5. The overall performance comparison is given in Table 4, using the Caremedia dataset, and selecting different tracker sets to get the result with estimated w and α weights in the F2F step and the T2T step. Figure 5 shows the tracking results(F1 score) distribution with different parameter settings, where $w \in [0, 1]$ and $\alpha \in [0, 1]$. When $\omega = 0.7$ and $\alpha = 0$, the proposed method achieves the best tracking performance.

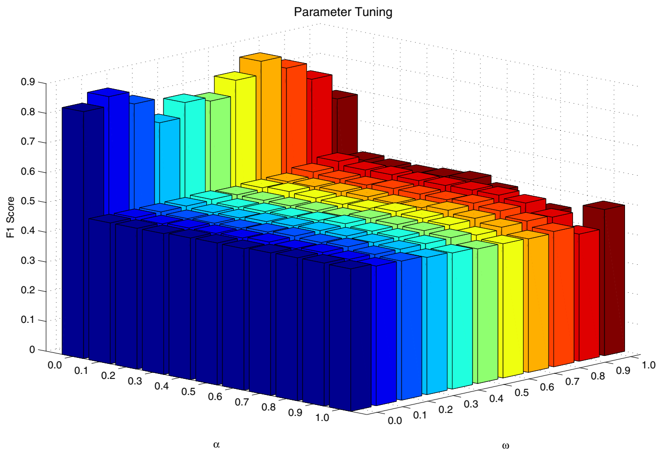


Fig. 5. The tracking results in terms of F1 score with different parameter settings

4.3 Comparison with Other Methods

To evaluate the effectiveness of the proposed method, following methods are employed as the compared methods.

- Individual tracker (best). In this method, the best tracking result from all trackers for each frame is selected.
- Trackers with average score. In this method, all tracker are fused with the same confidence score.
- Simple fusion. Simple fusion [17] uses the average distribution of each tracker result for the final hypothesized bounding box.
- Kalman filter.

Table 3. The tracking performance improvement by SBB in terms of F1 score compared with other methods in "Caremedia" dataset

	SBB's performance improvement	
	Tracker Set 1	Tracker Set 2
Individual tracker(best)	+28.89%	+26.63%
Simple fusion	+41.55%	+38.6%
Kalman filter[17]	+10.53%	+10.16%

Comparisons with other methods on the two testing datasets are provided in Table 3 and Table 4. As shown in these two tables, for two tracker sets, SBB has the best performance among individual trackers, Simple fusion, and Kalman filter.

In Table 5, we estimate the SBB performance in two tracker sets. This SBB tracker using the F1 score with a setting of weight w =0 and α = 0.3. "Lost target" indicates

Table 4. The tracking performance improvement by SBB in terms of F1 score compared with other methods in “Caviar” dataset

	SBB’s performance improvement	
	Tracker Set 1	Tracker Set2
Individual tracker(best)	+20.64%	+28.38%
Simple fusion	+55.55%	+52.23%
Kalman filter	+28.74%	+26.53%

the average number of frames where SBB loses the target. “Trackers lost aver.” shows the average number of frames when the target is lost. “Trackers lost min.” is from the tracker which loses the least frame of target. “Trackers lost max.” is the opposite. “SBB improvement over max.” denotes the percent improvement in the SBB’s F1 score over the best individual tracker’s F1 score. “SBB improvement over aver.” denotes the percent improvement in the SBB’s F1 score over the individual trackers’s average F1 socre. Time cost per frame is SBB algorithm’s processing time duration per frame. SBB processes 6 tracker hypotheses in one frame when using tracker set 1, and processes 10 tracker hypotheses when using tracker set 2.

Table 5. The tracking performance comparison of the proposed SSB method and other methods on “Caremedia” dataset

	Tracker set 1	Tracker set 2
Time cost per frame (s)	0.0084	0.0203
SBB Lost target (frame numbers)	4.53	3.07
Trackers lost aver.(frame numbers)	202.09	226.7462
Trackers lost min.(frame numbers)	10.53	10.5385
Trackers lost max.(frame numbers)	585.38	606.46
SBB F1 Score	0.9028	0.9029
SBB improvement over max.	+28.89%	+26.63%
SBB improvement over aver.	+53.08%	+53.08%

5 Conclusion

In this paper, we propose a Symbiotic Black-Box (SBB) approach for object tracking. Compared to other individual trackers that run symbiotically, SBB focuses exclusively on the bounding boxes that are generated by each individual tracker. The weights of the tracking hypotheses of each tracker are generated from both an F2F intra-tracker prediction and T2T inter-tracker propagation. The best results are obtained using the maximum confidence individual tracker for the current frame. Experiments conducted on the “Caremedia” dataset and “Caviar” dataset show that SBB improves the tracking results significantly . In the 5 and 10 trackers experiments, SBB achieves better tracking performance than the best individual tracker by more than 20%. In contrast to other multiple sensor fusion tracking methods, SBB does not require significant time learning a model based on detailed of tracking features. Therefore, it is appropriate for real-time applications.

Acknowledgments. This material is based upon work supported by the National Science Foundation under Grants No. IIS-0917072, IIS-0812465, and CNS-0751185, and by the National Institutes of Health (NIH) Grant No. 1RC1MH090021-0110. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or the National Institutes of Health.

References

1. Yilmaz, A., Javed, O., Shah, M.: Object Tracking: A Survey. ACM Journal of Computing Surveys 38(1) (2006)
2. Reid, D.: An algorithm for tracking multiple targets. IEEE Trans. on Automatic Control, IEEE society 24(6), 843–854 (1979)
3. Avidan, S.: Ensemble Tracking. IEEE Transaction of Pattern Analysis and Machine Intelligence 29(2), 261–271 (2007)
4. Rasmussen, C., Hager, G.: Joint probabilistic techniques for tracking objects using multiple visual cues. In: Porc. Intelligent Robots and Systems, vol. 1, pp. 191–196 (1998)
5. Badrinarayanan, V., Perez, P., Clerc, F.: Probabilistic color and adaptive multi-feature tracking with dynamically switched priority between cues. In: Proc. International Conference on Computer Vision (2007)
6. Stenger, B., Woodley, T., Cipolla, R.: Learning to track with multiple observers. In: Proc. Computer Vision and Pattern Recognition, pp. 2647–2654 (2009)
7. Zhong, B., Yao, H., Chen, S., Ji, R., Yuan, X., Liu, S., Gao, W.: Visual tracking via weakly supervised learning from multiple imperfect oracles. In: Proc. Computer Vision and Pattern Recognition, pp. 1323–1330 (2010)
8. Du, W., Piater, J.H.: A Probabilistic Approach to Integrating Multiple Cues in Visual Tracking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 225–238. Springer, Heidelberg (2008)
9. Grabner, H., Bischof, H.: On-line boosting and vision. In: Proc. Computer Vision and Pattern Recognition, vol. (1), pp. 260–267 (2006)
10. Babenko, B., Yang, M., Belongie, S.: Visual Tracking with Online Multiple Instance Learning. In: Proc. Computer Vision and Pattern Recognition, pp. 983–990 (2009)
11. Tang, F., Brennan, S., Zhao, Q., Tao, H.: Co-Tracking Using Semi-Supervised Support Vector Machines. In: Proc. International Conference on Computer Vision, pp. 1–8 (2007)
12. Stalder, S., Grabner, H., Gool, L.: Beyond Semi-Supervised Tracking:Tracking Should Be as Simple as Detection, but not Simpler than Recognition. In: Proc. International Conference on Computer Vision 2009 Workshop on On-line Learning for Computer Vision (2009)
13. Perez, P., Vermaak, J., Blake, A.: Data fusion for visual tracking withparticles. Proceedings of the IEEE 92(3), 495–513 (2004)
14. Moreno-Noguer, F., Sanfeliu, A., Samaras, D.: Dependent multiple cue integration for robust tracking. IEEE Trans. Pattern Analysis and Machine Intelligence. 30(1), 670–685 (2008)

15. Leichter, I., Lindenbaum, M., Rivlin, E.: A generalized framework for combining visual trackers - the black boxes approach. International Journal of Computer Vision 67(2), 91–110 (2006)
16. Beugnon, G., Singh, T., Llinnas, J., Saha, R.K.: Adaptive Tracking Fusion in a Multisensor Environment. In: Proc. International Conference on Information Fusion, vol. (1), pp. 24–31 (2000)
17. Chee, C.Y., Mori, S., Barker, W.H., Chang, K.C.: Architectures and Algorithms for Track Association and Fusion. IEEE Aerospace and Electronic Systems Magazine 15(1), 5–13 (2000)
18. Comaniciu, D., Ramesh, V., Meer, P.: Real-time tracking of non-rigid objects using mean shift. In: Proc. Computer Vision and Pattern Recognition (2000)
19. Nummiaroa, K., Meierb, E., Gool, L.: An adaptive color-based particle filter. Image and Vision Computing 21(1), 99–110 (2003)
20. Stevens, S., Chen, D., Wactlar, H., Hauptmann, A., Christel, M., Bharucha, A.: Automatic Collection, Analysis, Access and Archiving of Psycho/Social Behavior by Individuals and Groups. In: Proc. of the 3rd ACM Workshop on Continuous Archival and Retrival of Personal Experences (2006)
21. http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/

Identifying Objects in Images from Analyzing the Users' Gaze Movements for Provided Tags

Tina Walber[1], Ansgar Scherp[1,2], and Steffen Staab[1]

[1] Institute for Web Science and Technology
[2] Institute for Information Systems Research
University of Koblenz-Landau, Germany
{walber,scherp,staab}@uni-koblenz.de
http://west.uni-koblenz.de/

Abstract. Assuming that eye tracking will be a common input device in the near future in notebooks and mobile devices like iPads, it is possible to implicitly gain information about images and image regions from these users' gaze movements. In this paper, we investigate the principle idea of finding specific objects shown in images by looking at the users' gaze path information only. We have analyzed 547 gaze paths from 20 subjects viewing different image-tag-pairs with the task to decide if the tag presented is actually found in the image or not. By analyzing the gaze paths, we are able to correctly identify 67% of the image regions and significantly outperform two baselines. In addition, we have investigated if different regions of the same image can be differentiated by the gaze information. Here, we are able to correctly identify two different regions in the same image with an accuracy of 38%.

Keywords: region identification, region labeling, gaze analysis, tagging.

1 Introduction

To describe the semantics of images on social media platforms such as Flickr[1] and social networking sites like Facebook[2] users can allocate tags to the images. Nevertheless, tagging describes the semantics of the images in a limited way. One step towards improving the understanding of image semantics is to annotate specific image regions instead of the entire image. Although tagging image regions is in principle possible on these platforms and sites, the annotation is manual and thus quite tedious. In this paper, we are investigating if it is in principle possible to automatically assign tags to objects by analyzing the users' gaze paths. In order to analyze the gaze paths in a controlled manner, we have designed an experiment in which 20 subjects have viewed a sequence of 51 tag-image-pairs each. For each tag shown to the subjects, they had to decide whether or not an object corresponding to that tag can be found in the image. During the experiment, the users' gaze paths with the fixations are recorded. Fixations

[1] http://www.flickr.com/

[2] http://www.facebook.com/

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 138–148, 2012.

are the phases of highest visual perception in the movements of the eyes, which are briefly focused on a particular point on the screen. A fixation measure is a function on the users' gaze path. It is calculated for each image region over all users viewing the same image-tag-pair. The tag is assigned to the region with the highest fixation measure value. We have applied 13 fixation measures to explore their performance on determining these tag-to-region assignments. The results show a maximum precision of 67% that significantly outperforms two baselines. In addition to finding specific objects in images, we have investigated if it is possible to differentiate different objects shown in the same image by looking at the gaze paths. The results show an accuracy of 38% of two correctly identified objects in the same image and show potential for future improvements.

2 Related Work

The simplest approach for annotating image regions is *manual labeling*. For example, the photo sharing platform Flickr allows its users to manually mark image regions by drawing rectangle boxes on it and writing a comment to it. Other web platforms like LabelMe [9] allow for the more precise creation of regions by drawing polygons on the images. These regions are annotated with a tag. "Games with a purpose" trigger the human play instinct in order to obtain manually created image regions [12].

With respect to the automatic *segmentation and labeling of images*, Rowe [8] presents an approach to find the visual focus of an image by applying image processing in terms of segmentation and low-level features. Goal is to link the visual focus with the image caption. This approach is designed for images with a single object only [8]. In addition, it has many limitations concerning the position and characteristics of the shown object.

Usability studies are a standard use case for applying gaze information. For detailed analysis, regions of interest (ROI) are marked on the investigated medium, e.g., a web page or an image showing a commercial. Based on these ROIs, the users' attention is analyzed in order to optimize the object that is under examination [1]. These ROIs are manually created, have usually simplified shapes like rectangles, and do not aim at correlating image regions with tags for the purpose of region annotation.

In *information retrieval*, several approaches use eye-tracking to identify images in a search result as attractive or important and use this information as implicit user feedback to improve the image search, e.g., [6,2,5]. Jaimes et al. [3] carried out a preliminary analysis of identifying common gaze trajectories in order to classify images into five, predefined semantic categories. They do not consider image regions and the categories are very general. Santella et al. [10] present a method for semi-automatic image cropping using gaze information in combination with image segmentation. Goal is to find the most important image region but not to conduct a general identification of image objects. Klami et al. [4] present an approach to identify image regions relevant in a specific task using gaze information. Based on several users' gaze paths, heat maps are created that identify the regions of interest. The work revealed that the region identified depends on the task given to the subject before viewing the image. However the given task was very general and thus the work does not aim at identifying single objects in the images from the generated heat map. Finally, the work of Ramanathan et al. [7] aims at localizing affective objects and actions in images by using gaze information. Thus, the image regions that are affecting the users are identified and correlated with given concepts from an affection model. The affective image regions are identified using segmentation and recursive clustering of the gaze fixations. General identification of image regions showing specific objects is not conducted.

The related work shows that it is in principle possible to relate image regions with gaze path information. In contrast to our work, current research does not tackle the identification of objects in images based on the users' gaze information.

3 Experiment Design

The setup of our experiment was designed such that the users' gaze paths are obtained in a controlled manner. In our experiment application, we show tags to the subjects instead of asking them to enter own tags. In addition, the experiment application is designed such that first a tag and subsequently an image is shown to the subjects. The subjects were asked to decide whether or not an object described by the tag is shown on the image. 20 subjects (4 female) have participated in our experiment. The age of the subjects is between 23 to 40 years (average: 29.6 years). Their professions are undergraduate students (6), PhD students (12), and office clerks (2).

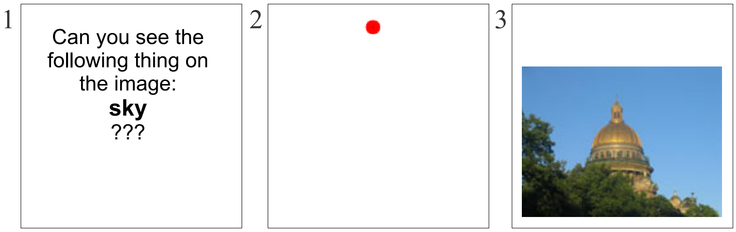
As data set, we use LabelMe[3] with 182.657 user contributed images (download August 2010). The LabelMe community has manually created image regions by drawing polygons into the images and tagging them. These manually created and annotated regions are used as ground truth in our experiment. The labels are used as tags and the regions as a manual, thus high quality image segmentation. For our experiment, we have randomly selected 51 images from the LabelMe data set. The images selected for our experiment have a minimum resolution of 1000x700 pixels and contain at least two labeled regions. We have created two sets of 51 tags and assigned one tag of each set to one image. Thus, each image has two tags. The two sets of tags are needed for the second part of our experiment aimed at discriminating two different objects shown on the same image. For every tag selected and assigned to the images, we have randomly decided if it should be a "true" or "false" tag. Here, "true" means that an object described by the tag can actually be seen on the image. The true tags are obtained from the labeled regions belonging to an image. The other tags were "false" and cannot be seen on the image. They were randomly selected from other LabelMe images. We had to manually replace images from the selected ones when a) the randomly selected false tags by coincidence correlate to some actually visible parts of the image and thus were true tags. We also replaced images where b) the tags where

[3] http://labelme.csail.mit.edu/

incomprehensible or expert knowledge is required and nonsense tags. In some cases there is c) a tag associated to a region like bicycle but multiple bicycles are depicted on the image and not all regions are explicitly marked as such. Thus, not all instances of the object the tag is referring to are actually labeled in the image. Finally, we have also removed images, where d) the object of interest is obstructed by other objects like a bicycle behind a car. Please note that the purpose of creating true and false image-tag-pairs is to keep the subjects concentrated during the experiment.

The experiment was performed on a screen with a resolution of 1680x1050 pixels. The subjects' gaze was recorded with a Tobii X60 eye-tracker at a data rate of 60Hz and an accuracy of 0.5 degree. The experiment was running as a simple web page in Microsoft's Internet Explorer. For each image-tag-pair, the following three steps are conducted as illustrated in Figure 1.

1. First, the tag with the question "Can you see the following thing on the image?" is presented to the subjects (see Figure 1, left). After pressing the "space" button, the application continuous with the next screen.
2. In this screen, a small blinking dot in the upper middle is displayed for one second (see Figure 1, middle). The subjects were asked to look at that point in order to let all subjects start viewing the images from the same position. The red dot let all subjects start viewing the image (which is shown next) from the same gaze position. The dot is placed above the actual image that is shown in the third screen.
3. Finally, the image is shown to the subjects (see Figure 1, right). Viewing the image, the subjects had to judge whether the thing shown in the first screen can be seen on the image or not. The decision is made by pressing the "y" (yes) or "n" (no) key.



Can you see the following thing on the image:
sky
???

Fig. 1. Steps Conducted for Identifying Image Objects

The first image-tag-pair is used to introduce the application to the subjects and is not used for the analysis. Each subject did evaluate one of the two sets consisting of 51 image-tag-pairs from the data set described above. The subjects were told that the goal of the experiment is not to measure their efficiency in conducting the experiment task. They could take as much time as they like to make the decision.

Besides recording the raw gaze data, we have also measured the time the subjects took to make a decision per image and the correctness of the answers.The average answer time over all images and users is about 3,003 ms. 5.7% of the given answers of all subjects were incorrect. The proportion of wrong answers is the same for given true and false tags. Subsequently to the experiment, the subjects were asked to provide subjective feedback in a questionnaire. The eye tracker and the experiment situation did not much influence the users' comfort. 85% of the subjects strongly agreed or agreed on the statement that they felt comfortable during the evaluation.

4 Analysis of Gaze Fixations on the Images

The preprocessing of the raw eye-tracking data was performed with the fixation filter offered by Tobii Studio with the default velocity threshold of 35 pixels and a distance threshold of 35 pixels. The extracted fixations are the base for our measure analysis. We have analyzed the gaze paths for images with a true tag and where the subjects gave a correct answer. In cases where the subjects gave incorrect answers, we do not know if the subjects did not took enough time to examine the image, did not understand the given tag, or if they had other problems. 547 gaze paths have been collected during the experiment that fulfill our requirement. 476 (87 %) of these gaze paths have at least one fixation inside or near a correct region. With this data, we are able to investigate the best fixation measure to identify the correct region in the image, i.e., finding the region of the image the tag shown in the experiment refers to. Please note that we do not use the images with the false tags, as the false image-tag-pairs have only been created in order to keep the subjects concentrated during the experiment (see Section 3). Investigating if it is possible to detect from the gaze path whether a subject had looked at a true image-tag-pair or false image-tag-pair is part of future work.

4.1 Calculating the Precision of Tag-to-Region-Assignments

The procedure for calculating the precision of the tag-to-region assignments is illustrated in Figure 2. The single steps performed for this calculation are:

1. For every LabelMe region in an image (b) a value for a fixation measure is calculated for every gaze path (c).
2. For every region, the measure results for every gaze path are summed up. From this, we obtain an ordered list of image regions for a fixation measure that determines the favorite region (d).
3. The label of the favorite region is compared with the tag (a) that was given to the subject in the experiment. If the label and tag match, the assignment is true positive (tp) otherwise it is a false positive (fp). We have summed up the total number of correct and incorrect assignments over all images and calculate the precision P for the whole image set using the following formula:

$$P = \frac{tp}{tp + fp} \tag{1}$$

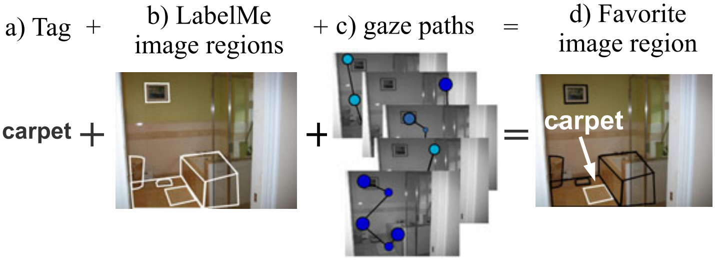


Fig. 2. Overview of Calculating the Tag-to-Region-Assignments

4.2 Considered Fixation Measures

We have selected 13 fixation measures and compared their performance to identify the correct favorite region. The measures including their units are presented below. The way the favorite region is calculated using the measure is summarized in brackets after the measure. It can be, e.g., the minimum of fixation counts on the different image regions (min count), the maximum distance between two fixations in centimeters (max centimeter), or the maximum fixation duration on the regions in milliseconds (max millisecond).

The standard measure (1) firstFixation (min count) computes the number of fixations on the image before fixating on a region r. The favorite is the region that was fixated first that means the region with no previous fixations on the image. The measure (2) secondFixation (min count) ignores the first fixation, because this fixation is influenced by the first visual orientation on an image [13]. We have also used a modification of the secondFixation measure called (3) fixationsAfter [4] (min count) to examine also the fixations on the image after the subjects made their decision, i.e., have pressed the "n" or "y" key. 96% of the gaze paths contain fixations after making the decision by pressing the button on the keyboard. This is due to the inherent reaction time of the experiment setup. The average duration of the recording after making the decision is 834 milliseconds. We have investigated the fixations around the moment of decision with the new measures (4) fixationsBeforeDecision (min count) and (5) fixationsAfterDecision (min count). The last measure includes also fixations at the moment of decision. The (6) fixationDuration (max millisecond) describes the sum of the duration of all fixations on a region r. The Tobii measure (7) firstFixationDuration (max millisecond) considers the order of the fixations and describes the duration of only the first fixation on a region r. Also the measure (8) lastFixationDuration (max millisecond) was investigated. It provides the duration of the last fixation on the region. The last fixations were taken into consideration in [11]. The standard measure (9) fixationCount (max count) counts the fixations on a region r. The three measures (10) maxVisitDuration (max millisecond), (11) meanVisitDuration (max millisecond) and (12) visitCount (max count) are based on visits. A visit describes the time between the first fixation on a region and the next fixation outside. The last measure (13) saccLength (max centimeter) [6] provided good results for the relevance feedback in image search. Thus, we have also considered it in our experiments. The assumption is that moving the gaze focus over a long distance (i.e., long saccade) to reach an image region r shows high interest in a region.

For our analysis, only fixations on the image are considered. Fixations on the experiment screen but outside the evaluated image are ignored.

4.3 Extending Object Boundaries and Weighting Small Objects

When comparing the fixation measures, we have investigated two further parameters: The first parameter is an extension of region boundaries to deal with the inaccuracy of eye-tracking data. Based on our prior investigations [13], we use an extension of 13 pixels. The second parameter deals with the fact that larger image regions have the advantage of being more likely fixated than smaller images. To support smaller regions, we investigate a linear weighting function with the highest weighting factor 4 [13]. The weighting depends on the image region size in relation to the total image size. All image regions smaller than 5% of the image size are weighted. The detailed analysis of the region extension and weighting parameters can be found in [13].

5 Results of Finding Objects in Images

Comparing the different fixation measures, we have received the best results for the measure (11) meanVisitDuration with precision $P = 0.54$ (cf. Figure 3). That means, 54% of the image regions selected by the gaze analysis belonged to the tag that was shown to the subjects. Two measures reach the second best value ($P = 0.53$): (4) fixationsBeforeDecision and (8) lastFixationDuration. With $P = 0.50$, the measure (6) fixationDuration provides the third best result. The lowest precision values are 0.21 and 0.26 for (1) firstFixation and (2) secondFixation.

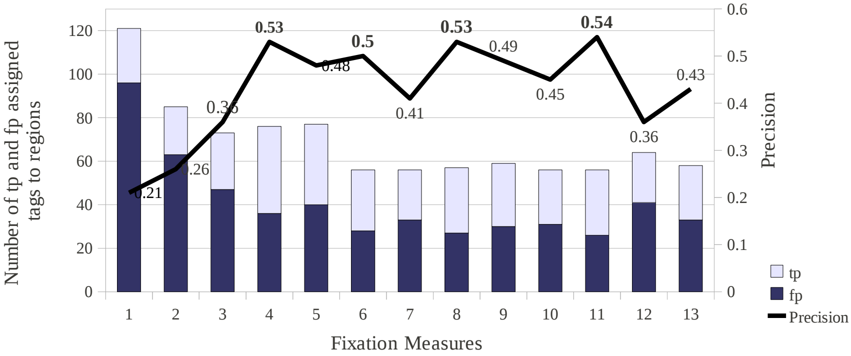
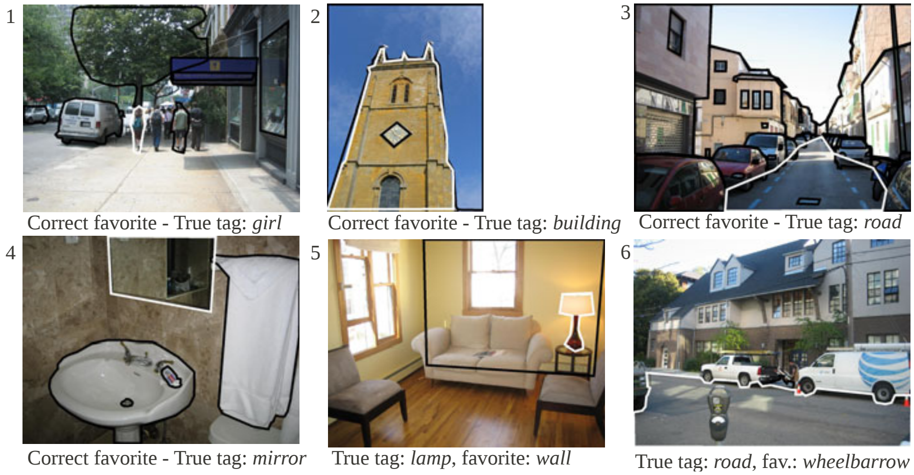


Fig. 3. Precision Values for the Fixation Measures from Section 4.2

Taking the image region extension and the weighting from Section 4.3 into account, we receive for meanVisitDuration the best precision value $P = 0.67$. The following analysis and computations are based on this measure and parameters. Figure 4 shows some positive and negative examples. As we have investigated, the size or the position of an object in the image does not have in principle an influence of the correctness of the assignments (see [13] for details). However, we have identified some characteristics of the images with incorrect assignments. First, in some scenes with a small given object the wrongly selected favorite object is also small and located next to the correct object. This problem can be based on the accuracy of the eye-tracker (5 of 19 wrong assignments belong to this category). Second, the object is sometimes located within another object (cf. Figure 4, image 5). In these cases, the outer region is identified as favorite (5 of 19 wrong assignments). Finally, further images show scenes with an object that seems to be very easy to identify. For example large objects like *road* (cf. Figure 4, image 6) or *sky* might be perceived even in the corner of the human eye or based on context knowledge (7 of 19 wrong assignments).



1 Correct favorite - True tag: *girl*

2 Correct favorite - True tag: *building*

3 Correct favorite - True tag: *road*

4 Correct favorite - True tag: *mirror*

5 True tag: *lamp*, favorite: *wall*

6 True tag: *road*, fav.: *wheelbarrow*

Fig. 4. Correctly (1. - 4.) and incorrectly (5., 6.) identified favorite objects

5.1 Compare with Baselines

We use two baselines that have been applied to evaluate relevance feedback from gaze information in [6] and [5]. We compare the precision P for image-tag-pair assignments calculated from the baseline "naive" (a) and the baseline "random" (b) with the mere measure meanVisitDuration (c) and the meanVisitDuration measure including region extension and weighting (d). The naive baseline makes the assumption that the largest area in an image should be the favorite one. The random baseline randomly chooses one of the labeled regions of the image as favorite. The results in Figure 5 show, the naive approach has a precision of 0.16 and the random baseline of 0.21 compared to the gaze-based approach with a precision of 0.54 and the extended and weighted of 0.67. The identification of assignments based on gaze information or on gaze information including extension and weighting performs better than both baseline approaches. Applying Chi-square tests shows that the gaze assignments are significantly better than the baselines (all with $\alpha < 0.001$).

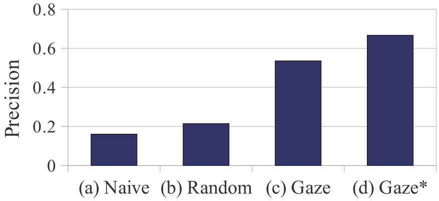


Fig. 5. Precision for two baselines and gaze based analysis

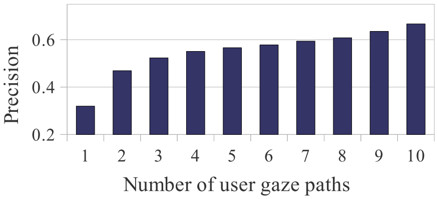


Fig. 6. Effect of aggregation of gaze paths from one up to ten users

5.2 Effect of Aggregation of Gaze Paths on Precision

We have investigated how strong the influence of the aggregation over multiple subjects on the precision. We present precision values for aggregations of 1 to 10 subjects for the measure meanVisitDuration, including extension and weighting. Precision P is calculated for every possible subset of subjects and averaged for all subgroups of the same size. As Figure 6 shows, the influence of the number of users is very high. With the gaze paths of only a single user, we have received an average precision (over all users and all images) of $P = 0.31$. For the aggregated data for all 10 users we got a precision $P = 0.67$. This corresponds to an improvement of 109%. The biggest improvements take place between the first group sizes. For example between one and two users per group we have an improvement of 46%. Between nine users and ten users per group, there is only an improvement of 4%.

The results based on multiple gaze paths are considerably better than the ones calculated from only a few gaze paths. However, the improvement of the precision gets lower when aggregating more gaze paths. Compared with the two baselines from Section 5.1, the results for single users are still significantly better than the naive or random baseline. The Chi-square test provides for the naive approach $\alpha < 0.001$ and for the random approach $\alpha < 0.002$.

6 Results of Discriminating Objects in Images

As an extension to the experiment described above, we have investigated if it is possible to differentiate objects by analyzing the users' gaze paths given that different tags of the same image are shown to the subjects. For this experiment, we have used the two tag sets assigned to the 51 images as described in Section 3. We use the measure meanVisitDuration, including extension and weighting, to calculate the results. For 16 images with two correct tags, the favorite image regions were calculated. In 6 images, two correct image regions were identified. This is a proportion of 38%. In Figure 7, some examples with two correctly identified regions are shown. As the figure shows, the two tags *sky* and *sea* could be distinguished in the upper image. Also the tags *water pot* and *teas* in the lower image could be identified using gaze information. The average probability to identify the correct region in one image is 67% (see Section 5). For two images, the probability of identifying correct assignments for both tags is 44%. With a value of 38% for two image regions in one image, the probability is close to the probability for two image regions in two different images. Thus, it is possible to identify different image regions in one image with an accuracy close to the accuracy of the single assignments.

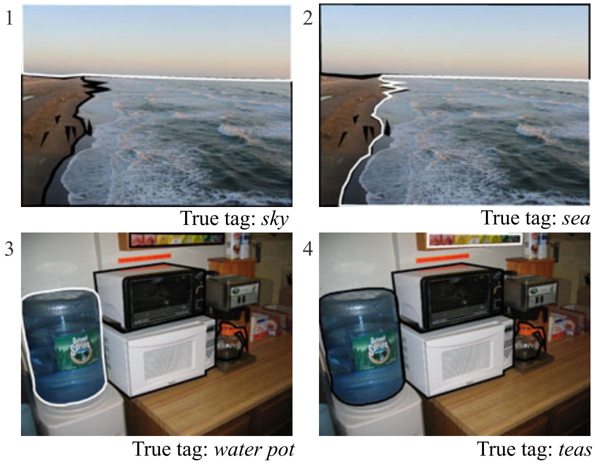


Fig. 7. Example images with two correctly identified regions (white borders)

7 Conclusions

In this paper, we have shown that it is possible to identify image regions by analyzing the gaze paths of users viewing the image with a given tag and given image regions at a precision of 67%. In addition, we have shown that two different regions can be differentiated in the same image with an accuracy of 38%. The results are gained in a controlled experiment with manually segmented images from the LabelMe data set. We have used LabelMe instead of applying automatic segmentation based on low-level features because of the additional error that would have been introduced in the experiment by automatic segmentation. The next step will be to apply the experiment on automatically segmented images. Such automatic segmentation can be improved by using the gaze information [10].

Acknowledgement. The research leading to this paper was partially supported by the EU projects Petamedia (FP7-216444) and SocialSensor (FP7-287975).

References

1. Castagnos, S., Jones, N., Pu, P.: Eye-tracking product recommenders' usage. In: Proceedings of the Fourth ACM Conference on Recommender Systems, pp. 29–36. ACM (2010)
2. Hajimirza, S.N., Izquierdo, E.: Gaze movement inference for implicit image annotation. In: Image Analysis for Multimedia Interactive Services. IEEE (2010)
3. Jaimes, A.: Using human observer eye movements in automatic image classifiers. In: SPIE (2001)
4. Klami, A.: Inferring task-relevant image regions from gaze data. In: Workshop on Machine Learning for Signal Processing. IEEE (2010)
5. Klami, A., Saunders, C., De Campos, T.E., Kaski, S.: Can relevance of images be inferred from eye movements? In: Multimedia Information Retrieval, ACM (2008)
6. Kozma, L., Klami, A., Kaski, S.: GaZIR: gaze-based zooming interface for image retrieval. In: Multimodal Interfaces. ACM (2009)
7. Ramanathan, S., Katti, H., Huang, R., Chua, T.-S., Kankanhalli, M.: Automated localization of affective objects and actions in images via caption text-cum-eye gaze analysis. In: Multimedia (2009)
8. Rowe, N.C.: Finding and labeling the subject of a captioned depictive natural photograph. IEEE Transactions on Knowledge and Data Engineering, 202–207 (2002)
9. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: LabelMe: a database and web-based tool for image annotation. Journal of Computer Vision 77(1), 157–173 (2008)
10. Santella, A., Agrawala, M., DeCarlo, D., Salesin, D., Cohen, M.: Gaze-based interaction for semi-automatic photo cropping. In: CHI, p. 780. ACM (2006)
11. Shimojo, S., Simion, C., Shimojo, E., Scheier, C.: Gaze bias both reflects and influences preference. Nature Neuroscience 6(12), 1317–1322 (2003)
12. von Ahn, L., Liu, R., Blum, M.: Peekaboom: a game for locating objects in images. In: CHI. ACM (2006)
13. Walber, T., Scherp, A., Staab, S.: Towards improving the understanding of image semantics by gaze-based tag-to-region assignments. Technical Report 08/2011, Institut WeST, Universität Koblenz-Landau (2011), http://www.uni-koblenz.de/~fb4reports/2011/2011_08_Arbeitsberichte.pdf

On Video Recommendation over Social Network

Xiaojian Zhao[1], Jin Yuan[2], Richang Hong[3],
Meng Wang[2,3], Zhoujun Li[1], and Tat-Seng Chua[2]

[1] State Key Laboratory of Software Development Environment,
Beihang University, Beijing 100191, China
[2] School of Computing, National University of Singapore, 117590, Singapore
[3] School of Computer and Information,
Hefei University of Technology, Hefei Anhui 230009, China
{zhaoxj01,hongrc.hfut,eric.mengwang}@gmail.com
{yuanjin,chuats}@comp.nus.edu.sg, lizj@buaa.edu.cn

Abstract. Video recommendation is a hot research topic to help people access interesting videos. The existing video recommendation approaches include CBF, CF and HF. However, these approaches treat the relationships between all users as equal and neglect an important fact that the acquaintances or friends may be a more reliable source than strangers to recommend interesting videos. Thus, in this paper we propose a novel approach to improve the accuracy of video recommendation. For a given user, our approach calculates a recommendation score for each video candidate that composes of two parts: the interest degree of this video by the user's friends, and the relationship strengths between the user and his friends. The final recommended videos are ranked according to the accumulated recommendation scores from different recommenders. We conducted experiments with 45 participants and the results demonstrated the feasibility and effectiveness of our approach.

Keywords: Video Recommendation, Relationship Strength, Activity Domain.

1 Introduction

Driven by the proliferation of digital capture and the advent of near-ubiquitous broadband Internet access, videos on the internet are growing at an explosive rate [3]. For example, it is estimated that the most popular video sharing website YouTube[1] stores over 150,000,000 videos in its repository. Therefore, today's online users always face a daunting volume of video clips when they search for interesting videos from the repositories. As a result, there is an increasing demand of video recommendation service which is able to help users to find "*interesting*" or "*highly related*" videos [10].

Currently, there are three prevalent approaches widely used in video recommendation, namely, *content-based filtering* (CBF), *collaborative filtering* (CF), and *hybrid filtering* (HF). The CBF approaches recommend videos based on the similarity between the unseen videos and the past videos viewed by the user [6], while the CF approaches

[1] http://www.youtube.com

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 149–160, 2012.

predict video preference of the given user based on the ratings of the other users whose tastes are similar to him/her. Combining both of the above two approaches, HF approaches could achieve a better performance [1].

However, the above approaches ignore an important fact that a user's friends can be a more reliable source to recommend interesting videos rather than strangers. For example, a user is more likely to be interested in the videos recommended by his friends than that from strangers since he and his friends may have more common interests and know each other well. Moreover, for different friends, the user may share interests on different domains. This will affect video recommendation results. For example, a user usually discusses with friend A about the "*sport*" topic, then the sport videos recommended by friend A may be of more interest to the user. While he may share "*diet*" topic with friend B, and thus the diet videos from friend B may be good candidates to be recommended. Not only the relationship strength between users, the video interest is another important factor to be considered in video recommendation task. For example, a user has a strong relationship with a friend on the domain "*sport*", but his friend may be more interested in "*football*" than "*basketball*", thus the videos about "*football*" viewed by his friend are more important than those of "*basketball*". Therefore, the interest degree of video is another important factor in video recommendation task.

Based on these motivations, in this paper, we propose a novel video recommendation framework by considering both user relationship strength and the interest degree of video. For a given user, we calculate a recommendation score for each video candidate. The recommendation score is composed of two parts: the interest degree of this video by the user's friends, and the relationship strengths between the user and his friends. We measure the interest degree of each video based on its textual and visual similarity with the other viewed videos. The relationship strengths between users are inferred through online social network. For each pair of users, we employ a graph model to estimate the relationship strength by considering the users' profile information, the interaction activities as well as the activity domains. The final recommended videos are ranked according to the recommendation scores.

We summarize the main contributions of this paper as follows:

1. To the best of our knowledge, this is the first work that integrates the relationship strength information derived from online social network into a personalized video recommendation framework. We not only utilize relationship strength between users, but also consider them in different activity domains.
2. We also propose an approach to identify the interest degree between a candidate video and the recommender. The interest degree is calculated using the textual and visual similarity between the video and the other videos viewed in the past.

The rest of this paper is organized as follows. In section 2, we review the related work. Section 3 details of the proposed methodology of interest degree estimation and the relationship strength estimation. The initial experiment results and evaluations are provided in Section 4. Finally, we conclude the paper and discuss the directions of future works in Section 5.

2 Related Work

There are three prevalent approaches widely used in video recommender system, namely *content-based filtering* (CBF), *collaborative filtering* (CF), and *hybrid filtering* (HF) that combines the above two approaches. For CBF approaches, videos can be recommended based on the contents of previously viewed videos. For example, Mei et al. [11] presented an online video recommendation system, VideoReach, using multi-modal relevance between videos and users' click-through data. They considered three modalities, textual, visual and aural, and combined the relevance scores from them by using the attention fusion function. The CF approaches compare a user's ratings of videos with those of hundreds of others, find people who share similar preferences, and then recommend videos that are interesting for those people with similar preferences [12]. For example, Baluja et al. [2] built a user-video graph which represents the co-view information among different users and its recommendation was performed by a graph propagation in which the label of each node was obtained from its neighbors. However, the CBF approach neglects the fact that different users may share similar interestingness and the performance of CF approach is decreased when there is a shortage of user's ratings such as user can only view a very small portion of the videos from a large-scale online video database. Thus, HF approaches, which combine both of the above two approaches in a single framework, are proposed. For example, Burke [4] employed mixture models which build the recommendation based on a linear combination of voting, the content-based prediction and the collaborative prediction.

The growth and popularity of online social networks, such as Facebook and Google+[2], have led to a surge in research focusing on estimating the relationship strength between different users in online social network. Gilbert et al. [7] presented a predictive model that maps the social media data to the strength of ties between friends. However, these works only consider the binary prediction task of distinguishing the strong ties from the weak ties. Xiang et al. [16] developed an unsupervised model to estimate the relationship strength between different users from the interaction activity (e.g., communication, tagging) and the user similarity. However, it mixes all the interaction activities from various activity domains together and did not consider the fact that the relationship strengths between the same user pair may be different in various activity domains.

In this paper, we utilize the relationship strength between different users in different domains to help recommend videos to the user. Meanwhile, we also consider the interest degree of video candidates from each recommender's viewing history.

3 Approach

Give a set of users $\mathcal{U} = \{u_1, u_2, \ldots, u_K\}$ associated with a dataset of viewed videos $\mathcal{V} = \{v_1, v_2, \ldots, v_H\}$, the video recommendation service aims to provide a video list for each user u_k. Here, we calculate a recommendation score for each video with respect to the user u_k ($1 \leq k \leq K$), and then return a rank list of videos to this user according to the recommendation scores. As shown in Fig. 1, for a given video v_h ($1 \leq h \leq H$), its

[2] http://plus.google.com/

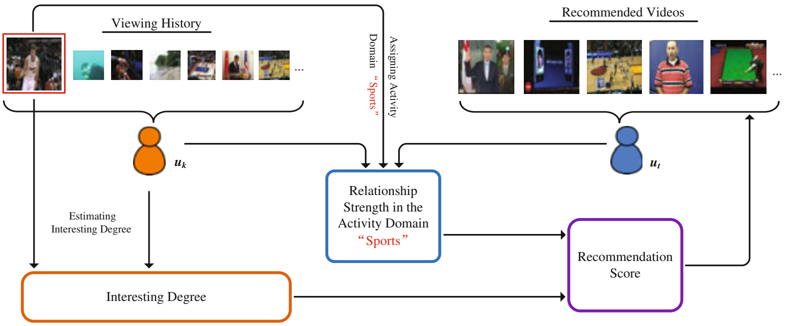

Viewing History
...
Assigning Activity Domain "Sports"
Recommended Videos
...
u_k
u_t
Estimating Interesting Degree
Relationship Strength in the Activity Domain "Sports"
Recommendation Score
Interesting Degree

Fig. 1. The schematic illustration of the proposed video recommendation strategy that explores the user's viewing history and the relationship strengths in various activity domains

recommendation score to the user u_k is determined by two factors: the interest degree of the video v_h by the user u_t $(t \neq k)$ who has viewed it before, and the relationship strength between the user u_k and u_t in the specific activity domain in which video v_h belongs to. Next, we will describe these two factors in detail.

3.1 Interest Degree Estimation

Give a set of videos $\mathcal{V} = \{v_1, v_2, \ldots, v_H\}$, we estimate the interest degree $I(v_h, u_k)$ of the video v_h by the user u_k as follow:

$$I(v_h, u_k) = \mu(v_h, u_k) U(v_h, u_k) \tag{1}$$

where $\mu(v_h, u_k)$ is an indicator whether u_k has viewed v_h before; $U(v_h, u_k)$ reflects the importance of the video v_h among all the viewed videos by u_k.

Generally, for a given user u_k, the importance of the video v_h $U(v_h, u_k)$ could be estimated according to the textual and visual information of all the viewed videos by u_k. Take the textual information as an example, if some words of the video v_h, such as "*football*" etc, frequently appear in the other viewed videos by u_k, then this video v_h is very important for user u_k since user u_k likes this topic "*football*". Based on this idea, in our approach, we adopt a linear function to calculate the importance value $I(v_h, u_k)$ based on textual and visual sources as:

$$U(v_h, u_k) = \alpha U_t(v_h, u_k) + (1 - \alpha) U_v(v_h, u_k) \tag{2}$$

where $U_t(v_h, u_k)$, $U_v(v_h, u_k)$ denote the importance of v_h measured by the textual and visual information respectively, and α is the balance weight which we empirically set in experiments.

Textual Importance Estimation. For a user u_k, the textual importance of video v_h is calculated by averaging the textual similarities between v_h and the other viewed videos by u_k, which we express it as:

$$U_t(v_h, u_k) = \frac{1}{num(v_o|u_k)} \sum_{o=1}^{L} \mu(v_o, u_k) S_t(v_h, v_o) \tag{3}$$

where $num(v_o|u_k)$ is the number of videos viewed by the user u_k, and $S_t(v_h, v_o)$ is the similarity between the video v_h and v_o measured based on textual information [14]. Here, the textual information includes video's title, description, tag and category etc. We represent each video v_o as a set of words $\mathcal{W}_o$, and the the similarity between video v_h and v_o is calculated as:

$$S_t(v_h, v_o) = \frac{1}{|\mathcal{W}_o||\mathcal{W}_l|} \sum_{w_o \in \mathcal{W}_o, w_l \in \mathcal{W}_l} exp(-\frac{NGD(w_o, w_l)}{\sigma}) \tag{4}$$

where $NGD(w_o, w_l)$ is the normalized Google distance [5] between the word w_o and w_l, and σ is a scaling parameter.

Visual Importance Estimation. The visual importance $U_v(v_h, u_k)$ is calculated by averaging the visual similarities between v_h and the other viewed videos by u_k, which we express as:

$$U_v(v_h, u_k) = \frac{1}{num(v_o|u_k)} \sum_{o=1}^{L} \mu(v_o, u_k) S_v(v_h, v_o) \tag{5}$$

where $num(v_o|u_k)$ is the number of videos viewed by user u_k, and $S_v(v_h, v_o)$ is the similarity between video v_h and v_o measured based on visual information. We represent each video as a set of key-frames. For each key-frame, we extract 428-dimensional global visual features, including 255-dimensional block-wise color moment, 128-dimensional wavelet texture, and 75-dimensional edge direction histogram [8][17][18]. The visual similarity $S_v(v_h, v_o)$ is calculated by averaging the similarities between their key-frames:

$$S_v(v_h, v_o) = \frac{1}{|v_h||v_o|} \sum_{\mathbf{x}_i \in v_h, \mathbf{x}_j \in v_o} (1 - \cos(\mathbf{x}_i, \mathbf{x}_j)) \tag{6}$$

where $\mathbf{x}_i$, $\mathbf{x}_j$ are key-frames in v_h and v_o respectively, $|v_h|$, $|v_o|$ represent the key-frame numbers contained in the corresponding videos, and $\cos(\mathbf{x}_i, \mathbf{x}_j)$ is the cosine distance between these two key-frames [13].

3.2 Relationship Strength Estimation

As Fig. 2 shows, the relationship strength estimation is composed of three steps: data preprocessing, activity domain assignment and graph-based relationship strength estimation. We will introduce these three steps in the rest of this subsection.

Data Preprocessing. Given the set of the interaction activities (messages, news feeds and events) downloaded from the Facebook website, there are three main sequential steps in our data preprocessing: word correction by Aspell[3], stop word removed, and stemming by WordNet[4]. After that, we obtain the dataset composed of the interaction

[3] http://aspell.net

[4] http://wordnet.princeton.edu/

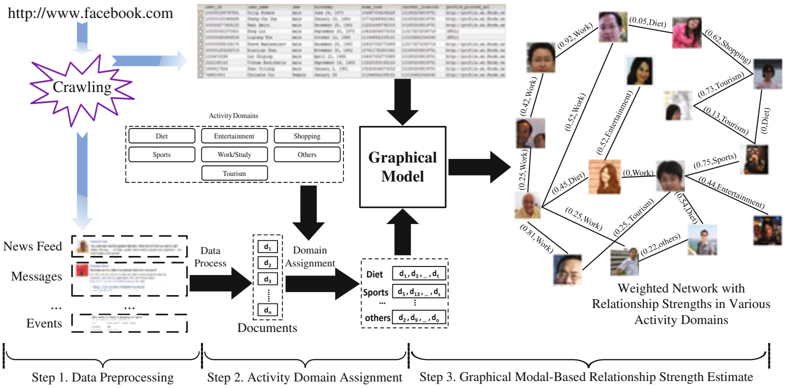


Fig. 2. A general framework to measure the relationship strength between different users in various activity domains in social network, where the (number,domain) pairs on the edges of the right network represent the value of relationship strength in that activity domain

activity documents $\mathcal{D} = \{d_1, d_2, \dots, d_N\}$, where N is the number of documents. For each interaction activity document d_n, its related users refer to those users sending or receiving this document. We record this user-document relationship by a matrix $\mathbf{UD} = \{ud_{kn}\}_{k,n=1}^{k=K,n=N}$, where ud_{kn} indicates whether document d_n is related to user u_k.

Activity Domain Assignment. Given the activity domains $\mathcal{A} = \{A_1, A_2, \dots, A_L\}$, we assign an activity domain A_l to each document d_n in $\mathcal{D}$ before estimating the relationship strength. Here, we define seven activity domains as "*diet*", "*entertainment*", "*shopping*", "*sports*", "*work*", "*tourism*", and "*others*". We represent each document d_n as a set of words. Let $Sem(d_n, A_l)$ be the relatedness degree between document d_n and activity domain A_l, which is calculated as:

$$Sem(d_n, A_l) = \sum_{w_n \in \mathcal{W}_n} tf_n * NGD(w_n, A_l) \tag{7}$$

where tf_n is the normalized frequency of the word w_n in $\mathcal{W}_n$, and $NGD(w_n, A_l)$ is the normalized Google distance [5] between w_n and the domain name A_l. For each document d_n, the domain A_l with the highest relatedness degree is assigned only if $Sem(d_n, A_l)$ is larger than a threshold, otherwise, this document belongs to "others".

Graphical Model Based Relationship Strength Estimation. To estimate the relationship strength between different users in various activity domains, we build a graphical model (described in Fig. 3) based on two observations:

1. For two users u_i and u_j, given a specific activity domain A_l, the relationship strength $T_l^{(i,j)}$ in this domain is determined by $S^{(i,j)}$, the profile similarity between these two users.
2. The relationship strength $T_l^{(i,j)}$ between u_i and u_j in activity domain A_l impacts their interaction activities on this domain (denoted as $D_l^{(i,j)}$).

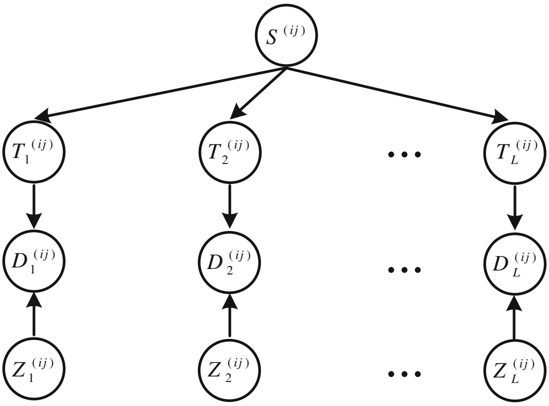


Fig. 3. Graphical model for estimating the relationship strength in various activity domains

Furthermore, to increase the accuracy of the graphical model, we introduce an auxiliary variable $Z_l^{(i,j)}$ for each $D_l^{(i,j)}$. The detailed descriptions of these variables in Fig. 3 are as follow:

- $S^{(i,j)} = (s_1^{(ij)}, s_2^{(ij)}, \ldots, s_P^{(ij)})$ is the similarity vector between the users u_i and u_j, where P is the number of attributes in the profile. For the p-th attribute f_p with discrete values, we set $s_p^{(ij)} = 1$ if u_i and u_j have the same values on f_p, and $s_p^{(ij)} = 0$ otherwise. On the other hand, if the values on f_p are continuous, $s_p^{(ij)}$ is determined according to:

$$s_p^{(ij)} = 1 - \frac{|f_p^i - f_p^j|}{\max\limits_{1 \le k_1, k_2 \le K} |f_p^{k_1} - f_p^{k_2}|} \tag{8}$$

  where f_p^i represents the value of user u_i on the p-th attribute.
- $T_l^{(ij)}$ is the relationship strength between users u_i and u_j in activity domain A_l.
- $D_l^{(ij)}$ is the strength of interaction activities between users u_i and u_j in activity domain A_l. We measure it based on their related documents in this domain, which is calculated as:

$$D_l^{(ij)} = \sum_{n=1}^{N} Sem(d_n, A_l) * ud_{in} * ud_{jn} \tag{9}$$

- $Z_l^{(ij)}$ is an auxiliary variable. We set $Z_l^{(ij)}$ to 1 in our experiment.

As illustrated in Fig. 3, our graphical model represents the likely causal relationships among all the variables by modeling their conditional dependencies. Based on these dependencies, the joint distribution decomposes as follows:

$$\begin{aligned} &P(T_1^{(ij)}, \ldots, T_L^{(ij)}, D_1^{(ij)}, \ldots, D_L^{(ij)} | \mathbf{S}^{(ij)}, Z_1^{(ij)}, \ldots, Z_L^{(ij)}) \\ &= \prod_{l=1}^{L} P(T_l^{(ij)} | \mathbf{S}^{(ij)}) P(D_l^{(ij)} | T_l^{(ij)}, Z_l^{(ij)}) \end{aligned} \tag{10}$$

In this work, we adopt the widely-used Gaussian distribution to model the conditional probabilities $P(T_l^{(ij)}|\mathbf{S}^{(ij)})$ and $P(D_l^{(ij)}|T_l^{(ij)}, Z_l^{(ij)})$, which are expressed as:

$$\begin{aligned} P(T_l^{(ij)}|\mathbf{S}^{(ij)}) &= \mathcal{N}(\mathbf{w}_l^T\mathbf{S}^{(ij)}, v) \\ P(D_l^{(ij)}|T_l^{(ij)}, Z_l^{(ij)}) &= \mathcal{N}(\alpha_l T_l^{(ij)} + \beta_l Z_l^{(ij)}, v) \end{aligned} \tag{11}$$

where $\mathbf{w}_l$ is a P-dimensional weight vector to be estimated, α_l, β_l are two coefficients, and v is the variance in Gaussian model, which is configured to be 0.5 in our experiments. To avoid over-fitting, we put L_2 regularizes on parameters $\mathbf{w}_l$ and α_l, β_l, which can be regarded as Gaussian priors:

$$\begin{aligned} P(\mathbf{w}_l) &\propto e^{-\frac{\lambda_1}{2}\mathbf{w}_l^T\mathbf{w}_l} \\ P(\alpha_l, \beta_l) &\propto e^{-\frac{\lambda_2}{2}(\alpha_l^2+\beta_l^2)} \end{aligned} \tag{12}$$

Among all the variables, $\mathbf{S}^{(ij)}$, $D_l^{(ij)}$, $Z_l^{(ij)}$ are all visible, and $\mathbf{w}_l$, α_l, β_l are to-be-learned parameters. Given the samples of the user pairs $\mathcal{P} = \mathcal{U} \times \mathcal{U}$, the joint probability is expressed according to Eq. (10)-Eq. (12):

$$\begin{aligned} &\prod_{l=1}^{L} P(\mathcal{P}|\mathbf{w}_l, \alpha_l, \beta_l)P(\mathbf{w}_l)P(\alpha_l, \beta_l) \\ &= \prod_{l=1}^{L} \prod_{(i,j)\in\mathcal{P}} P(D_l^{(ij)}|Z_l^{(ij)}, T_l^{(ij)}, \alpha_l, \beta_l)P(T_l^{(ij)}|\mathbf{S}^{(ij)}, \mathbf{w}_l)P(\mathbf{w}_l)P(\alpha_l, \beta_l) \\ &\propto \prod_{l=1}^{L} \prod_{(i,j)\in\mathcal{P}} e^{-\frac{1}{2v}(\mathbf{w}_l^T\mathbf{S}^{(ij)}-T_l^{(ij)})^2} e^{-\frac{1}{2v}(\alpha_l T_l^{(ij)}+\beta_l Z_l^{(ij)}-D_l^{(ij)})^2} e^{-\frac{\lambda_1}{2}\mathbf{w}_l^T\mathbf{w}_l} e^{-\frac{\lambda_2}{2}(\alpha_l^2+\beta_l^2)} \end{aligned} \tag{13}$$

Since the joint probabilities of L activity domains in Eq. (13) are independent, we can divide Eq. (13) into L independent joint probabilities, and infer the solution for each activity domain separately. In our implementation, we use a gradient-based method to optimize it over the parameters $\mathbf{w}_l^T$, α_l, β_l, and variable $T_l^{(ij)}$. Due to the limited space, the detailed implementation is not presented here.

3.3 Video Recommendation

Given a user u_k, in this step, we calculate the recommendation score $R(v_h, u_k)$ for each video v_h. Aforementioned, the recommendation score $R(u_k, v_h)$ is determined by two factors: the interest degree of video v_h by user u_t ($t \neq k$) (denoted as $I(v_h, u_t)$, see Section 3.1), and the relationship strength between u_t and u_k in the special domain A_l that v_h belongs to (denoted as $T_l^{(tk)}$, see Section 3.2). We multiple these factors as:

$$R(v_h, u_k) = \sum_{t=1, t\neq k}^{K} I(v_h, u_t)T_l^{(tk)}\mu(v_h, A_l) \tag{14}$$

where $\mu(v_h, A_l)$ is an indicator to represent whether v_h belongs to domain A_l, $\mu(v_h, A_l) = 1$ indicates that video v_h belongs to domain A_l, and $\mu(v_h, A_l) = 0$ otherwise. In our approach, we represent video v_h as a word set, where the words inside are collected from the textual description associated with v_h. Based on the approach in Section 3.2, we can assign a domain to video v_h. According to the recommendation scores in Eq. (14), we return a rank list of videos to each user.

4 Experiments

4.1 Experimental Settings

The dataset is downloaded from the Facebook website, which is a popular online social network site with over 600 million members worldwide. To download data from Facebook, we first selected 9 active users from three countries (Singapore, China and America) as the seed nodes. After obtaining their consents, we collected all the friends of these 9 users, which results in a total of 632 people. Since it is hard to collect the viewing history of all people, we only sampled 45 persons who have at least three common acquaintances. We downloaded each user's profile information, including location, language, religion, interests and etc. Besides, we downloaded the interaction activities (messages, news feed, etc.) for each user between Sep. 2010 and Oct. 2010. This results in a total of 22,500 interaction activity documents. Meanwhile, the video viewing behaviors of these users on YouTube were tracked in a one-month period (from Dec. 2010 to Jan. 2011). Video links from each user's log were extracted. The video itself and the corresponding title, tag, category, description information were downloaded and stored in our database. It is shown that there are about 150 videos viewed per user on average.

For each user, we split the viewed videos into two parts, the first part is the videos viewed in the previous two weeks and the second part is the videos viewed in the next two weeks. The second part is used for testing. In other words, we regard videos in the second part as the relevant samples for recommendation. We assign the relevant score of 1 to the videos in the second part, and 0 to the other videos for a user. It is worth noting that this setting actually underestimates the performance, as the user may also be interested in the videos out of the second part. Though a more rigorous approach for ground truth establishment is to let users label all videos with interestingness, our approach is still reasonable for comparing different algorithms. For performance evaluation metric, we adopted the normalized discounted cumulative gain (NDCG) [9] [15].

4.2 Experimental Results

To comprehensively evaluate our approach, we consider two types of classical recommendation methods as baselines:

1. Content-based filtering method (CBF): the videos are recommended based on the similarity between the unseen video and the videos previously viewed by the user. The similarity between these two videos is calculated using Eq. (4) and Eq. (6).
2. Collaborative filtering method (CF): it predicts the preference to a video based on the ratings of similar users. The distance of vectors which represent the user's viewing history is adopted to measure the similarity between different users.

Fig. 4 illustrate the comparison of average NDCG@20 in various activity domains. We can see that our approach outperforms the other methods in almost every domain except for "*work*" domain. One possible reason is that the representative words in "*work*" domain are diverse. On the other hand, "*sports*" domain contains very highly repetitive words such as "*basketball*", "*swimming*" and "*jogging*", and thus the relationship

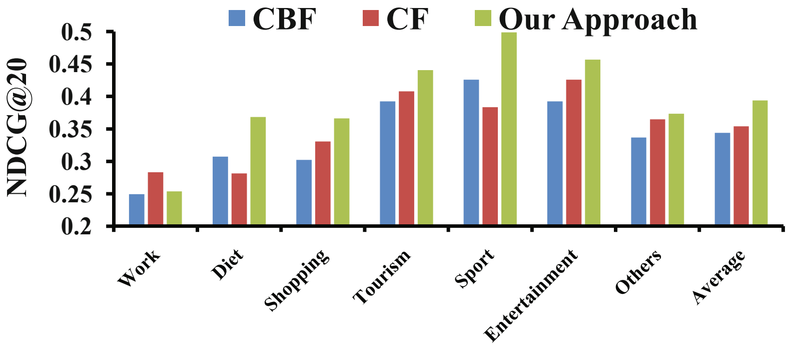


Fig. 4. The performance of the three video recommendation methods in different active domains

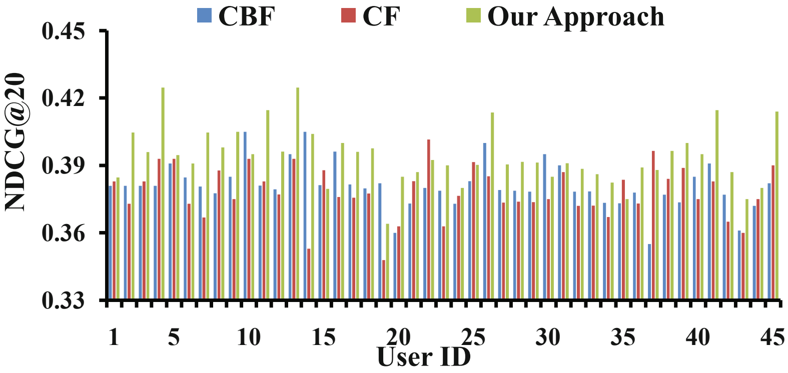
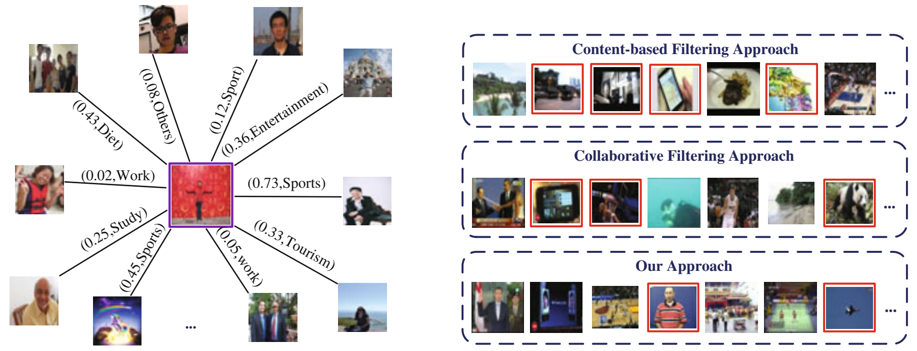


Fig. 5. The performance of the three recommendation methods for each user

strength in the "*sports*" domain can be estimated more accurately. In addition, people rarely use the social network to discuss work related topics.

Fig. 5 illustrates detailed NDCG@20 results for the 45 users. We can see that for most users our approach achieves better results than the other two methods. The reason is that the two main components in the ranking score function Eq. 14 integrates the positive aspects of both CBF and CF methods. One component is the relationship strength in the activity domain in which the video to be recommended belongs to. It is estimated based on user's profile information and the interaction activities between different users. The other component is the interest degree to the video given by the recommender. It is estimated based on the recommender's viewing history. So its performance is obviously better than the other two methods. Fig. 6 illustrates the top relationship strength in different activity domains for a user in a social network and the different ranking lists produced by three video recommendation strategies for this user. We can see that the user's main interest is "*sports*", which is estimated by our proposed relationship strength measurement. From Fig. 6(b), we can see that our proposed approach recommends more relevant videos given the user's interest.



(a) The relationship strengths in different activity domains for a user (the center one).

(b) The ranking lists of the three video recommendation methods for the user in subfigure (a).

Fig. 6. The relationship strength network for a user and the recommended videos to him by the three video recommendation strategies

5 Conclusion and Future Works

In this paper, we proposed a novel approach to improve the accuracy of video recommendation by utilizing the relationship strength information from social network. First, the interest degree of each viewed video by a user's friends was calculated. Second, the relationship strengths between different users were measured, taking into consideration not only the user's profile information, interaction activities, but also the activity domains. Third, the recommended videos viewed by the friends of a user were ranked based on their interest degree of each video and the user's relationship strengths with the friends in different domains. We conducted experiments with 45 participants and the results demonstrated the feasibility and effectiveness of our approach. In our future work, we will conduct experiments with more users and will also consider integrating more contextual factors in video recommendation, such as the time and location.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (61170189, 60973105), the Fund of the State Key Laboratory of Software Development Environment under Grant No. SKLSDE-2011ZX-03 and the Singapore National Research Foundation & Interactive Digital Media R&D Program Office, MDA under research grant (WBS:R-252-300-001-490).

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 734–749 (2005)
2. Baluja, S., Seth, R., Sivakumar, D., Jing, Y., Yagnik, J., Kumar, S., Ravichandran, D., Aly, M.: Video suggestion and discovery for youtube: taking random walks through the view graph. In: ACM WWW, pp. 895–904 (2008)

3. Boll, S.: Multitube–where web 2.0 and multimedia could meet. IEEE Multimedia 14(1), 9–13 (2007)
4. Burke, R.: Hybrid Web Recommender Systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 377–408. Springer, Heidelberg (2007)
5. Cilibrasi, R., Vitanyi, P.: The google similarity distance. IEEE Transactions on Knowledge and Data Engineering, 370–383 (2007)
6. Gibas, M., Canahuate, G., Ferhatosmanoglu, H.: Online index recommendations for high-dimensional databases using query workloads. IEEE Transactions on Knowledge and Data Engineering, 246–260 (2008)
7. Gilbert, E., Karahalios, K.: Predicting tie strength with social media. In: ACM CHI, pp. 211–220 (2009)
8. Hong, R., Wang, M., Xu, M., Yan, S., Chua, T.: Dynamic captioning: video accessibility enhancement for hearing impairment. In: ACM MM, pp. 421–430 (2010)
9. Hu, X., Tang, L., Liu, H.: Enhancing accessibility of microblogging messages using semantic knowledge. In: ACM CIKM (2011)
10. Mei, T., Aizawa, K.: Video recommendation. In: Chapter of Internet Multimedia Search and Mining. Bentham Science Publisher (2011)
11. Mei, T., Yang, B., Hua, X., Yang, L., Yang, S., Li, S.: Videoreach: an online video recommendation system. In: ACM SIGIR, pp. 767–768 (2007)
12. Park, J., Lee, S., Kim, K., Chung, B., Lee, Y.: An online video recommendation framework using view based tag cloud aggregation. IEEE Multimedia (99), 1 (2010)
13. Wang, M., Hua, X., Tang, J., Hong, R.: Beyond distance measurement: constructing neighborhood similarity for video annotation. IEEE Transactions on Multimedia 11(3), 465–476 (2009)
14. Wang, M., Hua, X., Tang, J., Qi, G., Song, Y.: Unified video annotation via multi-graph learning. IEEE Transactions on Circuits and Systems for Video Technology 19(5) (2009)
15. Wang, M., Yang, K., Hua, X.-S., Zhang, H.-J.: Towards a relevant and diverse search of social images. IEEE Transactions on Multimedia, 12 (2010)
16. Xiang, R., Neville, J., Rogati, M.: Modeling relationship strength in online social networks. In: ACM WWW, pp. 981–990 (2010)
17. Yang, Y., Xu, D., Nie, F., Yan, S., Zhuang, Y.: Image clustering using local discriminant models and global integration. IEEE Transactions on Image Processing (2010)
18. Yang, Y., Zhuang, Y., Wu, F., Pan, Y.: Harmonizing hierarchical manifolds for multimedia document semantics understanding and cross-media retrieval. IEEE Transactions on Multimedia, 10 (2008)

Improving Item Recommendation Based on Social Tag Ranking

Taiga Yoshida[1], Go Irie[1], Takashi Satou[1],
Akira Kojima[2], and Suguru Higashino[1]

[1] NTT Cyber Solutions Laboratories, NTT Corporation, 1-1 Hikari-no-oka,
Yokosuka, Kanagawa, Japan
[2] NTT Cyber Space Laboratories, NTT Corporation, 1-1 Hikari-no-oka,
Yokosuka, Kanagawa, Japan
{yoshida.taiga,irie.go,satou.takashi,
kojima.akira,higashino.suguru}@lab.ntt.co.jp

Abstract. Content-based filtering is a popular framework for item recommendation. Typical methods determine items to be recommended by measuring the similarity between items based on the tags provided by users. However, because the usefulness of tags depends on the annotator's skills, vocabulary and feelings, many tags are irrelevant. This fact degrades the accuracy of simple content-based recommendation methods. To tackle this issue, this paper enhances content-based filtering by introducing the idea of tag ranking, a state-of-the-art framework that ranks tags according to their relevance levels. We conduct experiments on videos from a video-sharing site. The results show that tag ranking significantly improves item recommendation performance, despite its simplicity.

Keywords: recommendation, content-based filtering, tag ranking.

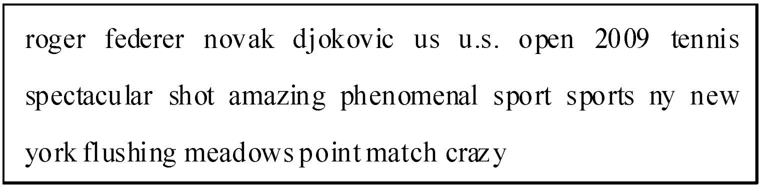
1 Introduction

The number of multimedia contents on the Web is dramatically increasing. This is making it more and more difficult for users to find interesting items. Many recommendation approaches have been proposed to support users in reaching their goal. The most popular approach, collaborative filtering, measures the similarity between items based on users' logs [1] [2]. If the log amount is sufficient, collaborative filtering works well. However, it fails when the log amount is small[3].

Content-based filtering that measures the similarity between item contents is a promising approach for resolving this issue. Multimedia contents sharing web services such as YouTube[1] and Flickr[2] use keywords associated with item

[1] http://www.youtube.com/
[2] http://www.flickr.com/

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 161–172, 2012.



roger federer novak djokovic us u.s. open 2009 tennis spectacular shot amazing phenomenal sport sports ny new york flushing meadows point match crazy

Fig. 1. Tags associated with a YouTube video

meta-data called "tags" for classification of items. Typically, these methods measure the similarity between a pair of items based on the tags associated with them. For instance, the number of co-occurring tags can be used [4]. As two items share a greater number of common tags, the similarity between them increases.

However, using tags in a naïve manner does not always work well because some are irrelevant to the item [5]. Because tags are provided by users, their quality depends on the users skills, vocabulary, and feelings. Fig. 1 shows an example of a list of tags associated with the video uploaded to YouTube titled "Federer Amazing Shot at the US Open 2009 Semifinal"[3]. For instance, "2009" and "crazy" are clearly less relevant to the video than "tennis" and "federer". The first two degrade the performance of item recommendation, because items sharing such irrelevant tags are not similar.

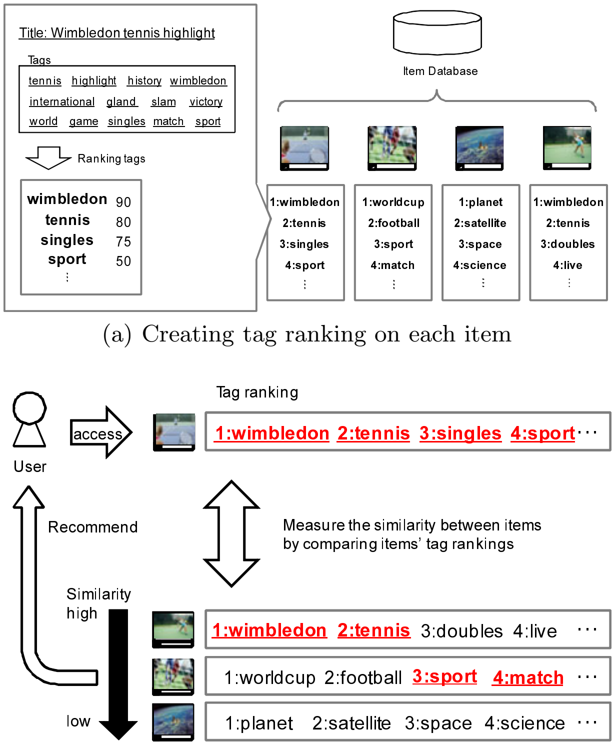
Recently, Liu [6] et al. reported that the performance of auto-tagging and image search can be improved by ranking the tags associated with an image according to their relevance levels to the image content. The idea is very simple and suitable for practical usage.

We here raise a question: is the tag ranking approach effective in the context of item recommendation? If tag ranking is effective for item recommendation, degradation by irrelevant tags can be suppressed in a simple manner. In this paper, we introduce tag ranking into content-based filtering. A ranking of tags is created based on item relevance, and the similarity between items is determined by comparing tag rankings of items. For validating the effectiveness of tag ranking, we conducted some experiments. The results show that tag ranking is effective for recommendation, and content-based filtering can be improved simply by introducing tag ranking.

2 Recommendation Based on Tag Ranking

Fig. 2 overviews a recommendation scheme based on tag ranking. At first, the tags set on each item are ranked according to their importance. If items share several tags that have high ranking, the similarity between them is assessed to be high. Items that are highly similar to those in a user's log are recommended to the user.

[3] http://www.youtube.com/watch?v=RJuEzJEQ9N4


Title: Wimbledon tennis highlight
Tags
tennis highlight history wimbledon
international gland slam victory
world game singles match sport
Ranking tags
wimbledon 90
tennis 80
singles 75
sport 50
Item Database
1:wimbledon 2:tennis 3:singles 4:sport
1:worldcup 2:football 3:sport 4:match
1:planet 2:satellite 3:space 4:science
1:wimbledon 2:tennis 3:doubles 4:live
User
access
Tag ranking
1:wimbledon 2:tennis 3:singles 4:sport ···
Recommend
Measure the similarity between items
by comparing items' tag rankings
Similarity
high
low
1:wimbledon 2:tennis 3:doubles 4:live ···
1:worldcup 2:football 3:sport 4:match ···
1:planet 2:satellite 3:space 4:science ···

(a) Creating tag ranking on each item

(b) Measuring similarities between items based on tag ranking

Fig. 2. Overview of recommendation based on tag ranking

2.1 Creating Tag Ranking

There are several approaches on which tag ranking can be based. One simplest approach is TF-IDF [7], a general indicator of the importance of keywords based on the frequency of the keyword in the item (TF) and the inverse of the number of items that were assigned the keyword (IDF). In the case of social tagging, the same tag is not usually associated with an item more than once, so TF value is always $\{0, 1\}$. The IDF value of a keyword rises if it is contained in fewer items. However, such rare tags are not always relevant to the item. Therefore, TF-IDF is not appropriate for social tags.

The tag ranking proposed by Liu [6] estimates tag relevance levels by applying a probabilistic method and random walk-based refinement. This approach is effective for ranking tags according to their relevance to items. However, it requires image features for ranking tags, so some modifications would be needed to apply this approach to other media such as videos.

A more typical approach for capturing the relevance of tags is co-occurrence. This approach is based on the idea that semantically related tags co-occur frequently. For instance, "wimbledon" frequently co-occurs with "tennis", because these tags are related to each other. This co-occurrence based approach is very simple and is applicable to various media, so we adopt it for tag ranking. We extract two different bits of information from tag co-occurrence data: Co-occurrence Depth and Co-occurrence Width.

If many tags that are semantically related to each other are associated with an item, they may be important tags for it. This is because if these tags are strongly related to the item, users may associate it with the item even when other semantically related tags are already associated with the item.

Co-occurrence Depth scores are calculated based on the co-occurrence between tags associated with the same item. As an example, we describe the calculation of Co-occurrence Depth score for the tags associated with the item "Wimbledon tennis highlight". Some tags such as "tennis", "wimbledon", "game", "singles" or "sport" are associated with the item. "Tennis" often co-occurs with many tags associated with the same item such as "wimbledon" , "singles" or "sport". On the other hand, "game" often co-occurs with tags for video games and more rarely found with tags about tennis. The score of "tennis" should be higher than that of "game", because "tennis" is more relevant to the item than "game". Co-occurrence Depth score increases if the tag co-occurs with tags on the same item more frequently.

We denote a certain item in the item database as i_n, and tags associated with i_n as $T^{i_n} = \{t_m^{i_n} | m = 1, 2, ..\}$. Co-occurrence Depth score $S_d(i_n, t_m^{i_n})$ of tag $t_m^{i_n}$, which is associated with item i_n, is calculated as follows.

$$S_d(i_n, t_m^{i_n}) = \sum_{t_l \in T^{i_n} \, s.t. \, t_l \neq t_m^{i_n}} C(t_l, t_m^{i_n}) \tag{1}$$

$C(t_x, t_y)$ is a chi-square value whose null hypothesis is that tags appear independently in the item database. The value of $C(t_x, t_y)$ increases when tags t_x and t_y have high positive correlation. If there is a negative correlation between t_x and t_y, the sign of $C(t_x, t_y)$ flips to minus.

By calculating scores based on only Co-occurrence Width, both scores of "wimbledon" and "sport" can be high in the same way. However, "wimbledon" is more relevant to the item than "sport", because "wimbledon" is more specific than "sport". We also use the specificity of the tag for creating tag ranking.

Co-occurrence Width scores are calculated based on the variety of co-occurring tags in the item database. For instance, the score of "wimbledon" should be high because it is associated with items only about tennis and the variety of co-occurrence tags is small. On the other hand, the score of "sport" should be low because it is also associated with items other than tennis such as football, baseball, golf and so on. Co-occurrence Width score increases if the tag co-occurrence with other tags in the database do not vary widely.

The Co-occurrence Width score of a tag is calculated from entropy, which is an indicator of degree of variability. Co-occurrence Width score $S_w(t_m^{i_n})$ of tag $t_m^{i_n}$ is calculated as follows.

$$E(t_m^{i_n}) = -\sum_{t_l \in T_m^{i_n}} \frac{N_{t_m^{i_n},t_l}}{N_m^{i_n}} \log \frac{N_{t_m^{i_n},t_l}}{N_m^{i_n}} \tag{2}$$

$$S_w(t_m^{i_n}) = e^{-E(t_m^{i_n})} \tag{3}$$

$T_m^{i_n}$ is a set of tags associated with the same items with $t_m^{i_n}$. $N_{t_m^{i_n},t_l}$ is the number of items with which both $t_m^{i_n}$ and t_l are associated. $N_m^{i_n}$ is calculated by $\sum_{t_l \in T_m^{i_n}} N_{t_m^{i_n},t_l}$.

Although tag ranking can be created from either of these co-occurrence scores, we simply combine them by multiplying them first. The importance score $S(i_n, t_m^{i_n})$ of tag $t_m^{i_n}$ associated with item i_n is calculated as follows.

$$S(i_n, t_m^{i_n}) = S_d(i_n, t_m^{i_n}) S_w(t_m^{i_n}) \tag{4}$$

Tag ranking for item i_n is created by ordering the tags associated with it in descending order of importance score.

2.2 Ordering Items Based on Tag Ranking

If two items share tags that are placed high in their tag rankings, those items may be similar. Thus similarity $R_i(i_m, i_n)$ between items i_m and i_n is calculated as follows.

$$R_i(i_m, i_n) = \sum_i \sum_j \frac{1}{ij} \delta(t_i^{i_m}, t_j^{i_n}) \tag{5}$$

$t_i^{i_m}$ is the i-th tag in the tag ranking associated with i_m. $\delta(t_i^{i_m}, t_j^{i_n})$ is a function whose value is 1 when $t_i^{i_m}$ is equal to $t_j^{i_n}$ and 0 otherwise. The value of $R_i(i_m, i_n)$ increases when i_m and i_n share many common tags with high ranking.

Items that have high similarity with items in the user's access log are recommended to the user. When items $I = \{i_k | k = 1, 2, ..\}$ are in the user's access log, the recommend score $R(I, i_n)$ of item i_n is calculated by the following equation.

$$R(I, i_n) = \sum_{i_k \in I} R_i(i_k, i_n) \tag{6}$$

Items are ordered in descending order of their recommend scores and recommended to the user.

3 Experiments

3.1 Experimental Conditions

We conducted experiments to validate the performance of tag ranking in the context of item recommendation. We used 14,159 videos downloaded from a

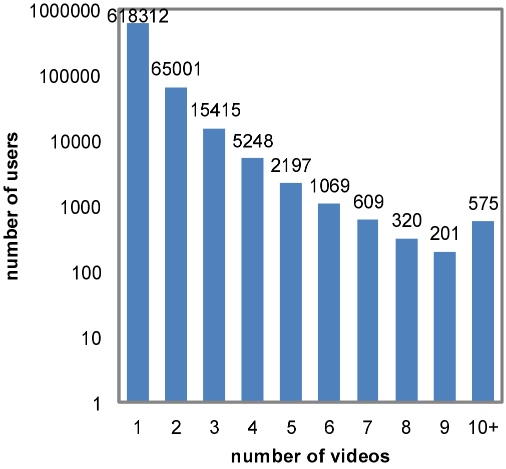


Fig. 3. Distribution of number of videos user viewed

popular video sharing site in the experiments. They contained 20 categories of genre, and one genre was assigned to each video. Evaluations of recommendation methods generally use the precision of predictions as discerned from users' access logs [9]. In this work, we performed an evaluation by taking the comment logs of users as access logs. In our dataset, 850,881 comments were attached to videos by 708,947 users. Fig. 3 shows the distribution of the number of videos viewed by individual users. In the experiments, we used the logs of 2,774 users, each of whom watched over 6 videos, for evaluating the dependency of precision on the amount of users' logs. We divided them into 1,387 learning users and 1,387 test users. Recommendation precision was taken to be the precision with which the user's 6th video (called target video) in the test user's log was predicted; each method assessed the 1st to the 5th item in the user's log to predict the 6th item, which is viewed next by the user. We evaluated the precision of each method by mean average precision (MAP) [10]. We compared the following 6 methods.

- **Content-based Methods**
  Tag-Rank: bases recommendations on tag ranking
  Jaccard: calculates similarities between items from Jaccard coefficients of tags
  Genre: recommends items whose genre is the same as the item randomly selected from the user's access log
- **Log-based Methods**
  Item-CF: calculates similarities between items using accessed user logs of items (item-based collaborative filtering)
  User-CF: calculates similarities between users using accessed item logs of users (user-based collaborative filtering)
  Ranking: recommends items that have higher rank in access ranking but have not yet been accessed by the user

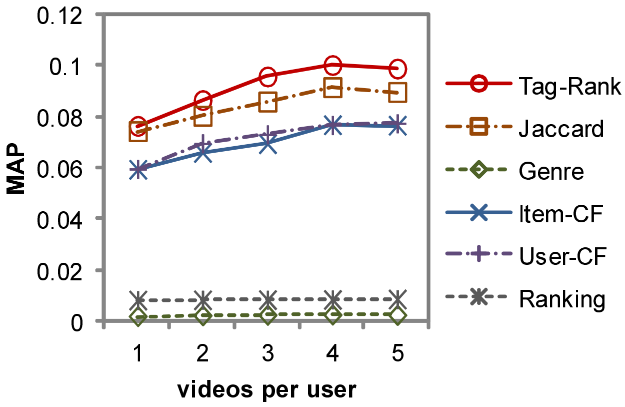


Fig. 4. MAP vs. user log number (single)

We conduct two different experiments. First, we compare each single method. Next, we validate accuracy when combining Tag-Rank with log-based methods. In each experiment, we measure MAP under the conditions of changing the number of items accessed by a user and MAP under the condition of changing the popularity of the target video.

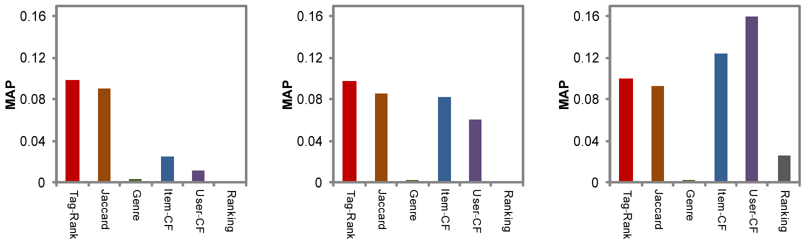
3.2 Comparing Single Methods

In the experiments, we compared three content-based methods and three log-based methods.

MAP vs. User Log Number. We evaluated MAP values of the target video while varying the number of entries in each user's log from 1 to 5. Fig. 4 shows the results of the experiment. The horizontal axis of the figure is the number of accessed items per user, and the vertical axis is MAP value.

The results show that Tag-Rank was the best of the 6 methods. The 3 log-based methods do not work when the users' access logs had few items. On the other hand, because Tag-Rank uses tags associated with videos for recommendation, it is able to measure similarity between videos precisely even when the users' access logs have few items. If each user's access log held many items, log-based methods are expected to top Tag-Rank. However, this situation is not common and Tag-Rank provides adequate performance for practical numbers of items. Moreover, precision can be improved by combining Tag-Rank with a log-based method as described below.

Among the 3 content-based methods, Tag-Rank and Jaccard are better than Genre. Since many videos were assigned the same genre, genre information was not discriminatory enough to assess the similarities between videos. Tag-Rank had higher MAP than Jaccard. We performed the Wilcoxon signed rank test for



(a) Tail (b) Mid (c) Head

Fig. 5. MAP vs. view number (single)

Tag-Rank and Jaccard while varying the items in each user's log from 1 to 5. There was a significant difference at the 1% significant level in all conditions.

MAP vs. View Number. We evaluated MAP values of target videos with different levels of popularity.

In this experiment, we divided test users into 3 groups according to the number of the target videos viewed.

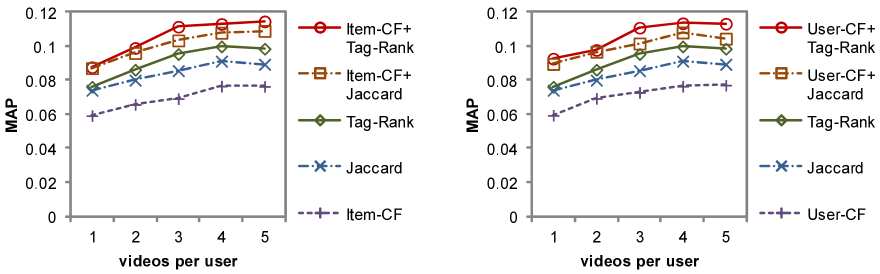
Tail: the target item is in the bottom one third of all videos
Mid: the target item is in the mid third of all videos
Head: the target item is in the top one third of all videos

We evaluated MAP values of the target videos in each group. Fig. 5 shows the results. The vertical axis is MAP value calculated from the 1st to the 5th videos in each user's log.

The results show that Tag-Rank had the best MAP values for the Tail group and the Mid group. User-CF had the best MAP values for the Head group. When using log-based methods for item recommendation, prediction accuracy is relative high if user logs contain many items, i.e. training data is sufficient. Since Tag-Rank does not depend on the number of times the target video is viewed, MAP offers high performance even when log amount is small.

Comparing the methods, Tag-Rank has higher MAP than Jaccard in all groups. Because Jaccard does not consider relevance levels of tags, it frequently recommends unsuitable videos. On the other hand, Tag-Rank emphasizes tags relevant to the video, so Tag-Rank offers high MAP. Item-CF and User-CF have high MAP when recommending videos in the Head group, but low MAP when recommending videos in the Tail group.

The precision of log-based methods varies according to log amount, but that of Tag-Rank is high and does not depend on item popularity. Tag-Rank is especially effective when recommending items with low view counts.



(a) Combination with Item-CF (b) Combination with User-CF

Fig. 6. MAP vs. user log number (hybrid)

3.3 Hybrid Methods

Combining log-based methods with content-based filtering approaches is a simple way of achieving high performance in a wider variety of situations [11] [12].

We evaluated MAP values when combining a content-based method: Tag-Rank or Jaccard with a log-based method: Item-CF or User-CF. The scores were calculated by summing up the normalized scores of the methods used.

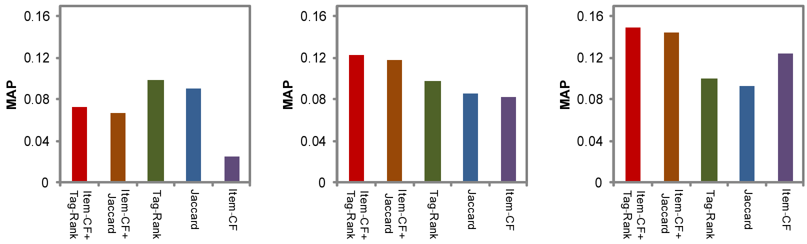
MAP vs. User Log Number. We evaluated MAP values of the target video while varying the number of items in each user's log from 1 to 5. Fig. 6 shows the results. The horizontal axis is the number of items accessed per user, and the vertical axis is MAP value.

The MAP values show that Tag-Rank combinations are superior to the Jaccard combinations in all conditions. Combining Item-CF or User-CF with Tag-Rank yields MAP values above the MAP values of the constituent methods used in isolation.

MAP vs. View Number. We also evaluated the MAP values of target videos with different levels of popularity. Fig. 7 shows the results for a content-based method with Item-CF. Fig. 8 shows the results for a content-based method with User-CF. The vertical axis is the MAP value when each user's log contains from 1 to 5 videos.

The results show that the Tag-Rank combinations are superior to the Jaccard combinations in all conditions. Tag-Rank well compensates the weak point of log-based methods with regard to recommending items in the Tail group. Even in the Mid group and the Head group, Tag-Rank can improve the MAP values of log-based methods.

From the results of the above experiments, we conclude that Tag-Rank improves recommendation performance despite its simplicity.

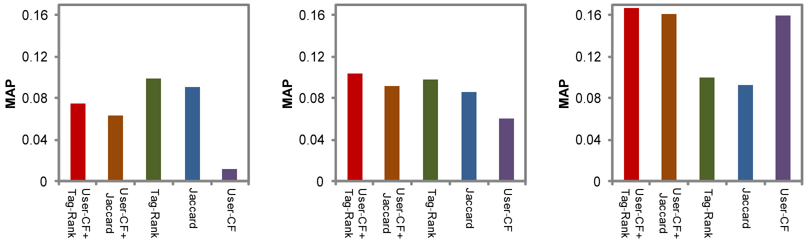


(a) Tail

(b) Mid

(c) Head

Fig. 7. MAP vs. view number (hybrid, combination with Item-CF)



(a) Tail

(b) Mid

(c) Head

Fig. 8. MAP vs. view number (hybrid, combination with User-CF)

3.4 Effect of Co-occurrence Depth Score and Co-occurrence Width Score

In this work, we create tag ranking based on two co-occurrence scores: Co-occurrence Depth score and Co-occurrence Width score. Co-occurrence Depth score indicates the relevance of the tag to the item. Co-occurrence Width score indicates the specificity of the tag. Tag ranking can be created based on combining them and also based on either of them. We compared the MAP value of each co-occurrence score when each user's log contained from 1 to 5 items. Fig. 9 shows the results.

For 1 or 2 items, Co-occurrence Width score yields higher MAP but above 2, the combination of Co-occurrence Depth score and Co-occurrence Width score offers the best performance. In this experiment the combination of co-occurrence scores is calculated by simple multiplication. For example, the MAP values might be improved by attaching a high weight to Co-occurrence Width scores when the log amount is small.

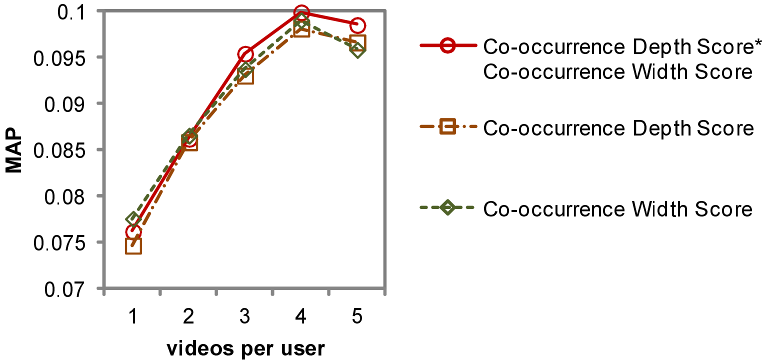


Fig. 9. Comparing co-occurrence scores

4 Conclusions

We proposed herein the idea of introducing tag ranking to improve recommendation precision. Tag ranking reflects tag importance as calculated by their co-occurrence. The similarity between items is measured by comparing their tag rankings. Items similar to those in the user's log are recommended to the user. In order to validate the effectiveness of tag ranking, we performed experiments on data from logs of a video sharing site. The experiments showed that our simple tag ranking approach can well improve the precision of content-based filtering. We also confirmed that the precision is improved by combining content-based methods with our proposed simple tag ranking method. We plan to validate the effect of tag ranking in detail by performing experiments on larger datasets and on other type of datasets. We also plan to examine other tag ranking methods.

References

1. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-Based Collaborative Filtering Recommendation Algorithms. In: 10th International Conference on World Wide Web (WWW), pp.285–295 (2001)
2. Breese, J.S., Heckerman, D., Kadie, C.: Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In: 14th Conference on Uncertainty in Artificial Intelligence (UAI), pp. 43–52 (1998)
3. Maltz, D., Ehrlich, K.: Pointing the Way: Active Collaborative Filtering. In: SIGCHI Conference on Human Factors in Computing Systems (CHI), pp. 202–209 (1995)
4. Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., Stumme, G.: Evaluating Similarity Measures for Emergent Semantics of Social Tagging. In: 18th International Conference on World Wide Web (WWW), pp. 641–650 (2009)
5. Golder, S., Huberman, B.A.: Usage patterns of collaborative tagging systems. Journal of Information Science 32(2), 198–208 (2006)

6. Liu, D., Hua, X.-S., Yang, L., Wang, M., Zhang, H.-J.: Tag Ranking. In: 18th International Conference on World Wide Web (WWW), pp. 351–360 (2009)
7. Salton, G.: Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley (1989)
8. Sigurbjörnsson, B., Zwol, R.V.: Flickr tag recommendation based on collective knowledge. In: 17th International Conference on World Wide Web (WWW), pp. 327–336 (2008)
9. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating Collaborative Filtering Recommender Systems. ACM Transactions on Information Systems 22(1), 5–53 (2004)
10. Yates, R.-B., Neto, B.-R.: Modern Information Retrieval. Addison Wesley (1999)
11. Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D., Sartin, M.: Combining Content-Based and Collaborative Filters in an Online Newspaper. In: ACM SIGIR Workshop on Recommender Systems (1999)
12. Pazzani, M.: A Framework for Collaborative, Content-Based, and Demographic Filtering. Artificial Intelligence Review, 393–408 (1999)

Double Fusion for Multimedia Event Detection

Zhen-zhong Lan, Lei Bao, Shoou-I Yu, Wei Liu, and Alexander G. Hauptmann

School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA
{lanzhzh,iyu,alex}@cs.cmu.edu.com, {lei.bao.cn,lwbiosoft}@gmail.com

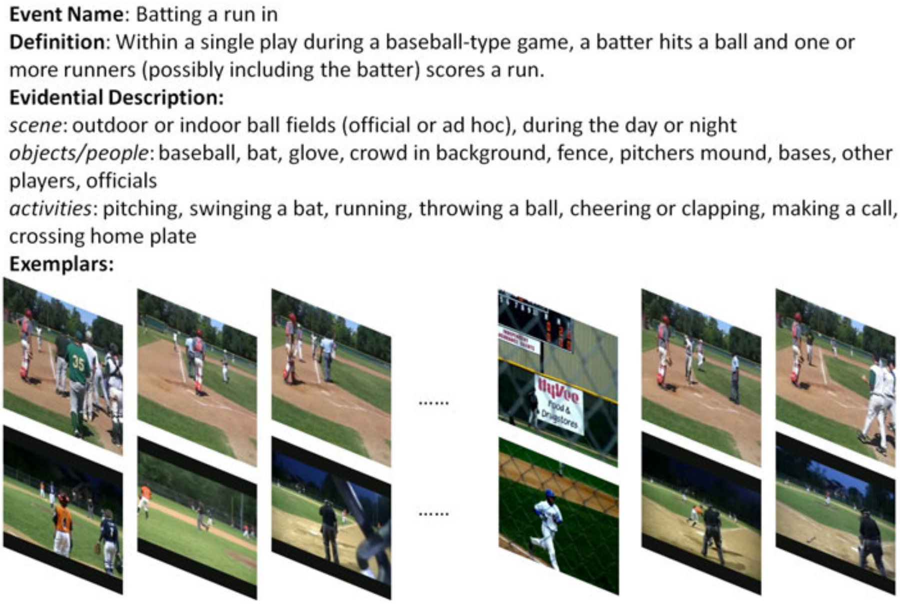
Abstract. Multimedia Event Detection is a multimedia retrieval task with the goal of finding videos of a particular event in an internet video archive, given example videos and descriptions. We focus here on mining features of example videos to learn the most characteristic features, which requires a combination of multiple complementary types of features. Generally, early fusion and late fusion are two popular combination strategies. The former one fuses features before performing classification and the latter one combines output of classifiers from different features. In this paper, we introduce a fusion scheme named double fusion, which combines early fusion and late fusion together to incorporate their advantages. Results are reported on TRECVID MED 2010 and 2011 data sets. For MED 2010, we get a mean minimal normalized detection cost (MNDC) of 0.49, which exceeds the state of the art performance by more than 12 percent.

Keywords: Feature Combination, Early Fusion, Later Fusion, Double Fusion, Multimedia Event Detection.

1 Introduction

In recent years, due to its great potential for many applications, the explosive growth of the user generated online videos and the prevailing online communities such as YouTube, Hulu etc., automatic detection of complex events in unconstrained videos has received a lot of interest from the research community [1] [2] [3]. However, most current tools only focus on single modality such as automatic transcription of speech from audio signal, scene recognition using color features or action detection based on time-related features. How to combine these state-of-the art approaches to build an accurate, fast and robust multimedia system to help users to study these overwhelming video data is still an open question. Many research in progress during the past few years still focus on the following two tracks: the design of highly discriminative and robust features [4] and the combination of multiple complementary features based on different modalities such as visual, audio and text [5] [6] [7] [9] [10]. For example, in 2010, NIST held the first Multimedia Event Detection (MED) evaluation [7] [10], which emphasis the importance of combining multiple modalities for event detection. As shown in Fig. 1, the task is: given an Event Kit (including an description of the concepts and some example videos), find videos that belong to the event defined by the Event Kit. In this paper, we will deal with the same task.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 173–185, 2012.



Event Name: Batting a run in

Definition: Within a single play during a baseball-type game, a batter hits a ball and one or more runners (possibly including the batter) scores a run.

Evidential Description:

scene: outdoor or indoor ball fields (official or ad hoc), during the day or night

objects/people: baseball, bat, glove, crowd in background, fence, pitchers mound, bases, other players, officials

activities: pitching, swinging a bat, running, throwing a ball, cheering or clapping, making a call, crossing home plate

Exemplars:

Fig. 1. The illustration MED task

Many research papers [5] [7] [8] [9] [10] state that a multimodal approach helps to obtain an effective retrieval/classification performance on image and video. In general, there are two types of combination strategies, namely early fusion and late fusion [9]. Early fusion combines feature before performing classification, such as multi-kernel learning [11] [12]. Late fusion combines output of classifiers from different features, such as average fusion, committee voting [13] and co-regularized least squared regression [14]. There is no universal conclusion of which strategy is the preferred method for multimedia content analysis and retrieval. [9] found that early fusion is better than late fusion in semantic indexing based on their results on TRECVID 2004 benchmark. While studying data on TRECVID 2006, [15] found that early fusion gets better results on most of concepts while late fusion is more robust and can tackle some harder concepts. To incorporate the advantages of both methods, we introduce a simple yet efficient fusion strategy called double fusion. In double fusion, we first perform early fusion to generate different combinations of features from subsets on the single features pool. After that, we train classifiers on each feature or feature combination and carry out late fusion on the output of these classifiers. For example, as shown in Fig. 2, we first extract three kinds of features (visual, audio and text) from three training and three testing videos. After that, pairwise early fusion (visual+audio, visual+text) are carried out in these three features based on their kernel matrice. In the training step, five classifiers are trained based on five features and feature combinations (visual, audio, text, visual+audio, visual+text). For each video, there are thus five output scores indicating how likely it is that this video belongs to the event. In the last step, late fusion is used to fuse five output score vectors into one score vector, on which the final interpretation can be executed. Experimental results on the TRECVID MED 2010 and MED 2011 data sets with about 484 hours' video clips for 18 events show the effectiveness of double fusion. For MED 2010 we get a mean minimal normalized detection cost (MNDC) of 0.49, which exceeds the state of the art performance [7] by more than 12 percent.

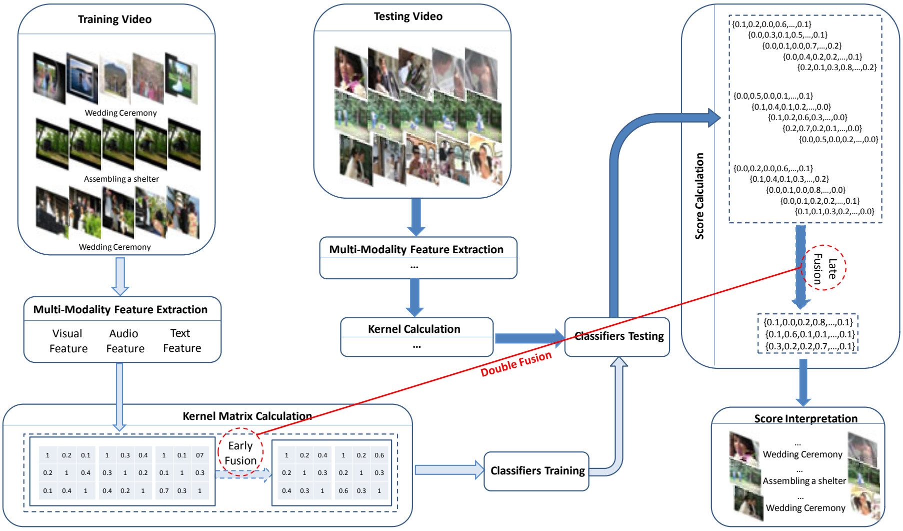


Fig. 2. The illustration of our MED system

The remainder of the paper is organized as follows. Section 2 briefly introduces different fusion strategies. Section 3 presents the details of our implementation, including feature representation, bag-of-words scheme, classifiers and fusion schemes. Section 4 demonstrates and analyzes experimental results on MED 2010 and MED 2011. Finally, section 5 concludes the paper and outlines our future work.

2 Fusion Scheme

Early Fusion [9] is a combination scheme that runs before classification. Both feature fusion and kernel space fusion are example of early fusion. The main advantage of early fusion is that only one learning phase is required. However, it is hard to combine features into a common representation [9]. Multiple kernel learning is one of the most popular early fusion technologies. Its drawback is the curse of high dimensionality, usually accompanied by limited training data.

In contrast to early fusion, late fusion [9] happens after classification. While late fusion is easier to perform, in general, it needs more computational effort and has potential to lose the correlation in mixed feature space. Normally, another learning procedure is needed to combine these outputs, but in general, because of the overfitting problem, simply averaging the output scores together yields better or at least comparable results than training another classifier for fusion.

As shown in paper [9], there is no conclusion about which fusion scheme will get better performance. For some concepts such as stock quotes, early fusion get better result, for other concepts such as road, late fusion get better performance. Could we come up a solution to combine the strengths of both early and late fusion? In this paper, we introduce a method called double fusion, which combines early fusion and late fusion together. Specifically, for early fusion, we fuse multiple subsets of single features by using standard early fusion technologies; for late fusion, we combine output of classifiers trained from single and combined features. By using this scheme, we can freely combine different early fusion and late fusion techniques, and get benefits of both.

Two early fusion strategies, i.e., rule-based combination and multiple kernel learning [12], are used to combine kernels from different features. For rule-based combination, we use the average of the kernel matrix. Multiple kernel learning [12] is a natural extension of average combination. It aims to automatically learn the weights for different kernel matrix. Our experimental results show that the performance of multiple kernel learning is slightly better than average combination. However, because of the explosive number (the number of combination is $2^n - 1$, n is the number of features) of combination, it is time consuming to use all possible feature combination when the feature space becomes large. To address this problem, our first possible solution is by combing features belonging the same categories. For each category or single feature, we train one classifier. The number of classifiers for late fusion will be n+c, in which c is the number of category. Our second solution is to combine all features together in early fusion and perform late fusion with all single feature classifiers that results in n+1 classifiers need to be fused in later fusion. In this paper, we use both approaches and train n+c+1 classifiers for late fusion, in which there are c early fusion classifiers built on category-based features, n single feature classifiers and one early fusion classifier trained on the combination of all features. This allows us to exploit the advantages of single feature classifier, category-based classifier and complete-feature classifiers.

3 Implementation

As shown in Fig. 2, there are four key steps in our system. In step one, we perform feature extraction on visual, textual and audio modality. After modality specific data processing, bag-of-words representation is used to aggregate the point features into whole video features. Early fusion is applied in step two after calculating the kernel matrix. In step three, classifiers are trained to perform the classification. The outputs of different classifiers are combined by using late fusion strategies in step four.

Feature Extraction and Feature Representation. Feature representation is critical for video content understanding. In TRECVID MED System, we explore three feature modalities including visual features, audio features and text features.

Visual Feature. We use five visual features, namely SIFT [20], CSIFT [16], MoSIFT [17], STIP [18] and GIST [4] .

For SIFT feature and CSIFT, the harris-laplace key point detector is used to detect key points. As processing all MED video frames will be computationally expensive, we only extract features from key frames extracted by a shot boundary detection algorithm. Specifically, the algorithm calculates the color histogram for every five frames and subtracts the histogram with the histogram of the previous frame, if the subtracted value is larger than a certain threshold, which is empirically setted, the key frame will be a shot boundary. After detecting the shot, we use the frame in the middle of the shot to represent that shot. By using this algorithm, we extracted 114992 key frames from MED 2010 and 364747 key frames from MED 2011 development data.

While SIFT and CSIFT describe 2D local structure in images, space-time interest points (STIP) and MoSIFT capture space time volumes where the image values have significant local variations in both space and time. STIP and MoSIFT are different in both key points detector and descriptor. STIP uses 3D Harris corner detectors and its key points are represented in two parts: the first part is HOG (Histograms of Oriented Gradients; 72 dimensions) which indicates the spatial appearance and the second part is HOF (Histograms of Optical Flow; 90 dimensions) describing the motion information. MoSIFT uses a Difference of Gaussian (DoG) based detector and represents by another descriptor which is also concatenated from two parts: the first part is SIFT (128 dimensions) which indicates the spatial appearance and the second part is also HOF (128 dimensions).

For the GIST feature, we follow the suggestion from [4] and set the dimension of feature points to 960.

Audio Feature. For the audio feature, we used Automatic Speech Recognition (ASR) feature, which is extracted as described in [10].

Textual Feature. Following the work of [10], we use Optical Character Recognition (OCR) feature extracted by the Informedia system to represent the text feature.

Bag-of-words Representation. After extracting above features from given videos, a formal Bag-of-words representation is adopted to cast features of key frames into fixed length feature vector. First, vector quantization (VQ) technique is used to cluster feature descriptors into a large number of clusters (i.e. 'words') using k-means clustering algorithm. For visual features, the code book size is 4096 except for GIST, which has 960 dimensions. Second, by mapping these features into their cluster centroid, we can get a feature representation for each key frame. Here, we adopt a soft-weight strategy in which we choose the ten nearest clusters and assigned a rank weight for them. For using these words to represent the videos, we need to cast image feature into video feature. For SIFT, CSIFT and GIST, we first normalize feature vectors of each key frame in a video and then sum them together to represent the video. For STIP and MoSIFT, we just sum all the feature points in a video together and normalize it. As for ASR and OCR, we simply count the number of words or tokens found in videos. There are a total of 11618 unique words and 180228 unique tokens extracted for ASR and OCR, respectively.

Spatial Pyramid Matching. Since the classic bag-of-words method loses all information about the spatial layout of features,[19] adopt the pyramid matching scheme by repeatedly subdividing the image and computing histograms of local features for each sub-regions. Specifically, besides the bag-of-word representation for the whole image, we divided the keyframe into 2x2 and 1x3 sub-regions, and computed the bag-of-word representation for each sub-region. Thus, the feature dimension for the spatial pyramid matching is 8x4096=32768. We applied this simple yet effective method for SIFT and CSIFT features.

Classifier. A large variety of classifiers exist for mapping the feature space into score space. In this paper, we adopt two classifiers, i.e. non-linear support vector machine (SVM) [21] and kernel regression (KR) [14]. SVM is one of the most commonly used classifier due to its simple implementation, low computational cost, relatively mature theory and high performance. In TRECVID MED 2010, most of the teams [7] [8] use SVM as their classifiers. Compared to SVM, KR is a simpler but less used algorithm. However, our experiment shows that the performance of KR is consistently better than the performance of SVM.

Fusion. In our feature set, only visual feature set has multiple features, while all other features represent each category by its own. By performing visual feature (SIFT, CSIFT, MoSIFT, STIP, GIST) combination and all-feature (SIFT, CSIFT, MoSIFT, STIP, ASR, OCR, GIST) combination, we have two feature combination and seven single features (SIFT, CSIFT, MoSIFT, STIP, ASR, OCR, GIST). For late fusion, we use two rule-based fusion methods to combine the output of above 9 classifiers. One is average combination, another one is weighted combination using weight learned from cross-validation. The detail of the weight calculation will be given in the experimental part.

4 Experiment

4.1 Data

For TRECVID MED 2010, we used both the annotated training and testing data, which consists of 114 hours of video clips and three event kits, i.e., "Making a cake","Batting a run" and "Assembling a shelter". For MED 2011, currently, we only have the annotated development data of MED 2011, which consists of about 370 hours of video clips and 15 events including 5 training events (Attempting a board trick, Feeding an animal, Landing a fish, Wedding ceremony and Working on a woodworking project) and 10 testing events (Birthday party, Changing a vehicle tire, Flash mob gathering, Getting a vehicle unstuck, Grooming an animal, Making a sandwich, Parade, Parkour, Repairing an appliance and Working on a sewing project). To test the performance of our system on MED 2011 dataset, we manually split the 10 testing events into same size of training and testing data. After the splitting, we have 3135 video clips for training and a 6687 video set for testing on MED 2011.

We ran our program on the Carnegie Mellon University Parallel Data Lab cluster, which contains 300 cores and it took us about 57000 CPU hours to extract features and perform the bag-of-words mapping.

4.2 Evaluation

For performance comparison, two evaluation schemes are adopted: the first one is the MNDC, which, as indicated in formula 1, is an evaluation criteria for NIST to evaluate MED 2010 and MED 2011. Lower MNDC indicates better performance. For better understanding, we also use maximum F1 Score by using test label to search the best threshold. Considering that we have 100 times negative sample than positive samples for each event, MNDC is still a better criteria for evaluation since it gives more weight on the cost of false alarm. However, both of above two criteria are highly depended on threshold and are not stable for evaluation.

$$NDC(S,E) = \frac{C_M * P_M(S,E) * P_T + C_{FA} * P_{FA}(S,E) * (1 - P_{FA}(S,E))}{MINUMUM(C_M * P_T, C_M * (1 - P_T))} \quad (1)$$

where $P_M(S,E)$ is the missed detection probability for system S, event E while $P_{FA}(S,E)$ is the false alarm probability for system S, event E. $C_M = 80$ is the cost for missed detection, $C_{FA} = 1$ is the cost for false alarm and $P_T = 0.001$.

4.3 Parameter Selection

For both SVM and KR, we used a χ^2 kernel [22] since all of our features are histogram features and the χ^2 kernel has been extensively used for histogram features. A parameter γ is needed for χ^2 kernel. For SVM, we have one additional regularization parameter C. To optimize these parameters, we ran two-folded cross-validation 10 times by randomly spliting the training data into two folds. Then, the average MNDC of two folds are used to choose the best parameters. We also use the average MNDC to generate weights to perform weight averaging for late fusion. The search ranges for both C and γ are 10^{-3} to 10^3, in multiples of 10. We did try small step size search for parameter selection suggested by [23], but didn't find much difference.

4.4 Results

To get a statistically meaningful experiment, for each setting, we run 10 times and calculate the mean and standard deviation for that setting. Because running

Table 1. Comparison of single features on TRECVID MED2010. Two evaluation criteria including MMNDC and MMF1 are adopted. For MMNDC, lower score indicates better performance; for MMF1, higher score means better performance.

Feature	MMNDC %± STD	MMF1%± STD
CSIFT	60.6 ± 0.7	52.5 ± 0.6
SIFT	**60.5 ± 1.4**	**53.3 ± 1.1**
MoSIFT	63.9 ± 1.4	50.6 ± 0.9
STIP	69.1 ± 0.5	48.2 ± 1.8
GIST	82.9 ± 1.5	33.7 ± 0.7
ASR	89.1 ±4.7	22.5 ± 4.1
OCR	85.7 ± 0.1	28.8 ± 0.8

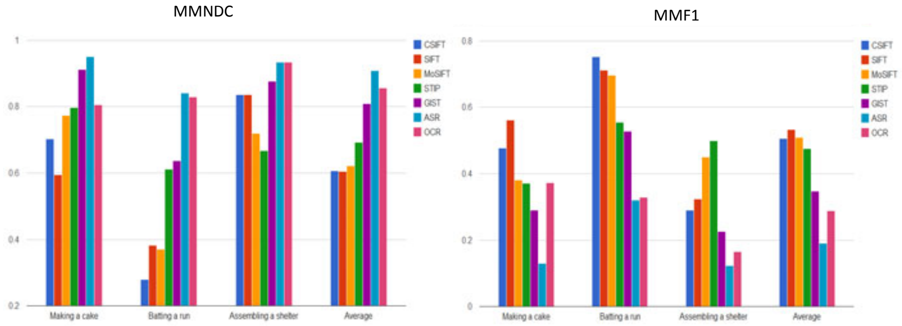


Fig. 3. Comparison of single feature on TRECVID MED2010. Two evaluation criteria including MMNDC and MMF1 are adopted. For MMNDC, lower score indicates better performance; for MMF1, higher score means better performance.

all the combination of fusion strategies and classifiers will be computational expensive and meaningless to our concern, we first compare all the classifiers, early fusion and late fusion strategies on MED 2010 and choose the best strategy for each step to perform further experiments on MED 2011.

Single Feature Comparison. First, we compare the mean MNDC (MMNDC) (lower MMNDC indicates better performance) and mean MF1 (MMF1) (higher MMF1 indicates better performance) of single features on MED 2010. As shown in Table 1 and Fig. 3, the performance of different features vary dramatically from event to event. Generally, four local features including CSIFT, SIFT, MOSIFT and STIP consistently outperform other three features. In these four features, motion based features including MOSIFT and STIP get much better results than static features including SIFT and CSIFT in "Assembling a shelter " event, which has a lot of motion. Contradictorily, static features are obviously superior to other features in "Batting a run" event and "Making a cake", because of their relatively monotonous background. Different matched situation for different features shows that above features are complementary to each other.

Table 2. Comparison of classifiers, early fusion and late fusion strategies on TRECVID MED 2010. Two evaluation criteria including MMNDC and MMF1 are used. For MMNDC, lower score indicates better performance; for MMF1, higher score means better performance.

	Classifiers		Early Fusion		Late Fusion	
	KR	SVM	MKL	Average Fusion	Weighted Fusion	Average Fusion
MMNDC% ± STD	**60.5 ± 1.4**	62.3 ± 1.1	**50.6 ± 0.8**	50.7 ± 0.6	**52.5 ± 1.5**	57.6 ± 1.9
MMF1% ± STD	**53.3 ± 1.1**	50.7 ± 2.9	**61.4 ± 0.1**	61.2 ± 0.6	**59.7 ± 1.1**	54.4 ± 1.6

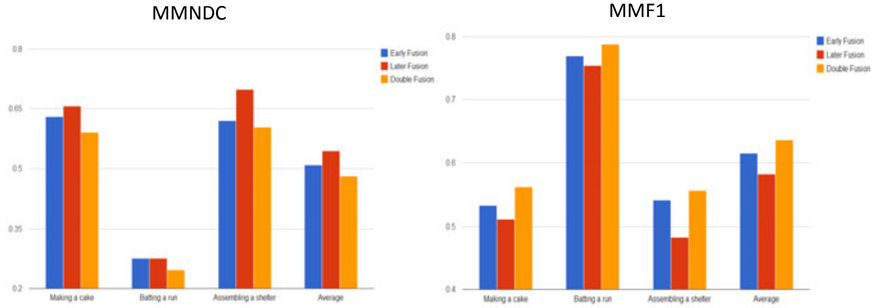


Fig. 4. Comparison of double fusion with early fusion and late fusion on MED 2010. Two evaluation criteria including MMNDC and MMF1 are adopted. For MMNDC, lower score indicates better performance; for MMF1, higher score means better performance.

Also, the performances of ASR and OCR features are much worse than those visual feature. All of these indicate that giving different weights for different features is a promising fusion strategy.

KR versus SVM. We further compared the performance of different classifiers by simply using the best single feature, which is SIFT. From Table 2, we can see that, compared to SVM, KR has lower MMNDC and higher MMF1, which indicate that KR is a better classifier for TRECVID MED task. From now on, we will use KR as our classifier for further experiments in this paper.

Early Fusion Strategies Comparison. For early fusion, we choose either multiple kernel learning oraverage fusion. As indicated in Table 2, we can see that MKL only gets comparable results to simple average fusion, this is consistent with what was suggested by [12]. Considering that the performances of some features are much worse than other features, it is quite unreasonable to give them equal weight. However, finding a better weight strategy is still an open question.

Late Fusion Strategies Comparison. Table 2 shows the results of late fusion using weighted fusion and average fusion. The result of weighted late fusion is

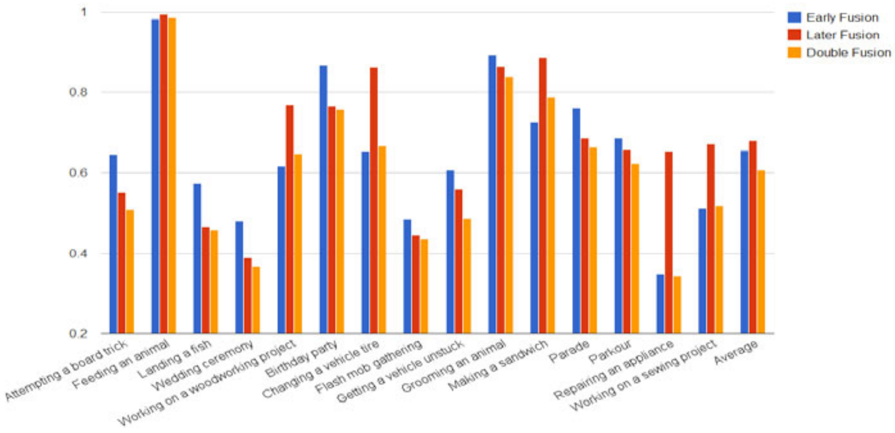


Fig. 5. Comparison of double fusion with early fusion and late fusion on MED 2011 by suing MMNDC criteria. Lower MMNDC indicates better performances.

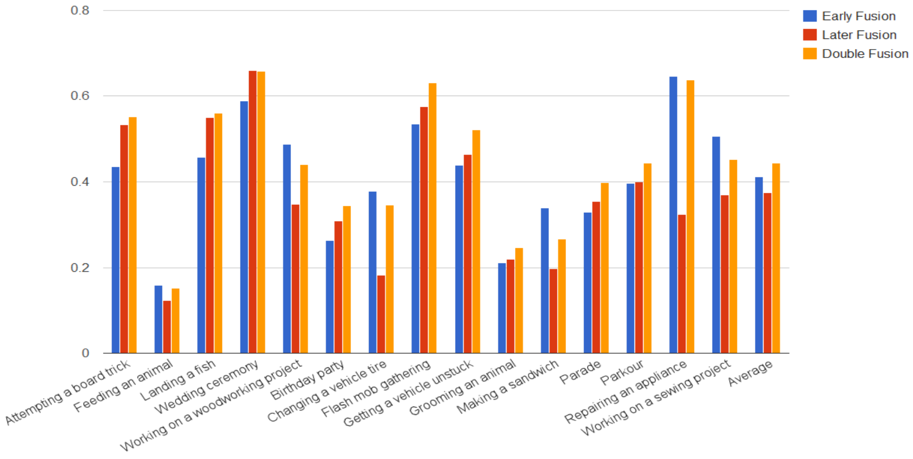


Fig. 6. Comparison of double fusion with early fusion and late fusion on MED 2011 by using MMF1 criteria. Higher MMF1 indicates better performances.

much better than the result of average late fusion. This indicates that different features have different contributions to the final results, especially when the performance varies dramatically between features. We will only use the weighted combination for late fusion for further comparison.

Double Fusion Versus Early Fusion and Late Fusion. The result of double fusion is shown in Table 3. From the table, we can see that double fusion gives much better results than both early and late fusion. The current best result

Table 3. Comparison of double fusion with early fusion and late fusion on MED2010. Two evaluation criteria including MMNDC and MMF1 are adopted. For MMNDC, lower score indicates better performance; for MMF1, higher score means better performance.

	MED 2010			MED 2011		
	Early Fusion	Late Fusion	Double Fusion	Early Fusion	Late Fusion	Double Fusion
MMNDC% ± STD	50.6 ± 0.8	52.5 ± 1.5	**48.9 ± 0.7**	65.6 ± 0.7	68.2 ± 1.3	**60.6 ± 0.8**
MMF1% ± STD	61.4 ± 0.1	59.7 ± 1.1	**62.9 ± 0.6**	41.1 ± 0.5	37.4 ± 3.8	**44.3 ± 0.9**

on TRECVID MED 2010 was achieved [7] using the MMNDC criteria and the performance was 0.565. Compared to this result, we get more than 12 percentages improvements in MMNDC, though results are not perfectly comparable due to different features and machine learning methods. Fig. 4 shows that double fusion gets consistently better performance than early fusion and late fusion on all of three events in MED 2010. MED 2011 is much harder and more diverse than MED 2010 since we have 15 events now, but Fig. 5 and Fig. 6 indicate that double fusion still gets better performance than early fusion and late fusion on 11 of 15 events. For the other 4 events, double fusion still gets similar results to the best methods for those events, which indicates that double fusion does capture advantages of both early fusion and late fusion.

5 Conclusion

In this paper, we presented an analysis of early fusion and late fusion which aims at combining features from different modalities for multimedia event detection and introduced a double fusion scheme which combines early fusion and late fusion together. Our experiments on about 484 hours of videos come from TRECVID MED 2010 and 2011 showed that this simple strategy is very effective and had a substantial advantage over both early fusion and late fusion strategies. Moreover, we found that weighted combination is better than average combination for late fusion but not for early fusion. How to learn weight for early combination is still an open question, our future work will focus on learning weight for early fusion.

Acknowledgments. This work was supported in part by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20068. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes not withstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government. Support was also provided, in part, by the National Science Foundation, under award CCF-1019104, and the Gordon and Betty Moore Foundation, in the eScience project. We thank the Parallel Data Lab for the use of their resources.

References

1. Smeaton, A.F., Over, P., Kraaij, W.: Evaluation campaigns and TRECVid. In: 8th ACM International Workshop on Multimedia Information Retrieval, MIR 2006 (2006)
2. Liu, J., Luo, J., Shah, M.: Recognizing realistic actions from videios 'in the wild'. In: IEEE Conference on Computer Vision and Pattern Recognition CVPR 2009 (2009)
3. Hauptmann, A., Yan, R., Lin, W., Christel, M., Wactlar, H.: Can high- level concepts fill the semantic gap in video Retrieval? A case study with broadcast news. IEEE Transaction on Multimedia 9(5), 958–966 (2007)
4. Oliva, A., Torralba, A.: Modeling the shape of the scene: a holistic representation of the spatial envelope. Int. J. Computer Vision 42(3), 145–175 (2001)
5. Yang, Y., Zhuang, Y., Wu, F., Pan, Y.: Harmonizing hierarchical manifolds for multimedia document semantics understanding and cross-media retrieval. IEEE Transactions on Multimedia (TMM 2008) 10(3), 437–446 (2008)
6. Liu, J., Yang, Y., Shah, M.: Learning semantic visual vocabularies using diffusion distance. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009 (2009)
7. Jiang, Y.G., Zeng, X.H., Chang, S.F., et al.: Columbia-UCF TRECVID 2010 multimedia event detection: combining multiple modalities, contextual concepts, and temporal matching. In: Proceeding TRECVID Workshop (2010)
8. Iyengar, G., Nock, H., Neti, C.: Discriminative model fusion for semantic concept detection and annotation in video. In: Proceedings of 11th Annual ACM International Conference Multimedia, MM 2003 (2003)
9. Snoek, C.G.M., Worringm, M., Smeulders, A.W.M.: Early versus late fusion in semantic video analysis. In: Proceedings of 13th Annual ACM International Conference Multimedia, MM 2005 (2005)
10. Li, H., Bao, L., Hauptmann, A., et al.: Informedia@ TRECVID 2010. In: Proceedings of TRECVID Workshop (2010)
11. Gehler, P., Nowozin, S.: On feature combination for multiclass object classification. In: Proceedings of International Conference Computer Vision, ICCV 2009 (2009)
12. Cortes, C., Mohri, M., Rostamizadeh, A.: L_2 regularization for learning kernels. In: Proceedings of Uncertainty Artitical Intelligence, UAI 2009 (2009)
13. Erp, M.V., Vuurpijl, L.G., Schomaker, L.: An overview and comparison of voting methods for pattern recognition. In: Proceedings of the 8th International Workshop on Frontiers in Handwriting Recognition, IWFHR-8 (2002)
14. Brefeld, U., Gaertner, T., Scheffer, T., Wrobel, S.: Efficient co-regularized least squares regression. In: Proceedings of the 23rd International Conference of Machine Learning, ICML 2006 (2006)
15. Ayache, S., Quénot, G., Gensel, J.: Classifier Fusion for SVM-Based Multimedia Semantic Indexing. In: Amati, G., Carpineto, C., Romano, G. (eds.) ECIR 2007. LNCS, vol. 4425, pp. 494–504. Springer, Heidelberg (2007)
16. van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluation of color descriptors for object and scene recognition. In: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008 (2008)
17. Chen, M.Y., Hauptmann, A.: MoSIFT: Recognition human actions in surveillance videos. Technological report, CMU-CS-09-161, Carnegie Mellon University (2009)
18. Laptev, I., Lindeberg, T.: Space-time interest points. In: Proceedings of International Conference Computer Vision, ICCV 2003 (2003)

19. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Computer Vision and Pattern Recognition 2006, CVPR 2006 (2006)
20. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (IJCV 2004) 60(2), 91–100 (2004)
21. Chang, C.-C., Lin, C.-J.: LIBSVM: a library for support vector machines (2001)
22. Vedaldi, A., Fulkerson, B.: VLFeat: An Open and Portable Library of Computer Vision Algorithms (2008)
23. Bernhard, S., Burges, C.J.C., Smola, A.J.: Advances in kernel methods: Support Vector Learning. MIT Press, Cambridge (1999)

Multi-modal Solution for Unconstrained News Story Retrieval

Ehsan Younessian and Deepu Rajan

School of Computer Engineering,
Nanyang Technological University, Singapore
{ehsa0001,asdrajan}@ntu.edu.sg

Abstract. We propose a multi-modal approach to retrieve associated news stories sharing the same main topic. In the textual domain, we utilize Automatic Speech Recognition (ASR) and refined Optical Character Recognition (OCR) transcripts while in the visual domain we employ a Near Duplicate Keyframe detection method to identify stories with common visual clues. In addition, we adopt another visual representation namely semantic signature, indicating pre-defined semantic concepts included in the news story, to improve the discriminativness of visual modality. We propose a query-class weighting scheme to integrate the retrieval outcomes gained from visual modalities. Experimental results show the distinguishing power of the enhanced representation in individual modalities and the superiority of our fusion approach performance compared to existing strategies.

Keywords: Semantic signature, Refined Optical Character Recognition, News video retrieval.

1 Introduction

Associated news stories refer to news stories addressing the same main topic. Examples of associated news stories are shown in Figure 1. For instance, in Figure 1(a), there are three news stories from ABC, CCTV and CNN channels discussing the same topic of "Bush press conference". The objective of this research is to retrieve associated news stories from different channels in daily broadcast news videos through the multi-modal approach. This task can be seen as a prior stage for other tasks like event-based information organization, topic detection and tracking (TDT), news story summarization and etc.

The broadcast news stories include enriched auditory, textual and visual cues which can be utilized for the news story retrieval. For example, in *reader*, which is a type of news article read without accompanying video or sound, the spoken words carry the major part of semantics. Accordingly, applying ASR and retrieving spoken words would be essential for the story retrieval task. In addition to ASR transcript and OCR transcript which generally can be extracted from textual regions, visual elements play a critical role especially since humans receive much of their information of the world through their sense of vision. However,

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 186–195, 2012.

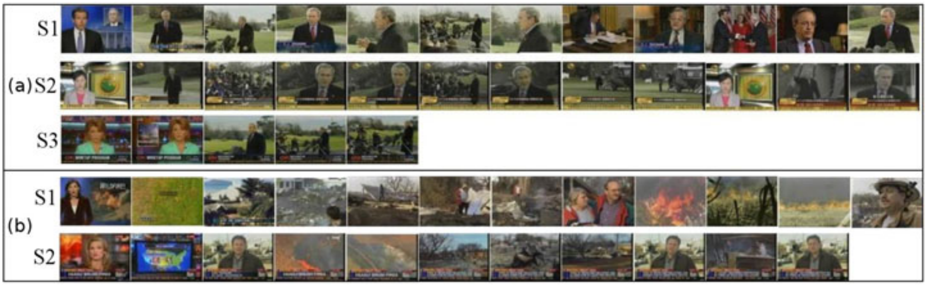


Fig. 1. Associated news story categories

finding the visual similarity of associated news stories often can be a challenging task.

To clarify this issue we visually categorize the associated news story into two major groups. (i) Associated news stories covering the same news object in usually the same venue and within the same context. They may come from the same or different video footages as shown in Figure 1(a). Finding mutual visual clues across associated news stories from this category can get problematic due to the high degree of variation of camera angles, camera lens settings and etc. (ii) Associated news stories addressing an event which is not visually related to specific objects or occurs at a specific venue like news story addressing “Katrina storm” or “Fire in Oklahoma” as shown in Figure 1(b). Visually, the storm or fire will be similar irrespective of where it happens. Modalities like textual information or high-level visual annotations could possibly be more informative and discriminative than routine visual representations like local signatures in these cases.

The rest of this paper is as follows. In Section 2 we explain related work. In Section 3 we explain our proposed approach to measure enhanced textual and visual similarities across news stories. In Section 4 we evaluation the retrieval performance through different uni- and multi-modal frameworks.

2 Related Work

During last years multi-modal fusion has absorbed much attention due to the benefit it provides for various multimedia analysis tasks. The integration of multiple media features is referred to as the early fusion while the integration of the intermediate decisions is referred to as the late fusion [1]. In the late fusion framework, the textual and visual units provide the local decisions which later combined through a decision fusion unit to obtain the final decision. The late fusion strategies can be categorized into two major groups of (i) Rule-based (e.g MIN, MAX, Ranked list, query (in)dependent weighting fusion and etc), (ii) Classification-based fusion (e.g. SVM , Bayesian inference and etc) as comprehensively discussed in [1]. In this paper, we focus on query-class weighting solutions to fuse different visual modalities.

Query-dependent or query-class weighting can be considered an evolution of query independent weighting since it tackles many query independent weighting failures. The focal point of this approach is that given training references and an appropriate set of training queries, query clusters (i.e. query-classes) can be found such that queries within each cluster share some similar properties which differentiate them from other queries in the collection (where properties may be artifacts such as semantic similarity, performance similarity, distance and etc). By partitioning a set of training queries into discrete query classes, it is possible to optimize for each query-class an instance of the weighting matrix RC, such that each class should have a different set of weights for local decisions. The query-class concepts was introduced by Yan et al. [11] for content-based video retrieval where four classes of Named person, Named object, General object, and Scene were defined based on which they assign different weights to different low-level classifiers. Later in [10] they developed probabilistic latent query analysis (pLQA) which is able to discover latent query classes automatically without using prior human knowledge, to assign one query to a mixture of query classes, and to determine the number of query classes under a model selection principle. Xie et al. [9] propose a query-dependent fusion strategy that dynamically builds a class using training queries that are the closest to the testing query, based on light-weight query features defined on the semantic analysis results on the query text.

Using late fusion framework in the textual domain, authors in [2] use an advanced pre-processing approach for each individual frame to provide qualified inputs to the OCR engine, the output of which is coupled with the ASR transcript for the search task. This approach is computationally expensive due to its inefficient text box verification step. In [3] authors compare n-gram analysis and dictionary look up techniques to correct OCR error for the text-based video retrieval. The former generates a new set of n-gram strings to match the unedited OCR outputs. These n-gram strings include strings with an edit distance of 1 character and all substrings with at least 3 characters. The second method uses the global dictionary to correct spelling errors. In [13] authors propose the keywords expansion method to compensate noisy ASR and OCR transcripts for TV commercial classification task. They also used the encyclopedia and English dictionary to correct misspelled terms and come up with keywords. Note that unlike TV commercial where background music degrades the ASR transcript quality, in the news domain ASR transcripts usually possesses higher accuracy. However, in the news domain the quality of OCR transcript varies depending on the video resolution, the font size and the complexity of the background of the text region. To refine the OCR output, we also propose the local dictionary concept using determined ASR transcripts.

3 Enhanced Multi-modal Content Similarity

In our proposed multi-modal solution, in the textual domain we propose a novel early fusion strategy applied in the feature-level to suppress the OCR errors and boost up the retrieval performance by generating enhanced textual features using both ASR and refined OCR transcripts. In the visual domain, we utilize the local-feature-based visual similarity and the semantic similarity. In the former, we use keypoint matching scheme to determine the visual similarity between keyframes of stories, while in the latter we calculate the visual similarity between news stories based on the pre-defined visual concepts they include. Then we fuse those two visual modalities through a query-class weighting scheme to address failings of the former and shorten the existing semantic gap. At the end, we combine the enhanced visual and textual similarities through different fusion strategies to improve the retrieval performance.

3.1 Enhanced Textual Content Similarity

In the textual domain, we aim to utilize the spoken words transcript obtained by an ASR engine and OCR transcripts together. Unlike documents, overlaid texts in news videos possess a wide range of sizes, fonts, colors, and mostly complex, dynamic, or/and transparent backgrounds. These complicating factors cause the OCR output to be highly erroneous. For instance, the word accuracy for detected text was estimated to be only 27% for VOCR used in [3]. Hence, first we aim to correct the OCR errors to reach the enhanced textual representation.

Optical Character Recognition. In news videos, overlaid text is mostly located at the bottom of the screen. We create a profile for each of 7 channels in the dataset that specifies the spatial information of overlaid components in the screen. This area is highlighted as a box in Figure 2(I). For the overlaid text extraction, first we spatially filter the gray-scale keyframe with respect to the position of overlaid text box in the broadcasting channel as shown in Figure 2(I)(a). The gray-scale text box is binarized using Otsu method (Figure 3(b)) and input to the OCR engine [5]. The output of the OCR is a series of highly erroneous terms for each keyframe, if any (Figure 2(I)(c)).

OCR Output Recovery. We use the spell-checker engine, called ASPELL [4], which generates a group of candidate words for each incorrectly spelled OCR output. The candidate words are ranked in the ascending order of their similarity to the raw OCR output. This similarity score is determined by considering typo analysis. In Figure 2(I)(d), three words “FEmAL”, ”wnNEssES ” ,and ”IRAOI” are input to ASPELL and it generates 28, 28, and 8 candidates, respectively. In addition, the correctness of some terms (i.e. GUARDS) can be confirmed by the spell-checker. Eventually, there is a portion of incorrectly spelled terms in OCR output which can not be recovered by the spell checking procedure. For instance, there is no “ELECTROCUTION” among candidates words generated for term “ELEMRUTION” in the above example.

In order to pick the right word among the generated candidates, we utilize the fact that both spoken words and overlaid text address the identical story content and most likely share some common important words. Hence, we build a local dictionary for each story using ASR transcript (Figure 2(II)). The local

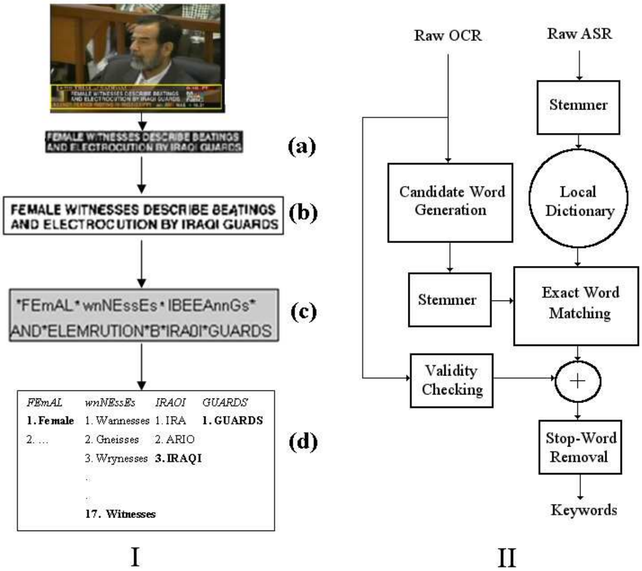

(a)
FEMALE WITNESSES DESCRIBE BEATINGS
AND ELECTROCUTION BY IRAQI GUARDS
(b)
*FEmAL*wnNEssEs*IBEEAnnGs*
AND*ELEMRUTION*B*IRA0I*GUARDS
(c)
FEmAL wnNEssEs IRAOI GUARDS
1. Female 1. Wannesses 1. IRA 1. GUARDS
2. … 2. Gneisses 2. ARIO
3. Wrynesses 3. IRAQI
17. Witnesses
(d)
I
Raw OCR
Raw ASR
Stemmer
Candidate Word Generation
Local Dictionary
Stemmer
Exact Word Matching
Validity Checking
+
Stop-Word Removal
Keywords
II

Fig. 2. Overview of the OCR (I) pre- and (II) post-processing

dictionary is basically the raw ASR transcript with the words converted to their roots by the stemmer. In the used dataset the local dictionary has around 50 words in average for each story. The local dictionary is checked for the words that are output by the spell checker and converted to the root form by a stemmer, and a set of words common to the local dictionary and the stemmer output is created. Note that if we use the global dictionary, each of OCR terms should be converted to the closest word in the global dictionary. In Figure 2(I), "Wannesses" and "IRA" are the closest words to "wnNEssEs" and "IRAOI" respectively according to the global dictionary and they are not correct. Those words that have been correctly recognized by OCR, e.g., GUARD in the above example, and may not necessarily exist in the ASR transcript, are also detected by validity checking process as indicated in Figure 2(II). The compilation of these valid words and common words are passed through the stop word removal filter and the result is a set of keywords. Next, we determine the enhanced *tfidf* representation as

$$tfidf(i,j) = tf_A(i,j)/df_A(i) + tf_O(i,j)/df_O(i), \qquad (1)$$

where $tf_A(i,j)$ and $tf_O(i,j)$, called the term frequency, are the number of times $term_i$ appears in the j^{th} ASR and OCR transcripts, respectively, and $df_A(i)$ and $df_O(i)$ called document frequency, are obtained by dividing the number of ASR/OCR transcripts containing the $term_i$ by the number of all documents in ASR/OCR lexicon. We calculate the enhanced textual similarity across stories using cosine similarity between their enhanced *tfidf* representations.

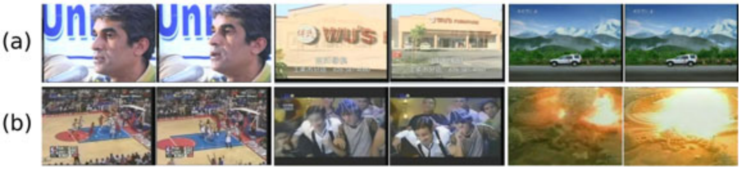


Fig. 3. Sample scenes with (a) high and (b) low *t-score*

3.2 Enhanced Visual Content Similarity

We simply detect Near Duplicate Keyframes(NDK) across two news stories keyframes using Bag-of-Words representation of SIFT descriptors followed by geometric verification [12] and cosine similarity as the similarity measure. Next, we utilize normalized set difference to measure between stories (S_i and S_j) similarity as

$$Sim\_KF(S_i, S_j) = \frac{|KF_i \cap KF_j|}{2}.(\frac{1}{KF_i} + \frac{1}{KF_j}), \qquad (2)$$

where KF_i refers to the set of keyframes contained in the S_i. Although the local-feature-based similarity is effective and robust to limited degree of certain variations, but it still suffers from significant object/camera movements occurred in scenes with dynamic concepts and also the well-known semantic gap as shown in Figure 3(b). On the other hand, they perform properly in scenes with mostly static concepts like mountains, speakers, building and etc as shown in Figure 3(a). This observation motivates us to study relation between concepts represented in a scene and the capability of the local-feature-based algorithm to catch the visual similarity across scenes. We use TRECVID 2006 dataset including around 160 hours news video from 7 different channels. Every shot is semantically indexed using 374 concepts by [7] and presented by 374-dimensional vector, called *SemSig*, each element of which shows the probability of the existence of the corresponding concept in the shot. Overall, there are around 21k shots each of which includes several keyframes.

Next, we determine matching keypoints between keyframes within each shot using the method mentioned earlier. Accordingly, we categorize shots into two categories of the detectable and non-detectable groups if the number of matching keypoints between their keyframes exceeds a specific threshold. Then we employ the *t-test* [8] between these two categories according to the included concepts to see what are the concepts existence of which lead to general failure of NDK detection within the shots. In Figure 4, the *t-score* for all concepts are shown. Concepts like sitting, US flag and address or speech have high *t-score* which implies that NDK detection algorithm is generally capable of finding scenes having these static concepts. On the other hand, concepts like shooting, dancing, ruins and natural disaster have low *t-score* which means there is a general difficulty to detect NDK including these concepts.

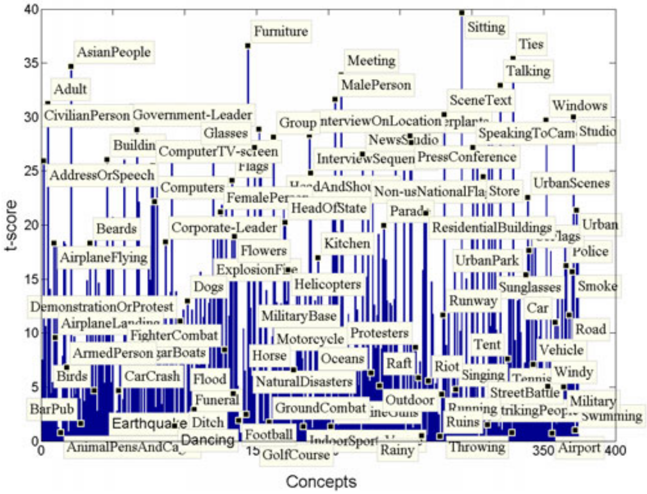


Fig. 4. The determined *t-score* for 374 concepts

Accordingly we compute Detectability Score (DS) for each concept as

$$DS(i) = 1 + 1/(1 + t\text{-}score(i)), i = 1, 2, .., 374. \tag{3}$$

In practice, semantic signature using predefined concepts are not discriminative enough to help us out to retrieve failure cases due to the general incapability of the semantic indexing to describe different scenes uniquely and limited number of concepts, detector of which is determined. However, we can improve the retrieval performance by fusing these two sources of visual knowledge which are local-feature-based and semantic signatures similarity. To this end, for each story we determine the semantic signature, $Sem(S_i)$, simply as the summation of $tfidf$ representations of its shots $SemSig$. Then we calculate between-story semantic similarity as

$$Sim\_Sem(S_i, S_j) = (DS \otimes Sem(S_i)).(DS \otimes Sem(S_j))^t, \tag{4}$$

where $\otimes$ denotes the element-wise production. Hence, we assign higher weights on concepts with the lower *t-score*. Next, we can determine the final visual similarity for each story pair based on the trained SVM classifier using determined Sim_KF and Sim_Sem as two input features.

4 Experimental Results

In this Section we evaluate different uni- and multi-modal approaches for associated news story retrieval. We use news videos from the TRECVID 2006 corpus from 7 channels addressing world news occurred in December 2005. The length of stories changes between 30 seconds and 5 minutes. We use ASR transcripts

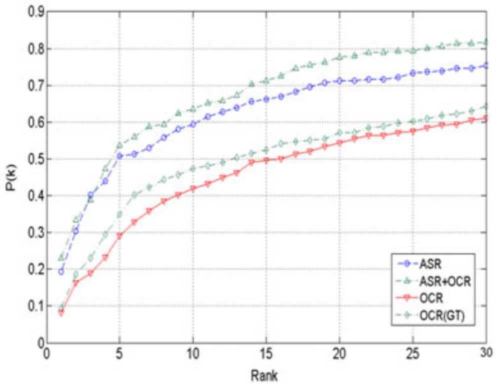


Fig. 5. Top-K retrieval result using the textual modality

provided by [6]. We manually segment the news stories as a group of keyframes and label them based on their main topic. The dataset contains 830 news stories out of which 296 pairs of associated news stories are labeled. More than half of associated news stories belong to the second category of associated news stories explained earlier in Section 1. We consider each associated news story as the query and measure the similarity between the query and all reference stories using different uni- and multi-modal representations and different fusion strategies and then rank them, accordingly.

The retrieval performance is quantified by the probability of retrieving the associated news stories in the top-k position of the ranked list given as $P(K) = Z_c/Z$, where Z_c is the number of queries that rank their associated news stories within the top-k position and Z is the total number of queries (296). In Figure 5, we show *top-k* news story retrieval results using ASR, refined OCR, OCR ground truth, OCR(GT), and enhanced textual representation. It should be mentioned that through OCR output recovery explained in Section 3.1, we could improve OCR accuracy from 18% to 42%. In Table 1, we compare our OCR post-processing method with others. In this paper we determine the local dictionary based on this assumption that story boundaries are given.

Table 1. Precision, Recall, and F-measure for different OCR post-processing methods

Methods	Raw OCR	our method	n-gram	Global Dic.
Precision(%)	18.40	**42.25**	20.35	21.78
Recall(%)	20.67	**48.14**	24.44	22.54
F-measure	19.46	**44.99**	22.21	21.15

In Figure 6, we compare retrieval results using different visual similarities and their fusion as discussed in Section 3.2. As explained earlier, using semantic similarity solely results in a poor retrieval performance but it is still above random retrieval plot. Using keyframe set similarity ends up with better retrieval

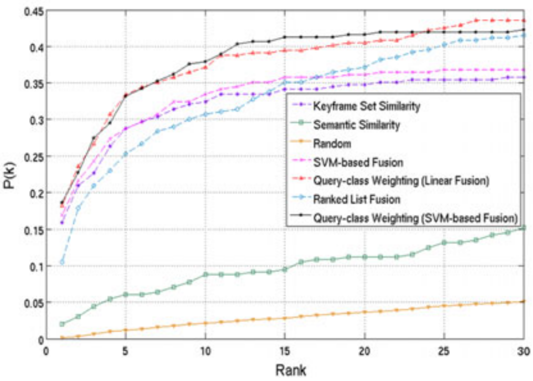


Fig. 6. Top-K retrieval result using the visual modality

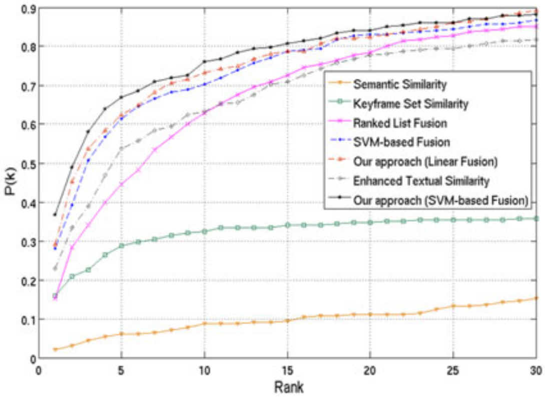


Fig. 7. Top-K retrieval result based on different multi-modal late fusion strategies

result. This relatively low retrieval accuracy comes from this fact that significant portion of the associated news stories belong to the second category mentioned in Section 1, where this visual representation is not discriminative enough. We also compare different fusion strategies to integrate these two visual similarities retrieval results such as *ranked list* and *SVM-based* fusion approaches. Through the *ranked list* fusion we generate the ranked list for each query using each visual modality then the final rank for each reference data is calculated as the *min* of its calculated ranks. In the *SVM-based* fusion approach, we employ leave-one-out framework and train the RBF kernel using training data including pairwise Sim_KF and Sim_Sem scores. Next, for the query of interest we retrieve similar reference news stories using trained RBF kernel with determined parameters. As shown in Figure 6, the best results are obtained by our query-class weighting scheme followed by the SVM-based/linear fusion.

In Figure 7, we compare retrieval results using different multi-modal fusion strategies as discussed earlier. The best result is obtained by the SVM-based fusion of modified semantic similarity explained in Section 3.2, keyframe set similarity and enhanced textual similarity. Note that even linear fusion of mentioned similarity scores outperforms ranked list fusion and SVM-based fusion using original semantic similarity, keyframe set similarity and enhanced textual similarity. This superiority explains the key role of our proposed Detectability Score (as determined in equation 3) to refine the semantic similarity.

5 Conclusion

In this paper, we determine enhanced textual and visual similarity between news stories using novel early and late fusion strategies, respectively. The experimental results confirm the effectiveness of our uni- and multi-modal associated news story retrieval approaches.

References

1. Atrey, P.K., Hossain, M.A., El Saddik, A., Kankanhalli, M.S.: Multimodal fusion for multimedia analysis: a survey. Multimedia Systems 16(6), 345–379 (2010)
2. Das, D., Chen, D., Hauptmann, A.G.: Improving multimedia retrieval with a video ocr. In: Gevers, T., Jain, R.C., Santini, S. (eds.) Society of Photo-Optical Instrumentation Engineers (SIPE) Conference, vol. 6820, p. 68200B. SPIE (January 2008)
3. Hauptmann, A.G., Jin, R., Ng, T.D.: Multi-modal information retrieval from broadcast video using ocr and speech recognition. In: JCDL 2002, pp. 160–161. ACM (July 2002)
4. `http://aspell.net` (last visited August 2010)
5. `http://jocr.sourceforge.net` (last visited August 2010)
6. `http://www-nlpir.nist.gov/projects/tv2006/tv2006.html` (last visited August 2010)
7. Jiang, Y.G., Yang, J., Ngo, C.W., Hauptmann, A.G.: Representations of Keypoint-Based Semantic Concept Detection: A Comprehensive Study. IEEE Transactions on Multimedia 12(1), 42–53 (2009)
8. Rice, J.A.: Mathematical Statistic and Data Analysis, 3rd edn. Duxbury, Belmont (2007)
9. Xie, L., Natsev, A., Testic, J.: Dynamic multimodal fusion in video search. In: IEEE International Conference on Multimedia and Expo (ICME), pp. 1499–1502 (July 2007)
10. Yan, R., Hauptmann, A.G.: Probabilistic latent query analysis for combining multiple retrieval sources. In: SIGIR 2006, pp. 324–331. ACM (August 2006)
11. Yan, R., Yang, J., Hauptmann, A.G.: Learning query-class dependent weights in automatic video retrieval. In: ACM MM 2004, pp. 548–555. ACM (2004)
12. Zhao, W.-L., Ngo, C.-W.: Scale-rotation invariant pattern entropy for keypoint-based near-duplicate detection. IEEE Transactions on Image Processing 18, 412–423 (2009)
13. Zheng, Y., Duan, L., Tian, Q., Jin, J.: Tv commercial classification by using multi-modal textual information. In: 2006 IEEE International Conference on Multimedia and Expo, pp. 497–500 (July 2006)

Workflow Activity Monitoring Using Dynamics of Pair-Wise Qualitative Spatial Relations

Ardhendu Behera, Anthony G. Cohn, and David C. Hogg

School of Computing, University of Leeds,
Woodhouse Lane, Leeds, LS6 4JZ, UK
{A.Behera,A.G.Cohn,D.C.Hogg}@leeds.ac.uk

Abstract. We present a method for real-time monitoring of workflows in a constrained environment. The monitoring system should not only be able to recognise the current step but also provide instructions about the possible next steps in an ongoing workflow. In this paper, we address this issue by using a robust approach (HMM-pLSA) which relies on a Hidden Markov Model (HMM) and generative model such as probabilistic Latent Semantic Analysis (pLSA). The proposed method exploits the dynamics of the qualitative spatial relation between pairs of objects involved in a workflow. The novel view-invariant relational feature is based on distance and its rate of change in 3D space. The multiple pair-wise relational features are represented in a multi-dimensional relational state space using an HMM. The workflow monitoring task is inferred from the relational state space using pLSA on datasets, which consist of workflow activities such as 'hammering nails' and 'driving screws'. The proposed approach is evaluated for both 'off-line' (complete observation) and 'on-line' (partial observation). The evaluation of the novel approach justifies the robustness of the technique in overcoming issues of noise evolving from object tracking and occlusions.

Keywords: Qualitative Spatio-temporal Relations, Workers Instructions, Activity Recognition, Hidden Markov Model (HMM), Probabilistic Latent Scemantic Analysis (pLSA).

1 Introduction

A *workflow* is a temporally ordered set of procedural steps for accomplishing a task in which people and tools are involved in each step of the process. In an industrial environment, the aim of workflow monitoring is to assist operators unfamiliar with a workflow by providing *on-the-fly* instructions from an automatic system. This enables continual interaction between operators and the system while performing a workflow. In an on-going workflow, the proposed monitoring system should be able to anticipate the next possible tasks and recognize the deviations from the correct workflows which may lead to quality and/or health and safety problems. In our case, the operators' instructions will be provided via augmented reality, video clips and/or text using a see-through Head Mounted Display (HMD)[27]. Therefore, the monitoring system requires a general ability

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 196–209, 2012.

to learn, analyze and model workflow patterns. This associates to a problem of activity recognition.

The more general problem of activity recognition is widely studied within Computer Vision. Much of this work has focused on the development of probabilistic models over object configuration spaces and estimated from training data. Examples include Hidden Markov Models [5,25,17,6], stochastic context free grammars (SCFG) [15,1], echo state networks (ESN) [22], propagation networks (P-nets) [21], Past-Now-Future networks (PNF-networks) [19] and Bayesian networks [12,11]. Very often the configuration space is confined to the location and motion of objects within a scene based frame of reference [14,9]. Most of these models consider only the behaviour of an individual object, such as location and speed in the image plane. Though an activity recognition using a trajectories-based model is powerful, the model complexity increases quadratically with an increase in interactions between multiple objects participating in a task. Furthermore, the tracking algorithm often fails due to occlusion and inability to distinguish between foreground and background.

In this paper, we explore the activity recognition problem in the context of workflow by using qualitative spatio-temporal pair-wise relations between human body parts, tools and objects in a workspace. These relations are established using a relational feature vector representing distance and the rate of change of distance between pairs of objects in 3D space. The motivation for using relational features is to enable the model to follow the ongoing workflow, even though an object is missing due to occlusion or scene complexity. This is possible by considering the spatio-temporal configurations of other observed objects. For example, during the task of hammering, if the individual's hand moves towards the nail box and back to the work bench, it is most likely that the he/she has picked up a nail, by considering the spatio-temporal configurations between nail box and hand during the 'retrieve-nail' subtask. Similarly, if the participant's hand moves towards the screw box, the system should then assert a violation of workflow since the ongoing task is *hammering of nails*.

In the present study, we consider all possible pair-wise relations among objects in a given workspace. These relations are then represented in a relational state space. We propose a novel method to model workflow from this relational state space by using probabilistic *Latent Semantic Analysis* (pLSA) [10]. We evaluate our proposed technique with the workflows of *hammering nails* and *driving screws*. In this model, each workflow sequence consists of multiple sub-sequences of *primitive events*.

2 Related Work

Activity recognition in the context of workflow is still an active field of research. In this section, a brief description of related work on workflow monitoring and computer vision-based activity recognitions most associated to the context of workflow, is presented.

Veres *et al.* [22] proposed a method for monitoring workflows in a car assembly line. The method uses a global motion descriptor by sampling an input image sequences by a fixed overlapping spatial grids over whole image. Each grid is represented by local motion descriptor based on pixel intensity. The global motion descriptor for an image at a given timestamp is the concatenation of these local motion descriptors. Eco state networks (ESN) [13] are used as a time series predictor for workflow monitoring. Pody *et al.* [18] uses a hierarchical-HMM with observation of 3D optical flow-features for monitoring a hospital's operating rooms. The 3D flow-features are extracted by quantisig the optical flow of pixels inside a spatio-temporal cell of fixed volume. The top-level topology of the hierarchical-HMM is temporally constrained and the bottom level sub-HMM is trained independently with labelled sub-sequences. Pinhanez and Bobick introduced the Past-Now-Future networks (PNF-networks) [19] using Allen's temporal relations [2] to express parallelism and mutual exclusion between different sub-events. In order to gain a detection of actions and sub-actions, Allen's interval algebra network is mapped into a simpler three-valued PNF-network representing temporal ordering constrained between the start and end timing of event instances. Shi *et al.* [21] presented propagation networks (P-nets) to model and detect primitive actions from videos by tracking individual objects. P-nets explicitly model parallel streams of events and are used for classification. The detailed topology is handcrafted and trained from partially annotated data. Moore and Essa [15] use stochastic context-free grammars (SCFG) to recognize separable multi-tasked activities from a video illustrating a card game. All relations between the tracked events are described using manually-defined production rules.

In another context, event recognition in meetings using layered-HMMs is proposed by Oliver *et al.* [17]. The HMMs operate in parallel at different levels of data granularity which allow event classification using multi-modal features. An integrated system for modelling and detecting both high- and low-level behaviours is demonstrated by Nguyen *et al.* [16]. The system uses the trajectories of occupants in a room consisting of pre-defined multiple cells in a given zone. The goal is to recognize behaviours that differ in the occupied cells and in the sequence of their occupation.

In most of the above-mentioned models: 1) object trajectories in the image plane are used as a feature descriptor. However, tracking algorithms often fail to detect and track objects efficiently due to variations in workspace settings, occlusions as well as dynamic or cluttered background. We partially address this issue by using spatio-temporal relational configurations of the objects involved. 2) The models take into consideration a limited number of objects at a given time. The complexity of the learning algorithm increases with the involvement of more objects or interactions, thereby hindering 'real-time' monitoring. This is overcome by the proposed probabilistic *Latent Semantic Analysis* (pLSA). 3) Additionally, we employ view-invariant relational feature for our model whereas view-dependent features are used in most models [22,18,17].

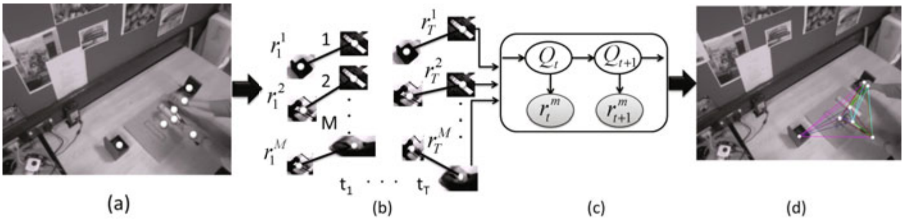


Fig. 1. Workflow monitoring model overview: a) tracked objects in a workspace, b) pair-wise relational feature, c) state space representation of each pair-wise relations, and d) reflections of pair-wise relations (state space) in the workspace

3 Qualitative Relations to Workflow Patterns

The proposed model for workflow activity monitoring comprises of four steps. The systematic procedure for this is shown in Fig. 1. In the first step, the relevant objects in a given workspace are tracked. The tracking system provides instantaneous 3D positions of objects of interest at each time frame (Fig. 1a). Secondly, a view-invariant relational feature vector for each pair of objects for each time point, is computed (Fig. 1b). In the third step, these relations are quantised into a finite number of states using an HMM (Fig. 1c and Fig. 1d). In the final step, the framework uses a generative process of pLSA for monitoring and recovering workflow activity from the relational configuration of quantised pair-wise relations as shown in Fig. 1d.

3.1 Feature for Qualitative Spatial Relations

Our model is based on the joint motion of a collection of N *key objects* relevant to the task at hand. Let $(\mathbf{x}_t^1, \mathbf{x}_t^2, ..., \mathbf{x}_t^N)$ be the respective 3D positions of these objects at time t, where $\mathbf{x}_t^i = (x, y, z)_t^i$. The joint motion is described in a view-invariant fashion as the set of spatial and kinematic relations between every pair of *key objects*. At each time step, the relation between a pair of objects i and j is represented by a real valued vector composed of the separation and the first derivative of separation with respect time *i.e.* $\mathbf{r}_t^{i,j} = (d_t^{i,j}, \dot{d}_t^{i,j}) \in \Re^2$, where $d_t^{i,j} = \|\mathbf{x}_t^i - \mathbf{x}_t^j\|$ for $\forall i < j$. For convenience, we order the set of pair-wise relations $\{\mathbf{r}_t^{i,j}, i < j\}$ and express as $R = [\mathbf{r}_t^m]_{M \times T \times 2}$, where $m = 1 \ldots M$ and $M = N(N-1)/2$ and T is the number of time steps. We now discretise the pair-wise feature vectors using an HMM to capture the temporal dependencies, and after discretisation it will be represented by corresponding HMM states $S = [s_t^m]_{M \times T}$.

3.2 State Space Representation of Spatial Relations

The state space $S = [s_t^m]$ representation of the corresponding relational feature set $R = [\mathbf{r}_t^m]$ is carried out using an HMM (Fig. 1c). This is defined as a quintuple

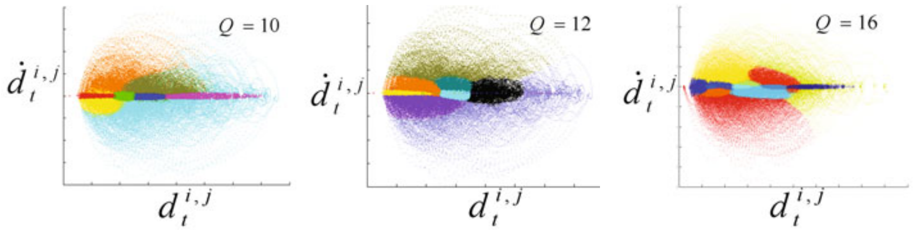


Fig. 2. State space (Viterbi path) representation of pair-wise relations using an HMM of 10, 12 and 16 states respectively (from left). Each colour represents a particular state in the HMM.

(Q, R, π, A, B), where Q is a finite non-empty set of 'relational' states, R is the input relational feature, $\pi = \{\pi_q\}$ is the starting probability for an element $q \in Q$, $A = \{a_{q,q'}\}$ are the state transition probabilities from the state q to state q' and $B = \{b_q(\mathbf{r}) = N(\mathbf{r}, \mu_q, \sum_q)\}$ is the output function, which is represented as a Gaussian density with mean vector μ_q and covariance matrix $\sum_q$ for the state q emitting feature $\mathbf{r}$. The optimal parameter $\lambda^* = (\pi^*, A^*, B^*)$ of the HMM is estimated using Baum-Welch forward-backward algorithm [3] from a training dataset consisting of $\mathcal{W}$ workflow sequences, where each workflow sequence is represented by M parallel sequences of pair-wise relational features:

$$\lambda^* = \underset{\lambda}{\operatorname{argmax}} \prod_{w=1}^{\mathcal{W}} \prod_{m=1}^{M} P(\mathbf{r}^{m,w}|\lambda)$$
$$P(\mathbf{r}^{m,w}|\lambda) = \sum_{\text{all } Q} P(\mathbf{r}^{m,w}|Q, \lambda)P(Q|\lambda) = \sum_{\text{all } Q} \prod_{t=2}^{T} \pi_q b_{q_t}(\mathbf{r}_t^{m,w}) a_{q_{t-1},q_t} \tag{1}$$

where $\mathbf{r}^{m,w}$ denotes the m^{th} series of pair-wise relational features from the w^{th} workflow sequence and consists of T time steps. The Viterbi algorithm [24] is used to find the most likely hidden states sequence from a given observed sequence of relational feature using the optimal parameter λ^*. Fig. 2 demonstrates the pair-wise relations with the varying number of states Q in the HMM.

3.3 pLSA for Modeling Spatial Relations

A workflow sequence can be decomposed into multiple sub-sequences. The decomposition granularity often depends on type of the workflows and the methods used for its realisation. In the case of 'hammering nail' and 'driving screws' workflows, we use a set of *primitive events* (Table 1) which have been manually annotated. The generative model of pLSA is used for this multi-class classification problem instead of discriminative classification techniques such as support vector machine *i.e.* SVM.

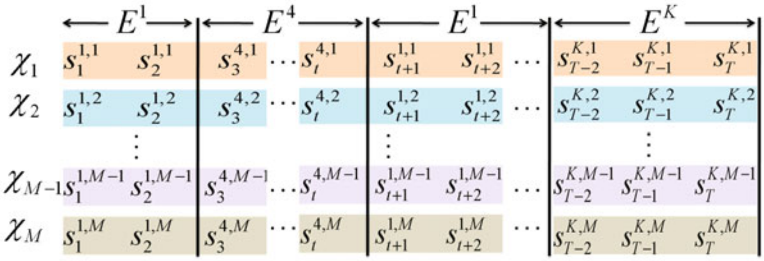

E^1
E^4
E^1
E^K
χ_1 $s_1^{1,1}$ $s_2^{1,1}$ $s_3^{4,1} \cdots s_t^{4,1}$ $s_{t+1}^{1,1}$ $s_{t+2}^{1,1} \cdots$ $s_{T-2}^{K,1}$ $s_{T-1}^{K,1}$ $s_T^{K,1}$
χ_2 $s_1^{1,2}$ $s_2^{1,2}$ $s_3^{4,2} \cdots s_t^{4,2}$ $s_{t+1}^{1,2}$ $s_{t+2}^{1,2} \cdots$ $s_{T-2}^{K,2}$ $s_{T-1}^{K,2}$ $s_T^{K,2}$
χ_{M-1} $s_1^{1,M-1}$ $s_2^{1,M-1}$ $s_3^{4,M-1} \cdots s_t^{4,M-1}$ $s_{t+1}^{1,M-1}$ $s_{t+2}^{1,M-1} \cdots$ $s_{T-2}^{K,M-1}$ $s_{T-1}^{K,M-1}$ $s_T^{K,M-1}$
χ_M $s_1^{1,M}$ $s_2^{1,M}$ $s_3^{4,M} \cdots s_t^{4,M}$ $s_{t+1}^{1,M}$ $s_{t+2}^{1,M} \cdots$ $s_{T-2}^{K,M}$ $s_{T-1}^{K,M}$ $s_T^{K,M}$

Fig. 3. Partition of a workflow sequence into *primitive events* E^p, pair-wise relations χ_m and spatial relations $s_t^{p,m}$. In [4], these are equivalent to 'corpus', 'document' and 'word' respectively.

Probabilistic latent space models [10,4] were initially proposed to automatically discover the recurrent themes or topics from a corpus of text documents. They are used to analyze topic distributions of documents and word distributions in a topic. The model is estimated from the co-occurrence of words and documents. In our work, we extract this by dividing a workfow into subsequences of *primitive events* ($E^{p=1\ldots K}$ in Fig. 3). Each pair-wise relation in a *primitive event* is represented by 'document' ($\chi_{m=1\ldots M}$ in Fig. 3). Every quantised relation ($s_t^{p,m}$) in a pair-wise relation is characterised by 'word'. In our framework, each *primitive event* symbolises a corpus and is modelled separately using a pLSA, namely a 'corpus-model'.

The spatio-relational structure of key objects changes over time with the progress of a *primitive event*. In our 'corpus-model', those underneath relational structures for that *primitive event*, are captured by the distribution of latent variable, which is known as 'topics' in the pLSA models. The latent variable itself characterised by probability distribution over the relational states in each pair-wise relation belongs to the *primitive event*. Therefore, we use all the instances of a given *primitive event* from all training sequences to train the corresponding 'corpus-model'. During evaluation of an unseen workflow sequence, the model uses a sliding window of duration $\mathcal{T}$ and decides its association with the seen *primitive events* based on maximum posterior probability.

We begin with some notations for our 'corpus-model'. A workflow sequence is a collection of K *primitive events* represented by $E = \{E^{p=1\ldots K}\}$. A *primitive event* is a group of M parallel pair-wise relations indicated by $\chi = \{\chi_{m=1\ldots M}\}$. The m^{th} pair-wise relation in the p^{th} *primitive event* $\chi_m^p = \{s_{t=1\ldots\tau}^{p,m}\}$ is a sequence of τ quantised spatial relations, where $s_t^{p,m} \in Q$ (Fig. 3). In fact, in a given *primitive event*, all pair-wise relations will have the same number of quantised spatial relations *i.e.* the same τ for $\forall m$. The pLSA-model parameter for each *primitive event* is learned from the training examples. This is done by considering all instances of the same *primitive event* appearing in all the training sequences (Fig. 3). For convenience, from here onwards χ^p represents the collection of M pair-wise relations and the corresponding quantised spatial relations s^p for all instances of *primitive event* E^p appeared in the training sequences.

For each *primitive event* E^p, our aim is to find the joint distribution $P_p(\chi^p, s^p)$ between the pair-wise relations χ^p and spatial relations s^p belonging to the E^p ($p = 1 \ldots K$). This is done by using a latent variable model for general co-occurrence of χ^p and s^p which associates an unobserved class variable $z^p = \{z_1^p, z_2^p, \ldots, z_Z^p\}$ [10]. The model assumes the conditional independence of χ^p and s^p given a latent variable z^p. The graphical representation of our pLSA is shown in Fig. 4. The joint probability $P_p(\chi^p, s^p)$ can be expressed as:

$$P_p(\chi^p, s^p) = P_p(\chi^p)P_p(s^p|\chi^p) \tag{2}$$

$$\text{where, } P_p(s^p|\chi^p) = \sum_{k=1}^{Z} P_p(s^p|z_k^p)P_p(z_k^p|\chi^p) \tag{3}$$

The conditional probabilities $P_p(s^p|z_k^p)$ and $P_p(z_k^p|\chi^p)$ are learned using the EM algorithm [8] by maximizing the following log-likelihood function:

$$L_p = \sum_{\chi^p}\sum_{s^p} n(\chi^p, s^p)\log(P_p(\chi^p, s^p)) \tag{4}$$

where the E-step is shown as:

$$P_p(z^p|\chi^p, s^p) = \frac{P_p(s^p|z^p)P(z^p|\chi^p)}{\sum_{z^{p'}} P_p(s^p|z^{p'})P(z^{p'}|\chi^p)} \tag{5}$$

and the M-step is:

$$P_p(s^p|z^p) = \frac{\sum_{\chi^p} n(\chi^p, s^p)P_p(z^p|\chi^p, s^p)}{\sum_{r^p}\sum_{s^{p'}} n(\chi^p, s^{p'})P_p(z^p|\chi^p, s^{p'})} \tag{6}$$

$$P_p(z^p|\chi^p) = \frac{\sum_{s^p} n(\chi^p, s^p)P_p(z^p|\chi^p, s^p)}{n(\chi^p)} \tag{7}$$

where $n(\chi^p, s^p)$ is the number of co-occurrences of the spatial relation s^p and the pair-wise relations χ^p in the *primitive events* E^p. The proposed 'corpus-model' computes the joint distribution $P_p(\chi^p, s^p)$ for each E^p ($p = 1 \ldots K$) by considering the temporally segmented subsequences representing the corresponding *primitive events* in the training *dataset* of workflow sequences (Fig. 3). During recognition of an unknown workflow sequence, the co-occurrences matrix of $n(\hat{\chi}, \hat{s})$ is computed by using a sliding window of duration $\mathcal{T}$ over it. At each time step, the likelihood of co-occurrences matrix $n(\hat{\chi}, \hat{s})$ with respect to each *primitive event* E^p is computed using the joint-distribution $P_p(\chi^p, s^p)$ of E^p via Eqn. 4. The unknown sliding window at each time step is assigned a *primitive event* $e^* = \text{argmax}(L)$, where $L = \{L_1, L_2, \ldots, L_K\}$ is the measured likelihood from all *primitive events*.

3.4 Activity Monitoring

For workflow activity monitoring, the model is not only for the recognition of ongoing activity but also for advising the agent on the next possible tasks. In order

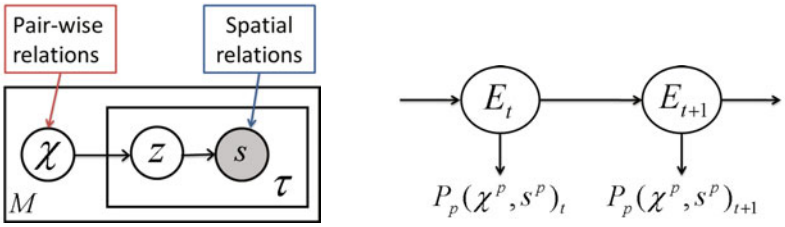


Fig. 4. The Generative pLSA model (left) and workflow monitoring-HMM (right)

to achieve this, a top-level workflow topology is required. Often, this top-level topology is provided manually for a well-defined structured workflow [21,15]. We achieve this by modelling event spaces with an HMM. The graphical structure is shown in Fig. 4. The monitoring-HMM consists of K hidden states denoting K *primitive events*. The observation likelihood for each hidden state E_t at time t is computed from the respective *primitive event's* likelihood via co-occurrence matrix $n(\chi^p, s^p)_t$ through a sliding window of duration $\mathcal{T}$.

$$P_p(n(\chi^p, s^p)_t|E_t) = \prod_{\chi^p}\prod_{s^p} P_p(\chi^p, s^p)^{n(\chi^p,s^p)_t} \tag{8}$$

We are interested in the transition probabilities from state E_t to state E_{t+1}; these are estimated via the Baum-Welch forward-backward algorithm [3] from the training sequences.

Our model can also be readily used for abnormal behaviour detection while monitoring a workflow. This can be achieved via examining the observation likelihood (Eqn. 8) of the ongoing activities. A lower score of this likelihood indicates higher abnormality of ongoing activities.

3.5 Handling of Occlusions

In general, the conventional HMM-based model faces difficulties in finding the most likely state sequences for missing observations. Therefore, a continuous most likely state sequences is not re-established once the observations reappear after a certain duration. A bottom-level HMM is used for the quantisation of pair-wise relations (section 3.2) and another top-level HMM is for monitoring workflows.

The quantisation HMM successfully handles the occlusions by treating the reappeared pair-wise relational observations $\mathbf{r}_t^m$ as a new sequence with a new starting point from the time it reappeared. For these reappeared sub-sequences, the model enforces the uniform starting probability $\pi = \{\pi_q\}$ of the HMM parameter $\lambda = (\pi, A, B)$ (section 3.2). As mentioned earlier, each pair-wise relational feature sequence belonging to a workflow sequence is treated separately for the quantisation. Therefore, the state space representation of pair-wise relation sequences corresponding to the observed objects are not affected by other occluded objects.

Table 1. *Primitive events* for 'hammering nails' and 'driving screws' workflow sequences

1. Grab nail baton	2. Place nail baton within marked region	3. Release nail baton	4. Grab hammer
5. Retrieve nail	6. Insert nail	7. Place hammer	8. Hammering nail
9. Release nail	10. Put down hammer	11. Grab screws baton	12. Placed screw baton within marked region
13. Release screw baton	14. Pick screwdriver	15. Retrieve screw	16. Insert screw
17. Release screw	18. Move screwdriver	19. Switch on screwdriver	20. Push down screwdriver
21. Turn off screwdriver	22. Put down screwdriver	23. Unknown	

The monitoring HMM tackles occlusions by taking advantage of the pLSA, which uses the co-occurrence matrix $n(\chi^p, q^p)$ to consider the occurrence frequency of quantised spatial relations in a pair-wise relation. In the event of an occlusion, pLSA masks off spatial relations corresponding to the occluded object.

4 Experiments

Our experimental datasets consist of two type of workflow sequence, 1) hammering 3 nails and 2) driving 3 screws. Two individuals are used to carry out the workflows on a bench. The sequences are captured using the vicon motion capture system [23]. Vicon markers are placed on all *key objects* utilized in the workflow including both wrists of the participants. This dataset consists of 9 objects (hammer, electric screwdriver, nail box, screw box, nail baton, screw baton, left wrist, right wrist and a piece of wood). The workflows are carried out on the workflow bench. Given the tools above, the user is asked to hammer 3 nails and drive 3 screws into the respective nail and screw batons. Using the setup above, a total of 16 (4 per participant per workflow) sequences are obtained. The vicon system provides the output at 50 Hz and 6 *DoF* (3D positions and orientations) for each tracked object while performing a task.

4.1 Evaluations

A total of 23 *primitive events* (Table 1) are identified for the 'hammering nails' and 'driving screws' workflows including an 'Unknown' event for time steps those are not labeled. We evaluated our approach for both off-line and on-line recognition. The off-line evaluation considers the whole workflow sequence for the recognition. The on-line evaluation takes into account the samples from the beginning until time step t, where $t = \{2, 3, \ldots, T\}$ and T is the total duration of the workflow sequence.

The frame-wise recognition rate is compared with the baseline approaches. The baseline evaluations use input as the 3D motion vectors $\mathbf{v}_t^o = (\dot{x}, \dot{y}, \dot{z})_t^o$ for individual object $o = 1, \ldots, N$ at each time step t. The final motion vector

Table 2. Performance comparison for leave-one-out experiment

Methods	Off-line	On-line
HTK-PaHMM	77.40%	12.20%
SVM-Multiclass	24.90%	24.90%
pLSA	36.84%	36.84%
M-HMM-pLSA	61.51%	61.10%

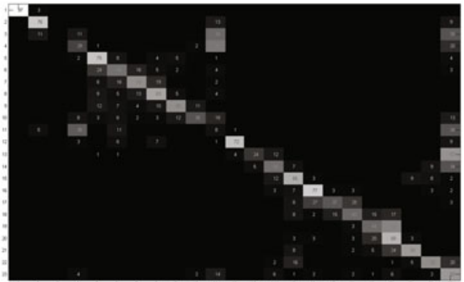


Fig. 5. Confusion matrix for the frame-wise evaluation of 23 *primitive events* for the leave-one-out experiment (off-line)

$\boldsymbol{v}_t = (\mathbf{v}_t^1, \mathbf{v}_t^2, \ldots, \mathbf{v}_t^N)$ at a given time t is a single vector by stacking the individual motion vector. In this experiment, the length of $\boldsymbol{v}_t$ is 27 for 9 objects. We compare our approach with HTK-PaHMM [28,25], SVM-Multiclass [20,7] and pLSA 'topic-model' [26,10].

In the HTK-PaHMM model, there are 23 parallel-HMM representing 23 *primitive events* in workflow sequences. Each HMM is trained separately with subsequences of corresponding *primitive events* from training workflow sequences. We use the HTK-toolkit [28] for this model.

For the SVM-Multiclass representation, each *primitive event* is treated as a class. A normalized $[-1, 1]$ 3D motion vector $\boldsymbol{v}_t$ at each time step t is used as a input feature. As in HTK-PaHMM, the model is trained on the training dataset comprising subsequences of *primitive events* using RBF-kernel. However, the temporal dependency of $\boldsymbol{v}_t$ is not considered. During testing of a workflow sequence, the class label of an unknown $\boldsymbol{v}_t$ at time t is inferred from the learnt model.

For the pLSA 'topic-Model', the input motion vectors $\boldsymbol{v}_t$ are represented as a word $w = \{w_1, w_2, \ldots, w_K\}$ by quantising it using k-means clustering algorithm. Each *primitive event* symbolises a topic $z = \{z_1, z_2, \ldots, z_{23}\}$. In [26], 'topic-model' is used for finding topics or themes corresponding to activities those are frequently occurring in a scene. In our model, we know these topics (*primitive events*) from the labelled workflow sequences. For each topic p, we compute $P(w|z_p)$, $p = 1 \ldots 23$ by counting the occurrence frequency of w. For an unknown document d, we assign a topic $z^* = argmax_z(P(z|d))$, where $P(z|d)$ is estimated using the procedure described in [10] without changing $P(w|z)$. For this model, document d is represented as a sequence of words w taken from a sliding window of duration $\hat{\mathcal{T}}$ (1 sec in this evaluation). This model gave better performance on our dataset for 100 clusters.

The performance of frame-wise comparison for the leave-one-out experiment on 16 workflow sequences is shown in Table 2. The HTK-PaHMM model performed better for the off-line evaluation. However, it gave very poor outcome for the on-line. SVM-Multiclass and 'topic-model' do not consider temporal

Table 3. Performance comparison of our model for leave-one-out experiment with the insertion of random noise to 1) both training and testing workflow sequences, 2) only testing sequences

Noise level	Inserted noise during training and testing		Inserted noise during testing only
	Off-line	On-line	Off-line
No noise $\sigma = 0$	61.51%	61.10%	61.51%
$\sigma = 4$	49.97%	48.90%	40.93%
$\sigma = 10$	52.00%	51.20%	34.56%
$\sigma = 15$	51.68%	50.40%	32.44%
$\sigma = 20$	50.95%	49.57%	28.44%

Table 4. Inter participants off-line performance comparison with random noise inserted in 1) both training and testing workflow sequences, 2) only testing sequences

Noise level	Inserted noise during training and testing		Inserted noise during testing only	
	Test on P_1	Test on P_2	Test on P_1	Test on P_2
No noise $\sigma = 0$	53.21%	59.31%	53.21%	59.31%
$\sigma = 4$	52.24%	48.86%	42.92%	52.10%
$\sigma = 10$	51.59%	53.39%	46.59%	08.63%
$\sigma = 15$	53.53%	54.13%	39.80%	12.42%
$\sigma = 20$	48.77%	52.23%	27.68%	12.15%

dependency and performed reasonably well. Our HMM-pLSA 'corpus-model' gave the best performance over all. The confusion matrix of our model for 23 *primitive events* is shown in Fig. 5. The confusion matrix reveals that some frames in the current *primitive event* are misclassified as either next or previous *primitive events*. This is typical synchronisation error as ground-truth for the evaluation is manually annotated.

Object trajectories captured in our motion capture system are reasonably clean in comparison to vision-based tracking. In order to validate the robustness of our approach, we injected random Gaussian noise of zero mean with varying standard deviation $\sigma = \{4, 10, 15, 20\}$ in centimeters to the 3D positions of objects in our workflow sequences. The frame-wise evaluations for both on-line and off-line is presented in Table 3. The declining performance is less than 12% for $\sigma = 20$ centimeters in both off-line and on-line experiments, when noise is inserted into both training and testing sequences.

In our dataset, two participants P_1 and P_2 carried out an equal number of workflows. We evaluated our method with workflows carried out by one participant in training and the rest for testing, and vice versa. The performance of frame-wise evaluation is shown in Table 4. Surprisingly, performance is comparable in most cases, although there is a large deterioration in performances for higher added noise levels in test data only and with training on a single participant.

4.2 Evaluation of Occlusions

The Vicon motion capture system [23] provides relatively clean data w.r.t. visual analysis and is not enough to validate our hypothesis about handling occlusions. Therefore, we evaluated our approach by removing one or more objects from the testing workflow sequences, whereas the model was trained on sequences by considering all objects. The average performance of complete removal of an

Table 5. Leave-one-out experiment with complete occlusion of an object in the testing sequences. The performance is 61.51% without occlusion.

Occluded objects	Off-line (Average)
screwdriver	58.61%
wood piece	60.98%
nail baton	56.03%
hammer	48.52%
nail box	59.90%
screw baton	55.11%
screw box	61.49%
left wrist	51.66%
right wrist	55.27%

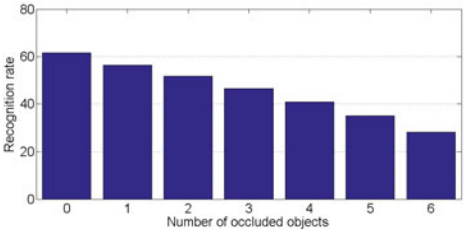


Fig. 6. Recognition performance (off-line) for the leave-one-out experiment with complete occlusion of 0-6 objects

individual object in testing sequences and a leave-one-out experiment is shown in Table 5. Removing static objects such as ‘wood piece’, ‘nail box’ and ‘screw box’, the drop off in performance is less than 1%. However, the model gave encouraging performance to the occlusion of actively involved objects such as ‘hammer’, ‘screwdriver’, ’left wrist’ and ‘right wrist’ (Table 5). We, then evaluated our model by removing two or more objects from the testing sequences. In this evaluation, while removing two or more objects all possible combinations of objects are considered and the average performance is shown in Fig. 6. The method gave accuracy > 50% for the complete occlusion up to two objects.

5 Conclusion

In this work, we proposed an innovative approach for real-time monitoring of workflows. The proposed method uses a novel view-invariant qualitative spatial feature, which is extracted by considering distance and rate of change of distance between a pair of objects in 3D space. The dynamics of this pair-wise relational feature is captured using an HMM. Realisation of workflows from the relational state space is carried out using a ‘corpus-model’, which is derived from probabilistic Latent Semantic Analysis (pLSA). Each *primitive event* in a workflow is modeled separately using our ‘corpus-model’. In order to predict the next possible *primitive event*, the approach uses a monitoring-HMM.

Acknowledgments. We would like to thank Dima Damen and Andrew Gee (Department of Computer Science, University of Bristol) for providing the Vicon dataset. The work is supported by EU grant COGNITO (www.ict-cognito.org, ICT-248290). We thank Elizabeth Carvalho (Center for Computer Graphics, Portugal) for supplying ground-truth for this dataset.

References

1. Ivanov, Y., Bobick, A.F.: of visual acticities and interactions by stochastic parsing. IEEE Trans. on PAMI 22(8), 852–872 (2000)
2. Allen, J.F.: Maintaining knowledge about temporal intervals. Communications of the ACM 26(11), 832–843 (1983)
3. Baum, L.E., Petrie, T., Soules, G., Weiss, N.: A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics 41(1), 164–171 (1970)
4. Blei, D.M., Ng, A., Jordan, M.: Latent dirichlet allocation. J. Mach. Learn. Res. 3, 993–1022 (2003)
5. Brand, M., Oliver, N., Pentland, A.: Coupled Hidden Markov models for complex action recognition. In: Proc. of IEEE CVPR, pp. 994–999 (1997)
6. Bui, H., Venkatesh, S., West, G.: Policy recognition in the abstract hidden markov model. Journal of Artificial Intelligence Research 17, 451–499 (2002)
7. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), http://www.csie.ntu.edu.tw/~cjlin/libsvm
8. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. Royal Statistical Society 39(1), 1–38 (1977)
9. Grimson, W.E.L., Stauffer, C., Romano, R., Lee, L.: Using adaptive tracking to classify and monitor activities in a site. In: Proc. of IEEE CVPR, pp. 22–29 (1998)
10. Hofmann, T.: Probabilistic latent semantic analysis. In: Proc. of Uncertainty in Artificial Intelligence, UAI 1999, pp. 289–296 (1999)
11. Hongeng, S., Nevatia, R.: Multi-agent event recognition. In: Proc. of ICCV, vol. 2, pp. 84–91 (2001)
12. Intille, S.S., Bobick, A.F.: A framework for recognizing multi-agent action from visual evidence. In: AAAI 1999, pp. 518–525 (1999)
13. Jaeger, H.: The "echo state" approach to analysing and training recurrent neural networks. Tech. rep., Fraunhofer Institute for Autonomous Intelligent Systems (December 2001)
14. Johnson, N., Hogg, D.C.: Learning the distribution of object trajectories for event recognition. Image Vision Comput. 14(8), 609–615 (1996)
15. Moore, D., Essa, I.: Recognizing multitasked activities from video using stochastic context-free grammar. In: Proc. AAAI National Conf. on AI, pp. 770–776 (2002)
16. Nguyen, N., Phung, D., Venkatesh, S., Bui, H.: Learning and detecting activities from movement trajectories using the hierarchical hidden Markov model. In: Proc. of IEEE CVPR, vol. 2, pp. 955–960 (2005)
17. Oliver, N., Garg, A., Horvitz, E.: Layered representations for learning and inferring office activity from multiple sensory channels. Computer Vision and Image Understanding 96(2), 163–180 (2004)
18. Padoy, N., Weinl, D.M.D., Berger, M.O., Navab, N.: Workflow monitoring based on 3D motion features. In: Proc. of ICCV Workshop on Video-oriented Object and Event Classification (2009)
19. Pinhanez, C., Bobick, A.: Human action detection using pnf propagation of temporal constraints. In: Proc. of IEEE CVPR (1998)
20. Schüldt, C., Laptev, I., Caputo, B.: Recognizing human actions: A local svm approach. In: Proc. ICPR, pp. 32–36 (2004)
21. Shi, Y., Huang, Y., Minnen, D., Bobick, A., Essa, I.: Propagation networks for recognition of partially ordered sequential action. In: Proc. of CVPR, vol. 2, pp. 862–869 (2004)

22. Veres, G., Grabner, H., Middleton, L., Van Gool, L.: Automatic Workflow Monitoring in Industrial Environments. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 200–213. Springer, Heidelberg (2011)
23. Vicon Systems, http://www.vicon.com
24. Viterbi, A.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans. on Information Theory 13(2), 260–269 (1967)
25. Vogler, C., Metaxas, D.: Parallel Hidden Markov Models for American sign language recognition. In: Proc. of IEEE ICCV, vol. 1, pp. 116–122 (1999)
26. Wang, X., Ma, X., Grimson, W.E.L.: Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models. IEEE Trans. PAMI 31, 539–555 (2009)
27. Wanstall, B.: HUD on the Head for Combat Pilots. Interavia 44, 334–338 (1989)
28. Young, S.J.: The htk hidden Markov model toolkit: Design and philosophy. Tech. rep., Cambridge University Engineering Department (September 1994)

Efficient Spatio-Temporal Edge Descriptor

Claudiu Tănase and Bernard Merialdo

EURECOM, 2229 Route des Crêtes, Sophia-Antipolis, France

Abstract. Concept-based video retrieval is a developing area of current multimedia content analysis research. The use of spatio-temporal descriptors in content-based video retrieval has always seemed like a promising way to bridge the semantic gap problem in ways that typical visual retrieval methods cannot. In this paper we propose a spatio-temporal descriptor called ST-MP7EH which can address some of the challenges encountered in practical systems and we present our experimental results in support of our participation at TRECVid 2011 Semantic Indexing. This descriptor combines the MPEG-7 Edge Histogram descriptor with motion information and is designed to be computationally efficient, scalable and highly parallel. We show that our descriptor performs well in SVM classification compared to a baseline spatio-temporal descriptor, which is inspired by some of the state-of-the-art systems that make the top lists of TRECVid. We highlight the importance of the temporal component by comparing to the initial edge histogram descriptor and the potential of feature fusion with other classifiers.

Keywords: spatio-temporal, descriptor, content-based video retrieval, high-level feature extraction, classification, concept, edge histogram.

1 Introduction

High-level feature extraction is a fundamental topic in multimedia research. Also known as semantic concept detection, its goal is to determine the presence or absence of semantic concepts in multimedia content. We investigate the presence of such concepts in video shots by using *concept classifiers*, which measure the relevance of a concept within a video shot. In the literature, much of the work in this domain is tested against the TRECVid benchmark, which provides large video databases, manual concept annotation that can be used for training and standardized evaluation measures, such as the widely used Mean Average Precision [1]. The task of each participant is to build a system that automatically identifies the video shots where a particular concept is shown (e.g. there is an occurence of concept 'dog' in shot 10 of video 244 in the set), and then rank them by relevance. The database is divided in the training set, which is the base for the experiments, and a test set, on which the system performances will be evaluated. Each participant must provide a list of 2000 shot IDs ranked by decreasing probability of detecting the particular concept. TRECVid organizers provide a shot decomposition, as well as a central keyframe for each shot of the video.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 210–221, 2012.

Highly performing [1] systems in TRECVid rely almost exclusively on image descriptors, computed over central keyframes in shot, so they are basically classifying and retrieving images. However, some [2] sample several keyframes in one shot, and others use local features computed around spatio-temporal interest points (STIP [3]) within the video. In spite of this, in the recent editions of the TRECVid Semantic Indexing task (SIN), progress seems to have slowed down because of several flaws in learning methods or dataset/annotation problems [4]. As each year more and more dynamic and motion-relevant concepts are added, the use of one or few keyframes per shot in concept detection is beginning to show its limitations.

Spatio-temporal descriptors have been used for various video detection tasks, most notably in human action recognition. These descriptors show very good discrimative features and are reliable in general, but have steeper computing requirements and sometimes need pre-processed video data [5,6]. For these reasons, their adoption in real systems at TRECVid has been slow and with mediocre practical results [1]. The most likely cause is the comparatively high computational cost that comes with descriptors on *xyt* space, but also because of dataset problems, such as high intra-concept variability and very few positive instances of concepts.

We propose a spatio-temporal descriptor that can work around these problems. Our ST-MP7EH descriptor is based on the Edge Histogram image descriptor, part of the MPEG-7 standard [7], and is basically analyzing the temporal evolution of edges in video. Our descriptor works by computing an edge histogram in each frame, and then calculating two simple statistic parameters on the distribution in time of each "bin" in the histogram. By subsampling frames at a reasonably low rate we can decrease computation time, which is essential given our large video datasets (TRECVid2010 has 200 hours of video for training and testing). This does little to impact the quality of the descriptor since our temporal statistics (moments) are theoretically invariant to this operation.

This paper is structured as follows: in section 2 we present some existing interesting approaches that rely on spatio-temporal detection, specially using edges. We present the MPEG-7 Edge Histogram, which is the starting point for our work, in section 3. In section 4 we present the method for our descriptor, while also motivating our design decisions. In section 5 we present the details of our testing and comparisons, we show how our spatio-temporal SIFT baseline is constructed and in section 6 we show results computed on actual TRECVid data. We conclude our paper with our comments on the descriptor's performance and future use in section 7.

2 Previous Work

In this section we describe known spatio-temporal video retrieval techniques that use features based on edges and have been successfully used in TRECVid or are part of the state of the art in concept video classification or action recognition. TRECVid systems seem to tend toward increasing the number of visual features and incorporating sophisticated fusion strategies while relying less on motion or edge information [1]. The most successful systems do incorporate spatio-temporal information by sampling multiple keyframes, however their number is extremely limited (MediaMill uses up to 6 additional I-frames distributed around the middle key frame of each shot) [2].

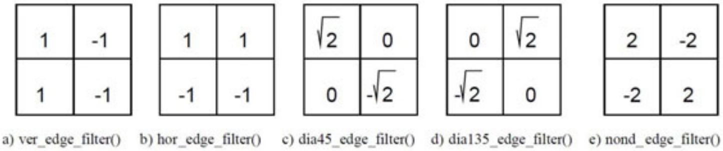
However, several TRECVid participants, mostly in the new Multimedia Event Detection (MED), do use edge features. In SIN (high level feature extraction), the MPEG-7 Edge Histogram has been used only in systems that work with the middle keyframe of the shot [8,9,10], thus without any spatio-temporal or motion information. On the other hand [11] compute edge histograms on a local level and use the BoSW (Bag of Spatiotemporal Words) strategy to track features in the space-time volume. An interesting approach is the TGC (temporal gradient correlogram) by [12], which computes edge direction probabilities over 20 frames evenly distributed in the shot. Their work is similar in principle to ours, but the temporal aspect is represented by a mere concatenation of the 20 vectors resulting from the 20 sampled shots. In spite of some temporal information, this approach is highly dependent on the shot length and has a higher computational cost. The EOAC [13] (edge orientation autocorrelogram) is practically identical. Another example of edge-based descriptor is MEHI (motion edge history image) from [14], which evolves from previous MHI and MEI (heavily used in human action recognition), and is a suitable descriptor for human activity detection. However, its use in general concept-based retrieval is questionable because of camera motion, broader range of possible motions and inherent video quality problems that come with Internet archive videos. Moreover, the exhaustive manner of computation could prove impractical for the high-level feature task.

3 MPEG-7 Edge Histogram Descriptor

The MPEG-7 standard describes an Edge Histogram descriptor for images, which is meant to capture the spatial distribution of edges, as part of a general texture representation. As with all color and texture descriptors defined in the MPEG-7 standard [7], this descriptor is evaluated for its effectiveness in similarity retrieval [15], as well as extraction, storage and representation complexity. The distribution of edges is a good texture signature that is useful for image to image matching even when the underlying texture is not homogeneous.

The exact method of computation for the MPEG-7 Edge Histogram descriptor can be found in [7] and [15]. The general idea is that the image is divided into 4×4 sub-images, and the local edge histograms are computed for each of the sub-images. There are 5 possible edge orientations that are considered: vertical, horizontal, 45° diagonal, 135° diagonal and isotropic (no orientation detected). For each sub-image and for each image type an edge intensity bin is computed, amounting to a total of $16\ images \times 5\ edges = 80$ bins.

Each sub-image is further divided into sub-blocks, which are down-sampled into a 2×2 pixel image by intensity averaging, and the edge-detector operators are applied using the 5 filters in the image below. The image blocks whose edge strengths exceed a threshold are marked as "edge blocks" and used in computing the histogram. These values are counted and normalized to $[0, 1]$ for each of the 80 bins. The value in each bin represents the "strength" of the corresponding edge type in that image block. According to its authors [7], this image descriptor is effective for representing natural images for image-to-image retrieval. It is not suited for object-based image retrieval. Moreover, the computation is efficient [15], and has low dimensionality and storage needs.



a) ver_edge_filter() b) hor_edge_filter() c) dia45_edge_filter() d) dia135_edge_filter() e) nond_edge_filter()

Fig. 1. MPEG-7 directional filters [15]

4 ST-MP7EH Spatio-Temporal Descriptor

In the context of video retrieval, visual descriptors that are traditionally used in CBIR are sometimes used to describe frame sequences instead of images, following a more or less elaborate extension process. A good example of a properly built 3D descriptor is the 3D extension [16] of SIFT (evidently the highest performing visual feature in image search) or the extension [17] of HOG used initially in human action detection. However these cases are rare, as most systems tend to use keyframe-based approaches or compute 2D descriptors at salient points in the ST volume (detected by spatio-temporal interest points).

Our opinion is that temporal and spatio-temporal features have a huge potential in content-based video retrieval, and at the same time we state that keyframe approaches miss a great deal of information on two aspects: the feature we are looking for may not be present on the selected keyframe (but present on other frames in the shot), and the feature is easier to recognize by means of its dynamics throughout the shot rather than its instantaneous visual characteristics [18]. Naturally ST methods require far more processing power than keyframe approaches, so a compromise between performance and computational cost must always be made. For that reason, we decided on MPEG-7 Edge Histogram thanks to its low computational cost [15].

In the same idea as the seminal work of Nelson and Polana [19] we tried to create a global and general descriptor that can detect the evolution in time of visual texture. In our case, the texture is characterized by the predominance of an oriented edge on a region of the image. The ST descriptor is computed in a simple manner: for each frame t of the analyzed video, we compute the (2D) MPEG-7 edge histogram descriptor, which gives an 80 value feature vector x_1^t to x_{80}^t. We compute this on every frame and put the data in an $N \times 80$ matrix, where N is the number of analyzed frames. We consider each column j

of this matrix as a time series $x_j^1, x_j^2, ..., x_j^N$. This series represents the evolution in time of a certain feature of the image (e.g. the 19th element represents the strength of horizontal edges on the 3rd sub-image). The feature vector is made from the average a_j and standard deviation σ_j of each of these 80 series, namely $[a_1, a_2, ..., a_N, \sigma_1, \sigma_2, ..., \sigma_N]$ which gives it a fixed dimension of 160. The order of magnitude is conserved between the two descriptors by means of average and variance. The spatial information given by the edge distribution and the division into grids is inherited from the underlying edge histogram descriptor. Temporal information is present in the form of a shot-wide average edge histogram plus the temporal variance in each histogram bin. Formally, the mean represents the first moment of the discrete distribution, and the standard deviation is the square root of the second central moment (the variance). We use the standard deviation and not the variance in order to conserve the metric (to have the same measuring unit), which is essential to classification.

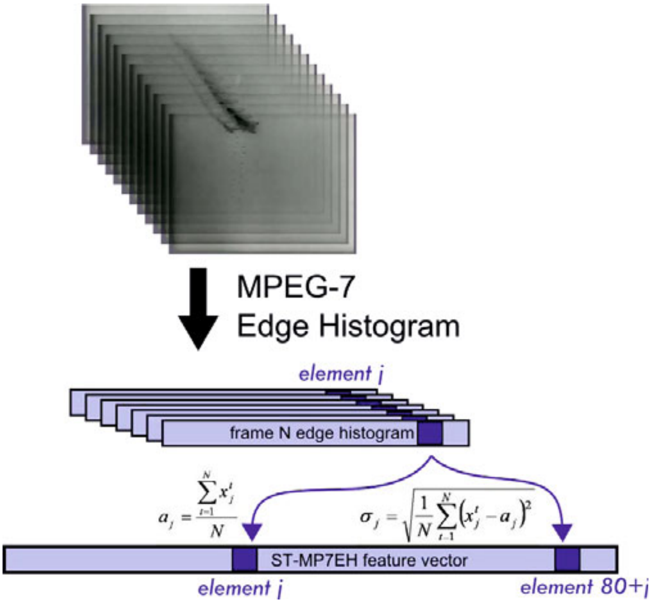

MPEG-7
Edge Histogram
element j
frame N edge histogram
$a_j = \frac{\sum_{t=1}^{N} x_j^t}{N}$
$\sigma_j = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(x_j^t - a_j)^2}$
ST-MP7EH feature vector
element j
element 80+j

Fig. 2. Overview of ST-MP7EH computation

ST-MP7EH has some interesting temporal properties. Firstly, it is a direct temporal extension for the MPEG-7 Edge Histogram: if we compute our descriptor on a sequence containing just one frame N=1, the resulting feature vector would hold the edge histogram for that frame, with all the extra variances equal to zero. We find this helpful in providing a comparison between the 2D descriptors extensively used in video retrieval and the corresponding 3D ST extensions. Secondly, this descriptor is robust to frame subsampling, as means and variances are statistically invariant to subsampling. Thirdly, this is a one-pass global descriptor, which is to say that concerning memory, it only touches each 3D point (in the XYT space) only once, and does so in a linear fashion. The consequence is that sliding-window implementations are simple and efficient (we can cut the ST volume anywhere on the T axis), that computation time is easy to estimate (directly proportional to the number of frames) and that its "global" property ensures that no region in the ST volume will be missed because of bad STIP detection [3], for instance.

Similar temporal strategies have been used in space-time analysis before, most notably in human gesture recognition. Darrell and Pentland propose the use of Dynamic Time Warping [20] for gesture recognition, and other authors have proposed frequency domain [19] (Fourier analysis) and wavelets [21] to detect repetitive patterns in walking motion. However, in the context of general motion of an object in a video sequence, possibly combined with noise and camera motion, periodicity would obviously not prove as robust. The computational overhead would also be significant by comparison to state-of-the-art video retrieval systems. One thing our approach shares in common with frequency domain representations is that both methods store the mean of the signal: ST-MP7EH computes it explicitly and Fourier analysis computes the mean as the first Fourier coefficient.

In order to minimize computation time we temporally sub-sampled the frames of the shot by a 1/5 ratio. We found the subsampling appropriate for two reasons: firstly, because the functions we use should be unaffected by the subsampling of the dataset, and secondly because we assume the continuity of the MPEG-7 edge strengths in time (as they are calculated from a continuous shot). Intuitively this property should hold true for a sequence of frames forming a continuous shot: since the difference between any 2 consecutive shots is small, so should be the difference between 2 elements of the edge histogram for the corresponding edges.

5 Experimental Setup

We have tested our descriptors on our TRECVid 2011 testbed platform. Eurecom's system uses a fusion of several visual classifiers (SIFT, GIST, color moments, wavelet features), of which SIFT is the earliest and still the most powerful, just like in most TRECVid systems. ST-MP7EH has been designed to complement the image descriptors by providing spatio-temporal information which would be invisible to keyframe-based descriptors. A computational constraint is also imposed by the amount of video data that is exponentially increasing from one edition of TRECVid to the next and the available hardware. The scoring metric for TRECVid SIN is the MAP (Mean Average Precision). Average precision (AP) is a standard performance metric of a concept classifier. Given the classifier's output as relevance scores on a set of shots, we rank the shots in descending order of their score, and compute AP as the average of the precisions of this ranked list truncated at each of the relevant shots. The mean of APs, or mean average precision (MAP), is a metric of the average performance of multiple concept classifiers.

5.1 ST-MP7EH Evaluation

We test the performance of our ST-MP7EH descriptor on TRECVid data. The chosen training and test sets are two subsets of the annotated TRECVid2010 data available at the moment of writing. Our experiments were conducted using the 10 concepts from the TRECVid 2010 ”light run” of the SIN task. The training set contains 59800 shots and the test set 59885 shots. The video data comes from approximately 8000 Internet Archive videos (50GB, 200 hours) with Creative Commons licenses in MPEG-4/H.264 with durations between 10 seconds and 3.5 minutes. We compute our descriptor on all available shots, which are segmented using an automated boundary detection mechanism. Given the fact that this is an ”embarrassingly parallel” problem, splitting the workload into a manageable number of jobs is trivial. Computation time for a single shot depends on shot length and frame size, as well as hardware-dependent considerations. On average for 10% of the tv10.shorts.B corpus (137327 shots) it takes approx. 28.23 hours, which makes for an average 7.4011 seconds per shot. This can also be approximated as 1.23× playback time. We estimate memory usage as 5.747 kB per shot.

The next step is the SVM training. We label our data using the available annotation. At this point the number of training examples becomes concept-dependent as the annotation is not complete over the entire dataset. We use a modified version of the SVM software available from LibSVM [22] that uses the $\chi 2$ kernel. As with all other SVM training experiments, we use one single-class SVM per concept that should differentiate between positive and negative samples. Given the disproportionate nature of the positive and negative examples (positive/negative ratio is $< 1\%$ for every concept), the label obtained in testing will always be negative. We use the assigned soft-boundary probability P as an indication of how likely the test vector is to actually be a positive instance, and call this the ”score” of the shot. We sort by this probability in order to obtain our ranked list on which we compute the AP for the 10 concepts. The average of APs over all concepts is the MAP.

5.2 Comparison with Spatio-Temporal Baseline Descriptors

We compare our descriptor with several baselines in terms of MAP. The first experiment is meant to compare direct retrieval quality, regardless of the nature of the descriptor. According to [1], the best individual visual descriptor in the current generation of video concept detection systems is still SIFT. We use a Bag of Words approach by clustering all SIFT features into 500 clusters (visual words) and constructing a histogram of visual word occurrences for each sample, based on the nearest visual word. Inspired by the work of MediaMill [2], we adopt a multi-keyframe approach, were we sample a large number of keyframes from the shot (1/5 regular frames), compute SIFT features, and finally create a single visual word occurrence histogram per keyframe. We classify using the same SVM method as for ST-MP7EH. At this point the 3 variants of our descriptor branch: we either consider the highest-scoring keyframe as the overall score for the corresponding shot (*mkfSIFT1*), average all scores to get the shot score (*mkfSIFT3*), or average the visual word histograms (*mkfSIFT2*) and finally compute the MAP. We consider these baseline descriptors as prototype spatio-temporal visual detectors for general concept classification. We motivate this by highlighting the fact that none of the descriptors that work well on human action [21,20], surveillance, etc. have made their way into general-concept systems because of their weak generalizing power.

5.3 Spatio-Temporal Performance Gain

We highlight the temporal quality of ST-MP7EH by directly comparing retrieval performance to its "predecessor", the MPEG-7 Edge Histogram. For that we compute Edge Histograms on one relevant keyframe in each shot and use the resulting 80-value vector in SVM classification. Results clearly indicate the gain of using multiple keyframes per shot. This experiment has been carried out on a subset of the training set, containing half of the training samples.

5.4 ST-MP7EH - SIFT Late Fusion

Following the multiple descriptor fusion paradigm that seems to dominate current state-of-the-art systems, especially in TRECVid, we made a late fusion between our SIFT descriptor and ST-MP7EH. The idea was to show that whilst both descriptors provide good concept recognition separately, each one represents a different type of visual information: SIFT is a very accurate image (spatial) descriptor, while ST-MP7EH focuses more on the temporal. In our experiment we tried to prove that fusing two different descriptors in such a manner could significantly improve the MAP. Since we use the same learning technique for the 2 descriptors, we have one SVM "score" from SIFT and another for ST-MP7EH for each shot. These 2 scores can be fused using a linear combination, with the mix variable α as a free parameter. We computed for each shot $score_{fusion} = \alpha \cdot score_{ST-MP7EH} + (1-\alpha) \cdot score_{SIFT}$ for 10 values of α in the interval [0,1]. For each value of α we computed the ranked lists and calculated the MAP.

6 Results

6.1 Comparison with Spatio-Temporal Baseline Descriptors

Table 1 shows the APs and the MAP obtained using the ST-MP7EH descriptor compared to the 3 variants of our SIFT multi-keyframe baseline described in 5.2. Improvement is evident for concepts containing motion (either of the object, such as Boat or Bus, or the camera, as on Cityscape or Singing). Cityscape has an exceptionally high score because of the dominant vertical edges, which can be seen as a discriminative feature for the concept. We justify the low score of Airplane_flying by pointing out the spatial inconsistency (the object can be anywhere in the frame, at any scale, whereas our descriptor is not scale-invariant) and the lack of background information (the background is either an edge-less sky, or ground that is irrelevant to the concept).

Table 1. Comparison between ST-MP7EH and spatio-temporal baseline

Descriptor	ST-MP7EH	mkfSIFT1	mkfSIFT2	mkfSIFT3
Airplane_Flying	0.00021851	0.00589261	0.01793414	0.02514655
Boat_Ship	0.02262003	0.01880984	0.02367281	0.01797646
Bus	0.00187492	0.00348848	0.00514326	0.00629265
Cityscape	0.21769612	0.17755185	0.13681920	0.15426416
Classroom	0.00857810	0.00785321	0.00900273	0.00465120
Demonstration_Or_Protest	0.01465306	0.03806428	0.06042413	0.03266605
Hand	0.00342396	0.00335580	0.00572085	0.00759707
Nighttime	0.01671574	0.02773094	0.04030243	0.06348906
Singing	0.07864447	0.06039517	0.07210428	0.07564689
Telephones	0.00006563	0.00969003	0.00302432	0.00346670
MAP	0.03644900	0.03528322	0.03741482	0.03911968

6.2 Spatio-Temporal Performance Gain

This test has been performed on a subset of the training and test datasets, consisting on half (30307) the number of shots. The experiment compares MAP for ST-MP7EH and MPEG-7 Edge Histogram and shows how many concepts that are lacking in spatial recognition (i.e. Demonstration_Or_Protest) perform far better in spatio-temporal analysis. The results can be seen in table 2. Since ST-MP7EH actually uses Edge Histogram, the improvement is a measure of temporal relevance given by the concept. Note that the MAP for ST-MP7EH differs from the one in the previous experiment because of the different datasets used.

Table 2. Comparison Between ST-MP7EH and MPEG-7 Edge Histogram

Descriptor	ST-MP7EH	MPEG-7 edge
Airplane_Flying	0.00041296	0.00210417
Boat_Ship	0.00437917	0.00527415
Bus	0.00039620	0.00006228
Cityscape	0.20201674	0.02011237
Classroom	0.02932826	0.00374221
Demonstration_Or_Protest	0.02609449	0.00629317
Hand	0.02561565	0.01700422
Nighttime	0.10152920	0.05869340
Singing	0.07730526	0.05053595
Telephones	0.00256774	0.00155078
MAP	0.04696450	0.01653720

6.3 ST-MP7EH - SIFT Late Fusion

Figure 3 shows how different mixes between ST-MP7EH and SIFT perform. The 10 columns represent 10 values for the α parameter, from 0 (*pure* ST-MP7EH) to 1 (*pure* SIFT). The first observation is that individually ST-MP7EH has a higher MAP (0.046964567) than SIFT (0.029211185) for these datasets. The fusion shows that there is clearly an optimum for each concept where the MAP from the fusion exceeds both descriptors. This is the result of complementary information that ST-MP7EH and SIFT are able to describe. The average gain in precision attributable to latent fusion is of 18.86782%, corresponding to a MAP of 0.055825 for a common value of $\alpha = 0.43$. We can also pick an optimum α value for each concept, which gives an upper bound of improvement of 22.823%, or a MAP of 0.057683244.

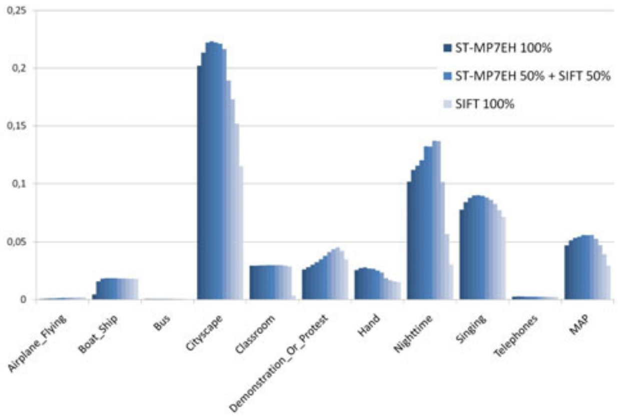


Fig. 3. Average Precision for late fusion between ST-MP7EH and SIFT

7 Conclusions

In this paper we presented a short overview of visual descriptors used in video retrieval concentrating on edge features, we proposed a novel spatio-temporal extension to the MPEG-7 Edge Histogram Descriptor, we described the computational method and provided experimental results of SVM-based retrieval in comparison to analogous spatio-temporal baseline descriptors, in comparison to its spatial predecessor and in fusion with SIFT.

Current large scale concept video retrieval systems show a slow adoption of dynamic features. In TRECVid, for example, only the Multimedia Event Detection task has provided significant research in spatio-temporal features. In the generalistic concept classifiers seen in the Semantic Indexing Task, the vast majority of systems still use only one keyframe to describe the entire shot. Only two different approaches have proven successful: local features (either in space or space-time) computed around STIP (spatio-temporal interest points) and the use of more than one keyframe in shot description. However these descriptors are part of complex systems where they participate in feature fusion. On all accounts, any method that analyzes the XYT volume is subject to a high processing and memory penalty. Our descriptor is invariant to changes in temporal scale, so that the frames of the shot can be subsampled up to a minimal rate. We can improve computation time by lowering the sampling rate with very little change in the feature vector.

Our experiment shows that MAP increases by almost 3 times (2.839) when we pass from the keyframe-based MPEG-7 Edge Histogram Descriptor to ST-MP7EH. This is remarkable since the two descriptors carry the same spatial information. The difference comes only from the temporal description via the usage of moments (mean and variance). We believe that these results should encourage the adoption of temporally-relevant description methods in such systems. The results of our late fusion experiment confirm that fusing a heterogeneous set of descriptors can yield a higher MAP thanks to the complementarity of the different representations. This is what we have been recently witnessing in TRECVid: large systems that collect information from color, texture, motion, audio, metadata features, etc. and perform a latent fusion similar to ours. To this end, EURECOM will use this descriptor in such a feature fusion scheme in the 2011 edition of TRECVid.

References

1. Over, P., Awad, G., Fiscus, J., Antonishek, B., Qu, G.: TRECVID 2010 - an overview of the goals, tasks, data, evaluation mechanisms and metrics, pp. 1–34 (2010)
2. Snoek, C.G.M., van de Sande, K.E.A.: The MediaMill TRECVID 2010 semantic video search engine. In: Proceedings of the TRECVID Workshop (2010)
3. Laptev, I., Lindeberg, T.: Space-time interest points. In: Proceedings of Ninth IEEE International Conference on Computer Vision, 2003, vol. 1, pp. 432–439 (October 2003)
4. Yang, J., Hauptmann, A.G. (Un)Reliability of video concept detection. In: Proceedings of the 2008 International Conference on Content-Based Image and Video Retrieval, CIVR 2008, pp. 85–94. ACM, New York (2008)
5. Weinland, D., Ronfard, R., Boyer, E.: A survey of vision-based methods for action representation, segmentation and recognition. Comput. Vis. Image Underst. 115, 224–241 (2011)
6. Ren, W., Singh, S., Singh, M., Zhu, Y.: State-of-the-art on spatio-temporal information-based video retrieval. Pattern Recognition 42(2), 267–282 (2009)
7. Manjunath, B.S., Rainer Ohm, J., Vasudevan, V.V., Yamada, A.: Color and texture descriptors. IEEE Transactions on Circuits and Systems for Video Technology 11, 703–715 (1998)
8. Shimoda, Y., Noguchi, A., Yanai, K.: UEC at TRECVID 2010 semantic indexing task (2010)
9. Naito, M., Hoashi, K., Matsumoto, K., Shishibori, M., Kita, K., Kutics, A., Nakagawa, A., Sugaya, F., Nakajima, Y.: High-level feature extraction experiments for TRECVID 2007. In: TRECVID 2007 (2007)

10. Tang, S., Dong Zhang, Y., Tao Li, J., Feng Pan, X., Xia, T., Li, M., Liu, A., Bao, L., Chang Liu, S., Feng Yan, Q., Tan, L.: Rushes Exploitation 2006 By CAS MCG (2006)
11. Moumtzidou, A., Dimou, A., Gkalelis, N., Vrochidis, S., Mezaris, V., Kompatsiaris, I.: ITI-CERTH participation to TRECVID 2010 (2010)
12. Rautiainen, M., Varanka, M., Hanski, I., Hosio, M., Pramila, A., Liu, J., Ojala, T.: TRECVID 2005 Experiments at MediaTeam Oulu (2005)
13. Mahmoudi, F., Shanbehzadeh, J., Eftekhari-Moghadam, A.-M., Soltanian-Zadeh, H.: Image retrieval based on shape similarity by edge orientation autocorrelogram. Pattern Recognition 36(8), 1725–1736 (2003)
14. Yang, M., Ji, S., Xu, W., Wang, J., Lv, F., Yu, K., Gong, Y., Dikmen, M., Lin, D.J., Huang, T.S.: Detecting Human Actions in Surveillance Videos
15. Park, D.K., Jeon, Y.S., Won, C.S.: Efficient use of local edge histogram descriptor. In: Proceedings of the 2000 ACM Workshops on Multimedia, MULTIMEDIA 2000, pp. 51–54. ACM, New York (2000)
16. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional sift descriptor and its application to action recognition. In: Proceedings of the 15th International Conference on Multimedia, MULTIMEDIA 2007, pp. 357–360. ACM, New York (2007)
17. Kläser, A., Marszalek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: BMVC 2008 (2008)
18. Efros, A., Berg, A., Mori, G., Malik, J.: Recognizing action at a distance. In: Proceedings of Ninth IEEE International Conference on Computer Vision, 2003, vol. 2, pp. 726–733 (October 2003)
19. Polana, R., Nelson, R.: Recognition of motion from temporal texture. In: 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Proceedings CVPR 1992, pp. 129–134 (June 1992)
20. Darrell, T., Pentland, A.: Space-time gestures. In: 1993 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Proceedings CVPR 1993, pp. 335–340 (June 1993)
21. Liu, F., Picard, R.: Finding periodicity in space and time. In: Sixth International Conference on Computer Vision, pp. 376–383 (January 1998)
22. Chang, C.-C., Lin, C.-J.: LIBSVM: a Library for Support Vector Machines (2001)

Recurring Element Detection in Movies

Maia Zaharieva and Christian Breiteneder

Interactive Media Systems Group
Vienna University of Technology
Favoritenstrasse 9-11/188-2, Vienna, Austria
{zaharieva,breiteneder}@ims.tuwien.ac.at

Abstract. Recurring elements in movies contribute significantly to the development of narration, themes, or even mood. The detection of such elements is impeded by the large variance of their visual appearance and usually relies on the experience and attentiveness of the viewer. In this paper, we present a new approach for the automated detection of recurring elements in movies such as motifs and main characters. Performed experiments show the reliability of the algorithm and its potential for automated high-level film analysis.

Keywords: film analysis, dominant object detection, motif detection.

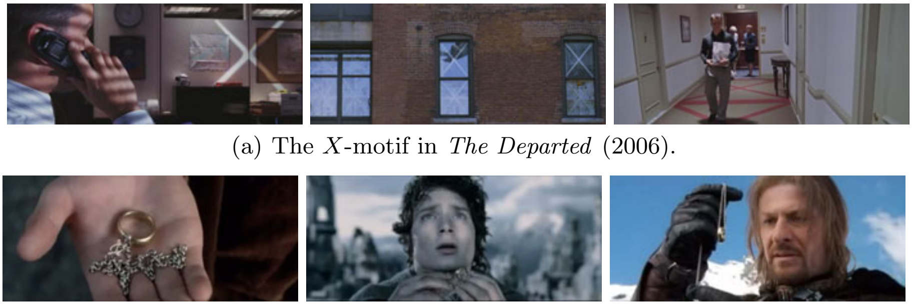
1 Introduction

Near-duplicate detection is a rapid emerging research field focused at the identification of identical or near-identical video sequences. Its vast development is mostly driven by requirements of large media providers, advertising agencies, and commercial companies. Near-duplicate detection facilitates application scenarios such as improved search and retrieval of videos by reducing the number of duplicated videos, the monitoring of commercials broadcastings, and copyright protection [12]. While currently near-duplicate detection explores video sequences as a whole, it does not allow for search on more detailed level such as the investigation of duplicated characters or objects. The detection of such recurring elements is a new requirement for automated film analysis stated by art and film experts.

Recurring elements are a common tool in visual arts such as painting, photography, and filmmaking. Examples for such elements can be found in the paintings by Salvador Dali (the piano is typical for his Surrealist compositions) or the films by Dziga Vertov (rails, spinning wheels, etc.) or Alfred Hitchcock (birds, cameo appearances, etc.). Recurring elements are often applied to convey a certain message, theme, or mood. They are usually called motifs and take very different shapes such as the use of a specific color or sound in a given context, a particular movement of an object or character, camera position, composition, or even a story line [1,2]. Motifs are often highly symbolic. Thus, their detection requires for semantic understanding and relies on the experience and attentiveness of the audience. An example for such visually highly varying motif is the X-motif in

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 222–232, 2012.

The Departed (2006) by Martin Scorsese (see Figure 1(a)). The X appears whenever a character is in mortal danger and takes very different shapes by means of lighting, color, and material. However, motifs can also be easily recognizable such as the ring in the *Lord of the Rings* (2001-2003) by Peter Jackson (see Figure 1(b)).



(a) The X-motif in *The Departed* (2006).

(b) The easily recognizable ring-motif in the *Lord of the Rings* (2001-2003).

Fig. 1. Examples for motifs in movies

In this paper we explore the feasibility of the detection of recurring elements by means of automated computer vision methods. A clear restriction for such an approach is the requirement for certain similarities in the visual appearance of present recurring elements. We propose a method based on local features that are robust to changes in illumination, rotation, and scaling. Furthermore, the system automatically learns recurring regions and creates links between related views of one and the same object. Following, detected elements may differ significantly in their position, orientation, and scale. In our approach, a region (or element) can be an object, a part of it, or recurring character (usually the main actors). Finally, the proposed method allows for the detection of recurring elements not only in a single movie but also among different works from the same author and, thus, can support a high-level film analysis currently performed tediously and manually by film experts. In summary, the main contributions of this paper are:

- We define a new research task motivated by the requirements of film experts.
- We propose an automated method to detect recurring elements in movies independently of their position, orientation, and scale.
- The output of the proposed system allows for a variety of summarizing visualizations of semantically related information.

This paper is organized as follows. In Section 2 we give an overview over related research. Section 3 describes the algorithm for recurring region detection. Section 4 presents experiments we performed as proof-of-concept for the evaluation of the proposed algorithm. We conclude in Section 5 and give an outlook for further research.

2 Related Work

To the best of our knowledge, recurring elements detection has not been subject to research so far. Related research areas compromise near-duplicate detection and object detection and tracking.

Near-duplicate detection aims at identifying images or video sequences showing slight variance due to editing or changes in lighting, viewpoint, motion, etc. [5,9,14]. This research area has emerged in recent years for a variety of applications such as the recognition of TV commercials, detection of duplicated news videos, media linking, and copyright infringement detection. Recently, Huang et al. proposed a method for scene recognition based on near-duplicates object detection [8]. The authors argue that shots of the same scene most probably share a large number of similar objects or background. However, the authors do not perform any object but simple keypoint detection and tracking. Following, a shot is represented by an average space-time feature, called imprecisely object key feature. In contrast to recent research in near-duplicate detection, our work performs on more detailed level. While existing approaches detect duplicated or reused media (images or video) as a whole we aim at the identification of recurring elements within a given medium and their reuse among different media.

Object detection and tracking usually requires a predefined appearance model of the salient object or a priori information about the scene for reliable background subtraction and motion tracking [6,7,10]. The application scenarios are manifold ranging from traffic control and surveillance to sport video analysis and the recognition of human action. Recently, Celik et al. proposed a method for unsupervised object detection in unlabeled surveillance video data [3,4]. The authors first detect salient object based on motion information and simple dimensional features. In the next step, similarity-based clustering allows for the grouping of objects according to the category they belong to. The approach is only applicable in a restricted scene with a static camera. Salient objects have to be moving and within a certain degree of perspective deformation due to the dimensional features in the initial step. Our approach differ significantly from existing methods for object detection and tracking in respect of available knowledge about both object and scene and in respect of the degree of detection, i.e. general category (a person, a car, etc.) vs. a specific subject.

3 Approach

The aim of the proposed system is to detect recurring regions within a video sequence. A video sequence can be a shot, a scene, or a whole movie. Detected regions have to meet two essential requirements. First, they have to be distinctive and not homogeneous regions such as the sky, or a wall. Second, detected regions should allow for a multiple view representation of the captured element. Thus, the proposed system includes two critical components, region detection and region representation, which will be discussed in the following sections (see Figure 2).

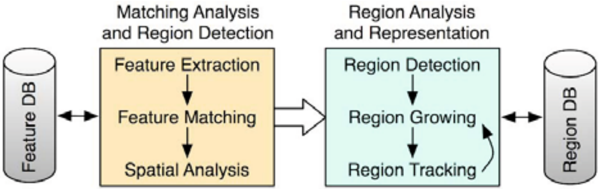


Fig. 2. Algorithm workflow

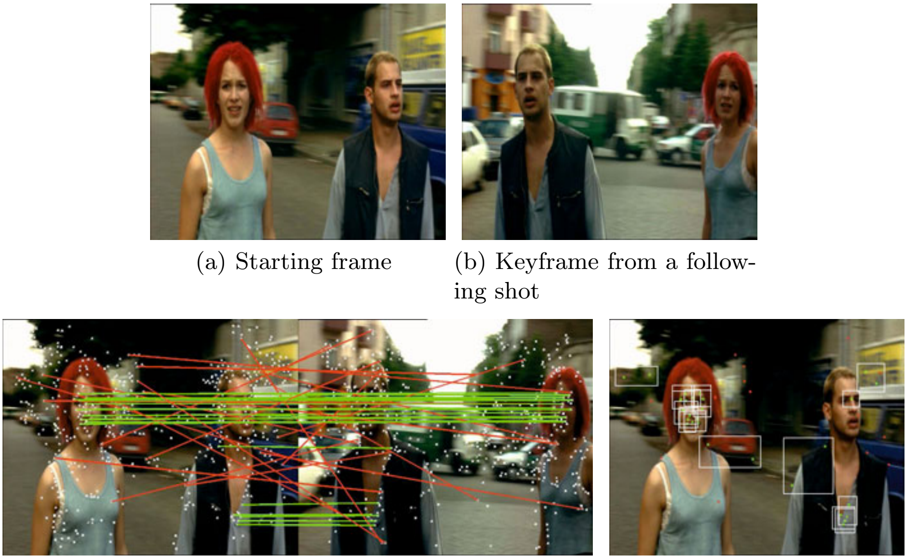
Given a video sequence for recurring object detection, the first step is, as in any general video analysis approach, the detection of shot cuts and the extraction of keyframes as shot representation. Both topics are well-investigated research areas resulting in numerous existing methods. For shot boundary detection we employed the method proposed by Truong et al. [13]. It is a simple adaptive thresholding technique detecting peaks in the histogram difference curve of consecutive frames. For each detected shot, we extract the first, middle, and last frames as keyframes. Despite the simplicity of both methods, they proved to work efficiently with the involved data set and achieved satisfactory results in the performed experiments. Since the input for the proposed system is a sequence of keyframes, they can be easily replaced by more sophisticated methods if needed.

3.1 Region Detection

For each keyframe K^{S_j}, where S_j is the corresponding shot, we detect distinct interest points and extract local features based on the Scale Invariant Feature Transform (SIFT) [11]. SIFT features are invariant to changes in translation, scale, and rotation and partially invariant to changes in illumination and affine distortions and, thus, allow for matching across different viewing conditions. Each feature F is described by a quadruple $\{K_i^{S_j}, x, y, D\}$, where K_i is the associated keyframe id, x and y the corresponding coordinates, and D the local feature descriptor.

Following, we perform initial, coarse region detection based on feature matching. Each keyframe is compared to each following keyframe in the input video sequence. Feature descriptors are matched by identifying the first two nearest neighbors in terms of Euclidean distances. A descriptor is accepted if the nearest neighbor distance is below a predefined threshold. The value of 0.8 was determined experimentally and used through the evaluation tests described in Section 4. To reduce the number of false matches we introduce a loose spatial constraint. Each match is considered within the cluster of its three nearest neighboring feature points. A match is accepted if there is at least one further match present in the cluster. Finally, all accepted clusters are set as initial regions.

Figure 3 shows an example for an initial region detection from the movie *Run Lola Run* (1998) by Tom Tykwer. Compared are two frames from two



(a) Starting frame

(b) Keyframe from a following shot

(c) Matched features: white dots identify detected interest points in the corresponding frame; red lines indicate false matches; green lines correct matched features

(d) Detected initial regions in the starting frame. Red dots indicate dropped features due to the spatial constraint.

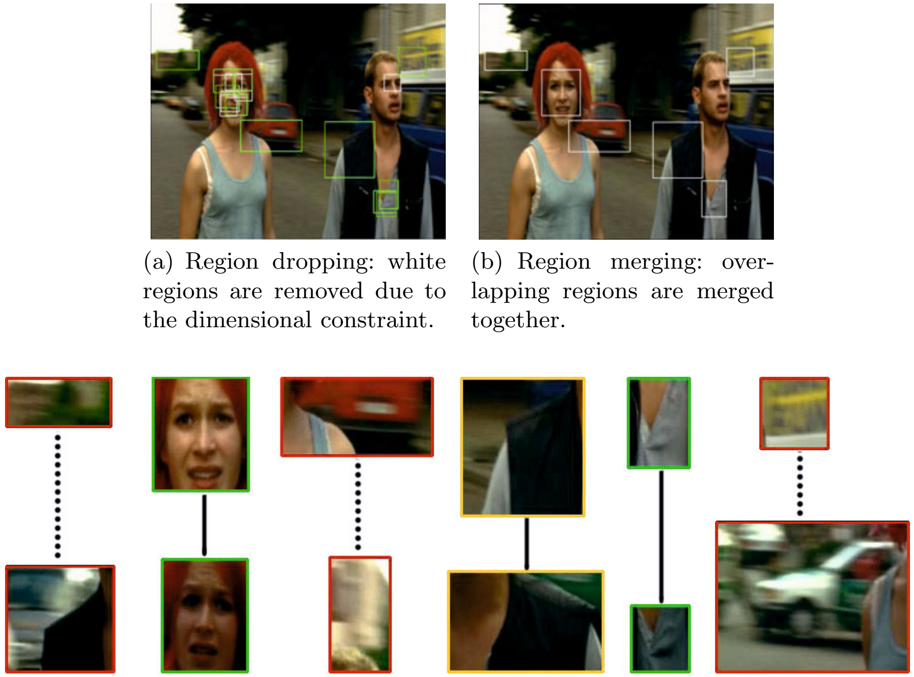
Fig. 3. Example for initial region detection (for better visualization some spacing is introduced within the detected regions)

different shots showing Lola and her boyfriend on the run from the police. The scene is shot from two different viewpoints (see Figures 3(a)-3(b)). Although the matching process produces a number of false positives (see the red lines in Figure 3(c)), most of the false matches are dropped due to the spatial constraint on the next stage of the algorithm (see Figure 3(d)).

3.2 Region Analysis and Representation

The first stage of the algorithm, coarse region detection, results in numerous regions. To reduce their number we first remove all regions with dimensions and area below a given threshold. Following, we perform region growing by detecting and merging all overlapping regions. Finally, each region R is defined as $\{K_i^{S_j}, x, y, w, h, R_M\}$, where w is the width of the region, h its height, and R_M is a set of links to matched regions $\{R_1, R_2, ...R_N\}$.

Figure 4 visualizes the process of region dropping and merging for the previous example from the movie *Run Lola Run*. From initially detected 28 regions, more than 50% were dropped due to the dimensional restriction (see Figure 4(a)). In our experiments we set the minimum for both width and height of detected region to 5 px. In such way a region can be visually perceived and interpreted by the viewer even if it only depicts a small part of an object. Following, all overlapping regions are merged together building preliminary final regions (see Figure 4(b)). Since the whole process of region detection for the starting frame is repeated for all following keyframes, detected regions are constantly updated in size, quantity, and the set of linked regions. For the detection of final recurring elements all linked regions can be recursively traversed. Eventually, some regions have few repetitions for the whole video sequence while others indicate recurring elements (see Figure 4(c)).



(a) Region dropping: white regions are removed due to the dimensional constraint.

(b) Region merging: overlapping regions are merged together.

(c) Final region linking: red borders indicate false positive linking; yellow borders show templates with similar parts of the same object; green borders indicate correct linked templates. Dotted lines shows elements with very few repetitions for the whole video sequence. Solid lines indicate detected recurring elements for the investigated video sequence.

Fig. 4. Example for region dropping and merging

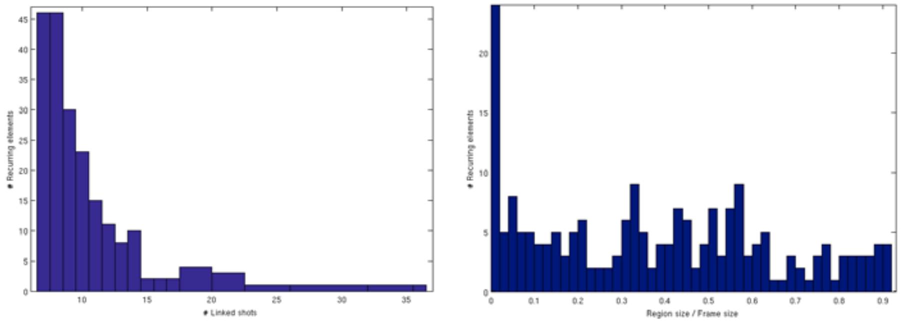
4 Experiments

As proof-of-concept for the proposed algorithm we perform two experiments. The first one focusses on the detection of recurring elements in a single, contemporary movie, and the second one explores the reuse of elements in and among several archived documentaries by the same filmmaker.

4.1 Contemporary Movie

For the first experiment we employed the German movie *Run Lola Run* (1998). The story follows Lola who has 20 minutes to raise 100.000 German marks and save her boyfriend's life. The film presents sequentially three possible scenarios about the story development and its outcome. All three scenarios share the same locations and characters. Following, the film involves many recurring elements (objects as well as characters) which makes it extremely suitable for our experimental tests.

Figure 5(a) depicts the decreasing distribution of the amount on linked shots per detected region. For a better visualization we only show the top 2% regions that have been linked to 7 or more shots. Approx. 98% of all detected regions are linked to less than 7 shots and, thus, considered as insignificant for our application scenario.



(a) Linked shots per recurring element

(b) Relative size to frame size

Fig. 5. Distributions of detected recurring elements

The definition of ground truth for recurring objects in a movie is a tedious process feasible probably by the filmmaker only. Therefore, we focus on the precision performance of the conducted experiments. In our evaluations, we define precision by the ratio of correct linked regions vs. all linked regions. The precision for the top 2% of all detected regions is approx. 75% which confirms the potential of the algorithm. In summary, we investigated over 200 regions with the corresponding associated regions. In average, for each detected region 10 shots has been linked (or 17 regions since multiple keyframes per shot are possible). The average area per region is 38% of the frame size (see Figure 5(b)). It turns out that detected regions should not be too small. Anything bellow 5% is not really a meaningful region but rather a part of an object such as a skin section, a shirt detail, a wall texture, etc. Following, the region cannot be tracked reliably since it is found in a large number of frames in spite of their non semantical relation.

Currently, few falsely linked regions reduce the overall performance. It is an implication of the approach that if a newly detected regions is matched to an existing one, the new region inherits all established links of the second region. Figure 6 shows four examples for detected recurring regions. The first example depicts two main characters, Lola and her father, from various scenes in the movie. The remaining examples show recurring objects: a huge dollar bill on the wall of the office of Lola's father, a phone, and a flying bag. The last example also demonstrates a false linking with Lola's hair since the texture of the bag and Lola's hair exhibit high similarities in their texture. Horizontal lines illustrate the level of linked elements. Especially noteworthy is the visual variance within the same level. While in the example with the dollar bill there is a high degree on visual similarity on the level bellow the top region, in the first example, Lola and her father are matched separately and the linked regions do not have any common visual information although they share the same semantical topic .

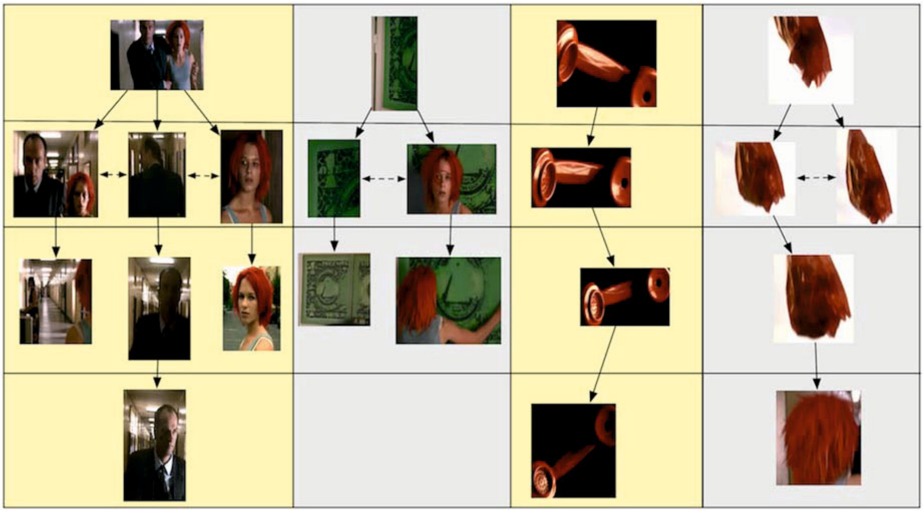


Fig. 6. Examples for detected recurring elements: solid lines depict directly linked regions; dashed lines show indirectly associated successors of the same region

Figure 7 shows a detected recurring element with its complete set of linked regions. The example shows Lola, running to raise money, her boyfriend looking at the clock on the wall, and a close up of the clock, which is an often occurring scene in the movie. All three objects (the two characters and the clock) are repeatedly found and linked together in various detail degree: from the initial close up, via a long shot of Lola, to an extreme close up of her trousers. Based on the region characteristics, next task could be the classification of regions according to the shooting length into e.g. a close up, a medium, and a long shot. The example illustrates the two main characteristics of the approach. *First*, due to the applied local features in the one-to-one keyframe comparison, directly linked regions (depicted by directional solid lines) share some common visual information. This does not necessarily hold true for the successors of a given region. Note, the green highlighted regions in Figure 7. Although, they are both successors of the same region, they do not share common visual information. However, they are both involved into the same semantical topic. *Second*, linked regions can be distributed over the entire movie. Next to the tree representations we used in the discussed examples, a variety of visualization methods can be applied to represent semantically related information based on detected recurring elements such as MPEG-7 collections, hierarchical and sequential summaries, etc.

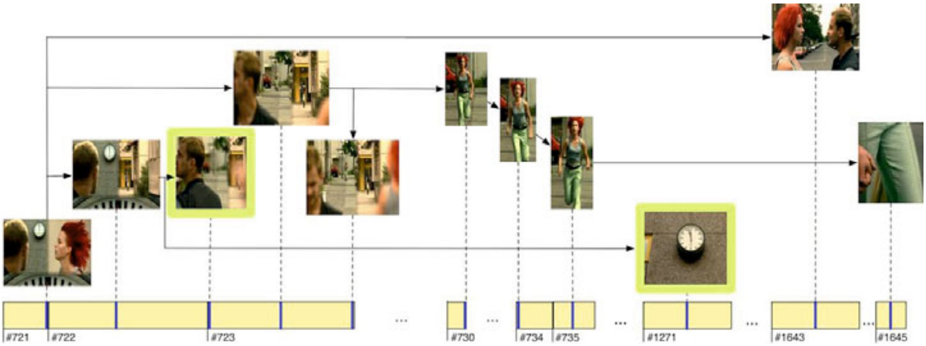


Fig. 7. A complete example for a detected region and the corresponding linked elements. Yellow boxes depict corresponding shots, blue lines indicate keyframe position within the shots. Solid lines between the regions show a direct linkage between regions. The two green highlighted regions are an example for siblings of the same region.

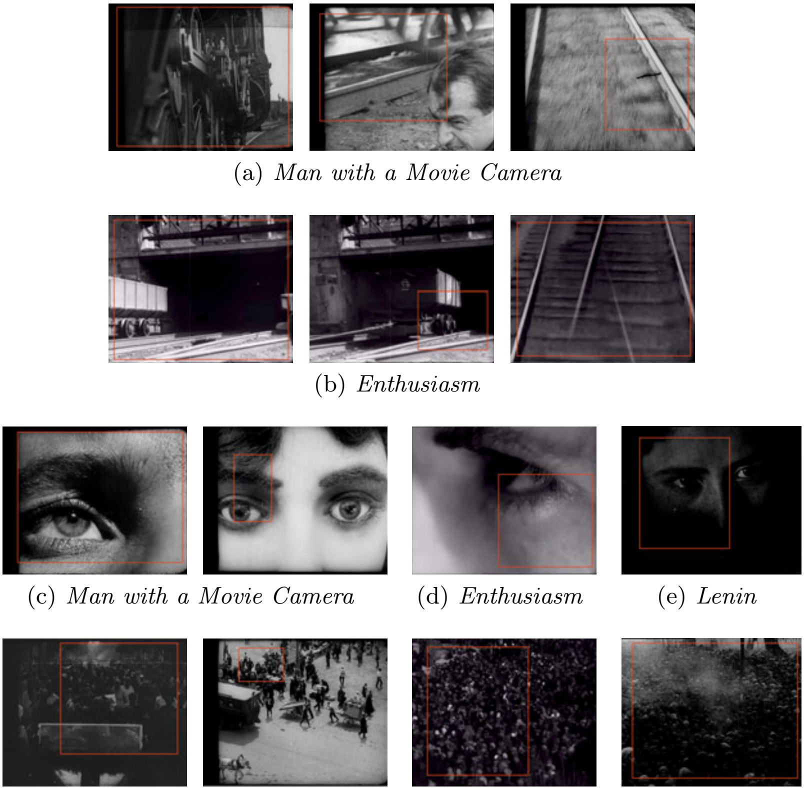
4.2 Archived Documentaries

The second experiment we performed in the context of recurring element detection investigates three archived documentaries by Dziga Vertov: *Man with a Movie Camera* (1929), *Enthusiasm* (1931), and *Three Songs about Lenin* (1934). The reason to choose the three documentaries is a suggested motif by film experts shared in all three movies: the rails. Hence, we first explore the movies separately and compare the results to those of the contemporary material. Following, we verify whether or not the proposed algorithm is able to detect recurring elements among different works of the same filmmaker.

In contrast to the contemporary movie from the previous experiment, the explored archived documentaries exhibit less recurring elements with much lower amount on linked shots per detected region (maximum of 7). This is mainly due to the fact that most documentaries care less about narration and actors, they often change locations, and characters do not necessarily recur. As a result, detected recurring elements within a single movie are mostly a long camera take that was cross-cut with a second scene. Hence, such shots bear a high visual similarity. In general, documentaries turn out to exhibit, to a greater extent, recurring scenes or sets rather that recurring elements (objects or characters).

Following, detected regions occupy predominantly (nearly) the full frame size. Finally, the precision performance is comparable to the results achieved in the first experiment. The average precision performance for the archived documentaries is approx. 70%.

Starting point for the cross-movie analysis are previously detected recurring regions in each movie. Similar to region tracking within a single movie, corresponding regions are matched using local features and a nearest neighbor ratio matching strategy. For our evaluations at least five matches per region are required to define a reasonable match. Figure 8 shows the top three elements detected in the explored movies: rails, eye, and crowd. Despite the partially high visual dissimilarities, all three detected elements represent typical Vertov motifs applied across different works.



(a) *Man with a Movie Camera*

(b) *Enthusiasm*

(c) *Man with a Movie Camera* (d) *Enthusiasm* (e) *Lenin*

(f) *Man with a Movie Camera* (g) *Enthusiasm* (h) *Lenin*

Fig. 8. Cross-movie analysis. 8(a)-8(b): rails-motif. 8(c)-8(e): eye-motif. 8(f)-8(h): crowd-motif. All detected regions are embedded into the original frame for better visualization.

5 Conclusion

In this paper we presented a new approach for the detection of recurring elements in movies. Since detected regions can be an object, a part of it, or a character, the system allows for the detection of visually similar motifs and recurring characters. The linking between detected regions shows possible different views of the recurring elements and facilitates the quick retrieval of relevant sequences. Performed experiments with different works by the same filmmaker demonstrate the potential of the proposed algorithm to assist experts in film analysis and film studies.

References

1. Beaver, F.E.: Dictionary of film terms: the aesthetic companion to film art. Peter Lang Publishing (2009)
2. Bordwell, D., Thompson, K.: Film art: an introduction. McGraw-Hill (2008)
3. Celik, H., Hanjalic, A., Hendriks, E.A.: Unsupervised and simultaneous training of multiple object detectors from unlabeled surveillance video. Computer Vision and Image Understanding 113(10), 1076–1094 (2009)
4. Celik, H., Hanjalic, A., Hendriks, E.A., Boughorbel, S.: Online training of object detectors from unlabeled surveillance video. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1–7 (2008)
5. Douze, M., Jégou, H., Schmid, C.: An image-based approach to video copy detection with spatio-temporal post-filtering. IEEE Trans. Multimedia, 257–266 (2010)
6. Ess, A., Schindler, K., Leibe, B., Gool, L.V.: Object detection and tracking for autonomous navigation in dynamic environments. International Journal of Robotics Research 29(14), 1707–1725 (2010)
7. Guo, W., Xu, C., Ma, S., Xu, M.: Visual Attention Based Motion Object Detection and Trajectory Tracking. In: Qiu, G., Lam, K.M., Kiya, H., Xue, X.-Y., Kuo, C.-C.J., Lew, M.S. (eds.) PCM 2010, Part II. LNCS, vol. 6298, pp. 462–470. Springer, Heidelberg (2010)
8. Huang, C.R., Chen, C.S.: Video scene detection by link-constrained affinity-propagation. In: IEEE Int. Symp. on Circuits and Systems, pp. 2834–2837 (2009)
9. Joly, A., Frélicot, C., Buisson, O.: Content-based copy detection using distortion-based probabilistic similarity search. IEEE Trans. MM 9(2), 293–306 (2007)
10. Leibe, B., Schindler, K., Cornelis, N., Van Gool, L.: Coupled object detection and tracking from static cameras and moving vehicles. IEEE Trans. Pattern Anal. Mach. Intell. 30(10), 1683–1698 (2008)
11. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
12. Shen, H.T., Liu, J., Huang, Z., Ngo, C.W., Wang, W.: Near-duplicate video retrieval: Current research and future trends. IEEE Multimedia (2011)
13. Truong, B.T., Dorai, C., Venkatesh, S.: New enhancements to cut, fade, and dissolve detection processes in video segmentation. In: ACM International Conference on Multimedia, pp. 219–227 (2000)
14. Zhao, W.L., Ngo, C.W., Tan, H.K., Wu, X.: Near-duplicate keyframe identification with interest point matching and pattern learning. IEEE Trans. Multimedia 9(5), 1037–1048 (2007)

Scalable Mobile-to-Mobile Video Communications Based on an Improved WZ-to-SVC Transcoder

Alberto Corrales-Garcia[1], José Luis Martínez[2],
Gerardo Fernández-Escribano[1], and Francisco Jose Quiles[1]

[1] Instituto de Investigación en Informática de Albacete (I3A),
University of Castilla-La Mancha,
Campus Universitario, 02071 Albacete, Spain
{albertocorrales,gerardo,paco}@dsi.uclm.es

[2] Architecture and Technology of Computing Systems Group, Complutense University
Ciudad Universitaria s/n, 28040 Madrid, Spain
joseluis.martinez@fdi.ucm.es

Abstract. Nowadays, video communications between mobile devices are one of the most demanded multimedia services. Since Wyner-Ziv coding provides low cost video encoding, it is a suitable codec to encode video with less resources. On the other hand, the video delivery provided by Scalable Video Coding covers the needs of a wide range of homogeneous networks and different devices. As a consequence, Wyner-Ziv to Scalable Video Coding transcoding can offer a suitable framework to support scalable video communications between low-cost devices. However, the complexity of the transcoder accumulates most of the complexity of both codecs and it must be reduced. In this paper, we introduced an improved Wyner-Ziv to Scalable Video Coding transcoding framework to support homogeneous mobile video communications. The experimental results show that the complexity is reduced around 83.5% without significant Rate-Distortion penalty.

Keywords: Transcoding, Scalable Video Coding, Wyner-Ziv Coding, Temporal Scalability.

1 Introduction

In most of traditional video communications, such as television broadcasting, the information is encoded once and then transmitted to many terminal devices. This is known as down-link model. The applications based on this model are supported by traditional video codecs, such as those adopted by all MPEG and ITU-T video coding standards [1]. The architectures of these codecs have most of the complexity in the encoding part whilst the decoding process is less complex. Taking into account the fact that existing networks are heterogeneous and the receivers have different features and limitations (such as power consumption, available memory, display size, etc), a new scalable standard has recently been proposed in order to support this variety of networks and devices. In particular, Scalable Video Coding (SVC) [2] has been standardized as a scalable extension of the H.264/AVC standard [1].

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 233–243, 2012.

Additionally, in the last few years, with the ever-increasing development of mobile devices and wireless networks, a growing number of emerging multimedia applications have required an up-link model. These end-user devices are able to capture, record, encode and transmit video with low-constraint requirements. Applications, such as low-power sensor networks, video surveillance cameras or mobile communications, present a different framework in which low-cost senders transmit video bitstreams to a central receiver. In order to manage this kind of applications efficiently, Distributed Video Coding (DVC) [3], and specially Wyner-Ziv coding (WZ) [4], proposed a solution in which most of the complexity is moved from the encoder to the decoder.

Taking into account the both low-cost coding parts, we can design a framework to support mobile-to-mobile video communications with both sender and receiver low-complexity video algorithms. These low-requirement video communications can be achieved by means of a WZ to traditional video coding transcoder such as H.264/AVC. With the introduction of this device in the network, the transcoder needs to perform the more complex algorithms such as WZ decoding and H.264/AVC encoding while, on the other hand, the user devices are responsible for the simpler ones: WZ encoding and H.264/AVC decoding. With this aim, in the literature several WZ to traditional video coding transcoders have been proposed, such as one based on H.263 [5] and another on H.264/AVC [6]. In addition, many applications such as multiuser video conferencing or live video streaming send video stream to a wide variety of networks and devices. As a consequence, a scalable solution is also desirable because of the variety of features for these end-user devices and networks. Then, on the down-link part an SVC bitstream which contains several layers (one base layer and one or more enhancement layers). As a result, with a WZ to SVC transcoding framework (Figure 1) a low-complexity mobile video communication can be provided between different devices.

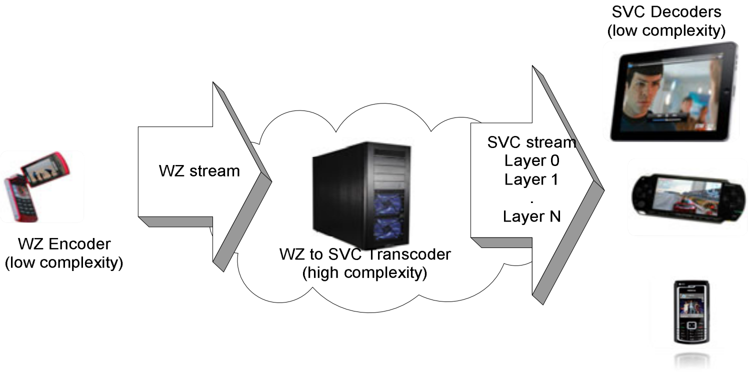


Fig. 1. WZ to SVC transcoder framework

Considering the high complexity of the transcoder, the idea of this paper is to perform this process as efficiently and as fast as possible by using information gathered in the first part in order to reduce the delay caused by the conversion. By including the SVC paradigm, different receivers can satisfy their requirements and the video can also be delivered over a variety of networks. The improved WZ-to-SVC transcoder with temporal scalability presented in this paper reuses the Motion Vectors (MVs) generated during WZ decoding, because they can give us an idea about the quantity of movement in the current frame and this information will be used to accelerate the Motion Estimation (ME) stage as part of the SVC decoding algorithm.

This paper is organized as follows: In Section 2, the technical background of this paper is discussed. Section 3 describes the state-of-the-art for WZ-based transcoders. In Section 4 our improved approach is presented, and some implementation results are shown in Section 5. Finally, in Section 6 conclusions are presented.

2 Technical Background

2.1 WZ Coding

One of the first practical Wyner-Ziv frameworks was proposed by Stanford in [3]. This approach was widely referenced and improved on by later proposals, and as a result, the DISCOVER project proposed an architecture based on the Stanford one in [4]. This architecture was later improved upon by the VISNET-II architecture [7]. The idea of WZ video coding relies on exploiting the source statistics in the decoder that are based on the availability of some decoder side information. For this purpose, the encoder splits the sequence into two kinds of frames: Key frames (K) and Wyner-Ziv frames. K frames are encoded by an H.264/AVC encoding algorithm and are used for generating this side information (therefore, in practice this bitstream provides some degree of H.264/AVC backward compatibility with a lower temporal resolution). On the contrary, WZ frames will follow the WZ video coding paradigm which consists of a DCT, a quantizer and a channel coding module on the encoder side and, for the decoder, apart from the side information generation module, it includes the channel decoder, IDCT and the reconstruction modules. The rate control is implemented in Stanford-based architectures by means of a feedback channel. More detail can be found in [4, 7, 8].

2.2 Scalable Video Coding

Scalable Video Coding is an extension of the H.264/AVC standard. SVC streams are composed of layers which can be removed to adapt the streams to the needs of end users or the capabilities of the terminals or the network conditions. The layers are divided into one base layer and one or more enhancement layers which employ data from lower layers for efficient coding. SVC supports three main types of scalability: 1) Temporal Scalability; 2) Spatial Scalability; and 3) Quality (SNR) Scalability. For a comprehensive overview of the scalable extension of H.264/AVC, the reader is referred to [9].

This work is focused on the WZ to SVC transcoding with temporal scalability. To provide temporal scalability, a bitstream is divided into a temporal base layer (with an identifier equal to 0) and one or more temporal enhancement layers (with identifiers that increase by 1 in every layer), so that if all the enhancement temporal layers with an identifier greater than one specific temporal layer are removed, the remaining temporal layers form another valid bitstream for the decoder. In this way, to achieve temporal scalability, SVC links its reference and predicted frames using hierarchical prediction structures [10] which define the temporal layering of the final structure. In this type of prediction structures, key pictures (typically I or P frames) are coded in regular intervals by using only previous key pictures as references. The pictures between two key pictures are hierarchically predicted and together with the succeeding key picture are known as Group of Pictures (GOP). The sequence of key pictures represents the lowest frame rate (temporal base layer) and the frame rate can be increased with the non-key pictures that are divided into enhancement layers. There are different structures for enabling temporal scalability. SVC provides both dyadic and non-dyadic temporal scalability using hierarchical pictures. In this paper the temporal scalability is achieved by means of P pictures. This technique provides lower latency and is particularly useful for multimedia communications such as mobile video broadcasting or mobile digital television where the transmission of a scalable bitstream is a good solution for mobile terminals with several restrictions.

3 Related Work

The transcoding solutions are well known in the literature throughout many transcoding approaches [11]. In addition, some of them have been based on scalable transcoders which make use of the SVC standard. All of them convert from H.264/AVC to SVC. In this framework, due to the fact that SVC supports different kinds of scalabilities the approaches can be divided into: 1) quality-SNR; 2) spatial; and 3) temporal capabilities. For quality-SNR scalability, in 2009, De Cock et al. presented different open-loop architectures for transcoding from a single-layer H.264/AVC bitstream to SNR-scalable SVC streams with Coarse-Grain Scalability (CGS) layers [12]. In 2010, Van Wallendael et al. proposed a simple closed-loop architecture that reduces the time of the mode decision process by analyzing the mode information from the input H.264/AVC video stream and using it to build a fast mode decision model [13]. Regarding spatial scalability, in 2009 a proposal was presented by Sachdeva et al. in [9]. The idea consists of a single information layer to SVC multiple-layer for adding spatial scalability to all existing non-scalable H.264/AVC video streams. The algorithm reuses available data by an efficient downscaling of video information for different layers. Finally, for temporal scalability, in 2010, Al-Muscati et al. proposed another technique for transcoding that provided temporal scalability in [14]. The method presented was applied in the Baseline Profile and reused information from the mode decision and motion estimation processes from the H.264/AVC stream. In the same year, R. Garrido-Cantos et al. presented an H.264/AVC to SVC video transcoder that efficiently reuses some motion information from the H.264/AVC decoding process in order to reduce the time consumption of the SVC encoding algorithm by reducing the motion estimation process time. The approach was developed for Main Profile and dynamically adapted for several temporal layers [15].

Additionally, in the past few years, another kind of transcoding approaches focused on WZ video coding has been proposed. In this framework, there are two kinds of approaches available in the literature: those which are based on H.263 [5], and those which are based on H.264/AVC [6]. In [5] the authors propose a WZ-to-H.263 video transcoder that reuses motion vectors from WZ to determine the starting point of the H.263 motion estimation. On the contrary, in [5] the authors propose a WZ-to-H.264/AVC transcoder which implements a faster variable-block-size motion estimation algorithm.

4 WZ to SVC Transcoding

As is shown in Figure 2, the WZ-to-SVC transcoder is composed of a WZ decoder concatenated with an SVC encoder. In the proposed architecture, the MVs are temporally stored in a buffer and sent to the Motion Prediction module of SVC, where they are processed as described in the following sub-sections.

4.1 Motion Vectors Extraction

During the WZ decoding process, the side information generation process is one of the most important tasks, because WZ frames are decoded by reconstructing the information provided by the SI. The two main approaches about the SI generation have been the hash-based motion estimation and the Motion Compensated Temporal Interpolation (MCTI). In the particular case of the SI generated by the VISNET-II codec is based on MCTI [3] with the following steps: Firstly, a forward ME is performed between the two K frames (Figure 3). In this step each 16x16 MB of the backward frame looks for the MB which generates the lowest residual inside the forward frame. This searching is carried out within a fixed search range of 32x32. Subsequently, the bidirectional ME calculates two MVs from the MV generated during the previous step. In order to improve the accuracy of the MVs generated, they are up-sampled for 8x8 blocks and a new bidirectional ME is done. Once the bidirectional motion field is obtained, it is observed that the motion vectors sometimes have low spatial coherence, so the MVs are improved by a spatial smoothing algorithm targeting the reduction in the number of false motion vectors. This is based on weighted vector median filters. Finally, bidirectional motion compensation is performed again. These steps are described in more detail in [16].

To obtain more accurate MVs, 16x16 MBs are divided into four 8x8 sub-blocks and for each 8x8 sub-block, two MVs (forward and backward) are calculated and stored. These MVs can help us to estimate the quantity of movement during the SVC stage.

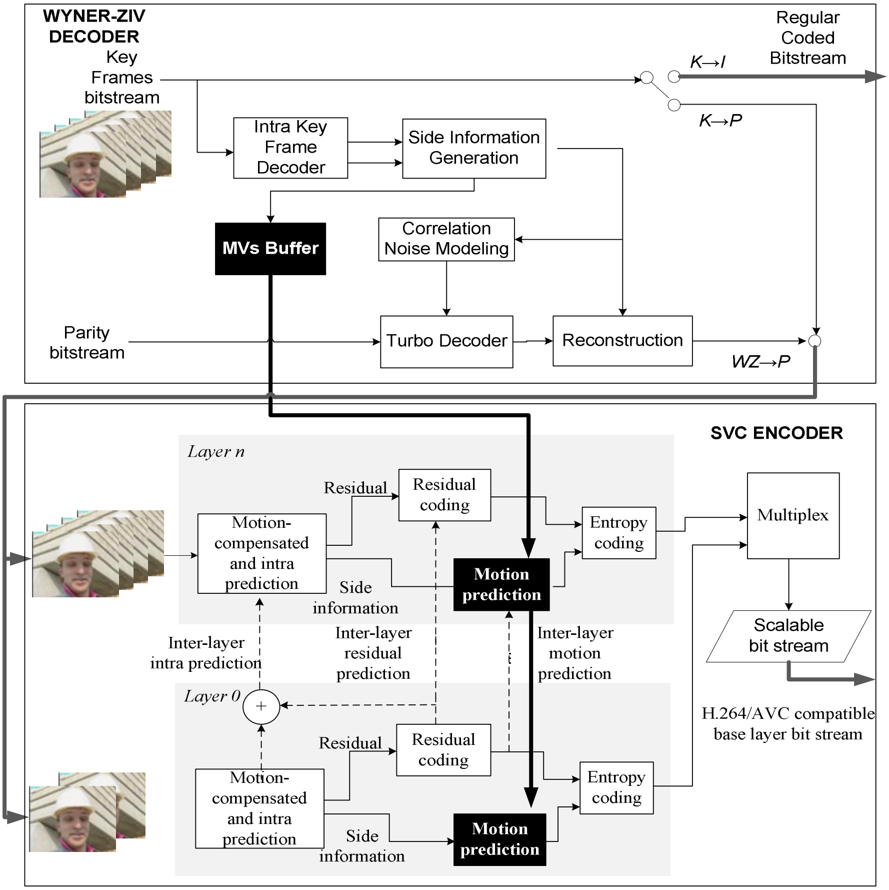


Fig. 2. Proposed WZ to SVC Video Transcoder Architecture

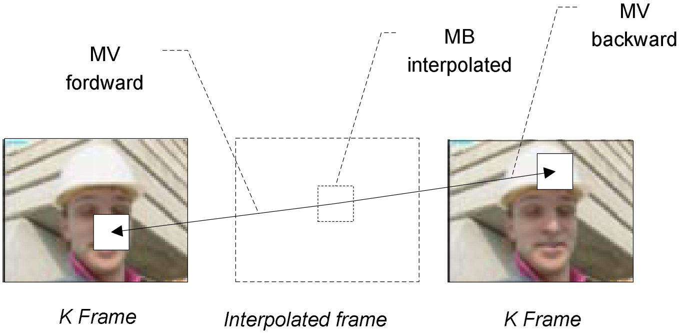


Fig. 3. Side Information generation process

4.2 Motion Vectors Mapping

After the MVs are obtained from the SI process, we must decide how we can use them to accelerate the SVC encoding. WZ and SVC are quite different video codecs. Thus, the first step is to decide the mapping needed to assign the MVs to the forward predicted (P) frames of SVC, and then reduce the ME process. Figure 4 represents the transcoding from a WZ GOP 2 to a H.264/AVC GOP IPPP. The first K frame is transcoded to an I-frame without any conversion, as was shown in Figure 4. On the other hand, for every WZ frame we have two MVs (forward and backward predicted). Then, the orientation of backward MVs is changed and each MV is assigned by keeping the position of the frame in the WZ sequence, as is shown by Figure 4.

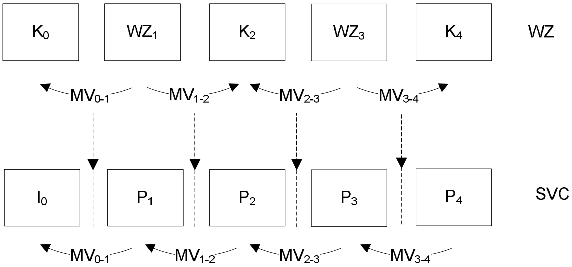


Fig. 4. Mapping MVs from WZ (GOP 2) to SVC (GOP IPPP)

4.3 SVC Fast Encoding

As is well known, most of the complexity of the SVC encoding depends largely on the search range used in the SVC ME process, which is a consequence of the quantity of positions checked. This paper is behind of the idea of accelerating this process by avoiding unnecessary checking without a significant impact on quality or bit rate. In this way, we use the MVs generated by WZ decoding (which contain information about the quantity of movement per MB) to reduce the search area used by the ME of the SVC encoder. In Section 4.1 we explained how the MVs are calculated and extracted, and in Section 4.2 we described how we map the MVs from GOPs of WZ to SVC. Once the MVs are mapped between frames, depending on the partition checked in the SVC encoding algorithm, we can use a different group of these 8x8 MVs. As is well known, in the SVC (as in the H.264/AVC standard) the inter prediction is carried out by means of the process of variable block size motion estimation. This approach supports motion compensation block sizes ranging between 16x16, 16x8, 8x16 and 8x8, where each of the sub-divided regions is a MB (MB) partition. If the 8x8 mode is chosen, each of the four 8x8 block partitions within the MB may be further split in 4 ways: 8x8, 8x4, 4x8 or 4x4, which are known as sub-MB partitions. For all these partitions, ME is carried out and a separate MV is generated. As is shown in Figure 5, WZ provides one backward MV for every 8x8 sub-partition in each MB. Although WZ side information gives us two pairs of MVs (one backward and one forward), the present approach only uses the backward ones because they are correlated with a P-frame in SVC. Then, depending on the sub-partition to be checked by SVC, the final MV predicted is calculated as follows: if the sub-partition is bigger than 8x8 (16x16, 8x16, 16x8), the predicted MV is calculated by taking the average of the MVs included in the sub-partition. For example, for the 8x16 MB-partition, only the MVs allocated at *Block_0* and *Block_1* will be used. If the sub-partition is equal to, or smaller than, 8x8, the corresponding MV is applied directly. Finally, this predicted MV is applied to reduce the ME search area dynamically. This dynamic search area is defined by a circumference with a radius that is dependent on the estimated MV (*Rmv*). In addition, this radius is multiplied by a factor equal to the distance of the reference frame in order to mitigate the effect of the longer distance between a frame and its reference. The dynamic search area can thus oscillate between a minimum (defined by *Rmin*, and set at 3) and a maximum (limited by the whole SVC search area).

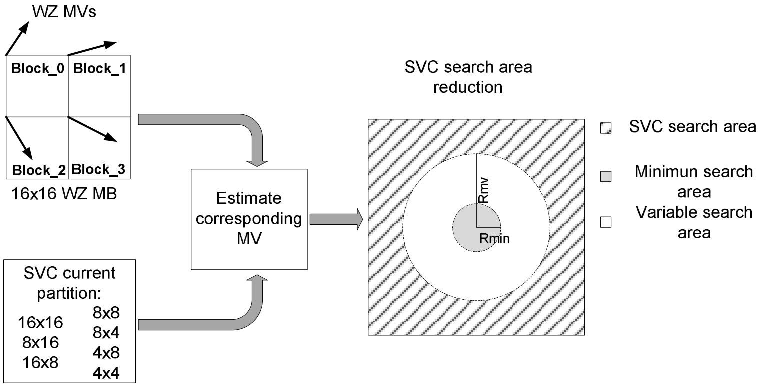


Fig. 5. Estimation of the dynamic ME search area for SVC

5 Experimental Results

The WZ video stream is generated by the VISNET II codec using a fixed matrix QP = 3 in pixel domain (which means that there are 3 bitplanes processed) and a GOP length of 2. While sequences are being decoded, the MVs are passed to the SVC encoder without any increase in complexity. Afterwards, the transcoder converts this WZ video input into an SVC video stream using QP = 28, 32, 36, and 40, as specified in *Bjøntegaard and Sullivan´s* common test rule [17]. The simulations were run by using the version JSVM 9.19.14 [18] and the baseline profile with the default configuration file. The baseline profile was selected because it is the most widely-used profile in real-time applications due to its low complexity. In order to check our proposal we have chosen four representative sequences with different motion levels at 15 fps and 30 fps, coding 150 frames and 300 frames, respectively; these are the same sequences that were selected in the DISOCOVER codec evaluation [8]. On the other hand, the *percentage of Time Reduction* (%TR) is calculated as is specified by

Equation 1. In tables 1 and 2, the reported TR (%) displays the average of the time reduction for the four SVC QP points under study, compared with the reference transcoder which is composed of the full WZ decoding and SVC encoding algorithms.

$$TR\ (\%) = 100 * \frac{\left(Time_{proposed} - Time_{reference}\right)}{Time_{reference}} \quad (1)$$

Table 1 shows the RD penalty measured for 15 fps sequences encoded using two temporal layers and the TR of the proposed transcoder. Concerning the RD results, we can observe that the drop penalty is negligible for every layer; even for the Hall sequence the coding efficiency is better than the reference transcoder. The TR achieved is -83.54% on average, so the time consumed by the SVC encoding is greatly reduced by using the MVs generated in the WZ decoding stage.

On the other hand, Table 2 includes the results for the 30fps sequences, and then 3 layers of temporal scalability. The results reported are similar to those of the 15 fps sequences, achieving a TR of -83.45% on average without a significant RD drop penalty.

Table 1. Performance for 15fps sequences and 2 temporal layers

Sequence	1 Layer (7.5 fps)		2 Layers (15 fps)		TR (%)
	ΔBitrate (%)	ΔPSNR (dB)	ΔBitrate (%)	ΔPSNR (dB)	
Foreman	1.57%	-0.06	2.03%	-0.07	-83.03%
Hall	0.09%	0.00	-0.11%	0.00	-84.95%
CoastGuard	1.14%	-0.06	1.17%	-0.05	-84.79%
Soccer	1.54%	-0.09	2.03%	-0.09	-81.38%
mean	**1.09%**	**-0.05**	**1.28%**	**-0.05**	**-83.54%**

Table 2. Performance for 30fps sequences and 3 temporal layers

Sequence	1 Layer (7.5 fps)		2 Layers (15 fps)		3 Layers (30 fps)		TR (%)
	ΔBitrate (%)	ΔPSNR (dB)	ΔBitrate (%)	ΔPSNR (dB)	ΔBitrate (%)	ΔPSNR (dB)	
Foreman	1.32%	-0.09	1.58%	-0.08	1.75%	-0.07	-82.02%
Hall	0.09%	0.00	0.16%	-0.01	0.14%	-0.01	-84.77%
CoastGuard	2.24%	-0.11	2.55%	-0.10	2.40%	-0.08	-84.55%
Soccer	1.22%	-0.05	2.67%	-0.12	2.41%	-0.09	-82.47%
mean	**1.22%**	**-0.06**	**1.74%**	**-0.07**	**1.68%**	**-0.06**	**-83.45%**

Finally, Figure 6 displays the RD obtained by using a QP = 28, 32, 36 and 40. As can be seen, there are no significant differences in quality or the bit rate obtained by the SVC reference codec and our proposed one. There are only tiny bitrate increases in several QP points and above all in high movement/complexity sequences. Similar RD results are obtained when comparing with PSNR, as is shown for 15 fps in Figure 6 (a) and for 30 fps in Figure 6 (b).

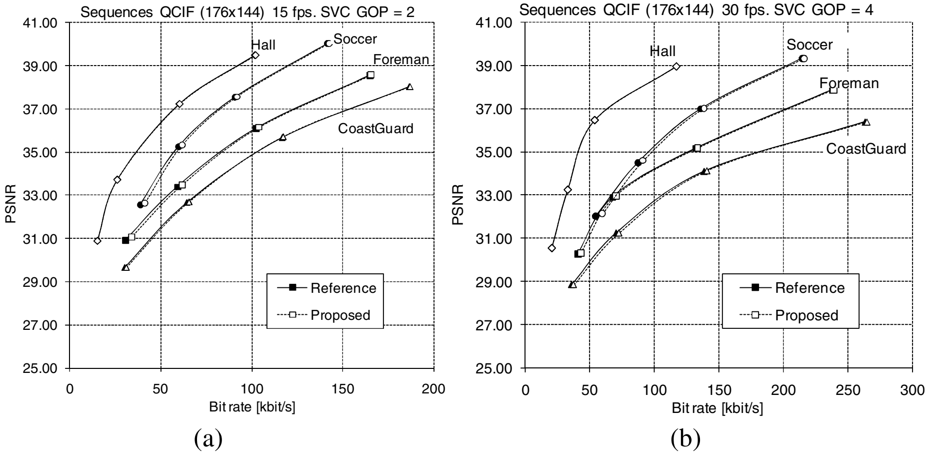


Fig. 6. PSNR/bitrate results for transcoding from WZ GOP = 2 to SVC IPPP GOP 2 (15 fps and 2 layers) and 4 (30 fps and 3 layers)
Reference symbols: ■Foreman ♦Hall ▲CoastGuard ●Soccer

6 Conclusions

This paper proposes a novel WZ-to-SVC transcoding framework. As a result, video communications between a wide range of mobile devices and over heterogeneous networks are supported with low complexity. Our proposed transcoder with temporal scalability analyzes and adapts the motion information generated during WZ decoding to accelerate the ME process of the SVC encoder. In order to manage this approach, several sequences were transcoded by using different scalability layers for 15 and 30 fps frame rates. In our results, the SVC encoding time is reduced by around 83.50% whilst maintaining the efficiency in RD terms.

Acknowledgments. This work was supported by the Spanish MEC and MICINN, as well as European Commission FEDER funds, under Grants CSD2006-00046, and TIN2009-14475-C04. It was also supported by JCCM funds under grant PEII09-0037-2328 and PII2I09-0045-9916, and the University of Castilla-La Mancha under Project AT20101802. The work presented was carried out by using the VISNET2-WZ-IST software developed in the framework of the VISNET II project.

References

1. ISO/IEC International Standard 14496-10:2003: Information Technology – Coding of Audio – Visual Objects – Part 10: Advanced Video Coding
2. Schwarz, H., Marpe, D., Wiegand, T.: Overview of the Scalable Video Coding Extension of the H.264/AVC Standard. IEEE Transactions on Circuits and Systems for Video Technology 17, 1103–1120 (2007)

3. Girod, B., Aaron, A.M., Rane, S., Rebollo-Monedero, D.: Distributed Video Coding. Proceedings of the IEEE 93, 71–83 (2005)
4. Aaron, A., Rui, Z., Girod, B.: Wyner-Ziv coding of motion video. In: Asilomar Conference on Signals, Systems and Computers, pp. 240–244 (2002)
5. Peixoto, E., Queiroz, R.L., Mukherjee, D.: A Wyner-Ziv Video Transcoder. IEEE Transactions on Circuits and Systems for Video Technology 20, 189–200 (2010)
6. Martínez, J.L., Fernández-Escribano, G., Kalva, H., Fernando, W.A.C., Cuenca, P.: Wyner-Ziv to H.264 Video Transcoder for Low Cost Video Encoding. IEEE Transactions on Consumer Electronics 55, 1453–1461 (2009)
7. Ascenso, J., Brites, C., Dufaux, F., Fernando, A., Ebrahimi, T., Pereira, F., Tubaro, S.: The VISNET II DVC Codec: Architecture, Tools and Performance. In: European Signal Processing Conference, EUSIPCO (2010)
8. Artigas, X., Ascenso, J., Dalai, M., Klomp, S., Kubasov, D., Ouaret, M.: The DISCOVER codec: architecture, techniques and evaluation. In: Picture Coding Symposium (PCS), pp. 1–4. Citeseer (2007)
9. Sachdeva, R., Johar, S., Piccinelli, E.: Adding SVC Spatial Scalability to Existing H.264/AVC Video. In: 8th IEEE/ACIS International Conference on Computer and Information Science (2009)
10. Schwarz, H., Marpe, D., Wiegand, T.: Analysis of Hierarchical B pictures and MCTF. In: IEEE Int. Conf. ICME and Expo. (2006)
11. Vetro, A., Christopoulos, C., Sun, H.: Video Transcoding Architectures and Techniques: An Overview. IEEE Signal Processing Magazine 20, 18–29 (2003)
12. De Cock, J., Notebaert, S., Lambert, P., Van de Walle, R.: Architectures of Fast Transcoding of H.264/AVC to Quality-Scalable SVC Streams. IEEE Transaction on Multimedia 11, 1209–1224 (2009)
13. Van Wallendael, G., Van Leuven, S., Garrido-Cantos, R., De Cock, J., Martinez, J.L., Lambert, P., Cuenca, P., Van de Walle, R.: Fast H.264/AVC-to-SVC transcoding in a mobile television environment. In: Mobile Multimedia Communications Conference, 6th International ICST (2010)
14. Al-Muscati, H., Labeau, F.: Temporal Transcoding of H.264/AVC Video to the Scalable Format. In: 2nd Int. Conf. on Image Processing Theory Tools and Applications (2010)
15. Garrido-Cantos, R., De Cock, J., Martínez, J.L., Van Leuven, S., Cuenca, P., Garrido, A., Van de Walle, R.: Video Adaptation for Mobile Digital Television. In: IFIP Wireless and Mobile Networking Conference (2010)
16. Ascenso, J., Brites, C., Pereira, F.: Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding. In: Speech and Image Processing, Multimedia Communications and Services, EURASIP (2005)
17. Sullivan, G., Bjøntegaard, G.: Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low-Resolution Progressive-Scan Source Material. In: ITU-T VCEG, Doc. VCEG-N81 (2001)
18. Joint Video Team JSVM reference software, Version 9.19.3

Place Recognition via 3D Modeling for Personal Activity Lifelog Using Wearable Camera

Hazem Wannous[1,2], Vladislavs Dovgalecs[1],
Rémi Mégret[1], and Mohamed Daoudi[2]

[1] IMS, UMR 5218 CNRS, University of Bordeaux, Talence, France
[2] LIFL, UMR 8022, University of Lille, Telecom Lille 1, Villeneuve d'Ascq, France
{hazem.wannous,mohamed daoudi}@telecom-lille1.eu,
{vladislavs.dovgalecs,remi.megret}@ims-bordeaux.fr

Abstract. In this paper, a method for location recognition in a visual lifelog is presented. Its motivation is the detection of activity related places within an indoor environment to facilitate navigation in the lifelog. It takes advantage of a camera mounted on the shoulder, which is primarily designed for the behavioral analysis of Instrumental Activities of Daily Living (IADL). The proposed approach provides an automatic indexing of the content stream, based on the presence in specific 3D places related to instrumental activites. It relies on 3D models of the places of interest that are built thanks to a lightweight semi-supervised approach. Performance evaluation on real data show the potential of this approach compared to 2D only recognition.

1 Introduction

First person audio-visual sensing has recently emerged as a way to record the actions and location of a person within a visual life-log in order to provide data for activity and behaviour monitoring as well as memory aid. Our work is motivated by the application to medical behaviour diagnosis, where such technology enables medical practitioners to enrich their study of the behavior of subjects at home in their ecological environment. In particular, for early diagnosis of dementia is still a great challenge to prevent insecurity and health worsening in aged people living at home. Generally, diagnosis of possible dementia is asserted by comparing evidences, added to an autonomy decline. This autonomy decline is frequently assessed by Activities of Daily Living (ADL) interviews in which impairments are known to be related to cognitive decline caused by dementia. The approach developed in this article is related to the paradigm of the IMMED project, where wearable video is used to capture the Instrumental Activities of Daily Living (IADL) of a patient in detail, to be further reviewed or analyzed by medical experts [13].

To efficiently browse such data, content based indexing is required. Location is a powerful information that can be then used as input for activity inference. In [3], Conaire et al. analyzed the performance of 2D local feature matching for place recognition in visual lifelogs. Most of the existing work for visual lifelog

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 244–254, 2012.

indexing has indeed focused on 2D approaches using image classification based on global signatures such as Bag-of-Features [14,5], possibly refined by direct image to image matching based on local 2D features such as SIFT or SURF keypoints [9].

In this paper, we will evaluate the gain of using a 3D model of the places of interest for detecting events of interest within a visual lifelog captured by a wearable wide-angle camera. The method we propose instead leverages the reconstruction of 3D models of the places of interest using Structure from Motion (SfM) techniques and then apply 2D to 3D matching in order to estimate the accurate pose of the query image with respect to the 3D scene models. This approach requires a one-time bootstrap acquisition for the modeling of places of interest, which are in limited number in a home. The gain lies in the possibility that the estimated 3D localization can be further analyzed to extract activities related location events that are used to index the lifelog, in a more precise way than pure image recognition.

The paper is organized as follows. In section 2, we review works related to the location detection problem and position our solution. In section 4, the wearable camera setup and general architecture of the system will be introduced. In section 4, the methodology used for building the environment model will be explained. In section 5, this model will be used to localize the person and provide event detection. In section 6, the results of the proposed methods on representative situations will demonstrate the usefulness of the approach, and show the gain compared to frame based 2D recognition.

2 Related Works

Life-logging systems aims at performing the concept of digitally capturing daily activities and personal memories for later retrieval [18,3,5]. Doherty et al. [4] augmented streams of Lifelog images with geographic data by including locational information provided by a GPS unit rather than estimated from the visual content. Torre et al. [19] recorded the actions and displacements of several subjects using fixed cameras, motion capture, inertial sensors and head-worn narrow angle cameras, that can provide precise data on the movements and ongoing instrumental activites in a restricted place. We consider in our indoor context that the only data available comes from the wearable imaging device, which allows light-weight deployment, by not requiring to equip the house with smart-home sensors, such as cameras, presence sensors, radio-frequency RFID or Wi-Fi beacons. GPS is also not available, due to multipath and fading signals.

Under these constraints, Blighe and O'Connor [2] described a framework for recognizing real-world locations from passively captured images by the Microsoft SensCam sensor. They have classified each image scene into a number of event categories using image keypoints descriptors (SIFT), providing a simple tool allowing the user to annotate the image sets based on user events. Kang et al. [9] proposed to refine the local feature based recognition using Re-Search, to decrease ambiguity in environments with visual aliasing such as offices. Sundaram

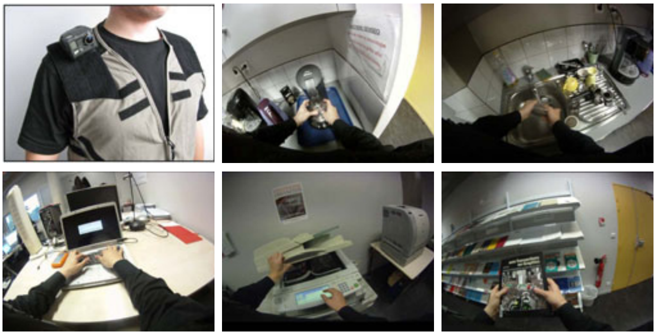


Fig. 1. Wearable camera and examples of places related to instrumental activities

et al. [18] extended image based lifelogging devices to the analysis of instrumental activity at video framerate. They infer location and manipulation activities using a Dynamic Bayesian Network, estimating location using the passage at doors and objects manipulations. Wearable video rate camera was also used by Dovgalecs and al. [5] who applied a Bag-of-Features approach to classify images according to predefined rooms. Kourogi et al. [10] use the fusion of 3D localization based on the alignement to geo-registered images of the environment and inertial sensors.

The work of Snavely et al. [17] have shown the possibility of estimating 3D models of scenes in a almost automatic way, thus enabling the creation of reference 3D models without the burden of geo-referencing the full environment. Its use for location recognition presents the advantage to necessitate only one model, instead of multiple images, as noted by Irschara et al. [8] who proposed outdoor localization based on 2D to 3D matching. Our work considers lifelogs taken with wearable video cameras, and evaluates the fitness of the 3D approach to extract place related events related to instrumental activities.

3 Overview of the System

3.1 Wearable Camera Setup

For a wearable camera to capture the instrumental activities with low motion, we opted for positioning the camera on the shoulder as in [18] and [13], integrated in a ergonomic clothing that is comfortable to the user. As the device is going to be used in capturing both the widest field of view of activities and the general context of the action, we chose to equip our prototype with a camera featuring a Fisheye lens with an effective diagonal angle of 150°and HD resolution (1280x960 pixels). This setup allows us to capture both the instrumental activities near the body and the environment as illustrated in Figure 1.

The captured images cover a very wide field of view, which help providing more varied matches for the image to image correspondences and thus improve

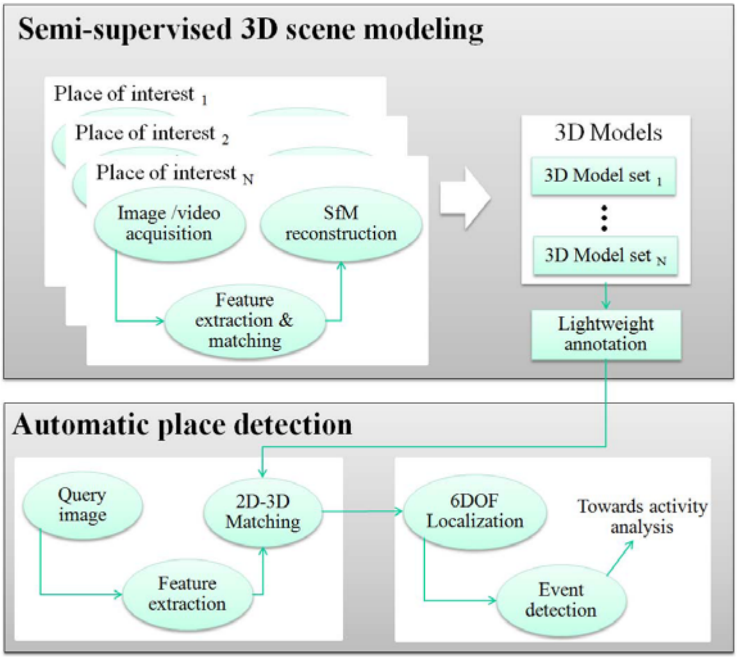


Fig. 2. Overview of the conceptual framework

the robustness of the SfM reconstruction. On the downside, radial distortion in such lens is usually very large and has to be taken into account. For this purpose, we calibrated the camera according to an omnidirectional camera model adapted for very wide angle lenses [15].

3.2 Global Architecture

The system architecture can be divided into two main modules: a semi-supervised 3D modeling of the places of interest and an automatic indexing of the lifelog (Figure 2). The first module deals with the recording of the image for 3D reconstructions and the generation of the 3D structures of the places of interest using SfM. Furthermore, we describe how to generate metric representation of the scenes and annotate the place of interest related to specific IADLs. The second module is composed of an automatic estimation of the 6 degrees of freedom (dof) with respect to the current place of interest, which is followed by an analysis in terms of events.

4 3D Scene Modeling

4.1 3D Reconstruction

We describe here the SfM-based reconstruction process. In orrder to have a lightweight protocol, the environment model has to be built using short training sequences in a way that does not require complex manipulations while still providing good recognition rates in the lifelog. The analysis is focused on the places of interest (kitchen working plane, in front of the library, ...), in order to index the video with respect to the presence in predefined activity zones, which is suitable for a higher level analysis of the IADLs. Since we are interested mainly in some places of interest, and not in the complete environment, the reconstructions of each place is done independently.

Initially, our system assumes multiple training image for the reconstruction in the form of several 2D views of a place of interest taken from different angles and grabbed from video sequence. This is easily done by moving lateraly in front of the places of interest, which has to be done only once for each place, while wearing the camera device.

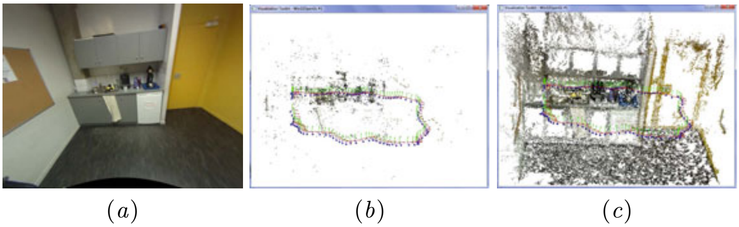
To extract the geometry between camera views, we need to match points between those views, which is possible using the SIFT local features [12] that exhibit good invariance properties. The captured images contain typically between 800 and 2000 SIFT keypoints. The SIFT features from image pairs are matched by considering the Euclidian distances between their 128-dimensional descriptors using the Approximate Nearest Neighbors / kd-tree package of [1]. Once an initial matching is obtained, the 3D reconstruction is done using the method proposed in [17]. The fundamental matrix for each pair images is robustly estimated using the RANSAC method [6]. During each RANSAC iteration, a candidate fundamental matrix is generated using the eight-point algorithm, followed by a non-linear refinement yielding a subset of geometrically consistent matches which will be chosen as an input to a SfM recovery algorithm in an incremental scheme. For each new camera, the back-projection error is minimized in order to calculate poses and 3D scene pointclouds by a generic Bundle Adjustment method, based on the sparse Levenberg-Marquardt algorithm [11].

Each model is represented as a sparse 3D point clouds with associated invariant visual descriptors. The initial training images are discarded in order to keep only a simplified representation of each place. Figure 3 shows the output of the SfM reconstruction applied on a sequence of 108 views captured from the kitchen place.

4.2 Annotation of the Places of Interest

Since the 3D point clouds obtained by SfM reconstruction are generally sparse, the scenes may be difficult to recognize directly of this model. To facilitate the annotation we use the Patch-based Multi-View Stereo method [7], which provides a denser 3D point cloud reconstruction from the sparser set of matched keypoints and estimated camera positions (see Figure 3). Although the sparser model is sufficient for the automatic estimation of location, the alignment between the two models being known.

The annotation of each place consists in defining the reference position in the related zone of interest; for example define the PC position in the office model, the working area in the kitchen, etc.



(*a*) (*b*) (*c*)

Fig. 3. Output of 3D reconstruction: (*a*) distorsion compensated image of the kitchen working area (*b*) sparse 3D model and camera poses obtained by SfM (108 cameras & 5741 points) (*c*) denser 3D point cloud obtained by PMVS (101198 points) for the interactive annotation of the environment.

First, a scale metric is applied manually to all reconstructed models using a graphical interface. Then, within each place model, the reference positionis directly selected in the 3D scene by clicking in the point cloud structure determining the center of gravity of the object. Another way to annotate an object consists in dragging a 2D box around a region of the current image containing this object; the center of the selected 3D points corresponds to the reference point of the corresponding place. Our approach has been applied to video capture, but it can be fully applicable to data captured at lower framerate, like those of [3], provided that the places of interest are captured from different points of view in order to reconstruct the reference 3D models.

5 Automatic Place Detection

5.1 3D Localization Using Natural Landmarks

Once the sparse 3D model has been annotated, it can be used in the automatic chain to index the video lifelog. SIFT keypoints are extracted from each query frame in the lifelog and compared to the models using robust 2D-3D matching, in a similar way as appending a new image to the 3D model [17]. Matching is first done on the SIFT descriptor features. The projective model is then used in the RANSAC framework for geometric verification. The estimation of the accurate pose of the query image with respect to the 3D scene models is then done iteratively. Tracking the natural landmarks of the environment by 2D-3D matching allows us to estimate a 6 dof trajectory in free movement of the person with respect to the reference models. These trajectories are then analyzed to detect events of interest.

5.2 Detection of Location Events

We aim to detect location events related to places that are meaningful with respect to the IADLs. The trigger of a location event of a place of interest depends primarily on the localization relative to the reference point associated to each place. If the current 2D frame image is matched with a 3D place model, thresholding the distance between the camera and the predefined reference point delineates areas defined as: close (manipulation zone), intermediate (approaching) and far (seeing the place, but too far to do instrumental activities). Since the wearable camera is located on the shoulder, just above the arm associated with the dominant hand, we use the camera position directly to evaluate this distance, as we consider that the distance between the camera and a point in the environment is representative of the distance for manipulation. In the current state, only the presence in the close zone is considered to trigger a location based event, although information is available to define additional types of events. The frame based classification is segmented into event intervals using connected components of consecutive same class frames. Each frame is therefore associated to zero or one event. Each event corresponds to a temporal interval representing the arrival, stay and exit in a specific place.

6 Experimental Evaluation

6.1 Dataset and Methods

For the evaluation of our approach we used a lifelog of 36 minutes (68047 frames at 30fps). A volunteer weared the camera on the shoulder with free movements. The activities of the subject are organized around 6 places of interest: 1) sitting at the PC, 2) in front of the library, 3) using the copier, 4) in the lounge, 5) fetching items in the office closet, 6) preparing food in the kitchen. The lifelog was manually annotated for the events of interest with respect to these places, which corresponds to 42 temporal intervals of interest, separated by transition intervals with displacement or activities not related to the places of interest. Instances of these events are shown in Figure 1.

The 3D models of the places of interest were obtained by applying the SfM method, presented in section 4, to six very short sequences shot with the wearable camera and lasting 3 to 5 seconds each. An example of the result of our indexing method is shown with the associated groundtruth in Figure 4; it can be seen that the overall structure of the video with respect to place detection is correctly estimated, which we now evaluate more precisely.

We compared the proposed 3D method to the standard approach based on 2D correspondences [16,3]. It uses the same 2D-2D pairwise matching as in the 3D reconstruction step, with outlier exclusion using the RANSAC procedure applied to fundamental matrix estimation. The temporal majority vote filtering was also applied on this approach. In the 2D approach, a query frame is matched to the class of the reference images, used in the construction of reference 3D models, for which it has the largest number of inlier matches. When the number of inliers is below a threshold of 100 inliers (best parameter found empirically), the frame is considered rejected.

6.2 Evaluation

The evaluation was done both at frame and event levels. The output of the location recognition module is an estimated location vector f that maps each

Table 1. Performances of 2D and 3D approaches for place recognition. Temporal window 5s.

	2D method		3D method	
Framewise global *Accuracy*	0.368		0.627	
Framewise class based $Prec_i$, $Recall_i$	Precision	Recall	Precision	Recall
1 - PC	0.018	0.232	0.403	0.934
2 - library	0.453	0.522	0.799	0.613
3 - copier	0.338	0.723	0.682	0.907
4 - lounge	0.053	0.038	0.682	0.832
5 - office	0.066	0.270	0.877	0.511
6 - kitchen	0.333	0.687	0.575	0.928
Framewise global *Prec* /*Recall*	0.451	0.2168	0.865	0.575
Eventwise global $Prec^{evt}$ / $Recall^{evt}$	0.520	0.190	0.814	0.523

frame $n \in \{1, \ldots, N\}$ either to a class of interest $f(n) > 0$ or to the reject class $f(n) = 0$. We also denote by $g(n)$ the corresponding ground truth location. Each frame can belong to one of the following cases; correct classification can be either a True Positive (TP : $f(n) = g(n) > 0$) or True Negative (TN : $f(n) = g(n) = 0$); incorrect classification can be either False Negative (FN : $f(n) = 0,\ g(n) > 0$), False Positive (FP : $f(n) > 0,\ g(n) = 0$), or, due to multiclass classification, Incorrect Positive (IP : $f(n) > 0,\ g(n) > 0,\ f(n) \neq g(n)$). The global accuracy metric is defined as the ratio of true positive frames to the total number of frames *Accuracy* = #TP/N. Since we are dealing with data that contain a large amount of reject class, we also use a multiclass global precision defined as $Prec = \#TP/(\#TP + \#FP + \#IP)$, and recall defined as $Recall = \#TP/(\#TP + \#FN + \#IP)$, Class specific precision $Prec_i$ is defined by considering only frames such that $f(n) = i$, class specific recall $Recall_i$ is defined by considering only frames such that $g(n) = i$.

An event E is defined as an interval of consecutive frames associated to the same class. An estimated event is considered correct if it overlaps on more than 50% of its length groundtruth events of the same class; a groundtruth event is considered retrieved if it overlaps on more than 50% of its length estimated events of the same class. Each estimated event can therefore be evaluated as True Positive or False Positive, yielding event based Precision $Prec^{evt} = \#TP^{est\_evt}/(\#TP^{est\_evt} + \#FP^{est\_evt})$. Each groundtruth event can be either True Positive or False Negative, yielding event based Recall $Recall^{evt} = \#TP^{gt\_evt}/(\#TP^{gt\_evt} + \#FN^{gt\_evt})$.Performance in place recognition was assessed on the precision, recall and accuracy rates at frame level, and precision and recall at event level (see Figure 5). The confusion matrix associated to the rate of correct detection for each place is given in Figure 6.

On all measured metrics, the proposed 3D approach outperforms the 2D approach. The additional constraints and completeness stemming from the use of

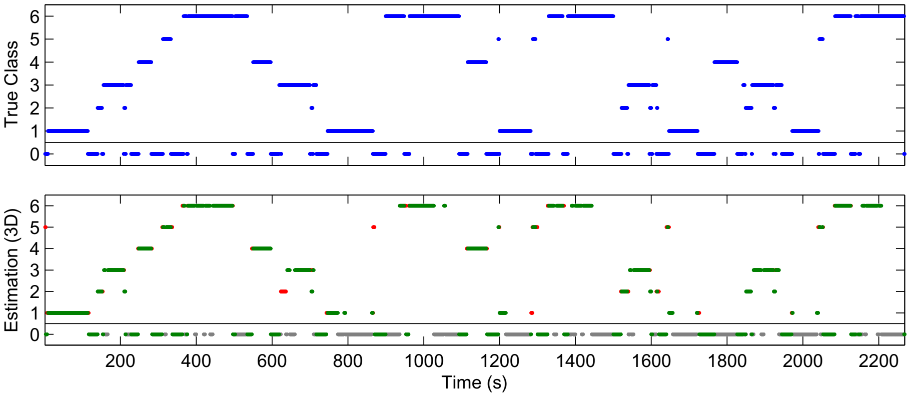


Fig. 4. Chronogram on the test lifelog. Top: true place. Bottom: estimated place using the proposed method (temporal window of 5s). Color-code: green=correct, red=incorrect, gray=false negative.

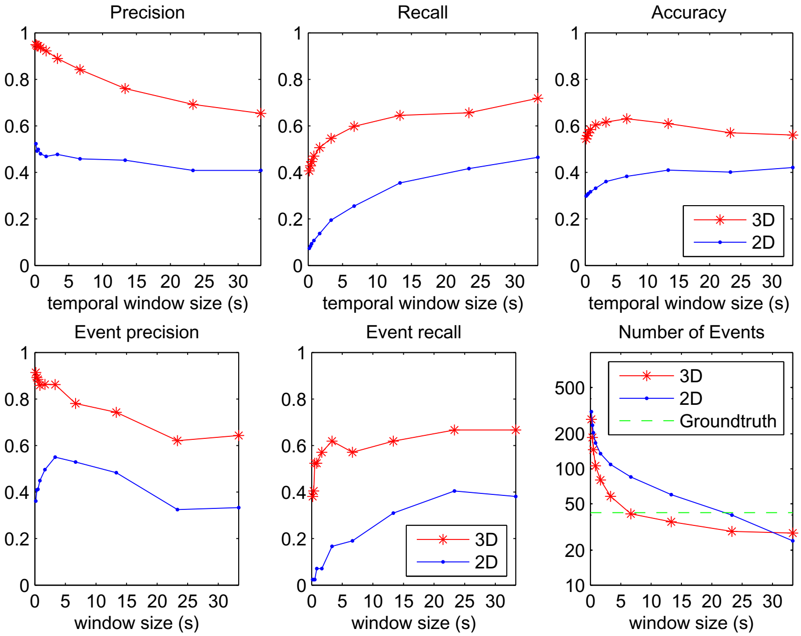


Fig. 5. Evolution of Precision, Recall, Accuracy and Number of Events as a function of the temporal window size. (Top) Frame-based metrics (Bottom) Event-based metrics.

3D models merging information from several reference images therefore help exclude cases that are incorrectly accepted by the 2D approach while providing a better recall that using a separate comparison with each reference frame. This is observed on classwise as well as global metrics (Table 1). This improvement is

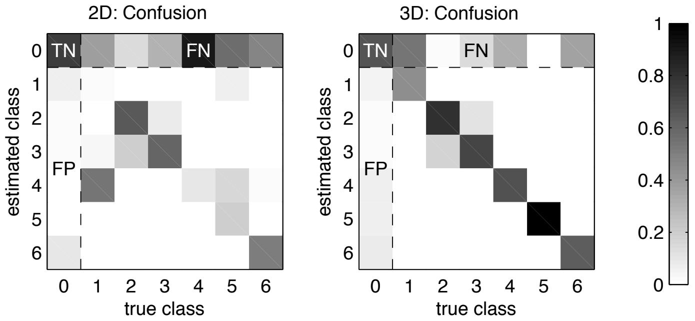


Fig. 6. Frame-based confusion matrix of place recognition for (left) 2D and (right) 3D based matching, both with temporal window of 5s

stable with respect to the choice of the temporal window (Figure 5). In particular, we can notice a sharp increase of the event recall for short-term temporal windows (Figure 5, bottom-center) which shows that some scattered false negatives, probably due to temporary occlusion or motion blur can be compensated. A longer temporal window is required when using the 2D approach. These observations can also be related to the lower number of events detected by the 3D approach (Figure 5, bottom-right), that corresponds to less oversegmentation. This allows the use of shorter temporal windows for postprocessing, thus generating less artifacts due to the temporal regularization.

7 Conclusion

In this paper, we have addressed the need of providing event based place detection for behavior monitoring in visual lifelog, by exploiting the same camera sensor that is used for observing the instrumental activities: a wearable camera fixed on the person shoulder. The proposed system is based on 3D models obtained using SfM techniques. It was shown suitable for the detection of situations of interest for the analysis of IADL, and improves the false alarm rate compared to purely visual recognition based approaches. The current vision only system is dedicated to video monitoring in an indoor home environment and showed promising results. We intend to expand the experimental part to a larger set of users, and extend the approach by hybridizing with inertial sensors in order to improve the accuracy of the localization and its robustness to occlusion and motion blur, which is the subject of future work.

Acknowledgement. This work is partly supported by a grant from the ANR (Agence Nationale de la Recherche) with reference ANR-09-BLAN-0165-02, within the IMMED project. It was initiated during the stay of M. Hazem Wannous at IMS.

References

1. Arya, S., Mount, D.M., Netanyahu, N.S., Silverman, R., Wu, A.Y.: An optimal algorithm for approximate nearest neighbor searching in fixed dimensions. Journal of the ACM 45(6), 891–923 (1998)
2. Blighe, M., O'Connor, N.: Myplaces: Detecting important settings in a visual diary. In: ACM International Conference on Image and Video Retrieval, Niagara Falls, Canada, July 7-9 (2008)
3. Conaire, C.Ó., Blighe, M., O'Connor, N.E.: SenseCam Image Localisation Using Hierarchical SURF Trees. In: Huet, B., Smeaton, A., Mayer-Patel, K., Avrithis, Y. (eds.) MMM 2009. LNCS, vol. 5371, pp. 15–26. Springer, Heidelberg (2009)
4. Doherty, A., O'Conaire, C., Blighe, M., Smeaton, A., O'Connor, N.: Combining image descriptors to effectively retrieve events from visual lifelogs. In: ACM Multimedia Information Retrievaly, Vancouver, Canada, October 30-31 (2008)
5. Dovgalecs, V., Mégret, R., Wannous, H., Berthoumieu, Y.: Semi-supervised learning for location recognition from wearable video. In: CBMI (2010)
6. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24(6), 381–395 (1981)
7. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. IEEE Trans. on PAMI 32, 1362–1376 (2010)
8. Irschara, A., Zach, C., Frahm, J.-M., Bischof, H.: From structure-from-motion point clouds to fast location recognition. In: CVPR, pp. 2599–2606 (2010)
9. Kang, H., Efros, A., Hebert, M., Kanade, T.: Image matching in large scale indoor environment. In: First Workshop on Egocentric Vision (2009)
10. Kourogi, M., Kurata, T.: A method of personal positioning based on sensor data fusion of wearable camera and self-contained sensors. In: IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems, pp. 287–292 (2003)
11. Lourakis, M., Argyros, A.: The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. Technical Report 340, Inst. of Computer Science-FORTH, Heraklion, Crete, Greece (2004), www.ics.forth.gr/~lourakis/sba
12. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)
13. Mégret, R., Dovgalecs, V., Wannous, H., Karaman, S., Benois-Pineau, J., El Khoury, E., Pinquier, J., Joly, P., André-Obrecht, R., Gaëstel, Y., Dartigues, J.: The IMMED project: wearable video monitoring of people with age dementia. In: ACM Multimedia, Firenze, Italy, pp. 1299–1302 (2010)
14. Nister, D., Stewenius, H.: Scalable recognition with a vocabulary tree. In: CVPR, vol. 2, pp. 2161–2168 (2006)
15. Scaramuzza, D., Martinelli, A., Siegwart, R.: A flexible technique for accurate omnidirectional camera calibration and structure from motion. In: ICVS (2006)
16. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: ICCV, vol. 2, pp. 1470–1477 (2003)
17. Snavely, N., Seitz, S.M., Szeliski, R.: Modeling the world from internet photo collections. IJCV 80(2), 189–210 (2007)
18. Sundaram, S., Mayol-Cuevas, W.: High level activity recognition using low resolution wearable vision. In: First Workshop on Egocentric Vision (2009)
19. Torre, F., Hodgins, J., Montano, J., Valcarcel, S.: Detailed human data acquisition of kitchen activities: the cmu-multimodal activity database. In: HCI Workshop (2009)

Sensor-Based Analysis of User Generated Video for Multi-camera Video Remixing

Francesco Cricri[1], Igor D.D. Curcio[2], Sujeet Mate[2],
Kostadin Dabov[1], and Moncef Gabbouj[1]

[1] Department of Signal Processing, Tampere University of Technology, Tampere, Finland
{Francesco.Cricri,Kostadin.Dabov,Moncef.Gabbouj}@tut.fi
[2] Nokia Research Center, Tampere, Finland
{Igor.Curcio,Sujeet.Mate}@nokia.com

Abstract. In this work we propose to exploit context sensor data for analyzing user generated videos. Firstly, we perform a low-level indexing of the recorded media with the instantaneous compass orientations of the recording device. Subsequently, we exploit the low level indexing to obtain a higher level indexing for discovering camera panning movements, classifying them, and for identifying the Region of Interest (ROI) of the recorded event. Thus, we extract information about the content without performing content analysis but by leveraging sensor data analysis. Furthermore, we develop an automatic remixing system that exploits the obtained high-level indexing for producing a video remix. We show that the proposed sensor-based analysis can correctly detect and classify camera panning and identify the ROI; in addition, we provide examples of their application to automatic video remixing.

Keywords: Sensor, compass, video, analysis, indexing, multi-camera.

1 Introduction

In recent years there has been a rapid convergence between Internet services and mobile technologies. Internet services are increasingly becoming more socially oriented, often allowing people to publish and share media files. Due to the easy portability of camera-enabled mobile phones, video recording with mobile phones has become one of the most popular means for capturing videos of interesting and unanticipated events. One of the most popular architectural patterns in the Web 2.0 is the *Participation-Collaboration* pattern [1], in which each user of a web service collaborates for achieving a certain task on a topic that is of mutual interest, by contributing with a small amount of information. A typical example of such a pattern is *Wikipedia*. This concept could be extended to the video domain by combining various user generated videos (recorded for example at the same public happening) in order to generate a video remix, i.e., a succession of video segments extracted from the contributing videos. When the amount of source videos is large an automatic approach for generating the remix would be preferable. Previous work has mainly concentrated on video content analysis for achieving this task, as we will discuss in Section II. However, content

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 255–265, 2012.

analysis (that needs to be applied to each video clip) typically requires high computational costs and does not always provide the necessary semantics. Alternatively, data from sensors embedded in modern mobile phones (such as electronic compass, accelerometer, gyroscopes, GPS, etc.) can be captured simultaneously with the video recording in order to obtain important context information about the recorded video.

In this paper, we present a set of novel sensor-based analysis methods for indexing multimedia content. We then use the indexing results for automatically generating a video remix of an event for which a number of users have recorded videos by using their mobile phones. The contribution of this work is the use of sensor data in order to detect camera panning and its speed. Also, we propose a sensor-based analysis for identifying the Region of Interest (ROI) of the whole event, to be applied to those cases in which a ROI exists (such as in live music shows). In this sense, we gather information about the content without performing content analysis. Instead, we use sensor data analysis, which is computationally less demanding. Section 2 introduces the related work on the subject; Section 3 describes the proposed automatic video remix system; Section 4 describes the proposed sensor-based indexing methods; Section 5 presents the experimental results; finally, Section 6 concludes the paper.

2 Related Work

The field of automatically summarizing video content has been studied quite intensively in the last fifteen years. Most of this work is based on the analysis of video and/or audio content (thus being computationally demanding) for video captured by a single camera [2], and also for multi-camera setups [3-5]. In [6], a case study that compares manual and automatic video remix creation is presented. In [7] a video editing system is described, which automatically generates a summary video from multiple source videos by exploiting context data limited to time and location information. Camera motion is determined in some automatic video editing systems either for detecting fast motion which can cause blurry images [8], or for detecting scene boundaries [9]. These methods are all based on video content analysis. Shrestha et al. [10] propose an automatic video remix system in which one of the performed analysis steps consists of estimating suitable cut-points. A frame is considered suitable as a cut-point if it represents a change in the video, such as at the beginning or end of a camera motion. Another interesting work is presented in [11], in which the authors analyze the content of user generated videos to determine camera motion, by assuming that people usually move their cameras with the intent of creating some kind of "story". In the work presented in this paper, we exploit the camera motion within an automatic video editing framework by using sensors embedded inside modern mobile phones, without the need of analyzing the video content as traditionally done in early research. In [12] the authors detect interesting events by analyzing motion correlation between different cameras. Finally, in [13] the authors exploit compass sensors embedded in mobile phones for detecting "group rotation", in order to identify the presence of something of interest. In contrast, we propose to analyze compass data from each mobile phone in order to detect and classify individual camera panning

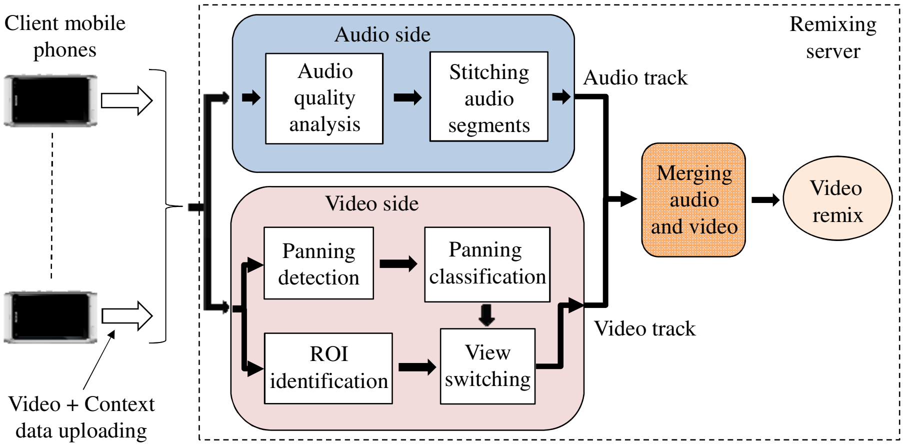


Fig. 1. Overview of the automatic video remixing system

movements, and also to identify the ROI of the recorded event. In this way, we are able to consider view changes performed by each user within the ROI, and also to account for the different types of camera motion which may have different semantics (e.g., a slow panning might have been performed to obtain a panoramic view).

3 Automatic Video Remixing System

The proposed automatic video remixing (or editing) system is implemented as a distributed system based on a client-server architecture (an illustration is given in Fig.1).

3.1 Context-Sensing Client

The client side of the system consists of the mobile camera device. We have developed a client application that enables to simultaneously record video and context information captured by the embedded sensors such as electronic compass, GPS, and accelerometer. These sensors are used for indexing the video content while it is being recorded, by storing both sensor data and the associated timestamps. In this work we exploit the electronic compass. This sensor (usually implemented in mobile phones by embedding two or three magnetometers) measures the orientation of the device with respect to the magnetic North (which is assumed to be at orientation equal to zero). When the camera is started, the client application (which runs as a background process) starts capturing data from the electronic compass. The sensor readings are sampled at a fixed sampling rate of 10 samples/second. The recorded compass data can be regarded as a separate data stream. Each sample is a time-stamped value representing the orientation of the camera with respect to the magnetic North. By using the timestamps, any orientation sample can be uniquely associated to a certain segment of video within the recorded media, and thus it is possible to understand the direction towards which the camera was recording at a particular time interval.

3.2 Automatic Video Remixing Server

The server of the automatic video remixing system exploits the availability of various user generated videos of the same event and the sensor-based low-level indexing performed on each client device by the context-sensing client. This system allows people who have recorded videos at the same event to collaborate in order to get an automatically generated video remix, which consists of segments extracted from the recorded videos. The participating users upload their videos to the server along with the associated context files. Firstly, all the source videos must be synchronized to each other so that they can be placed on a common timeline. This is achieved by means of a global clock synchronization algorithm such as the Network Time Protocol [14]. Regarding the audio side, the audio track consists of a succession of best quality audio segments extracted from the source videos. The audio analysis itself is not the focus of the work presented in this paper.

Regarding the visual side, the remix will consist of a sequence of video segments present in the original source videos. We group the criteria for switching view in two different categories: sensor-related criteria and temporal criteria. The sensor-related criteria are all based on compass data, and they include the camera panning detection, the classification of camera panning based on speed, and the identification of the Region of Interest. Thus, there is a need to perform a higher-level indexing at the server side, which uses the low-level indexing data (i.e., the instantaneous compass orientations) in order to detect and classify the camera panning and identify the ROI. The considered temporal criteria are lower-bound and upper-bound temporal thresholds. The timing for switching view and the specific videos to be used at each switching instant are decided by jointly evaluating the mentioned temporal and sensor based criteria. The lower-bound temporal threshold is used to avoid that two consecutive view switches (triggered by the sensor-related criteria) happen within a too short temporal window. If no view switches happen for a time interval greater than the upper-bound threshold, a switch to a different view is forced.

4 Sensor-Based Analysis

In the following sub-sections we provide a detailed description of how the analysis of the sensor data is performed to assist in generating semantic information from the compass data, and thus obtaining information about the content without analyzing it directly.

4.1 Camera Panning Detection

Camera panning is the horizontal rotational movement of a camera which is commonly used either for following an object of interest or for changing the focus point (or direction) of the recording. The most common techniques for detecting a camera panning or other camera motion are based on video content analysis (for example as described in [9] and [11]). In an automatic video editing system using a multi-camera setup there is a need to understand when to perform a view switch from one source

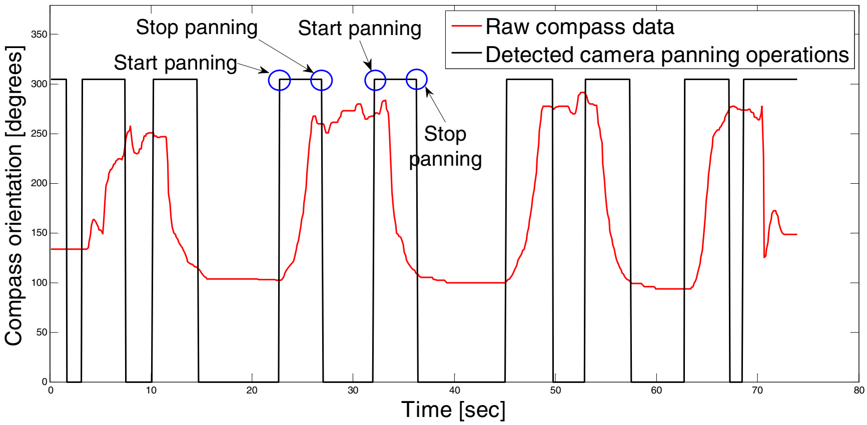


Fig. 2. Detection of camera panning movements

video to another, and also which specific video to use after the view switching. One of the criteria that we consider is based on the fair assumption that when a user intentionally performs a camera panning, then the obtained view is likely to be interesting. In fact, as also stated in [11], camera motion is an indicator of the camera user's interests in the scene and can also attract the viewer's attention. One reason for considering a panning as interesting is that it is performed to include something of interest. Also, it is unlikely that the view obtained after the panning will be exactly the same (for view-angle and position of the camera) as any of the views provided by the other cameras; thus, a new view of the scene is available to the system for being included into the remix. We mark all panning movements as potential switching points, which are to be evaluated when deciding how to perform the view switches.

In order to detect the camera panning, we analyze the data captured by the electronic compass during the video recording activity, instead of relying on video content analysis. One advantage of our method over content-based methods is that we analyze the real motion of the camera and not the effects of the camera motion on the recorded video content. Furthermore, motion of objects on the scene is a major obstacle for content-analysis methods. On the contrary, our method is not affected at all by such motions. For detecting a camera panning we perform the following steps:

1. *Apply low-pass filtering to the raw compass data.*
2. *Compute the first discrete derivative over time of the filtered compass signal.*
3. *Select the peaks of the obtained derivative by using a threshold T_P.*

The low-pass filtering in the first step is necessary to avoid obtaining peaks which are due to short or shaky camera movements, rather than to intentional camera motion.

Fig. 2 shows the detection of the camera panning movements by analyzing compass data. Each detected camera panning is represented by two timestamps: start- and stop-panning timestamps.

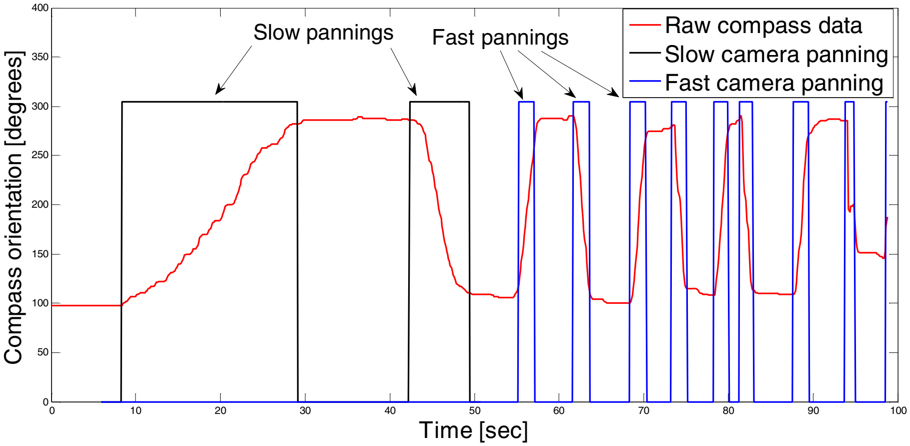


Fig. 3. Classification of camera panning movements based on speed

4.2 Classification of Camera Panning Based on Speed

We classify the detected panning movements with respect to their speed. Each panning can be a slow one, performed by the user with the intention of capturing the scene during the camera motion (e.g., for obtaining a panoramic panning or for following a moving object), or a faster one, which might have been performed for changing the focus point (or direction) of the video recording. This is a very important difference from a video editing point of view, because a too quick camera motion may result into blurry video content and should not be included into the final video remix, whereas a panoramic panning should be included for giving the viewer of the video remix a better understanding of the whole scene. We then classify the panning movements either as slow or as fast if their speed is respectively less or greater than a predefined threshold. Fig. 3 shows a sequence of panning movements and their classification.

4.3 Identifying the Region of Interest

In some use cases it is important to know which area of a public happening is considered as the most interesting by the audience. The approach considered in this work for identifying the ROI exploits the fact that most public events do have a single scene of common interest, for example the performance stage in a live music show (see Fig. 4). Therefore, most of the people who are recording such an event usually point their camera towards the stage, at least for most of the recording time, as it represents the main attraction area. The automatic video remix system identifies the ROI as the specific angular range of orientations (with respect to North) towards which the users have recorded video content for most of the time. The relative location of the users with respect to each other is not taken into account in this work. Instead, the proposed ROI identification method assumes that the stage of the recorded show is a

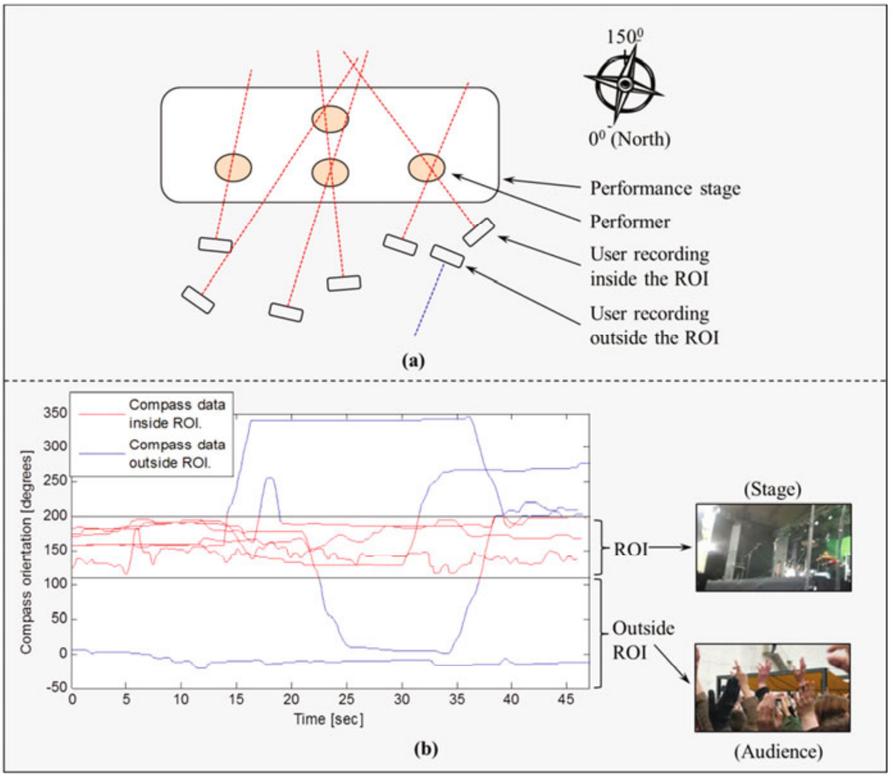


Fig. 4. ROI identification. (a) An example scenario where users use their cameras to record a live music performance. (b) The (unwrapped) compass data captured by seven users while recording a music show for one of our tests.

proscenium stage (i.e., the audience lies on only one side of the stage) which is the most common case at least for live music performances.

The algorithm for ROI identification works as follows:

1. *The preferred angular extent of the ROI is a parameter to be set manually.*
2. *For each recording user, compute the time spent recording in each orientation and update the histogram of recorded seconds over the orientations.*
3. *Analyze the obtained histogram and find the angular range (of user-specified extent) that maximizes the length of video content which has been captured towards such a range. This angular range represents the ROI.*

The extent of the ROI can be set to any reasonable value.

4.4 Use of Panning Movements and ROI for View Switching

The automatic video remix system decides about the timing of the view switching and about which view to select by considering the detected panning movements, their classification and the pointing direction of the cameras with respect to the identified ROI. The detailed description of the view switching algorithm is as follows:

```
FOR each detected panning event:
  IF panning ends inside ROI
    IF panning is classified as slow
      Switch view to the video containing the panning, when the panning starts.
    ELSE (panning is classified as fast)
      Switch view to the video containing the panning, when the panning ends.
  ELSE (panning ends outside ROI)
    No view-switching is performed.
```

A video segment containing a fast panning is not included in order not to have blurry video content in the video remix.

5 Results

An evaluation of the proposed sensor-based multimedia indexing techniques was performed using the following setup. We used a dataset consisting of 47 video recordings spanning an overall time duration of about 62 minutes and captured by nine different users during two live music shows. The users were not aware of the goal of the test. By visually inspecting the recorded video content, we identified and annotated 129 panning movements performed by the recording users. Table 1 summarizes the test results for the panning detection algorithm, and provides the accuracy in terms of precision (*P* – fraction of the detections which are indeed true panning movements), recall (*R* – fraction of the true panning movements which are detected correctly) and balanced F-measure (*F* – computed as the harmonic mean of the precision and recall). As can be seen from the table, our sensor-based panning detection performs well.

Regarding the panning classification, we considered the panning movements that were correctly detected by our detection algorithm. We manually classified them as either slow or fast depending on whether the video content captured during each panning was respectively pleasant or unpleasant to watch. Table 2 summarizes the performance of the proposed panning classification method. Among the 112 correctly detected panning movements, only four are misclassified.

We performed a test for the ROI identification using videos that seven users have recorded at one of the two music shows considered in our dataset. Fig. 4b shows the compass data captured by the users during the show. Six of the users have pointed their cameras towards a similar orientation for most of the recording time. We have specified a preferred ROI extent of 90 degrees. The proposed method identified the ROI to be in the range [110, 200] degrees, which is satisfactory, as it corresponds to orientations pointing towards the stage of the recorded event. As can be seen in Fig. 4b, during the recording session some of the users performed panning movements either inside (for recording the main area of interest) or outside the identified ROI (for

Table 1. Test results for the camera panning detection algorithm. *GT* stands for ground truth (manually annotated panning movements), *TP* for true positives, *FP* for false positives.

GT	*TP*	*FP*	*P*	*R*	*F*
129	112	12	0.90	0.87	0.89

Table 2. Confusion matrix for the classification of panning movements based on speed. Only the correctly detected panning movements (i.e., the true positives) are considered in the table.

	Automatically detected as fast	**Automatically detected as slow**
Manually annotated as fast (ground truth)	25	1
Manually annotated as slow (ground truth)	3	83

recording objects or people that have been found interesting). In particular, one user has been recording all the time something lying outside the ROI. Fig. 5 shows an example of using the proposed sensor-based analysis for automatically generating a video remix. For this experiment, we use the content and context data from two of the users positioned at different viewing angles. For each user, the plot of the compass data captured during the video recording is shown. Some relevant frames extracted from the recorded videos are displayed below the associated compass data. These frames have been extracted at time points indicated by the red arrows (mostly before and after each panning). The bottom-most part of the figure represents the video remix obtained by stitching together segments extracted from two source videos. The switching points are indicated by the vertical dashed lines that originate either from video 1 or video 2, depending on which video has triggered the view-switching. Panning detection, panning classification and ROI identification were performed. The identified ROI was the range [-41, 49] degrees. User 1 performed one slow and one fast panning, and has been recording always within the ROI. User 2 performed four slow pannings. The first and the third panning movements end outside the ROI, the second and the fourth end inside the ROI. At about 25 seconds user 2 turned the camera towards an orientation outside the ROI, thus the system switches view from video 2 (which is the initial state) to video 1. At 55 seconds user 2 turned the camera back towards the ROI, and a view switch to video 2 is performed. At about 62 seconds user 1 performed a slow panning, thus the system uses video 1 from the starting time of the panning. At about 86 seconds, user 2 performed a panning from outside to inside the ROI and a view switch is then triggered. Finally, as user 1 performed a fast panning, the remix will contain a segment of video 1 from the end of the panning.

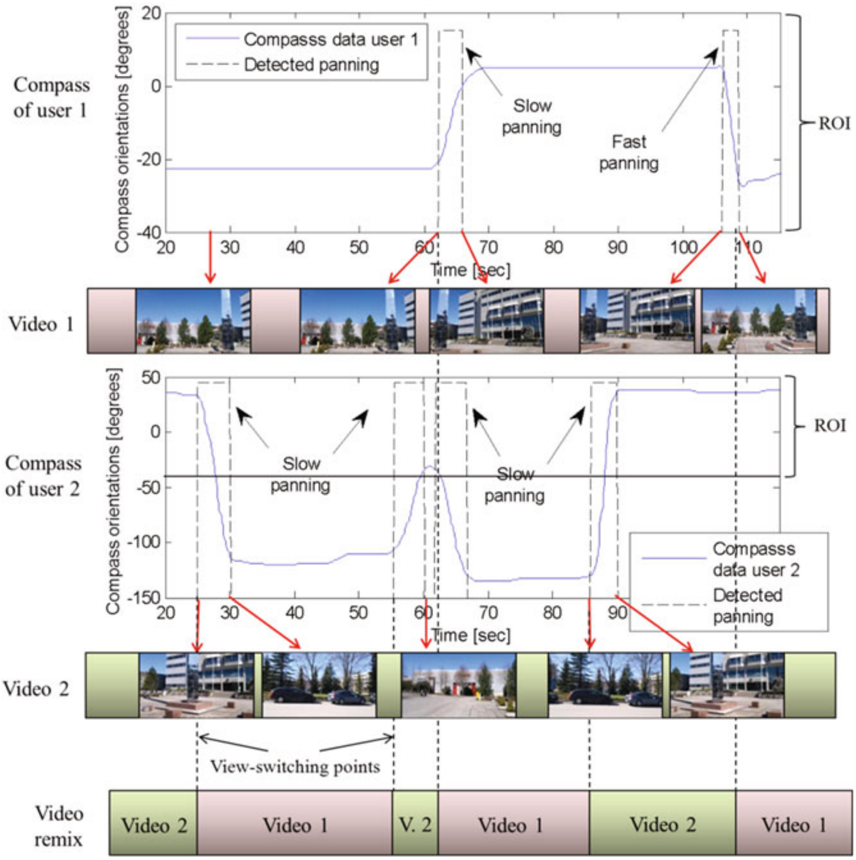


Fig. 5. Example of using the proposed sensor-based multimedia indexing techniques for automatically generating a video remix. Due to prior state, the excerpt of the remix starts with Video 2.

6 Conclusions

In this paper we presented methods for indexing user generated videos based on context sensor data. These methods are used to automate the video remixing process in a multi-camera recording scenario. The novelty of these methods is the use of sensor data for performing the multimedia indexing – thus avoiding computationally costly video-content analysis. In this work we have focused on analyzing compass data from each mobile phone and shown that the proposed methods can correctly detect and classify camera pannings, and identify the ROI of the recorded event. In this way, the system is able to generate a video remix that takes into account the panning movements performed by each user within the Region of Interest during the event. Furthermore, we are able to account for different semantics of camera motion.

References

1. Governor, J., Hinchcliffe, D., Nickull, D.: Web 2.0 Architectures. O'Really Media / Adobe Developer Library (2009)
2. Huang, C.H., Wu, C.H., Kuo, J.H., Wu, J.L.: A Musical-driven Video Summarization System Using Content-aware Mechanisms. In: IEEE International Symposium on Circuits and Systems, Kobe, Japan, vol. 3, pp. 2711–2714. IEEE (2005)
3. Kennedy, L., Naaman, M.: Less Talk, More Rock: Automated Organization of Community-Contributed Collections of Concert Videos. In: 18th International Conference on World Wide Web, Madrid, Spain, pp. 311–320. ACM (2009)
4. El-Saban, M., Refaat, M., Kaheel, A., Abdul-Hamid, A.: Stitching Videos Streamed by Mobile Phones in Real-Time. In: 17th ACM International Conference on Multimedia, Beijing, China, pp. 1009–1010. ACM (2009)
5. Zsombori, V., Frantzis, M., Guimaraes, R.L., Ursu, M.F., Cesar, P., Kegel, I., Craigie, R., Bulterman, D.C.A.: Automatic Generation of Video Narratives from Shared UGC. In: 22nd ACM Conference on Hypertext and Hypermedia, Eindhoven, The Netherlands, pp. 325–334. ACM (2011)
6. Vihavainen, S., Mate, S., Seppälä, L., Cricri, F., Curcio, I.D.D.: We Want More: Human-Computer Collaboration in Mobile Social Video Remixing of Music Concerts. In: ACM CHI Conference on Human Factors in Computing Systems, Vancouver, Canada, pp. 287–296. ACM (2011)
7. Järvinen, S., Peltola, J., Plomp, J., Ojutkangas, O., Heino, I., Lahti, J., Heinilä, J.: Deploying Mobile Multimedia Services for Everyday Experience Sharing. In: IEEE International Conference on Multimedia and Expo, Cancun, Mexico, pp. 1760–1763. IEEE (2009)
8. Foote, J., Cooper, M., Girgensohn, A.: Creating Music Videos using Automatic Media Analysis. In: 10th ACM International Conference on Multimedia, Juan les Pins, France, pp. 553–560. ACM (2002)
9. Peyrard, N., Bouthemy, P.: Motion-based Selection of Relevant Video Segments for Video Summarisation. In: IEEE International Conference on Multimedia and Expo, Baltimore, U.S.A, vol. 2, pp. 409–412. IEEE (2003)
10. Shrestha, P., de With, P.H.N., Weda, H., Barbieri, M., Aarts, E.H.L.: Automatic Mashup Generation from Multiple-camera Concert Recordings. In: ACM International Conference on Multimedia, Firenze, Italy, pp. 541–550. ACM (2010)
11. Abdollahian, G., Taskiran, C.M., Pizlo, Z., Delp, E.J.: Camera Motion-Based Analysis of User Generated Video. IEEE Transactions on Multimedia 12(1), 28–41 (2010)
12. Cricri, F., Dabov, K., Curcio, I.D.D., Mate, S., Gabbouj, M.: Multimodal Event Detection in User Generated Videos. In: IEEE International Symposium on Multimedia, Dana Point, U.S.A. IEEE (2011)
13. Bao, X., Choudhury, R.R.: MoVi: Mobile Phone based Video Highlights via Collaborative Sensing. In: 8th International Conference on Mobile Systems, Applications and Services, San Francisco, U.S.A, pp. 357–370. ACM (2010)
14. Network Time Protocol, Version 4, IETF RFC 5905 (2010)

Gait-Based Action Recognition via Accelerated Minimum Incremental Coding Length Classifier

Hung-Wei Lin[1], Min-Chun Hu[2,3], and Ja-Ling Wu[1,2]

[1] Dept. of CSIE, National Taiwan University, Taipei, Taiwan
[2] GINM, National Taiwan University, Taipei, Taiwan
[3] CITI, Academia Sinica, Taipei, Taiwan
{twinschild,trimy,wjl}@cmlab.csie.ntu.edu.tw

Abstract. In this paper, we present a novel human action recognition approach based on gait energy image (GEI) and minimum incremental coding length (MICL) classifier. GEIs are extracted from video clips and transformed into vectors as input features, and MICL is employed to classify each GEI. We also use multiple cameras to capture GEIs of different views, and the voting strategy is applied after the MICL classification results to improve the overall system performance. Experimental results show that the proposed approach can achieve approximately 95% of accuracy. For practical usage, we also speed up the classification time so that it can be accomplished in a very short time. Moreover, other classification methods are used to classify GEIs and the experimental result shows that MICL is the most suitable classifier for this approach. Besides our recorded action clips, the Weizmann dataset is also used to verify the capability of our approach. The experimental results show that our approach is competitive to other state-of-the-art action recognition methods.

Keywords: Action Recognition, Human Gait, GEI (Gait Energy Image), MICL (Minimum Incremental Coding Length), GPU, Visual Surveillance.

1 Introduction

The analysis of human activities can be applied to a variety of application domains, such as video surveillance, human-computer interaction systems, medical care, anti-terrorism, etc. Thus, recognizing human activities has become an important research topic in the fields of machine learning and computer vision, recently. Many approaches have been proposed for human activity recognition. For example, Bobick et al. [1] extracted motion energy images (MEI) and motion history images (MHI) as their features from aerobic exercise sequences, and used 7 Hu moments to yield reasonable shape discrimination in a translation- and scale-invariant manner. Various distance metrics were then used to separate different movements. They also combined two views of camera information to help improve the efficacy of their method. Hsieh et al. [2] represented the structure of the human body using a skeleton feature and a centroid context feature based on a triangulation-based technique. The key posture of human activities are selected and coded into a set of symbols. Finally, a string-based

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 266–276, 2012.

technique was applied for recognizing human activities. Zou et al. [3] classified human walking sequences with or without a briefcase on an elliptical path. The gait energy images (GEI) [4] was used as their input feature to co-evolutionary genetic programming, and they improved the classifier performance by further applying the strategy of majority voting to the Bayesian classifiers' results. There are also many researches related to GEI. For examples, Han and Bhanu [4] applied GEI to recognize individuals, and Lu et al. [5] conducted human age estimation based on GEI.

In this work, we propose a novel approach based on gait information and the criterion of lossy coding for classifying human actions directly from videos. The details of our approach are presented in Section 2. Section 3 demonstrates the experimental results of the proposed approach. Finally, Section 4 concludes this write-up.

2 Proposed Approach

Fig. 1 shows the block diagram of the proposed approach. First, we extract GEIs for each sequence with specific human activity as ground truth (or training) features. The minimum incremental coding lengths (MICLs) [6] of the training features are calculated and stored as classification references. The extracted GEIs of testing videos are transformed into vectors as inputs of the MICL classifier and a classification result would be returned. For boosting the overall performance, we recorded one human activity video from three different angles at the same time, and a better classification result can be obtained through majority voting from the three classification results. The details of each module will be addressed in the following subsections.

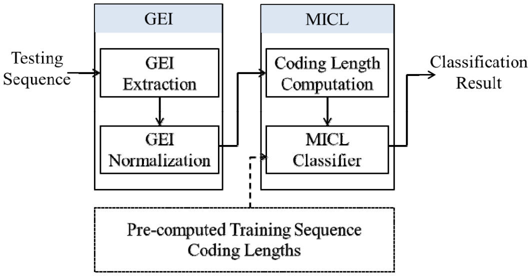


Fig. 1. Block diagram of the proposed GEI-based human activity recognition system

2.1 Gait Energy Image (GEI)

Gait Energy Image. According to our observation, the variations of silhouettes are quite different between distinct actions. Thus, GEI, the well-known representation formed by silhouettes, is an intuitive choice of feature. Moreover, in comparison with the gait representation by binary silhouette sequence, GEI saves both storage space and computation time for recognition and is less sensitive to silhouette noise in individual frames [4]. To design a real-world action classification system using gait cues, GEI is the best choice for its robustness against noise and its efficiency. We extract silhouettes from human activity sequences using running Gaussian average background modeling. Given the binary gait silhouette images, the gray-level GEI is then defined as

$$G(x, y) = \frac{1}{N} \sum_{t=1}^{N} B_t(x, y), \tag{1}$$

where $B_t(x,y)$ represents a gait silhouette image at time t, N is the number of total frames in a sequence, and x and y are values of the 2D image coordinates.

Normalization and Alignment of GEIs. There are two important issues on how to construct a proper GEI: 1) How to proportionally resize silhouette images so that every silhouette image would have a reasonable height/width? 2) How to align every silhouette image with respect to its centroid? In the past, GEI-related researches assumed that silhouette images are extracted only from walking sequences with a side view. Since the walking actions are cyclic, GEIs are easy to be normalized and aligned. However, in our work, silhouettes are extracted from video sequences with a variety of different actions and different angles. In this situation, how to normalize the extracted silhouettes into reasonable sizes and align them to construct a visually-reasonable GEI becomes a challenging issue. Fig. 2 shows GEIs for actions run, sidewalk, and walk formed by silhouettes without any preprocessing. It can be seen that these GEIs are noisy (involving the variation of different object sizes) and confusing (somewhat similar to one another). This fact will pose problems for later human action classifications.

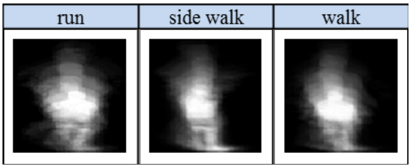


Fig. 2. Original GEIs for actions run, side walk, and walk

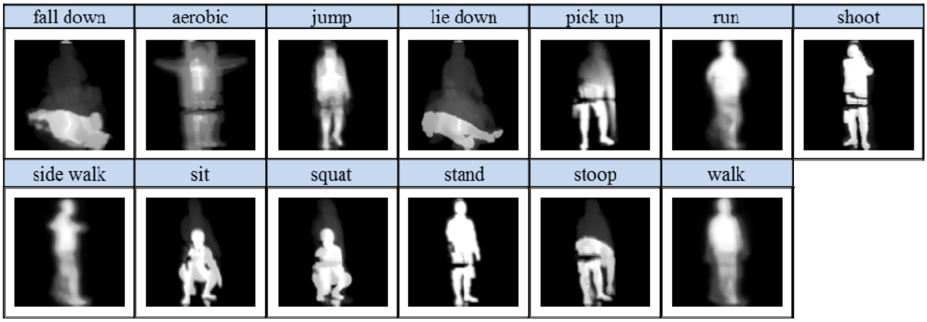


Fig. 3. Normalized GEIs for different actions captured from front-right side of human

Because the object aspect ratio in terms of conventional bounding box may vary a lot in the video sequences of actions like fall down, lie down, etc., we utilize squared bounding boxes to avoid object deformation during silhouettes normalization. Given a binary gait silhouette image, we first compute its squared bounding box. If the width (height) of this bounding box is the largest in this sequence until then, the width (height) is stored and we bind the object with this box size. If the width (height) is smaller than pre-stored bounding box size, we bind the object with the pre-stored bounding box size. All bounding boxes in the same video sequence are then aligned to the center-down of the object. After resizing boxed silhouette images into the given GEI size and accumulating these preprocessed silhouette images, a normalized and aligned GEI is formed. Fig. 3 shows GEIs aggregated by preprocessed silhouettes for different actions captured from front-right side of human (45°). In contrast to Fig. 2, it is obvious that the normalized GEIs reflect major shapes of human silhouettes and their temporal changes over human actions much better than the original ones.

2.2 Minimum Incremental Coding Length (MICL)

Minimum Incremental Coding Length. MICL, a new classification criterion proposed by Wright et al. [6], is founded on the principle of lossy data compression. The main idea is to measure how efficiently a new observation (test datum) can be encoded by each class of the training data subject to an allowable distortion, and to assign the new observation (test datum) to the class that requires the minimum number of additional bits. According to the criterion, we map the classification problem into an incremental coding length minimization one. Test data x would be assigned to the class which minimizes the number of additional bits needed to code $(x,\hat{y})$, subject to the given distortion ε, that is

$$\hat{y}(x) = \arg\min_{j=1,\dots,K} \delta L_{\varepsilon}(x, j). \tag{2}$$

For a multivariate Gaussian source $N(\mu,\Sigma)$, the average number of bits needed to code a vector subject to a distortion ε^2 can be approximated by

$$R_{\varepsilon}(\Sigma) = \frac{1}{2}\log_2 \det(I + \frac{n}{\varepsilon^2}\Sigma), \tag{3}$$

where n is the dimension of the input data. Then, given the data $\chi=(x_1,\dots,x_m)$ with sample mean $\hat{\mu} = \frac{1}{m}\sum_i x_i$, we can represent the data up to expected distortion ε^2 using on average $R_{\varepsilon}(\hat{\Sigma})$ bits, where

$$\hat{\Sigma}(\chi) = \frac{1}{m-1}(x_i - \hat{\mu})(x_i - \hat{\mu})^T \tag{4}$$

is the sample covariance, and so the number of bits needed for the m vectors is $mR_{\varepsilon}(\hat{\Sigma})$. Since the optimal codebook is adaptive to the data, we need additional $nR_{\varepsilon}(\hat{\Sigma})$ bits to represent the principle axes of the covariance matrix. In addition, we need extra $\frac{n}{2}\log_2(1+\frac{\hat{\mu}^T\hat{\mu}}{\varepsilon^2})$ bits to code the mean vector $\hat{\mu}$. Thus, the total number of bits required to code x becomes:

$$L_{\varepsilon}(\chi) = \frac{m+n}{2}\log_2 \det(I + \frac{n}{\varepsilon^2}\hat{\Sigma}(\chi)) + \frac{n}{2}\log_2(1+\frac{\hat{\mu}^T\hat{\mu}}{\varepsilon^2}). \tag{5}$$

Since the class label *y* is discrete, it can be coded losslessly. Because we can have a prior knowledge regarding the distribution of the class labels, we can code the labels using coding length

$$L_j = -\log_2 \pi_{j,} \tag{6}$$

where $\pi_j = \left|\chi_j\right|/m.$ Given the coding length function for the observations and the coding length for the class labels, we can compute the incremental coding length for each class as

$$\delta L_j(x, j) = L_{\varepsilon}(\chi_j \cup \{x\}) - L_{\varepsilon}(\chi_j) - \log_2 \pi_j. \tag{7}$$

Finally, the test sample would be assigned to the class in which the incremental coding length is the minimum.

Speed Up for MICL. Although the MICL classifier does have great classification power, it takes too much time for calculating the coding length. This fact makes the MICL classifier not suitable for applications in which timing performance is crucial. After some analysis, we found out the bottleneck for conducting the MICL classifier: the log-determinant computation. Notice that the second and the third term of Eq. (7) can be pre-computed offline during the training stage; however, the first term depends on the new test sample and requires computation of the log-determinant of an $n \times n$ or $m \times m$ matrix. The log-determinant computation can be computed either via Cholesky decomposition or singular value decomposition (SVD) and requires $\Theta(m^3)$ operations with straightforward numerically stable implementations [6]. In high dimensional spaces, i.e. when n>>m, the computation time would be very long and hinder the value of the proposed approach in real applications. We implement the MICL classifier using C language with the OpenCV library [7]. However, with the increasing use of graphics processing unit (GPU) [8], many general purpose processing tasks become faster and less difficult. An important reason why we choose MICL as our classifier, despite of its excellent classification power, is that all inputs for MICL are vectors, which can easily be moved onto the GPU memory and be computed in parallel. To speed up the computation of $L_{\varepsilon}(\chi)$, we compute the log-determinant via SVD with the aid of GPU instead of using the cvSVD() function in the OpenCV library. After moving all data vectors onto the GPU memory, we use a GPU-accelerated linear algebra library, CULA [9], with compute unified device architecture (CUDA) toolkit by NVIDIA [10], to do the SVD computation of input vectors. The CULA library can handle the GPU-side computations in a very short time. Notice that we need only the eigenvalues obtained from SVD, that is, we don't need to compute the eigenvectors of the input matrix. This fact makes the computation process much faster and more suitable for practical usage.

2.3 The Voting Strategy

A limit to use GEI as our key feature in human action recognition is that it cannot record overlapped information. Thus, it may cause information loss when GEIs aggregated from bad angles are used for classifications. An intuitive way to enhance the capability of the proposed approach is to recognize one human action with GEIs from more than one viewing angle. We equally divide 360° into eight discrete angles from a person's view, and set cameras at three angles having less overlapping information of the captured objects, i.e. 45°, 135°, and 225°. For each action, we extract three GEIs from the recorded sequences of the three view angles, and apply the MICL classifier to each GEI, respectively. To obtain the final recognition result, the decision rule is defined as follows: 1) each classifier (at each view angle) can be considered as a voting agent who votes for the best result. The majority result will be picked up. 2) If these three agents vote for three different classes, the one with the minimum incremental coding length, $\delta L_{\varepsilon}(x, j)$ is taken as the recognized class.

3 Experimental Results

3.1 Input Data and Computing Environment

We recorded video sequences of 13 kinds of human activities. Each activity is performed one or two times by 11 people (5 males and 6 females) and their heights vary from 5 to 6 feet. Some groups of similar actions are recorded to evaluate the effectiveness of the proposed system. For example, fall down vs. lie down, sit vs. squat, and pick up vs. stoop. To evaluate the power of detecting abnormal activities, we also recorded a set of shooting sequences. For medical care applications, we recorded the fall down and lie down activities as well. Three views of sequences are recorded, which are 45°, 135°, and 225° to the front view of the performer. The number of frames for each sequence varies from 10 to 540, and all frames are used to construct GEIs. The size of each GEI is normalized to 100×100 pixels. The data set contains 843 sequences with frame resolution of 320×240 pixels. All experimental results are conducted on an Intel® Dual Core™ 2.40 GHz PC and an NVIDIA GeForce GTX 460 graphics processor with CUDA toolkit 3.2 and CULA R11. The GTX 460 graphic processor has 7 multiprocessors with 336 cores and a total of 1GB memory.

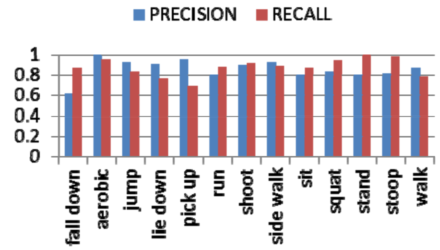


Fig. 4. Precision and recall of the proposed approach using the original GEIs as inputs

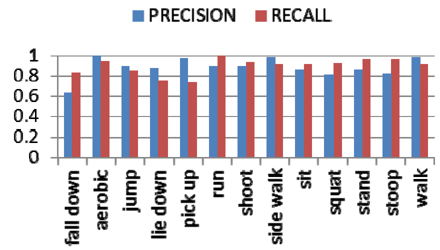


Fig. 5. Precision and recall of the proposed approach using normalized GEIs obtained by the proposed normalization method.

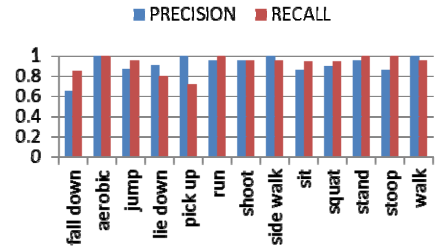


Fig. 6. Performance of the voting strategy with original GEIs

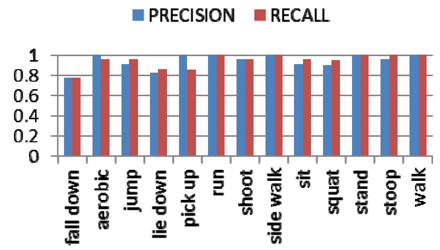


Fig. 7. Performance of the voting strategy and GEIs obtained by applying the proposed normalization method

3.2 Performance Evaluation

Improvement by Normalized GEI. After obtaining the original GEIs as well as the normalized and aligned ones, we transform GEIs into vectors as inputs for the MICL classifier. All experiments are evaluated with a 5-fold cross-validation. Fig. 4 shows the precision and recall of the proposed approach using original GEIs. The average precision and recall using the original GEIs are about 85% and 87%, respectively. It can be seen from Fig. 4 that many actions are misclassified as “fall down”. It is not surprising to have this result because actions of “lie down” and “fall down” are very similar even by human observation. In comparison with the original GEIs, Fig. 5 shows the precision and recall of the proposed approach using GEIs obtained by the proposed normalization method. The average precision and recall are about 88% and 90%, respectively. The classification results of the normalized GEIs outperform the original GEIs’ on almost every action class.

The Voting Strategy. Fig. 6 shows the experimental results after applying the voting strategy using the original GEIs as the inputs to MICL classifiers. Compared to the experimental results without the voting strategy in Fig. 4, the average precision is improved from 85% to 91%, and the recall is from 87% to 93%. Fig. 7 shows the results using the GEIs obtained by the proposed normalization method. With the help of the voting strategy, the averaged classification precision and recall can be improved to near 94%. The classification rate improves in almost every class. Even though some fall down sequences tended to be misclassified as lie down due to their high degree of similarity, this approach do perform well for classifying most of the similar human actions.

The Effect of Different GEI Sizes. In this work, the GEI size is set to be 100×100 pixels. In this subsection, we want to prove that this is a proper choice of GEI size concerning the issues of saving storage space, containing enough information and the classification speed at the same time. We generate GEIs with different sizes using the proposed normalization method. The tested sizes are 10×10, 20×20, 50×50, 100×100, 150×150, and 200×200 pixels. No GEI larger than 200×200 pixels is aggregated because most objects in our testing sequences are not larger than such size. Fig. 8 shows different sizes of GEIs extracted from the same squat sequence in 45° using the proposed normalization method. We resize every GEI to the same size for comparison. We can see that there are severe information losses in GEIs of sizes 10×10 and 20×20. The GEI of 50×50 pixels looks almost the same with that of 100×100 pixels; however, the image of 100×100 pixels GEI seems a little sharper. Moreover, the 150×150 pixels GEI and the 200×200 pixels GEI do contain more information than the 100×100 pixels GEI. Table 1 shows the comparison of precision and recall of the proposed approach (with the voting strategy) using the GEIs listed in Fig. 8.

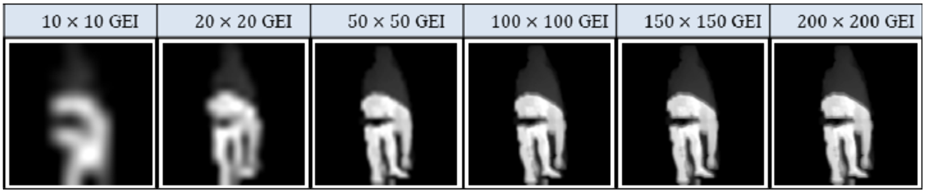


Fig. 8. The different size GEIs extracted from the same walking sequence. All GEIs are resized to the same size for comparison.

Table 1. Comparison of precision and recall when different sizes of GEIs are used

	10×10	20×20	50×50	100×100	150×150	200×200
PRECISION	51%	54%	90%	94%	91%	91%
RECALL	66%	49%	91%	94%	92%	92%

From Table 1, we can see that for the GEIs of sizes 10×10 pixels and 20×20 pixels, the accuracies are much lower than that of using other larger-size GEIs. The accuracy of the classification result may be higher when we use larger size of GEIs. However, with a larger GEI size, it takes more time to aggregate and classify a GEI. Moreover, both precision and recall are not significantly higher after the GEI size is larger than 50×50 pixels. Our proposed approach achieves the highest rate of precision and recall (both 94%) with a 100×100 pixels GEI size. In the next subsection, we would like to show the speed up results and compare the classification time between GEI size 50×50 pixels and 100×100 pixels, to see which GEI size is more suitable for our proposed classification approach.

Speed Up for MICL. With the help of the excellent parallel computing power of GPU and the well-designed linear algebra library, CULA, we could achieve a significant speed up for the SVD computation and thus our proposed approach is also speeded up. In Table 2, we list the SVD computation time, the classification time, and the speed up results using GEI size 100×100 pixels and 50×50 pixels. When using GEI size 100×100, we have improved the speed of the SVD computation from 7.33 seconds to 0.02 seconds per coding length computation and achieved a speedup about 300 times faster than the OpenCV version, in which the computation time for the MICL classifier is reduced from 180.00 seconds to 0.59 seconds per testing process. On the other hand, with a GEI of 50×50 pixels, the SVD computation time is reduced from 0.39 seconds to 0.01 seconds, and the classification time is also reduced from 6.25 seconds to 0.46 seconds. The required computation time depends on the number of activity classes to be classified and of course, the volume of data. In this work, we have 13 action classes and totally 843 data sequences. Comparing the classification times in the table, we can also observe that the classification time is only 0.13 second longer when we use a larger size GEI (100×100 pixels). However, from Table 1, the precision and recall using GEI size 100×100 pixels is about 4% higher than using GEIs of 50×50 pixels. Considering the trade-off between the precision, recall and the computation time, we suggest that the GEI of size 100×100 pixels is more suitable for our approach.

Table 2. The SVD computation time, the classification time, and the speed up result using GEI size of $100\times 100$and 50×50 pixels

	100×100			50×50		
	ORIGINAL	CULA	SPEED UP	ORIGINAL	CULA	SPEED UP
SVD	7.33s	0.02s	366.50	0.39s	0.01s	39.00
Classification	180.00s	0.59s	305.08	6.25s	0.46s	13.59

Comparison with Other Classification Methods. In this subsection, we want to compare the classification result between our classifier, MICL, with other known classifiers, support vector machine (SVM) [11] and K-nearest neighborhoods (KNN) [12]. Using the same GEI aggregated by the proposed normalization method, Table 3 shows the classification precisions using the MICL classifier as well as SVM and KNN. The results using single camera and the voting strategy are both listed and all experiments are conducted by 5-fold cross validation. From Table 3, we can see that the precisions using the MICL classifier are higher than SVM and KNN results, no matter the voting strategy is applied or not.

Table 3. The comparison of experimental results between MICL, SVM and KNN. The result using single camera and the voting strategy are listed.

Methods	MICL	SVM	KNN
Single Camera	88.42%	86.60%	82.68%
Voting Strategy	94.01%	92.17%	92.12%

Table 4. The experimental results using different classification methods to classify the Weizmann dataset. (*: All GEIs are transformed to the same direction.)

Methods	% of Correct	# of Actions Used
Blank et al. 2005 [14]	100%	9 (not including skip)
Our Proposed Approach	100%	9 (not including skip)
Gorelick et al. 2007 [15]	100%	10
Our Proposed Approach	97.78%	10
Our Proposed Approach*	100%	10

Classification Results with Other Dataset. Besides the actions sequences recorded by ourselves, we also conducted experiments based on the Weizmann dataset [13]. The Weizmann dataset contains 10 classes of actions: bend, jack, jump, pjump, run, side, skip, walk, wave1 and wave2. Each action is performed by 9 people. There are totally 90 sequences. Different from our recorded action sequences, the Weizmann dataset regards actions toward different directions (left and right) as the same class of action. Table 4 shows the experimental results classifying the Weizmann dataset using different classification methods. Notice that because the Weizmann dataset doesn't contain multiple views of an action, we do not apply the voting strategy in our approach here. All frames are used for classifying in all methods. The GEIs of size 100×100 pixels are used in our approach. From Table 4, we can see that our proposed approach is competitive to other state-of-the-are methods. The classification time of our approach is 0.35s per test. In the run, side, skip and walk actions, the actions are not set to be in the same direction. However, in our approach, we suppose that two actions in different directions, e.g. a man run toward the right and toward the left, should be classified as two classes of actions. In this experiment, we regard these different-direction-actions as the same class of actions and our approach achieves a precision at 97.78%. If all GEIs are transformed to the same direction, our approach can classify every action sequence correctly and a precision at 100% is obtained.

4 Conclusion

In this work, we present a novel approach for analyzing human actions from surveillance videos. Based on GEI and MICL, we designed a human activity recognition system. We also proposed a method to normalize and align silhouettes extracted from a variety of human activities, and GEI size 100×100 is suggested to be a more suitable choice for our approach. MICL is employed to learn the classifier, and multi-cameras are applied with the voting strategy after the classification process to enhance the overall system performance. Based on the normalized/aligned GEIs, the MICL classifier, and the voting strategy, our approach achieves accuracy at about 94% on average (on 13 classes of actions). With the help of GPU computation, we also achieved a speedup of 300 times for classification, from 180.00 seconds per test to 0.59 seconds, making the proposed approach valuable in practical usage. Clearly, there is still a need for improving the timing performance of the proposed approach before it can be applied to a real time surveillance system. Of course, this is one of the major directions of our future work. Sequences in our data set now contain only one action. In the future, we may try to recognize successive actions of a human in a single sequence based on the spatial-temporal correlation model, e.g. Hidden Markov Model, for gaining higher level semantic meaning of human actions.

References

1. Bobick, A.F., Davis, J.W.: The Recognition of Human Movement Using Temporal Templates. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(3), 257–267 (2001)
2. Hsieh, J.W., Hsu, Y.T., Liao, H.Y., Chen, C.C.: Video-Based Human Movement Analysis and Its Application to Surveillance Systems. IEEE Transactions on Multimedia 10(3), 372–384 (2008)
3. Zou, X., Bhanu, B.: Human Activity Classification Based on Gait Energy Image and Co-evolutionary Genetic Programming. In: The 18th International Conference on Pattern Recognition, vol. 3, pp. 556–559 (2006)
4. Han, J., Bhanu, B.: Individual Recognition Using Gait Energy Image. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(2), 316–322 (2006)
5. Lu, J., Tan, Y.P.: Gait-Based Human Age Estimation. In: International Conference on Acoustics, Speech and Signal Processing, vol. 5(4), pp. 761–770 (2010)
6. Wright, J., et al.: Classification via Minimum Incremental Coding Length (MICL). SIAM Journal on Imaging Sciences 2(2), 367–395 (2009)
7. OpenCV (Open Source Computer Vision), http://opencv.willowgarage.com/wiki/
8. Graphics Processing Unit (GPU), http://en.wikipedia.org/wiki/Graphics_processing_unit
9. EM Photonics. CULA Library R11 (2011), http://www.culatools.com/
10. NVIDIA Corporation. CUDA toolkit 3.2 (2010), http://developer.nvidia.com/cuda-toolkit-32-downloads
11. LibSVM, http://www.csie.ntu.edu.tw/~cjlin/libsvm/
12. K-Nearest Neighbor, http://note.sonots.com/SciSoftware/knn.html#1f488b13
13. The Weizmann dataset, http://www.wisdom.Weizmann.ac.il/~vision/SpaceTimeActions.html
14. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as Space-Time Shapes. In: Proc. Int'l Conf. Computer Vision, pp. 1395–1402 (2005)
15. Gorelick, L., Blank, M., Schechtman, E., Basri, R., Irani, M.: Actions as Space-Time Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(12), 2247–2253 (2007)

Hairstyle Suggestion Using Statistical Learning

Wei Yang, Masahiro Toyoura, and Xiaoyang Mao

University of Yamanashi
4-4-11 Takeda, Kofu, Yamanashi, Japan 400-8511
{g10mk039,mtoyoura,mao}@yamanashi.ac.jp

Abstract. Hairstyle is one of the most important features people use to characterize one's appearance. Whether a hairstyle is suitable or not is said to be closely related to one's facial shape. This paper proposes a new technique for automatically retrieving a suitable hairstyle from a collection of hairstyle examples through learning the relationship between facial shapes and suitable hairstyles. A method of hair-face image composition utilizing modern matting technique was also developed to synthesize realistic hairstyle images. The effectiveness of the proposed technique was validated through evaluation experiments.

Keywords: Hairstyle retrieval, example-based, statistical learning, non-parametric sampling, hairstyle image synthesis.

1 Introduction

With the advance of media acquisition and processing technologies, multimedia is now playing a very important role in enriching our daily life. This paper presents a new technology for suggesting a user with his/her suitable hairstyles by combing statistical learning and image processing techniques. Hairstyle is one of the most important features people use to determine their appearance and mood. People can look completely different simply by changing their hairstyles. Everyone would like to have a suitable hairstyle to make them look attractive, but it is usually difficult to find one as we cannot easily try out various styles with our real hair. Several commercials or free software have been developed allowing users to simulate how they look with different hairstyles by manually selecting hairstyle samples and superimposing them over their facial images. Although these systems do provide some general guidelines on choosing hairstyles, they do not provide any hints on what a suitable hairstyle is for a particular face. Therefore, users usually need to go through a very tedious process of trying out many different hairstyles before the one they like can be found. On the other hand, several papers have been published related to hairstyles [1,2,3,4] in the field of computer graphics. However, to the best of our knowledge, all these have focused on how to model and render hairstyles with computer graphics and have mainly been applied to create virtual characters and animations. In this paper, we propose a new technique for automatically retrieving a suitable hairstyle for a given face from a collection of successful hairstyle examples. What is a suitable hairstyle

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 277–287, 2012.

for someone? Suitability is a perceptual attribute and it is difficult to model it computationally. A hairstyle that appears to be attractive to one person may not look that acceptable to another. Personal aesthetics may be affected by many factors, such as one's cultural background and living environment. There are also many other factors that may affect how one looks in a particular hairstyle. Despite this, there is still some common aesthetics sense on hairstyles, and a skilled hair stylist can usually successfully create styles that conform to such common aesthetics sense. We started our project by interviewing several hair stylists. An important fact we observed is that although there are no stylists who can tell any explicit rules about their designs, they all viewed the shape of the face as the most important attribute in designing a hairstyle. This inspired us to adopt an example-based framework, which is an approach that has been successfully used for texture synthesis [5, 6] and style transferring [7,8,9,10] in recent years. Our new technique finds suitable hairstyles for a given face by learning the relationship between facial shapes and successful hairstyles.

The four major contributions of this paper can be summarized as follows:

1. A new framework for retrieving suitable hairstyles through learning the relationship between facial shapes and hairstyles from successful hairstyle examples.
2. The design of a compact feature vector space enabling fast non-parametric sampling in statistical learning.
3. A method of hair-face image composition utilizing modern matting techniques for synthesizing realistic hairstyle images automatically.
4. An evaluation experiment demonstrating the validity of the feature vector and the effectiveness of the example-based approach.

2 Example-Based Framework

Given a face image I_{input}, we want to create another image I_{output} with hairstyle S matching the face best. We achieve this in two steps:

Statistical Learning: Find the most suitable hairstyle S, through learning the relationship between facial shapes and suitable hairstyles.

Composition: Superimpose hairstyle S over face image I_{input} to obtain realistic image I_{output} of the face in a suitable hairstyle.

2.1 Statistical Learning

The statistical learning step can be described within the Bayesian inference framework. Based on Bayes theorem, posterior probability $P(S|I_{input})$, i.e., the probability for face image I_{input} to have its best hairstyle S, can be represented as:

$$P(S|I_{input}) = \frac{P(I_{input}|S)P(S)}{P(I_{input})}. \quad (1)$$

$P(S)$ is the prior probability of S, and $\mathrm{P(I_{input}|S)}$ is the probability of observed face image I_{input} given hairstyle S. Consequently, the aim of finding S can be turned into an optimization problem maximizing$P(S|I_{input})$. Since evidence $P(I_{input})$ can be treated as a constant, S is actually the one maximizing the product of likelihood $P\left(I_{input}|S\right)$and prior $P(S)$,

$$S = arg\,max_S\, P(I_{input}|S)P(S). \tag{2}$$

We obtain S by using non-parametric sampling, which has been proved to be an easy yet very efficient method in style transferring and texture synthesis applications [5,6,8,9]. Professionally designed hairstyle examples are used as the training data to learn prior $P(S)$ and likelihood $P(I_{input}|S)$.

To compute S, we first construct an approximation to conditional probability distribution$P(S|I_{input})$ and sample from this. Assuming $d(I^T, I_{input})$ is the distance between two facial images under some metric, if we define set

$$\Omega\left(I_{input}\right) = \left\{ I^T \,\middle|\, I^T \subset T, d\left(I^T, I_{input}\right) = 0\right\}. \tag{3}$$

containing all occurrences of I in training data set T, then the conditional probability distribution of $P(H)$ can be estimated with a histogram of all the hairstyles for faces I^T in $\Omega\,(I_{input})$. In other words, we can obtain a distribution of possible hairstyles forI_{input}. However, since we only have a finite number of examples from the training data set, we may fail to find any matching facial image I^T with $d\left(I^T, I_{input}\right) = 0$. To obtain an approximation to $\Omega\,(I_{input})$, let us specify small distance allowance e, and obtain $\Omega'\,(I_{input}) = \{I^T\,|I^T \subset T, d\left(I^T, I_{input}\right) < e\}$. The distance function d will be discussed in Section 3. Allowance e is determined by multiplying the minimum of $d\left(I^T, I_{input}\right)$ with a given constant, in our current implementation.

There are several possible ways of computing S from its conditional probability distribution:

1. Integrate the distribution and obtain an expected hairstyle image by compositing all the possible hairstyles.
2. Take the hairstyle with the highest probability. If there is no maximum value in the distribution, randomly choose a hairstyle.
3. Take the hairstyles of the *k*-nearest-neighbor of I_{input}.

Method 1 may produce some unrealistic hairstyle images due to the discontinuity in the hairstyles from the distribution. Since the training dataset usually consists of no more than one hairstyle for one single facial image, the result from Method 2 is usually a randomly chosen hairstyle from the set of hairstyles $I^T \subset \Omega\,'(I_{input})$. Method 3 is used in our current implementation and the hairstyles of the *k*-nearest-neighbor of I_{input} are recommended as candidates for the best suitable hairstyles. The *k* can be adjusted interactively.

2.2 System Overview

As shown in Fig 1, our system assumes n successful hairstyle images $I_i^T(i = 1,2,\ldots,n)$ are available. The three operations below are executed in the training phase to build a training data set $\mathrm{T}(V_i^T, \alpha_i^T)(\mathrm{i} = 1,\ldots \mathrm{n})$ where V_i^T is the feature vector characterizing the shape of each face I_i^T and α_i^T is the α-matte indicating the probability of a hair area in the image:

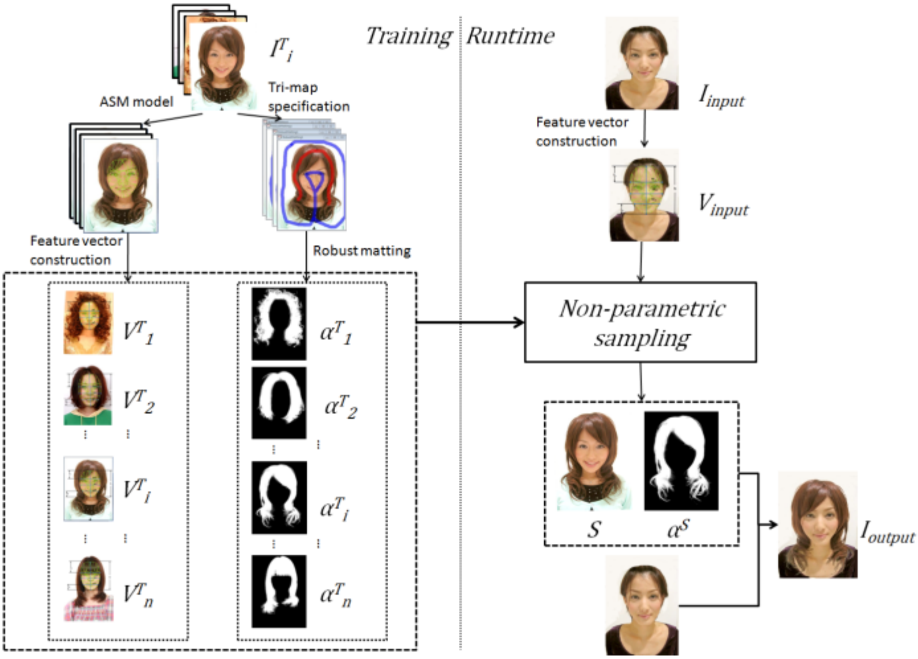


Fig. 1. System framework

1. Apply robust matting technique [11] to $I_i^T(i = 1,2,\ldots,n)$ to create $\alpha_i^T(i = 1,2,\ldots,n)$.
2. A trained Active Shape Models (ASM) [12] model is used to detect facial feature points on $I_i^T(i = 1,2,\ldots,n)$.
3. Construct feature vector $V_i^T(i = 1,2,\ldots,n)$ from the ASM model feature points.

While the first operation requires example strokes to be manually specified to create the tri-map for estimating the α-matte, the other two operations are performed fully automatically.

Given face image I_{input} in the runtime phase, the system performs six operations to compute a set of suitable hairstyles for I_{input}.

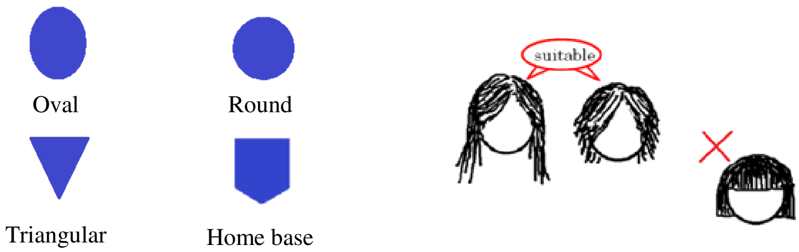
1. Apply a trained ASM to detect the facial feature points of I_{input}.
2. Construct feature vector V_{input} characterizing the shape of the face in I_{input}.
3. Search through all images in T in the feature vector space.

4. If $d\left(I^T, V_{input}\right) < e$ add I_i^T to $\Omega'\,(I_{input})$.
5. Sort images in $\Omega'\,(I_{input})$ by $d\left(I^T, V_{input}\right)$.
6. Take top k images $I_j^T\,(j = 1,2,\ldots,k)$ from $\Omega'\,(I_{input})$ and composite them with I_{input} using$\alpha_j^T\,(j = 1,2,\ldots,k)$.

The design of feature vector $V$is crucial to obtain good results, as well as to quickly searches through the training examples. We will discuss this in detail in the next section.

3 Feature Vector Design

Active Shape Models (ASM) [5] is one of the most popular techniques for detecting the geometric features of faces from images. Instead of using ASM directly, we want to have a more compact feature vector space, which can successfully model the relationship between facial shapes and hairstyles. As seen in Fig 2(a), it is known that human faces can be roughly classified into four categories by shape: *oval, round, triangular* and *home base*.



(a)Shapes of face

(b) Hairstyles for round faces

Fig. 2. Facial shapes and their relationships to hairstyles

A hairstyle giving an impression of an oval shaped face is likely to be suitable [13]. For example, as we can see from Fig 2(b), for the round face, the styles with long bangs flowing smoothly toward both sides or the back make the face look longer and hence are suitable, while the one with bangs cut straight across and a thick volume on top of the head is not suitable because it further emphasizes the impression of roundness. To find these kinds of relationships between facial shapes and hairstyles, we first compute six line segments (Fig 3(b)):

h : the center vertical line segment

w_1 : the horizontal line segment at the height of the eyebrows

w_2 : the horizontal line segment at the widest position of the facial area

w_3 : the horizontal line segment at the height of the mouth

h_t : the vertical line segment from the top of the face to the cross-section of h and w_1

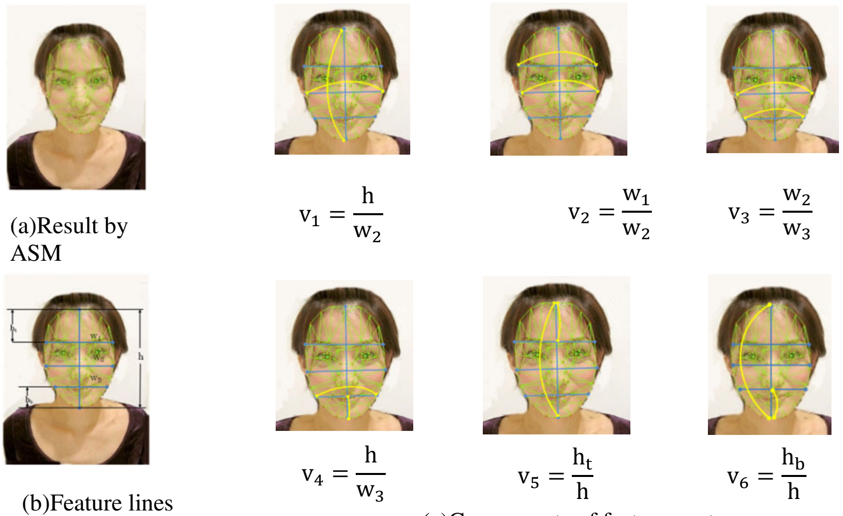
h_b : the vertical line segment from the bottom of the face to the cross-section of h and w_3

From the six line segments above, we define a six-dimensional feature vector $V(v_1,v_2,v_3,v_4,v_5,v_6)$ with $v_j(j=1,2,..6)$ being the ratio of two line segments, normalized to be in (0,1] (Fig 3(c)). To automatically compute the feature vector, we trained the ASM model with 81 points so that it includes the end points of the six line segments as the feature vector (Fig 3(a)).

The distance function for non-parametric sampling is defined as

$$\mathrm{d}\left(\mathrm{I}^{\mathrm{T}}, \mathrm{I}_{\mathrm{input}}\right) = \sum_{i=1}^{6} k_i \left(v_i^T - v_i^{I_{input}}\right)^2. \tag{4}$$

Coefficient $k_j(j = 1,2, \dots ,6)$ controls the weight of each dimension in determining the hairstyle. For example, if we use a large k_1 and a small k_2, then the system would treat the aspect ratio of the face as being more important and the shape of the cheek as being less important in determining a suitable hairstyle. Two faces with different shapes for cheeks but a similar aspect ratio may be suggested with similar hairstyles. We set all six coefficients to be equal to treat all dimensions homogenously in our current implementation. We plan to explore the best coefficients through machine learning in future work. A. Kagian et al used neural network to model the attractiveness of human face based on a 3240($_{81}C_2$) dimension feature vector [14]. Each dimension of the feature vector is the length of a line segment connecting two feature points of ASM model and the 3240 dimension corresponds to all the possible combination among 81 feature points. They trained the neural network through subjective an experiment. As the future work, we plan to adopt a similar approach explore the best feature vector to model the relationship between the geometric features of faces and hairstyles



(a)Result by ASM

$v_1 = \frac{h}{w_2}$ $v_2 = \frac{w_1}{w_2}$ $v_3 = \frac{w_2}{w_3}$

(b)Feature lines

$v_4 = \frac{h}{w_3}$ $v_5 = \frac{h_t}{h}$ $v_6 = \frac{h_b}{h}$

(c)Components of feature vector

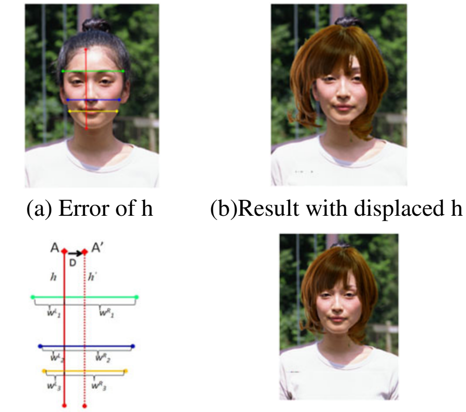
Fig. 3. Feature vector

4 Hairstyle Image Synthesis

Before we can superimpose obtained hairstyle image S over input face image I_{input}, position alignment and size adjustment between the two images are required. This is achieved by scaling H with $w_2^{I_{input}}/w_2$ in width, $h^{I_{input}}/h^H$ in height, followed by translation aligning the upper end point of h^H with that of $h^{I_{input}}$. As shown in Fig 4(a), due to fitting error with the ASM model, we may fail to obtain centered h^H or $h^{I_{input}}$, and this may result in an unnatural composition like the one in Fig 4(b). To correct error, we translate h horizontally by displacement D (Fig 4 (c)):

$$D = \frac{1}{2} \cdot \frac{1}{3} \cdot ((w_1^L - w_1^R) + (w_2^L - w_2^R) + (w_3^L - w_3^R). \tag{5}$$

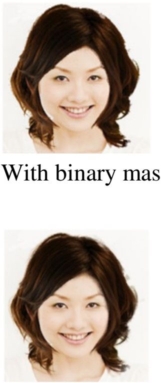
Fig 4(d) shows an improved result obtained by using the new position of h^H and $h^{I_{input}}$ for position alignment.



(a) Error of h (b)Result with displaced h

(c)Correction of h (d)Result with corrected h

Fig. 4. Position alignment



(a)With binary mask

(b) With α-matte

Fig. 5. Compare composition results using binary mask and α-matte

Finally, the α-matte of S is used to composite S and I_{input} *to ob*tain output image I_{output}

$$O_p = (1 - \alpha) \times H_p + \alpha \times I_p. \tag{6}$$

Here, p denotes a pixel, and O_p, H_p, and I_p, correspond to the pixel values for the output hairstyle image, input image, and suitable hairstyles suggested by the system, respectively. Fig 5 compares the results of binary mask based composition with our α-matting based method. We can see our method produces more realistic images, especially at regions near the boundary of hair.

5 Implementation and Evaluation

5.1 Implementation and Result

We built a training data set consisting of 84 hairstyle images collected through the courtesy of hair stylists from three hair salons in our current implementation. Through a preliminary user study, we found that in many cases users wanted to specify the length of their hair before searching for the best hairstyles, and hence it would be helpful if we could advise them of the best candidates for different lengths. Currently, both the hairstyle examples and the input face need to be frontal photos and the facial area in the input image should not be covered by strands of hair. In addition to the six dimensions representing the shape of a face, three additional dimensions, representing the length, hardness and volume of hair, are also added and a nine-dimensional feature vector is computed for each example. Each hair can take one of three values in the three newly added dimensions:

Length	: long, medium, short.
Volume	: large, medium, small.
Hardness	: hard, medium, soft.

The necessity to consider the properties (hardness and volume) of hair arose from the fact that the hairstyle one can actually have is constrained by the properties of his/her hair, even if one can find the best hairstyle making her look virtually attractive. Therefore, with the additional dimensions characterizing the properties of hair, we can constrain sampling to only hairstyles with similar hair properties. The hair properties of the input face are specified by the user.

Fig. 8 shows two examples of results. For the input face images at the left, the nearest face in the feature vector space for each of the long, medium and short hairstyle training sets are shown in the upper row and below it are the resulting hairstyles.

5.2 Evaluation

We conducted two experiments to validate the effectiveness of our approach. The first was aimed at investigating how the hairstyles recommended by our system to a person would look for other people, while the second experiment was aimed at how satisfactory the result would be for the person herself.

Ten female college students participated in the first experiment. We prepared nine sets of hairstyle images, each consisting of ten hairstyles with two of them recommended by the system as suitable hairstyles. At each trial, a subject was presented with one set of images and asked to mark the top three most suitable hairstyles out of the 10 hairstyles. Fig 6 shows an example of the hairstyle image set used in the experiment. Since hair color can largely affect the impression of hairstyle, we used the monochrome picture in the experiment to exclude the effect of color. There were a total number of 90 trials (9 sets×10 people) and we evaluated the probability of the occurrence of the following three cases.

Case 1 : The hairstyles recommended by the system were marked as the top suitable hairstyle.
Case 2 : At least one hairstyle recommended by the system was included in the top two suitable hairstyles.
Case 3 : At least one hairstyle recommended by the system was included in the top three suitable hairstyles.

We used a binomial test and our null hypothesis was that all 10 hairstyles in an image set would be marked with the same probability. Table 1 summarizes the experimental results. Out of the 90 trials, there were 20 trials for Case 1, 58 trials for Case 2 and 80 trials for Case 3. The probability for the number of occurrences above those observed ones upon the null hypothesis is listed at the rightmost column of Table 1. The null hypothesis was rejected at a significance level lower than 0.05 for the latter two cases. In other words, the experimental results suggested that the hairstyles recommended by our system at least could be viewed as the second best hairstyle even though it might not be the best.

The second experiment had the same setting as that for the first except that each subject was presented with hairstyle images of herself. The subjects were ten female college students and 3 image sets were prepared for each of them. Therefore, these were a total of 30 trials (3 sets×10 people). The number of trials for the three cases and the corresponding probability upon null hypothesis are summarized in Table 2. We can see the null hypothesis was rejected with a very low level of significance for the latter two cases, which is the same as the results for the first experiment.

Table 1. Result for Experiment 1: Viewed by Others (binomial test)

	Number of occurrence (Out of 90 trials)	Probability upon null hypothesis
Case 1	22%	0.25
Case 2	64%	0.00
Case 3	89%	0.00

Table 2. Results from Experiment 2: Viewed by self (binomial test)

	Number of occurrence (Out of 30 trials)	Probability upon null hypothesis
Case 1	17%	0.57
Case 2	53%	0.02
Case 3	83%	0.00

We can conclude from the experiment results that even though it may not be the best one, our system can advise users of good candidates for hairstyles viewed to be suitable both by themselves and others.

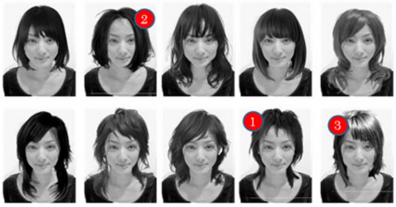


Fig. 6. Example of image set used for experiment

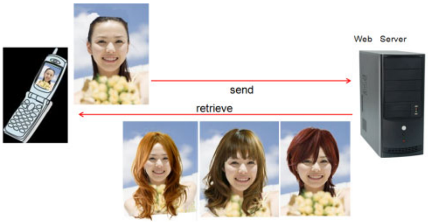


Fig. 7. Example of mobile application

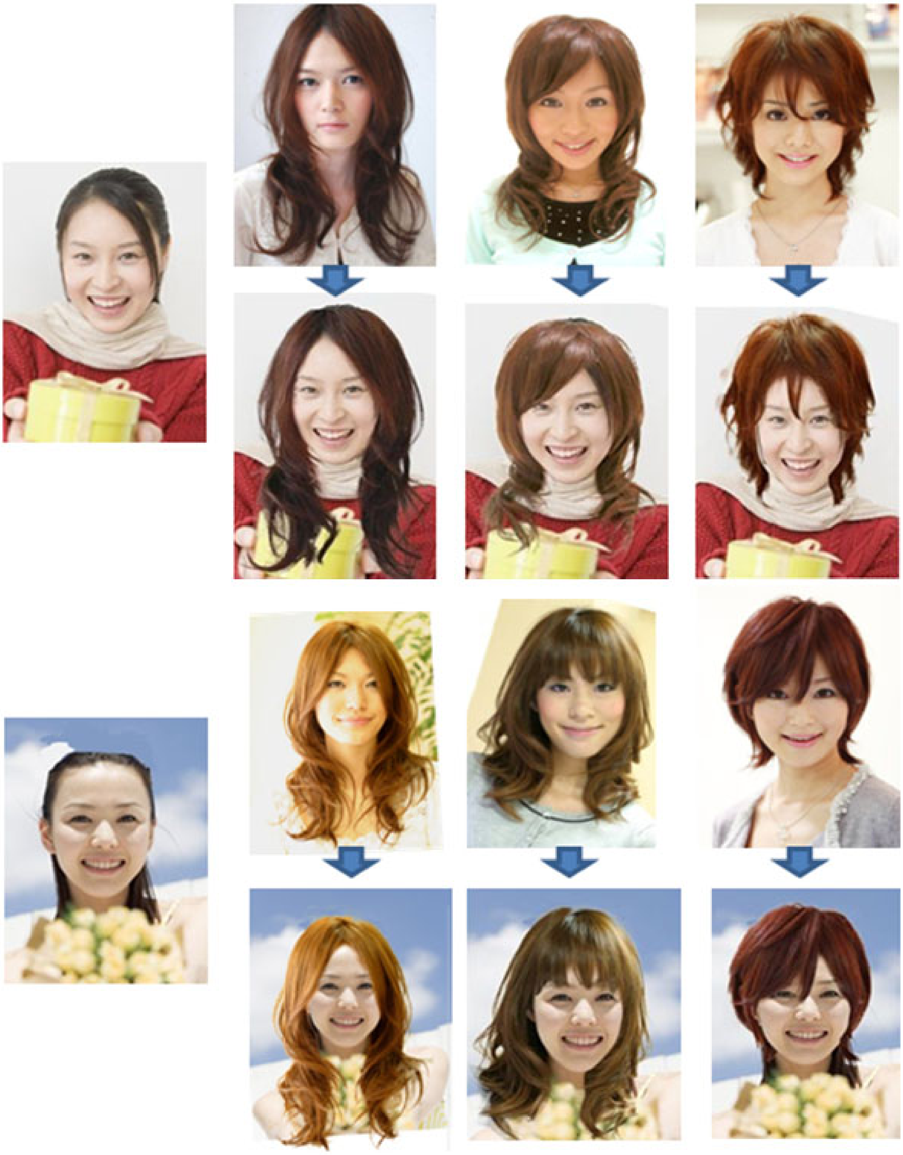


Fig. 8. Example of results

6 Concluding Remarks

We presented a new example-based framework for creating suitable hairstyles for a given face image. Suitability is a perceptual attribute and our evaluation experiments demonstrated the effectiveness of addressing these kinds of problems with an example-based approach. Since the proposed technique is fully automatic, a promising application is to implement it on mobile terminals with camera. For example, shows in Fig. 7, a user can retrieve a suitable hairstyle before going to hair salon simply by taking a photo of herself with the camera on her cell phone and then show it to the hair stylist after arrives the hair salon. In future research work, we want to apply the same framework to other fashion simulation problems, such as advising people of suitable attire by learning the relationships between body shape and successful choice in dress.

References

1. Paris, S., Chang, W., Kozhushnyan, O.I., Jorosz, W., Matusik, W., Zwicker, M., Durand, F.: Hair Photobooth: Geometric and Photometric Acquisition of Real Hairstyles. ACM Transactions on Graphics 27(3), article 30 (2008)
2. Wang, L., Yu, Y., Zhou, K., Guo, B.: Example based hair geometry synthesis. ACM Transactions on Graphics 28(3), Article 56 (2009)
3. Xu, Z., Dong, X., Yang, D.: V-hair studio: An interactive tool for hair design. IEEE Computer Graphics and Applications 21(3), 36–43 (2001)
4. Yu, Y.: Modeling realistic virtual hair styles. In: IEEE Pacific Conference on Computer Graphics and Applications(PG), p. 295. IEEE Computer Society, Washington (2001)
5. Efros, A., Leung, T.K.: Texture synthesis by nonparametric sampling. In: IEEE International Conference on Computer Vision (ICCV), vol. 2, pp. 1033–1038 (1999)
6. Hertzmann, A., Hacobs, C.E., Oliver, N., Curless, B., Salesin, D.H.: Image analogies. ACM Transactions on Graphics 25(4), 1360–1379 (2006)
7. Chang, C., Peng, Y., Chen, Y., Wang, S.: Artistic Painting Style Transformation Using Example-based Sampling Method. Journal of Information Science and Engineering 26(4), 1443–1458 (2010)
8. Chen, H., Xu, Y.Q., Shum, H., Zhu, S.C., Zheng, N.N.: Example-based facial sketch generation with nonparametric sampling. In: IEEE International Conference on Computer Vision (ICCV), pp. 433–438 (2001)
9. Chen, H., Liang, L., Xu, Y.Q., Shum, H.Y., Zheng, N.N.: Example-based automatic portraiture. In: IEEE Asian Conference on Computer Vision, pp. 23–25 (2002)
10. Suo, J., Min, F., Zhu, S., Shan, S., Chen, X.: A multi-resolution dynamic model for face aging simulation. In: IEEE Conference on Computer Vision and Pattern Recognition, Minnesota, USA (June 2007)
11. Wang, J., Cohen, M.F.: Optimized Color Sampling for Robust Matting. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8 (2007)
12. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models - their training and application. Computer Vision and Image Understanding 61, 38–59 (1995)
13. Best Hairstyle by Face Shape. Edited by Shufunotomo Co., Ltd (2007) (in Japanese)
14. Kagian, A., Dror, G., Leyvand, T., Cohen-Or, D., Ruppin, E.: A humanlike predictor of facial attractiveness. In: Schölkopf, B., Platt, J., Hoffman, T. (eds.) Advances in Neural Information Processing Systems 19, MIT Press (2007)

Topic Based Query Suggestions for Video Search

Kong-Wah Wan[1], Ah-Hwee Tan[2], Joo-Hwee Lim[1], and Liang-Tien Chia[2]

[1] Institute for Infocomm Research, 1 Fusionopolis Way, Singapore 138632
{kongwah,joohwee}@i2r.a-star.edu.sg
[2] School of Computer Engineering, Nanyang Technological University, Singapore
{asahtan,asltchia}@ntu.edu.sg

Abstract. Query suggestion is an assistive technology mechanism commonly used in search engines to enable a user to formulate their search queries by predicting or completing the next few query words that the user is likely to type. In most implementations, the suggestions are mined from query log and use some simple measure of query similarity such as query frequency or lexicographical matching. In this paper, we propose an alternative method of presenting query suggestions by their thematic topics. Our method adopts a document-centric approach to mine topics in the corpus, and does not require the availability of a query log. The heart of our algorithm is a probabilistic topic model that assumes that topics are multinomial distributions of words, and jointly learns the co-occurrence of textual words and the visual information in the video stream. Empirical results show that this alternate way of organizing query suggestions can better elucidate the high level query intent, and more effectively help a user meet his information need.

Keywords: Topic Modeling, Latent Dirichlet Allocation, Query Suggestion.

1 Introduction

Query formulation has long been a critical component in information retrieval, and most modern search engines now have mechanisms to help users refine their queries. One such mechanism is the auto-query completion that automatically suggests possible complete queries as a user type into the query box [1]. However, most existing query suggestion methods suffer from the following two problems:

First, the query suggestions in most systems are obtained by utilizing a large-scale query log [2,3]. These methods typically work by looking at the clickthrough patterns of similar queries. However, the applicability of such methods for customized search domains such as personal desktop search or intranet search is severely limited. This is because often times there is either no available query log, or the size of the user base and the number of past queries is too small for any meaningful query mining.

Second, query suggestions are mostly obtained using simple lexicographical matches and then ranked by their frequencies. The resulting suggestions often appear ad-hoc and disorganized. An example is shown on the left in figure 1. Our

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 288–299, 2012.

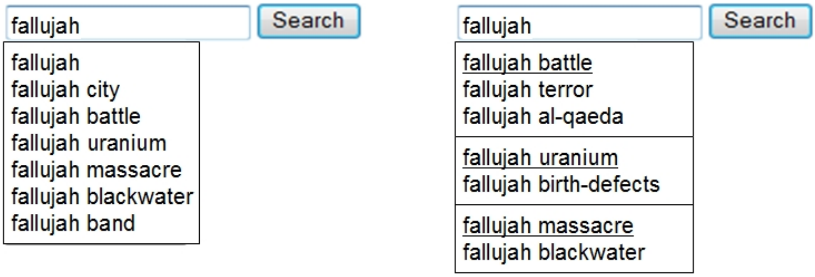


Fig. 1. Left: Existing query suggestions; Right: Presenting suggestions by thematic topics: three topics relevant to the query "fallujah" are shown, each titled and underscored by the highest probability word in the discovered topic

approach is to identify the *topics* among the documents that are relevant to the original query, and to group the query suggestions according to these topics. This new layout is shown on the right in figure 1. Note that query suggestions are now clustered into topics that are labeled by a keyword for easy navigation. The query suggestion "fallujah birth-defects" is also a new suggestion that gets presented as being part of the "fallujah uranium" topic. This allows a user to uncover possible new query interests that were hidden previously. The new presentation is clearly an improvement over the original.

Motivated by the above observations, in this paper, we propose a document-centric topic-based query suggestion method for the video search domain. We base our topic-elucidation methodology on the latent dirichlet allocation (LDA), which aims to uncover the hidden thematic structures in documents [4]. Using LDA to model query suggestions offers the following two advantages: (a). the relationships among the observed words, documents and latent topics is based on a theoretically robust probabilistic framework. (b). the learned topics are multinomial distributions of words, from which the high probability words are natural candidates for query word suggestion.

1.1 Challenges for the Video Domain

Despite provened capable of mining semantic topics in text collections, the use of topic model for the video domain poses some new challenges. First, naively feeding the speech transcripts (usually from Automatic Speech Recognition, or ASR) of video as textual input to the topic model will likely not yield good results. This is because ASR transcripts are noisy (ASR word error rates are generally above 20%), whereas most successful application of LDA are reported on clean text (e.g. newswire and publications). Apart from the few sporadic work in [5], the utility of LDA to noisy text source remains suspect.

The second challenge concerns the quality of LDA output: LDA often produces word distributions that are coarse, with no apparent meaning amongst high probability words. This can degrade the quality of detected topics. The common approach to deal with this problem is to introduce side-information into the modeling [6,4]. In this paper, we explore using the visual information in the shot keyframes to constrain topic development. There are two motivating intuitions: First, video footages are often repeated for similar or related news stories, and hence are highly correlated with the spoken (ASR) words. Second, different topics of a query may use different sets of words, and the same set of recurring visual shots become a bridge between these words, allowing them to be learned as distinct topics. To compute recurring visual shots, we use the Near Duplicate Image (NDI) detection method of Chum *et al* [7]. We derive a Gibbs Sampling-based inference and parameter estimation algorithm to jointly account for the NDI shots and the ASR words as distinct but correlated observations.

The third challenge relates to how to choose words from a learned topic to label suggestions. The difficulty arises because the semantic *theme* of a learned topic is only *collectively* conveyed by the high probability words, without any preferential order. Hence, a judicious choice of words from these high probability words is still needed to yield meaningful suggestions.

1.2 Our Contributions

In this paper, our contributions are:

(a) We propose a topic-based query suggestion method that can effectively help users refine and formulate their queries
(b) We develop a variant of the latent dirichlet allocation to constrain the development of topics during the inference process. The variant works by jointly modeling the text and visual information of a video stream.
(c) We develop a way to select high probability words in a topic to form suggestions.
(d) We perform extensive evaluation of our approach using real-life datasets and user studies.

In the remaining of this paper, we first discuss related works in Section 2. Section 3 provides details of our joint topic model. Experimental results are presented in Section 4, before we conclude the paper in Section 5.

2 Related Works

Query Completion: Machine-predicted text was first used mainly as an assistive mechanism for the physically handicap, but in recent years, has also found widespread benefit for mobile web search [8]. Today, query completion is featured in all major search engines. However, the focus of most work is on deriving a set of semantically related queries [9], rather than clustering them by topics. Jain and Mishne [10] proposed clustering query suggestions by simple phrasal keywords. There are only a few works on query suggestions without query log. Bhatia *et al* proposed a probabilistic model for generating query suggestions from the corpus, again using phrasal keywords. However, the limitations of phrasal keywords as a text summary have been highlighted in [12].

Table 1. Sample expansion words given the words on the left

Condolezza	state, bush, secretary, security, stanford
Jintao	china, brazil, sino, economic, aids
Basketball	points, conference, group, match, nba

Topic Modeling: Topic models are first derived as multinomial distributions of unimodal text data, and the joint modeling of multiple data types such as visual and text is not straightforward. The authors in [13,14] model annotated images using the visual features and text annotations, for automatic annotation and retrieval respectively. Two ways of combining the two modalities are explored: feature concatenation and hierarchical modeling. The former treats the two modalities equally, while the latter first models each individually and then fuse them at a later stage. Our work in this paper differs from the above in that we perform joint modeling of visual features and text in the video domain. Our modeling granularity is coarser: our visual features are not at the local patch-level but rather at the keyframe-shot-level. This most closely resembles Wu *et al*'s video representation with visual shot duplicates [15].

Because topic models are unsupervised methods, often best results are obtained by incorporating a priori knowledge about the desired output (e.g. must-link constrains in clustering). Adding observations from cross-media types as a way to constrain topic modeling is proposed by several authors. Jain *et al* guide topic formation of news photo caption by correlating the names with a face recognizer [6]. Blei and Jordan describe an image annotation model to learn the correspondence between an image region and a word in the caption [4].

3 Our Method

3.1 Overall Framework

In this subsection, we present a topic model to hypothesize a set of K topics $\{f_1, f_2, ..., f_k\}$ in a video collection D. Our overall approach is shown in figure 2. Two feature extraction tracks act on an input video simultaneously to compute text and visual features. These are then jointly combined using a generative model to compute topics. We discuss each of these key modules in the following.

Text Features: While important keywords are generally detected by ASR, the presence of the many misrecognized words can result in erroneous topic formation. Apart from stemming and discarding rare words, we use the document expansion approach of Wan to alleviate the problem [17]. The main idea is to introduce additional words to form an expanded text document vector. These words are selected based on their high mutual information in a parallel news corpus. Such a corpus is readily obtainable since news content is widely available

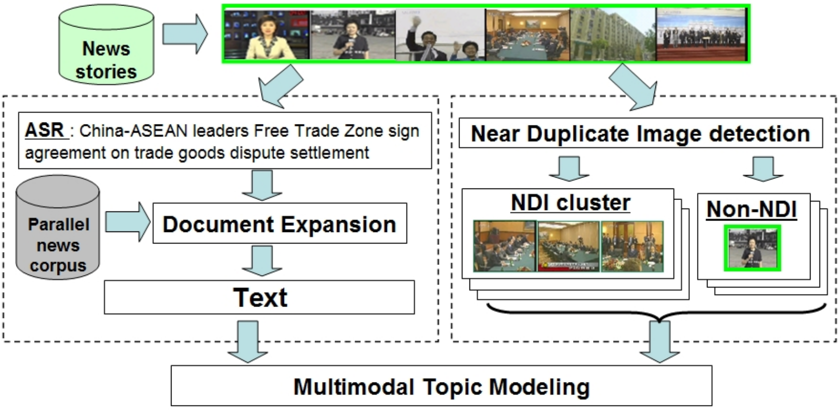

News stories
ASR : China-ASEAN leaders Free Trade Zone sign agreement on trade goods dispute settlement
Parallel news corpus
Document Expansion
Text
Near Duplicate Image detection
NDI cluster
Non-NDI
Multimodal Topic Modeling

Fig. 2. Multimodal topic modeling framework for hypothesizing topics in video

over the internet. For the TRECVID-2005 dataset used in this paper, we build the parallel corpus by issuing to Google Archives News our query description and restricting retrieval time-range to the period when the dataset was collected (Nov-Dec 2004). Table 1 shows some examples of expansion words.

Visual features: To capture a higher level of visual information, we generalize the common practice of modeling images as bag-of-features to represent video as *bag-of-keyframes*. Following the approach of Wu *et al* in [15], a keyframe is classified as whether it is a Near Duplicate Image (NDI) to other keyframe(s) or not. By assigning unique IDs to keyframes, they can be treated as visual words. All keyframes in a NDI-cluster are visually similar and are given the same ID. We can now generalize the TD-IDF weighting in text domain to these visual words. Figure 3 shows some examples of videos represented by the term frequencies (tf) and document frequencies (df) of visual words.

To compute near duplicates images, we use a color histogram combined with a spatial pyramid over the image to jointly encode global and local information. This approach is inspired by Chum *et al* [7], who applied the method to efficiently handle NDI detection amidst jitter and noise. Figure 4 shows the spatial pyramid configuration which is arranged so that an increasingly granular grid (i.e. from global to more localized) of color information is stored as we move up the level. The setup provides a highly compressed representation for each image that makes histogram comparison efficient. Given a query image, the NDIs are defined to be those within a specified Euclidean distance from the query. To compute a NDI cluster, we maintain a NDI list that initially only contains the query image. This list is then repeatedly populated with NDI of the new members in the list. The final NDI cluster is then given by the transitive closure of the NDI list.

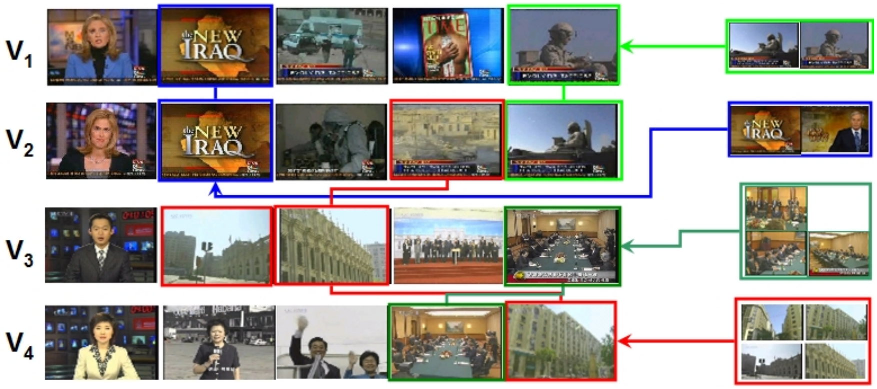


Fig. 3. TFIDF weighting of keyframe-based visual words. Within the 4 videos, four NDI clusters are shown and colored differently. For the red NDI cluster, it appears in V_2, V_3, V_4, hence its df=3, tf=2 for V_3, tf=1 for V_2, V_4. All non-NDI in a video have df=1 and tf=1. Best viewed in color.

3.2 Joint Modeling of Visual and Text

We now consider a corpus **D** of news video, each comprising of text **W** and visual words **V**. Each video d is modeled as a mixture of latent topics, to *simultaneously* account for **W** and **V** as distinct set of observations. Our model is motivated by Blei and Jordan's Corr-LDA model for text and images [4], and also Jain *et al*'s People-LDA model in [6]. We call our model cLDA-VT, denoting the use of both visual and text modality. Figure 5 shows the graphical representation of cLDA-VT. The generative process of cLDA-VT is as follow:

- Draw a multinomial ϕ over K topics: $\phi \sim \text{Dir}(\alpha)$
- For each topic $k = 1 \ldots K$,
  - draw multinomial $\theta_k \sim \text{Dir}(\eta_w)$ for text words
  - draw multinomial $\gamma_k \sim \text{Dir}(\eta_v)$ for visual words
- For each text word index n in d, $n = 1$ to N_d
  - Sample a topic z_n from ϕ: $z_n \sim \text{Multinomial}(\phi)$
  - Sample a text word w_n from θ_{z_n}
- For each visual word index m in d, $m = 1$ to M_d
  - Sample a topic y_m from ϕ: $y_m \sim \text{Multinomial}(\phi)$
  - Sample a visual word v_m from γ_{y_m}

where N_d and M_d is respectively the number of text words and visual words in video d, and η_w and η_v are Dirichlet priors for the text and visual words distribution respectively. The above cLDA-VT model results in the following joint distribution on text **W**, visual **V** and the latent topics:

$$P(\mathbf{W}, \mathbf{V}, \phi, \mathbf{z}, \mathbf{y}) = P(\phi|\alpha)\Big(\prod_{n=1}^{N_d} P(z_n|\phi)P(w_n|z_n, \theta)\Big) \Big(\prod_{m=1}^{M_d} P(y_m|\phi)P(v_m|y_m, \gamma)\Big) \quad (1)$$

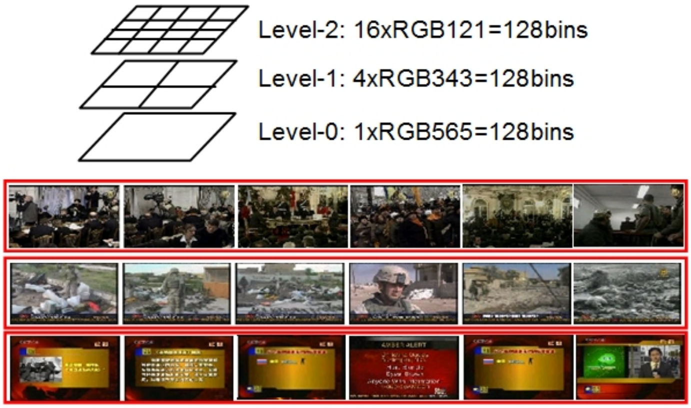

Level-2: 16xRGB121=128bins
Level-1: 4xRGB343=128bins
Level-0: 1xRGB565=128bins

Fig. 4. Top shows the spatial division of the image at each level of the RGB pyramid. At each level, the RGB suffix refers to the number of bits in the color channel. E.g. at level-1, we have 4 divisions each with 3 bits red, 4 bits green, 3 bits blue, totaling 128 bins. Each bin count uses 2 bytes. Therefore, each image is represented by 768 bytes. Bottom shows sample NDI given the leftmost image as query.

The main difference of our model from the Corr-LDA in Blei and Jordan [4] is that we use two multinomial distributions for the text and visual words. The sampling of visual words is essentially the same as the sampling text words. However, within a video, ϕ is a higher-level factor that is held fixed and it governs the ensemble of all text word and visual word observations. The topic-word multinomial θ and γ now learns the combined co-occurrence of important text words across video documents and also the complementary visual words.

Several approaches to learning the cLDA-VT parameters exist in the literature, such as Variational inference [4] and Gibbs Sampling [18]. We choose a simple extension of the latter by iterating over each text word, visual word and video document, each time resampling a single topic of the word (text or visual) based on the current topic assignment for the document and all other observed words (text and visual). A perplexity measure on a held-out set is used to determine learning convergence. On the 1028 TRECVID-2005 video documents comprising of 210K text words and 95K visual words, our implementation on a standard 3Ghz PC takes about 10 minutes to compute. We noticed that varying the Dirichlet priors η and α did not affect performance too much. We used the same value of 0.2 in the experiments below.

3.3 Selecting Topics and Terms for Query Suggestions

After the cLDA-VT learning has converged, the K latent topic distributions of both the text words and visual words are given by θ_k and γ_k, k=1..K. In particular, the probability of a text-word w in the kth latent topic is given by $P(w|\theta_k)$. We now make the assumption that the text multinomial θ_k represents the kth topic f_k in $\mathbf{D}$.

Topic Selection: From the K topics $\{f_1, f_2, ..., f_K\}$, we select a few as suggestion clusters. We can do this by a ranking approach, where the kth topic f_k is ranked by its relevance to the query as follow. Given a query Q, first note that $p(f_k|Q) \propto p(Q|f_k)p(f_k)$. By assuming $p(f_k)$ are uniformly distributed and query words occur independently, we can write

$$p(f_k|Q) \propto p(Q|f_k) = \prod_{q \in Q} p(q|f_k). \tag{2}$$

We select the top S topics with highest $p(f_k|Q)$ as suggestion clusters.

Terms Selection: Next, from each of the top S topics, we wish to select a few query suggestions as representative queries of the topic. For example, for the "fallujah battle" topic in figure 1, we have selected three suggestions: "battle", "terror" and "al-qaeda". While each of these suggestions is anchored on the "battle" topic of the query "fallujah", they also bear distinctive aspects within the topic. Our proposed way to achieve this is to select from amongst the high probability words candidate terms that are "compatible" with the current topic. By compatibility, we mean the following: suppose a candidate term t is also associated with a multinomial distribution of words $p(t|C)$, where C is a parallel context corpus. Then we can compare this multinomial distribution with the topic multinomial word distribution f_k using a suitable distribution measure such as the KL divergence:

$$\begin{aligned} \mathbb{KL}(t, f_k) &= -\sum_w p(w|f_k) \log \frac{p(w|f_k)}{p(w|t, C)} \\ &= \sum_w p(w|f_k) \log \frac{p(w, t|C)}{p(t|C)p(w|f_k)} \\ &\propto \sum_w p(w|f_k)\mathrm{PMI}(w, t|C) \end{aligned} \tag{3}$$

where PMI is the *pointwise mutual information* between the candidate term t and the terms in the topic model over the context corpus C. The PMI of two words is usually used as a measure of the semantic relationship between them. Intuitively, equation 3 assigns greater weights to a candidate term if it has a stronger semantic relationship to the important topic words. Hence, selected candidate terms are better representative of the entire topic f_k.

4 Experiments

To evaluate our methods, we used a subset of the TRECVID-2005 dataset [19]. It comprises of about 127 hours of Chinese and English news broadcast from 5 different sources (e.g. CCTV4, CNN). The dataset includes computed story boundaries from CMU Informedia. For queries, we used the annotated set of 33 queries from Wu *et al* [15]. Sample queries include "Bush visits Canada", "Mideast Peace", "Arafat health". For the full list of queries, refer to [15].

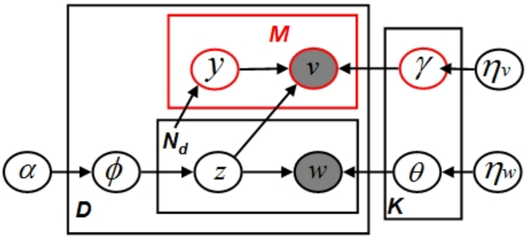


Fig. 5. Graphical representation of cLDA-VT. The red box encloses the additional observations from the visual modality, which constrain the topic formation in the text modality.

4.1 Perplexity Comparison

A commonly used quantitative evaluation of a topic model is how well it can predict the words in a test-set W_{test} after learning from a train-set W_{train}: that is, we are interested in which model provide a better predictive distribution $p(w \in W_{test}|w \in W_{train})$. We adopt the popular *perplexity* measure [4]:

$$\text{Perplexity}(\Psi) = \left(\prod_{d=1}^{D} \prod_{w \in W_{test}} p(w|\Psi, w \in W_{train}) \right)^{\frac{-1}{\sum_{d=1}^{D}(|doc_d|-|W_{train}|)}}, \quad (4)$$

where Ψ denotes the model parameters of a learned LDA or cLDA-VT model. Mathematically, the perplexity of a word distribution is defined as the inverse of the per-word geometric average of the probability of the observations. Informally, it can be thought of as the effective number of equally likely words according to the model. Note that a lower number denote more predictive power. Figure 6 compares the predictive perplexity obtained by the regular LDA and cLDA-VT. In the former, only the ASR words are fed to the inference process. As can be seen from the graph, there is less uncertainty in the cLDA-VT than in LDA. This indicates that the topics learned in cLDA-VT are more robust and descriptive.

4.2 Qualitative Inspection

We show sample topic-based query suggestions in Table 2. In Table 3, we qualitatively show how the text-words multinomial θ_k has benefited from the inclusion of visual features during topic learning. On each of two queries "War on Fallujah" and "Mideast peace", we introspectively pick a learned cLDA-VT topic that contains high probability words that are meaningful to the queries. Observe that words from multimodal model are more intuitive and correlate better to the query topic.

4.3 User Study

We are also interested to see if users would find it beneficial to be provided with our query suggestions. To do this, we conducted a small-scale user study.

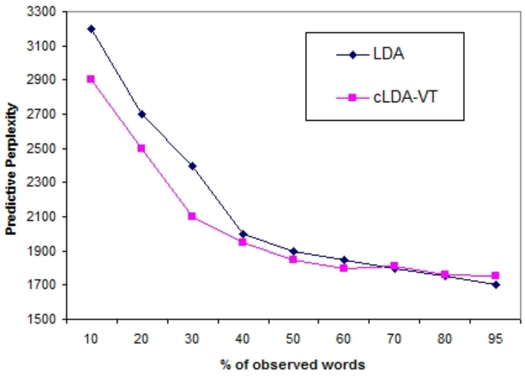


Fig. 6. Predictive Perplexity plot of LDA and cLDA-VT on a held-out set from the TRECVID-2005 video dataset. Lower is better.

Table 2. Sample topic-based query suggestions. The first line is the original query.

Fallujah	Middle-east	George Bush
fallujah battle	Middle-east Yasser Arafat	George Bush visits
fallujah terror	Middle-east Palestinian	George Bush canada
fallujah al-qaeda	Middle-east Israel	George Bush falungong
fallujah uranium	Middle-east peace	George Bush protests
fallujah birth-defects	Middle-east minister	George Bush pentagon
fallujah massacre	Middle-east election	George Bush rice
fallujah blackwater	Middle-east leader	George Bush cheney

Table 3. Top-10 probability words of 2 learned cLDA-VT θ_k topics, for "War on Fallujah" (top) and "Mideast peace" (bottom). The topics are picked introspectively. Meaningful words highlighted in bold.

LDA (ASR-text-only)	cLDA-VT (Visual+text)
iraq people time good meet **united states baghdad** chinese	**iraq iraqi** people **military** govern **arm kill force attack baghdad**
arafat know **yasser** leader thing **peace** minister people just **israel**	**palestinian arafat peace israel** president leader election **yasser east middle**

We asked three subjects to look at two versions of suggestions displayed for all 33 queries. The first version uses a simple lexicographical match and spelling corrector to find the top ten similar queries as query suggestions. They form the non-clustered suggestions as are currently obtained on most search engines. The second version uses the results from our automatic topic-based query suggestions. The subjects were then asked to put down on a scale of one to five their preference liking to both methods. We consider a method to be significantly better or worse liked for a user if the difference in his rating on the two methods is equal or more

Table 4. User preferences for query suggestions. See text for details.

Category	Percentage of time
Topic-based clustered suggestion significantly better	30
Topic-based clustered suggestion marginally better	22
No preference	28
Unclustered suggestions marginally better	18
Unclustered suggestions significantly better	2

than two, and marginally better or worse liked if the rating difference is one. The results are shown in Table 4. They indicate that our topic-based query suggestion method has potential value.

5 Conclusion and Future Works

Because the average length of a typical query is only two or three words long, most search engines are faced with a difficult task of discerning a user's search intent. Query auto-completion is an important mechanism to ameliorate the problem, and to facilitate a user in finding his information need. However, most existing query suggestion methods are based on simple lexicographical matching, and suggestions are placed in an ad-hoc manner. In this paper, we propose an alternative form of query suggestions that presents to users a list of possible complete queries grouped by their thematic topics. We argue for the benefits of such an arrangement, and demonstrate empirical results that show a high level of user acceptance and satisfaction with the idea.

As opposed to current query suggestion techniques, our method is document-centric and do not require a query log. Such a platform is particularly helpful when the user is not aware of how to phrase his query, or when the query words he chooses are not found in the documents. By mining topics on the documents directly, the resulting query suggestions are guaranteed to be found in the documents, making the suggestive framework useful not only for predicting what the user is likely to type, but also for uncovering new queries that may be of interest to the user.

Several aspects of the current implementation will be considered for future works. One relates to the run-time efficiency. Another pertains to the use of more sophisticated methods for query suggestion term selection.

Acknowledgments. The authors would like to express appreciation for the labeling help rendered by the three students from Temasek Polytechnic.

References

1. Feuer, A., Savev, S., Aslam, J.A.: Evaluation of phrasal query suggestions. In: Proc. ACM Conference on Information and Knowledge Management, pp. 841–848 (2007)

2. Baeza-Yates, R., Hurtado, C., Mendoza, M.: Query Recommendation Using Query Logs in Search Engines. In: Lindner, W., Fischer, F., Türker, C., Tzitzikas, Y., Vakali, A.I. (eds.) EDBT 2004. LNCS, vol. 3268, pp. 588–596. Springer, Heidelberg (2004)
3. Boldi, P., Bonchi, F., Castillo, C., Donato, D., Vigna, S.: Query suggestions using query-flow graphs. In: Proc. Workshop on Web Search Click Data, pp. 56–63 (2009)
4. Blei, D., Jordan, M.: Modeling annotated data. In: Proc. ACM Conference on Research and Development in Information Retrieval, pp. 127–134 (2003)
5. Cao, J., Li, J., Zhang, Y., Tang, S.: LDA-based retrieval framework for semantic news video retrieval. In: Proc. International Conference on Semantic Computing, pp. 155–160 (2007)
6. Jain, V., Learned-Miller, E., McCallum, A.: People-LDA: Anchoring topics to people using face recognition. In: Proc. International Conference on Computer Vision (2007)
7. Chum, O., Philbin, J., Isard, M., Zisserman, A.: Scalable near identical image and shot detection. In: Proc. ACM International Conference on Image and Video Retrieval, pp. 549–556 (2007)
8. Church, K., Thiesson, B.: The wild thing! In: Proc. Association for Computational Linguistics (2005)
9. Zhang, Z., Nasraoui, O.: Mining search engine query logs for query recommendation. In: Proc. International World Wide Web Conference (2006)
10. Jain, A., Mishne, G.: Organizing query completions for web search. In: Proc ACM International Conference on Information and Knowledge Management, pp. 1169–1178 (2010)
11. Bhatia, S., Majumdar, D., Mitra, P.: Query Suggestions in the Absence of Query Logs. In: Proc. ACM Conference on Research and Development in Information Retrieval (2011)
12. Madnani, N., Dorr, B.: Generating Phrasal and Sentential Paraphrases: A Survey of Data-Driven Methods. Journal of Computational Linguistics 36(3), 341–387 (2010)
13. Monay, F., Gatica-perez, D.: Modeling semantic aspects for cross-media image retrieval. IEEE PAMI 29, 1802–1817 (2007)
14. Lienhart, R., Romberg, S., Horster, E.: Multilayer plsa for multimodal image retrieval. In: Proc. ACM International Conference on Image and Video Retrieval, pp. 1–18 (2009)
15. Wu, X., Hauptmann, A., Ngo, C.: Novelty detection for cross-lingual news stories with visual duplicates and speech transcripts. In: Proc. ACM Multimedia, pp. 168–177 (2007)
16. Cutting, D., Karger, D., Pedersen, J., Tukey, J.: Scatter/Gather: A cluster-based approach to browsing large document collections. In: Proc. ACM Conference on Research and Development in Information Retrieval, pp. 318–329 (1992)
17. Wan, K.: Exploiting story-level context to improve video search. In: Proc. Internatonal Conference on Multimedia and Expo. (2008)
18. Griffiths, T., Steyvers, M.: Finding scientific topics. Proc. National Academy Science U.S.A 101 (Supp. 1), 5228–5235 (2004)
19. TRECVID (2005), http://www-nlpir.nist.gov/projects/trecvid

A Multimedia Retrieval Framework Based on Automatic Graded Relevance Judgments

Miriam Redi and Bernard Merialdo

Eurecom, Sophia Antipolis
{redi,merialdo}@eurecom.fr

Abstract. Traditional Content Based Multimedia Retrieval (CBMR) systems measure the relevance of visual samples using a binary scale (Relevant/Non Relevant). However, a picture can be relevant to a semantic category with different degrees, depending on the way such concept is represented in the image. In this paper, we build a CBMR framework that supports graded relevance judgments. In order to quickly build graded ground truths, we propose a measure to reassess binary-labeled databases without involving manual effort: we automatically assign a reliable relevance degree (Non, Weakly, Average, Very Relevant) to each sample, based on its position with respect to the hyperplane drawn by support vector machines in the feature space. We test the effectiveness of our system on two large-scale databases, and we show that our approach outperforms the traditional binary relevance-based frameworks in both scene recognition and video retrieval.

1 Introduction

CBMR systems aim to categorize and search for visual content in large collection of visual data, by exploiting statistical models that predict the presence of semantic concepts in images or videos. Such frameworks generally rely on supervised learning techniques, that require manually-assessed ground truth annotations associated with the samples in the dataset. When labeling a dataset, real assessors are asked to categorize an image or a shot according to its topical relevance with respect to a given concept. In most cases (e.g. the TrecVid collaborative annotation [1]), the notion of relevance is measured using a binary scale: a visual input is either "positive" or "negative" for the concept considered. This type of assessment assumes therefore that all the relevant elements are identically relevant and that all the irrelevant samples are equally non-relevant.

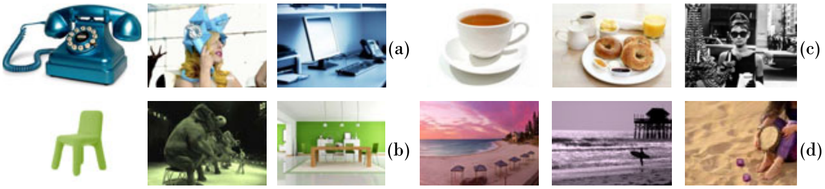


Fig. 1. Relevance is a relative notion: images labeled as positive for (a) "telephone" (b) "chair" (c) "cup" (d) "beach" actually have different visual evidences

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 300–311, 2012.

However,"as all human notions, relevance is messy and not necessarily perfectly rational"[16]. Each group in Fig. 1 shows a set of images that would be annotated as positive for the same corresponding concept: even if we can acknowledge that all the images are relevant with respect to the group label (e.g. images in group *a* contain the concept "Telephones"), the global semantic content of each image differs. Intuitively, we would say that each image is relevant for the associated concept with a different degree (for example, similar to web search engines, labels or grades such as "weakly relevant" or "very relevant" could be assigned). A distribution of relevance inferences over a graded scale would reflect better the human way of understanding concepts. From a learning system point of view, binary judgments imply that both marginally-relevant samples and very representative samples are treated equally when modeling the concept feature space: this might cause inconsistencies in the classification process. In a multimedia retrieval framework, concept models might be therefore less effective due to the contrast between the intra-class diversity and the binary relevance judgment. While graded relevance is widely used in web information retrieval (see [4], [23]), its use was rarely explored in CBMR: an attempt is represented by the graded-relevance system of Elleuch et Al. [3], that in the TrecVid 2010 edition outperformed the traditional binary-relevance frameworks proposed by the other participants at the Semantic Indexing Task.

When building a graded-relevance framework for information retrieval, the first step is to reassess the training samples, labeled as positive/negative, by assigning a "degree" of relevance. Generally [19] [3], the level of relevance of each sample is labeled manually. However, when dealing with large collections of visual data, e.g. the 400 hours of training videos for TrecVid [17] 2011, such re-assessment becomes time-consuming and practically unfeasible.

In this paper we propose an effective automatic graded-relevance based framework for image recognition and video retrieval. With our system, we can treat noisy and marginally relevant samples with less importance, achieving a better usage of our training set, thus improving the performances of traditional binary-relevance systems. Moreover, the key aspect of our framework is that, unlikely [3], the relevance degree of a training sample is assessed automatically: we assign to each sample a reliable and realistic relevance judgment, without involving any manual effort. To auto-reannotate each training sample in the database according to a non-binary relevance scale, we find a measure that first assigns a fuzzy membership judgment (i.e. how much a sample is representative/positive for a given concept), based on the position of the sample with respect to the hyperplane drawn by a Support Vector Machine (SVM) [2] in the feature space. Then, based on such relevance score, we re-categorize the training dataset into 4 groups for every concept: Very Relevant, Average Relevant, Weakly Relevant and Non Relevant samples. By training the system on such multiple repartitions, we then build a multi-level model for each semantic concept considered. When assigning labels to a new sample, the system outputs a set of concept prediction scores (one for each relevance-based layer of the model), that we weigh and combine to obtain a final label.

We test the effectiveness of our system by comparing it with traditional binary-relevance frameworks in two different tasks, namely scene categorization and video retrieval. For the first task we consider a large scale, noisy, database of tourism-related images, and we show that traditional categorization systems and features benefit from our automatic graded relevance-based multi-layer model when classifying this kind of biased data. We also consider the non-trivial Semantic Indexing Task of TrecVid 2010 [17] and we show that our non-binary reassessment combined with a multi-level prediction improves the recognition performances of a binary-scale video retrieval system by about 13%.

The remainder of this paper is structured as follows: in Sec. 2 we present an overview of the related work; in Sec. 3 we outline some background knowledge on traditional SVM-based retrieval systems; in Sec. 4 we show how to build an automatic relevance degree assignment scheme in a video retrieval framework. Finally, in Sec. 5 we compare our proposed framework with traditional image recognition and video retrieval systems and evaluate the results.

2 Related Work

Relevance is a fundamental notion for information retrieval: as pointed out in [16], while traditional bibliographic and classification frameworks aim to describe/categorize samples, retrieving information involves, besides description and categorization, the need for *searching*, and "searching is about relevance". Graded relevance-based learning methods first appeared for real Web search engines, where pages cannot be simply categorized as relative/non relative, but they need a multi-level relevance assignment. Several algorithms have been proposed to learn ranking functions from relative relevance judgments, like RankNet[21], based on neural networks, RankBoost[4], or the regression-based learning proposed in [23] by Zheng et al. How are these "grades" assigned? Generally, in traditional information retrieval such reassessment is done manually, either using real expert assessors [19], or using Amazon MechanicalTurk [18]. For web-based searches, the relevance judgment can be inferred in an automatic way, using the users' clickthroughs (see [7] for an overview of implicit relevance feedback method). In the image analysis and video retrieval field, graded relevance has been rarely explored. Traditional multimedia retrieval systems (see, for example, [14]) generally rely on binary-labeled keyframes or images. However, it was recently shown [3] that a video retrieval framework benefits from a graded-relevance annotated training set: in [3] the development set is reassessed by assigning, for each generally "relevant" frame, a degree of relevance from Somehow Relevant to Highly relevant. Three new training sets are then created based on different combinations of the relevance-based partitions.

Our work is somehow similar to the framework presented in [3]; however, in their work, the manual database re-assessment involves a lot of human effort and might increase the labeling noise. In this paper we automatize this process by automatically assigning a class membership degree to each sample. The idea is to exploit the learning methods traditionally used in video retrieval frameworks: the SVMs. Few works have indeed been presented in machine learning literature that reassess the samples in a binary-labeled training set based on the learnt feature space. Generally, they assign to the samples automatically a fuzzy membership score, namely a value representing their relevance for a given class. For example [9] defines an automatic membership measure as a function of the mean and radius of each class; this work is then extended by Lin et al in [8], that uses an heuristic strategy based on a confidence factor and a trashy factor in training data to automatically assign a score to each sample. An example of using automatic relevance assignment for image recognition is represented by the work of Ji et al. [5], where, to solve a face gender classification problem, the distance to the SVM hyperplane is used to measure the importance of each sample in a dataset for a given class. Another example can be found in [12], where the confidence of an image region label is again derived from the sample distance from the hyperplane. Similar to the work in [5], we use a SVM-based measure to identify a fuzzy relevance score for each class, that we then discretize, in order to label our training sets with three relevance degrees. However, instead of using the raw distance value, we prefer to use a calibrated, thresholded value, that still depends on the distance to the hyperplane, but it is expressed with the probability of a given sample to be positive with respect to a concept.

3 Binary Relevance Based Retrieval Systems

Traditional multimedia categorization systems associate a set of images or videos with a semantic label given a low-dimensional description of the input, namely a feature vector. Multimedia retrieval systems use categorization frameworks to build lists of pictures/shots ranked according to their pertinence with respect to a semantic concept or query. In both cases, the problem can be reduced to a multiclass classification problem, where each class represents the query/concept to be found in a visual sample. Generally, concept-specific SVMs are used to build models able to predict the presence of a given concept in a visual sample.

In order to build such system, a set of training samples $(x_i, y_i l)$, $i = 1 \dots n$ is required, where $x_1, x_2, \dots, x_n$ are the feature vectors extracted from the visual input data, and $y_i l$ the associated labels. For a set of concepts or categories $\{c_1, c_2, \dots . c_p\}$ (e.g. "Telephones"), each sample is labeled either as "positive", $y_{il} = +1$, $l = 1, \dots, p$, (the concept is present in the visual input represented by x_i) or "negative", $y_{il} = -1$ (no visual trace of the concept is found in x_i). A set of SVM-based classifiers, one for each concept/category, is used to learn the feature space and then to label new samples according to the same scheme. The idea behind the SVM is to find a hyperplane that optimally separates the two classes ($y_{il} = \pm 1$) in the problem feature space, given the distribution of the positive and negative samples with respect to a given concept. Such hyperplane satisfies the equation $\sum_i (\alpha_{il} y_{il} x_i)^T x - b_l = 0$.[1] When a new sample z needs to be categorized, the system assigns the corresponding label y_{zl} based on the

[1] Where $w_l = \sum_i \alpha_{il} y_{il} x_i$ has been proved in [2] to be the linear combination of the support vectors (i.e. the samples x_i for which the corresponding Lagrangian multiplier α_i is non zero).

sign of the dot product-based decision function $f_l(z) = w_l^T z + b_l$. For a retrieval framework (see, for example [14]), a confidence score $p(y_{zl} = 1|z)$ is obtained for sample z based on decision function values. Generally, a set of v visual features are extracted from each sample. v scores are obtained given such features, and then combined into one single confidence score. The results are then ranked according to such final score.

4 A Graded Relevance Based Retrieval System

As showed in Sec. 3, a SVM separates the feature space so that we are able to distinguish between positive and negative new samples for each given concept. This boundary is found based on a binary relevance judgment, y_{il}, that, as discussed before, might be too restrictive compared to the range of possible instances of a semantic concept in the visual input. In order to allow a better usage of our data, we go beyond the Relevant/Irrelevant subdivision, by reassessing our binary-relevance based training set with graded relevance judgments: in the new training set, a frame can be either Irrelevant (negative), Weak/Marginally Relevant, Average Relevant or Very Relevant. We then integrate the inferred relevance degree in a multi-layer concept classifier. The proposed framework works as follows (see Fig. 2):
(1)The features extracted from the training samples are processed by a set of binary p SVM-based classifiers (one for each concept). According to such models, we analyze the position of each training sample x_k with respect to the hyperplane, using a calibrated decision value, and extract, for each concept c_l, a fuzzy membership score σ_{kl}. This is a continuous value representing how much a given sample is representative for a semantic concept (see Sec 4.1 for more details).
(2) As shown in Section 4.2, for each concept, we sort the positive training samples according to their fuzzy relevance scores and we set two thresholds so that we are able to re-categorize the samples using discrete relevance degrees. We obtain three subsets of Strongly, Average and Weakly Relevant training samples. All the negatives are equally labeled as Non Relevant samples.
(3) Similar to [3], we then build a multi-layer model by training the system on three different, relevance-based training sets. Then, as presented in Sec 4.3, given a new test image, for all c_l we obtain from the multi-layer model three different concept prediction scores, that we then combine with weighted linear fusion to obtain one single output score. Such output score is then used for ranking and thresholded to determine the image label.

4.1 Decision Values as Relevance Indicators

As any traditional retrieval system, we start from an annotated training set of images/keyframes represented using low level features, namely our labeled samples. Given a set of non-negative samples, how to automatically define the fuzzy degree of relevance σ_{kl} of each sample with respect to a semantic concept? We tackle this problem by exploiting the SVM decision values of the training

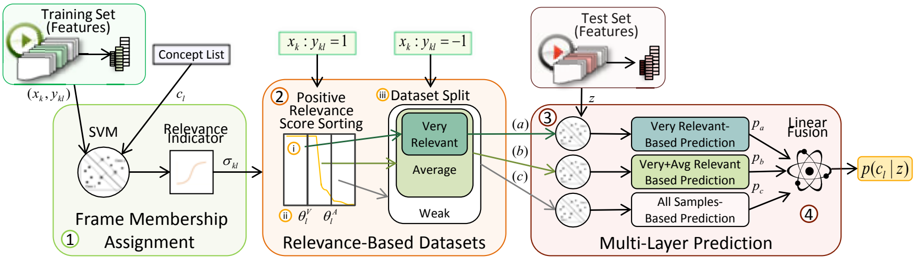


Fig. 2. Visual representation of our Relevance Based Framework

set. The idea is that if, for a concept c_l, we are able to define how "positive" the sample is, given its position with respect to the hyperplane, we can have a good estimation of its relevance degree for that given concept. As a matter of fact, various works [5,9,12] showed that there is a correlation between the distance to the hyperplane (or the distance to the class center) and how much each sample is representative for a given class (the bigger its distance from the boundary, the higher its relevance with respect to the positive/negative category).

In our approach, we use as a fuzzy membership measure for a training sample a thresholded version of the decision function, according to the solution proposed in [13] to translate the uncalibrated decision value into a probabilistic output. First, we calculate $f_l(x_k)$, namely the decision value for concept c_l, $\forall x_k, k = 1, \ldots, n$ in the training set samples. We then estimate the membership assignment as the positive class posterior probability $\sigma_{kl} = p(y_{kl} = 1|f_l(x_k))$ with a parametric model based on fitting a sigmoid function:

$$\sigma_{kl} = \frac{1}{1 + \exp(Af_l(x_k) + B)}, \tag{1}$$

Where A and B are parameters adapted in the training phase to give the best probability estimates.

4.2 A Multi-layer Training Set with Different Relevance Levels

Once the *continuous* value σ_{kl} is computed for each training sample x_k, the next step is to build a graded relevance retrieval framework. In order to achieve this goal, we need to have a *discrete* relevance degree for each training sample, so that we are able to perform a relevance-based split of the training set into smaller, consistent subsets with different degrees of relevance with respect to a concept c_l. As pointed out in [6], there is no universal rule to define such number of relevance degrees in a graded system. However, as shown in Sec 5.2, our experimental results suggest to set to 4 the number of relevance levels considered.

We therefore separate, for each concept, the positive/relevant training samples into three groups: Very Relevant Samples, that represent the most representative images/keyframes for a given class, Average Relevant Samples, and Weakly Relevant Samples; all the negatives are equally labeled as Non Relevant samples.

We then generate three repartitions of our training database, based on which a multi-layer model will be learnt (see Sec. 4.3). Having the fuzzy membership score σ_{kl} for each relevant sample, the discretization procedure is very simple:

(i) For each c_l, we take the *positive* $(x_k : y_{kl} = 1)$ training samples and sort them according to their corresponding σ_{kl}, in decreasing order.
(ii) We now want to find a partition of the positive samples in three classes, according to the relevance scale selected. Based on the shape of the curve drawn by the sorted fuzzy relevance scores, we identify two thresholds, θ_l^V and θ_l^A. We use and test three different approaches to choose such thresholds: (ii.a) we split the curve into equally spaced intervals, (ii.b) we choose the thresholds manually such that, intuitively, the intra-partition variance of the scores value is minimized (ii.c) we choose the values corresponding to 1/3 and 2/3 of the maximum membership score for the concept considered . For each concept c_l, the Very Relevant samples are then defined as the positive $x_k : 1 < \sigma_{kl} < \theta_l^V | y_{kl} = 1$; the Average Relevant samples as $x_k : \theta_l^V < \sigma_{kl} < \theta_l^A | y_{kl} = 1$; the Weakly Relevant as $x_k : \theta_l^A < \sigma_{kl} < 0 | y_{kl} = 1$.
(iii) Finally, similar to [3] we create three new training sets: (a) merges the Very Relevant Samples with all the Non Relevant (i.e. our *negatives*, $x_k : y_{kl} = -1$), (b) merges (a) with the Average Relevant Samples, and (c) considers all positives and negatives samples.

4.3 Multi-layer Prediction and Fusion

Once we have created the three concept-specific training subsets, for each concept we build our multi-layer model: it consists of three different SVM-based models, each of them learning a partition (a), (b), (c). Each level of the model separates the feature space in a different way, according to the annotations of the subset considered. When a new test sample z needs to be classified, we compute, using probabilistic SVM, three prediction scores for each concept (each of them is generated by a layer of the model). We therefore obtain , $\forall l$, $p_a(y_{zl} = 1|z), p_b(y_{zl} = 1|z), p_c(y_{zl} = 1|z)$.

Each of these predictions is generated by a different relevance-based partition, which gives a different, complementary type of information regarding the relevance degree of the new sample to be classified. In order to exploit such different cues and obtain a single output, we then merge the three outputs using weighted linear fusion, as follows:

$$p_{zl} = p(y_{zl} = 1|z) = \sum_t w_t p_t(y_{zl} = 1|z), \tag{2}$$

$t = a, b, c, \forall l$, where w_t is a concept-specific weight learnt with development data. For retrieval purposes, we then rank, for each query l the test samples according to p_{zl} in decreasing score, while for image categorization, the final label y_{zl} is assigned according to the following scheme:

$$y_{zl} = \begin{cases} -1 & \text{if } p_{zl} < 0.5 \\ +1 & \textit{otherwise} \end{cases}$$

5 Experimental Validation

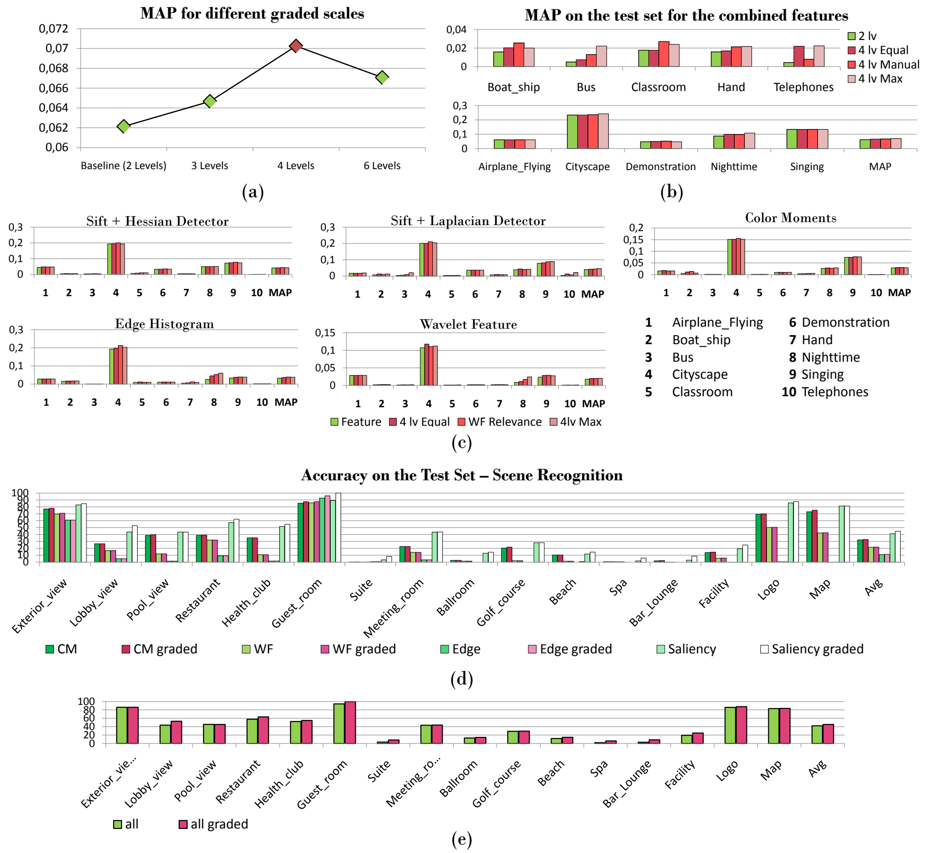
In this section, we use our proposed framework for both scene recognition and video retrieval: we compare the graded relevance framework with the classical binary-relevance systems (our baselines) for both tasks. First, in Sec. 5.1 we briefly summarize the composition of the large-scale databases considered and the experimental setup of the binary-relevance systems we use as baselines. We then explain in Sec 5.2 some details about our graded relevance framework setup and present some visual results that validate our automatic membership measure, presented in Sec 4.1. Finally, in Sec 5.3 we present the results obtained by comparing binary and graded relevance systems, for both the considered tasks.

5.1 Binary Relevance Framework Setup: Databases and Baselines

Scene Recognition. For this task, an automatic annotation system is required to assign a semantic category/concept to each image in the database. We choose for this task a large-scale database composed of around of 100,000 images coming from 100,000 touristic properties[2]. The database spans 16 between outdoor and indoor scene categories. For our binary-relevance baseline, we extract from such database the most widely used global features for content based image retrieval, namely Color Moments [20] , Wavelet Feature [11], Edge Histogram [22] and Saliency Moments [15] (respectively "CM", "WF", Edge" and "Saliency" in Fig. 3(d)). For every considered feature, a one-versus-all polynomial SVM-based model is built to separate each class from the others. Finally, the label confidence score of all the features are combined with linear fusion to obtain one single output.("all" in Fig. 3(d)).

Video Retrieval. Here, we focus on the Light Semantic Indexing Task (SIN), of TrecVid [17] 2010 where the retrieval system is required to produce a ranked list of relevant shots for a set of semantic concepts proposed. We use as a database the TrecVid 2010 IACC.1.tv10.dev set, which is composed of 3200 Internet Archive videos (a total of around 100,000 shots), that have been annotated with binary assignments. For our baseline, similar to our system in TrecVid 2010 [14], from each keyframe/shot, we extract a pool of visual features (Sift [10], Color Moments [20], a Wavelet Feature [11], and the MPEG7 edge histogram [22]). We then use them as input for a set of concept-specific classifiers, to build models that will predict the presence of a concept in each keyframe, and output a label and a concept score (the label confidence). All the concept scores coming from the different features are linearly combined to obtain the final concept score for each shot, that we will use to build the ranked list of shots.

[2] This is a randomly sampled subsed of 1 million images describing hotels amenities and surroundings, that have been manually labeled on the property owner's side before uploading them into a Hotel Management Platform.



(a)

(b)

(c)

(d)

(e)

Fig. 3. We compare our system with a traditional binary-relevance CBMR system. Video Retrieval task (a) Mean Average Precision values with different numbers of relevance-based categories (b) per-concept Average Precision on the TrecVid Database given the complete set of features (c) per-feature results. Image Categorization task (d) per-feature results and (e) Average Precision Accuracy on the test set for the combined set of features

5.2 Graded Relevance Framework Setup: Scale Selection and Relevance Visual Results

Our Graded Relevance frameworks are built on top of the baselines outlined in the previous section. As we already have binary annotated datasets, we need to (1) add a fuzzy membership score to each frame, (2) find proper thresholds to obtain a discrete relevance category assignment, and (3) build a multi-layer model as described in Sec 4.1-4.2-4.3.

(1)For each feature f, we can re-use the model built in the baseline to estimate the fuzzy membership score σ_{kl}^{f} of a keyframe/image in the training set x_k for a concept/category c_l. Instead of using directly feature-based membership scores, that might supply incomplete information (e.g. the most relevant samples given

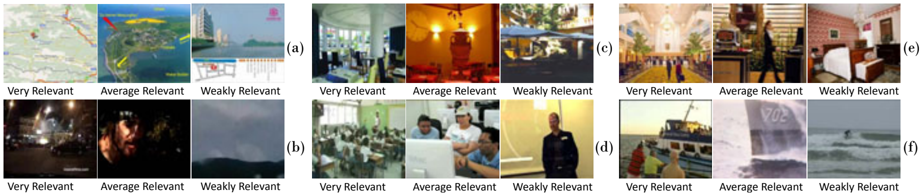


Fig. 4. Automatic relevance-based reassessment: for given semantic concepts, examples from the three relevance-based categories are shown (a)Map (b)Nighttime (c)Restaurant (d)Classroom (e)Hotel Lobby (f)Boat

the color or the edge distribution only), we combine them to obtain one single σ_{kl} for each sample.

(2)Now that we have a fuzzy score, how to select the number of discrete levels that we will use to re-categorize the training set? As shown in Fig. 3, we experimented with different subdivisions of the relevant samples of the training set and tested their respective performances on the video retrieval task. Results shown in Mean Average Precision (MAP) yield to the selection of a 4-level graded scale (namely Highly, Average and Weakly Relevant, and the Non Relevant label assigned to all the negatives) to reassess the training set. Is this subdivision reliable? Fig. 4 shows examples from the three relevance-based classes: as we can see, our proposed method actually separates samples according to their relevance with respect to the given category or query, and in some cases, among the "Weakly Relevant" samples we can even find wrongly annotated images. Given the trend of the fuzzy membership score curve, we select the thresholds θ^V and θ^A, according to methods (ii.a), (ii.b), (ii.c) mentioned in Sec. 4.2 (respectively "4lv Equal", "4lv Manual", "4 lv Max" in Fig. 3 b and c).

(3) Finally, for every feature and every concept, given the new training set repartitions, three models are created and then used to predict the presence of the concept, combining the three outputs as shown in Sec. 4.3. At the end of this step we will have, for a new sample z, a concept score p^f_{zl} for each feature. Such feature-specific concept scores are then fused with linear fusion, similar to the binary baseline.

5.3 Results

Scene Recognition. The scene recognition results in Fig. 3 (d-e) show the improvement obtained on a traditional binary relevance categorization system by introducing our graded-relevance reassessment, evaluated with the standard average classification accuracy on the test set. If we consider the whole set of descriptors combined together ("all" vs "all graded"), we can see that with our system we improve the overall categorization performances of about 8%. In particular, we can see that, when switching to graded relevance, we improve the discriminative power for some particular categories (e.g. Spa, +214%, Bar/Lounge, +197% and Beach, +24%) : analyzing such categories, we saw that those are the classes that are more affected by labeling noise, because they are often confused by the manual assessor with semantically similar classes (e.g. Bar-Restaurant, Spa-Health Club, Beach-Exterior View).

Video Retrieval. For the Video Retrieval Task, we present the results of both systems in terms of Mean Average Precision, the standard evaluation measure used for TrecVid assessments. We can see from Fig. 3 that the weaker features (e.g. Edge Histogram, +20% and Wavelet Feature, + 15%) benefit from our graded system. Moreover, we can see that the overall MAP increases of about 13%, when considering the ensemble of features combined together, with some peaks for those concepts for which the binary system was less performing, e.g. Classroom +53%, Telephones +420%, Bus +356% and BoatShip +60%.

6 Conclusions and Future Work

We presented a Multimedia Categorization and Retrieval Framework based on automatic graded relevance annotations. We automatically reassessed binary-labeled databases by assigning a degree of relevance to each sample based on its position with respect to the SVM hyperplane, and build an effective graded-relevance based CBMR system. We showed that our system, by allowing different degrees of relevance, outperforms the traditional binary-based frameworks for both image recognition and video retrieval.

Our simple approach can be improved in various ways. First, the automatic relevance fuzzy score assignment can be refined by using more complex machine learning-based measures, or by considering the combination of the relevance scores of a sample with respect to different concepts. Moreover, we can automatize the discretization procedure (from fuzzy to discrete relevance degrees) by designing a measure that infers the best thresholds from the shape of the positive membership scores curve. Finally, while in our framework, similar to traditional CBMR systems, we use simple SVM classifiers for ranking, we could explore the learning methods used for web page ranking (e.g. [23]), that are designed to support graded-relevance, achieving a higher discriminative power.

References

1. Ayache, S., Quénot, G.: Trecvid 2007 collaborative annotation using active learning. In: Proceedings of the TRECVID 2007 Workshop (2007)
2. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pp. 144–152. ACM (1992)
3. Elleuch, N., Zarka, M., Feki, I., Ammar, A.B., Alimi, A.: Regimvid at trecvid 2010: Semantic indexing. In: Proceedings of the TRECVID 2010 Workshop (2010)
4. Freund, Y., Iyer, R., Schapire, R.E., Singer, Y.: An efficient boosting algorithm for combining preferences. J. Mach. Learn. Res. 4, 933–969 (2003)
5. Ji, Z., Lu, B.-L.: Gender Classification Based on Support Vector Machine with Automatic Confidence. In: Leung, C.S., Lee, M., Chan, J.H. (eds.) ICONIP 2009, Part I. LNCS, vol. 5863, pp. 685–692. Springer, Heidelberg (2009)
6. Kekäläinen, J.: Binary and graded relevance in ir evaluations–comparison of the effects on ranking of ir systems. Information processing & management 41(5), 1019–1033 (2005)

7. Kelly, D., Teevan, J.: Implicit feedback for inferring user preference: a bibliography. SIGIR Forum 37, 18–28 (2003)
8. Lin, C., Wang, S.: Training algorithms for fuzzy support vector machines with noisy data. Pattern Recognition Letters 25(14), 1647–1656 (2004)
9. Lin, C.F., Wang, S.D.: Fuzzy support vector machines. IEEE Transactions on Neural Networks 13(2), 464–471 (2002)
10. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
11. Papageorgiou, C.P., Oren, M., Poggio, T.: A General Framework for Object Detection. In: Proceedings of the Sixth International Conference on Computer Vision, p. 555. IEEE Computer Society (1998)
12. Paterno, M.C.S., Lim, F.S., Leow, W.K.: Fuzzy semantic labeling for image retrieval. In: 2004 IEEE International Conference on Multimedia and Expo, ICME 2004, vol. 2, pp. 767–770. IEEE (2004)
13. Platt, J.: Probabilistic outputs for support vector machines. In: Bartlett, P., Schoelkopf, B., Schurmans, D., Smola, A.J. (eds.) Advances in Large Margin Classifiers, pp. 61–74
14. Redi, M., Merialdo, B., Wang, F.: Eurecom and ecnu at trecvid 2010: The semantic indexing task. In: Proceedings of the TRECVID 2010 Workshop (2010)
15. Redi, M., Merialdo, B.: Saliency moments for image categorization. In: Proceedings of the 1st ACM International Conference on Multimedia Retrieval, ICMR 2011, pp. 39:1–39:8. ACM, New York (2011)
16. Saracevic, T.: Relevance: A review of the literature and a framework for thinking on the notion in information science. part iii: Behavior and effects of relevance. Journal of the American Society for Information Science and Technology 58(13), 2126–2144 (2007)
17. Smeaton, A.F., Over, P., Kraaij, W.: Evaluation campaigns and trecvid. In: MIR 2006: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, pp. 321–330. ACM Press, New York (2006)
18. Snow, R., O'Connor, B., Jurafsky, D., Ng, A.Y.: Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 254–263. Association for Computational Linguistics (2008)
19. Sormunen, E.: Liberal relevance criteria of trec-: counting on negligible documents? In: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 324–330. ACM (2002)
20. Stricker, M.A., Orengo, M.: Similarity of color images. In: Proceedings of SPIE, vol. 2420, p. 381 (1995)
21. Svore, K., Vanderwende, L., Burges, C.: Enhancing single-document summarization by combining ranknet and third-party sources. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 448–457 (2007)
22. Won, C.S., Park, D.K., Park, S.J.: Efficient use of MPEG-7 edge histogram descriptor. Etri Journal 24(1), 23–30 (2002)
23. Zheng, Z., Chen, K., Sun, G., Zha, H.: A regression framework for learning ranking functions using relative relevance judgments. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 287–294. ACM (2007)

Effective Heterogeneous Similarity Measure with Nearest Neighbors for Cross-Media Retrieval

Xiaohua Zhai, Yuxin Peng*, and Jianguo Xiao

Institute of Computer Science and Technology, Peking University,
Beijing 100871, China
{zhaixiaohua,pengyuxin,jgxiao}@pku.edu.cn

Abstract. Emerging multimedia content including images and texts are always jointly utilized to describe the same semantics. As a result, cross-media retrieval becomes increasingly important, which is able to retrieve the results of the same semantics with the query but with different media types. In this paper, we propose a novel heterogeneous similarity measure with nearest neighbors (HSNN). Unlike traditional similarity measures which are limited in homogeneous feature space, HSNN could compute the similarity between media objects with different media types. The heterogeneous similarity is obtained by computing the probability for two media objects belonging to the same semantic category. The probability is achieved by analyzing the homogeneous nearest neighbors of each media object. HSNN is flexible so that any traditional similarity measure could be incorporated, which is further regarded as the weak ranker. An effective ranking model is learned from multiple weak rankers through AdaRank for cross-media retrieval. Experiments on the wikipedia dataset show the effectiveness of the proposed approach, compared with state-of-the-art methods. The cross-media retrieval also shows to outperform image retrieval systems on a unimedia retrieval task.

Keywords: cross-media, cross-modal, multimedia, retrieval, heterogeneous similarity measure.

1 Introduction

There has been rapid growth of multimedia content on the web. Many research efforts have been devoted to the content-based multimedia retrieval [4,5,9,15,3,8,2]. To facilitate effective retrieval method, several organizations have held a number of large-scale researches and evaluation efforts, such as TRECVID [13] and imageCLEF [7,14]. Recently, automatic image annotation and image retrieval based on text label or short image description have attracted research efforts [1,6,5].

However, the prevailing methods didn't well explore the textual information. Texts, images and videos always co-exist in a multimedia document to describe the same semantic concept [12,20]. For instance, we can quickly draw a vivid

* Corresponding author.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 312–322, 2012.

imagination about a concept through an image, but it is not complete and accurate enough. Different people always have different views. In contract, texts could accurately reflect the details of the concept, but it is not intuitive enough. Different media types are complementary to each other. A key requirement in the multimedia community is joint modeling for multimodality media contents. Therefore, a new topic of cross-media retrieval becomes increasingly important. By fully exploiting the rich information of multiple media types, we could understand the content of multimedia more accurately.

Generally, there are two major components of a retrieval task: feature representation and feature comparison. The representation step aims to transform the media objects into feature vectors. The comparison step aims to retrieve the results which are similar to the query by a similarity measure. Although many similarity measures are developed for information retrieval within a single media type[17,8,10], there is few work about the matching between heterogeneous features of different media types.

Zhuang et al. [22,21,20] explore the co-occurrence information in different modalities, that is, if a media object is shared by two multimedia documents, then they are of the same semantic. However, these methods heavily rely on the co-occurrence of query object and multimedia documents in dataset. When the query object is out of dataset, the performance will decrease dramatically. So users' feedback is required to ensure the performance. Rasiwasia et al. [12] proposed two kinds of supervised learning methods for cross-media retrieval. The first is to learn the subspace that maximizes the correlation between images and texts by canonical correlation analysis (CCA). The other is to represent the media objects by vectors of posterior probabilities with respect to a set of predefined categories, computed with multi-class logistic regression. The posterior probability vector indicates the high level semantic. The learning methods seek to find a homogeneous feature representation for media objects with different modalities instead of the co-occurrence information. After the media objects are represented as homogeneous feature vectors, comparison could be performed by traditional similarity measures. However, due to the gap between low-level features of media content and human understanding, the high level representations are not accurate enough. Then the inaccuracy will be propagated to the following matching step and decrease the performance of cross-media retrieval.

To address the above problems, in this paper we propose a novel heterogeneous similarity measure with nearest neighbors (HSNN). Unlike traditional similarity measures which are limited in homogeneous feature space, HSNN could compute the similarity between media objects with different media types. The heterogeneous similarity is obtained by computing the probability of two media objects belonging to the same semantic category, regardless of which category it is. The probability is obtained by analyzing the nearest neighbors of each media object. HSNN is flexible so that any traditional similarity measure could be incorporated, which is further regarded as the weak ranker. A ranking model for cross-media retrieval is learned from multiple weak rankers through AdaRank.

Experiments on the wikipedia dataset show the effectiveness of the proposed HSNN approach compared with state-of-the-art methods.

The rest of this paper will be organized as follows. In section 2, we demonstrates the proposed HSNN approach for cross-media retrieval. Section 3 demonstrates the approach of learning the ranking model. Section 4 shows the experimental results. Finally, we conclude this paper in section 5.

2 Heterogeneous Similarity Measure with Nearest Neighbors

In this section, we present the proposed heterogeneous similarity measure for cross-media retrieval. We restrict the discussion to multimedia documents containing images and texts as [12] and the fundamental ideas are applicable to any combination of media types. The goal is to retrieve text articles in response to the query of images and vice-versa.

Following the convention of previous research [12,22], we define a multimedia document as a set of co-occurring media objects. The multimedia dataset is denoted as $\mathcal{D} = \{D_1, ..., D_N\}$, in which D_i denotes a multimedia document containing images and texts, which are referred to as media objects. Images and texts are represented as feature vectors $I_i \in \mathcal{R}^I$ and $T_i \in \mathcal{R}^T$, respectively. Bag-of-words (BOW) model and topic model are utilized to represent the images and texts respectively. The goal of cross-media retrieval task is described as follows: given a image (text) query $I_q \in \mathcal{R}^I$ $(T_q \in \mathcal{R}^T)$, return the closest match in the text (image) space $\mathcal{R}^T(\mathcal{R}^I)$.

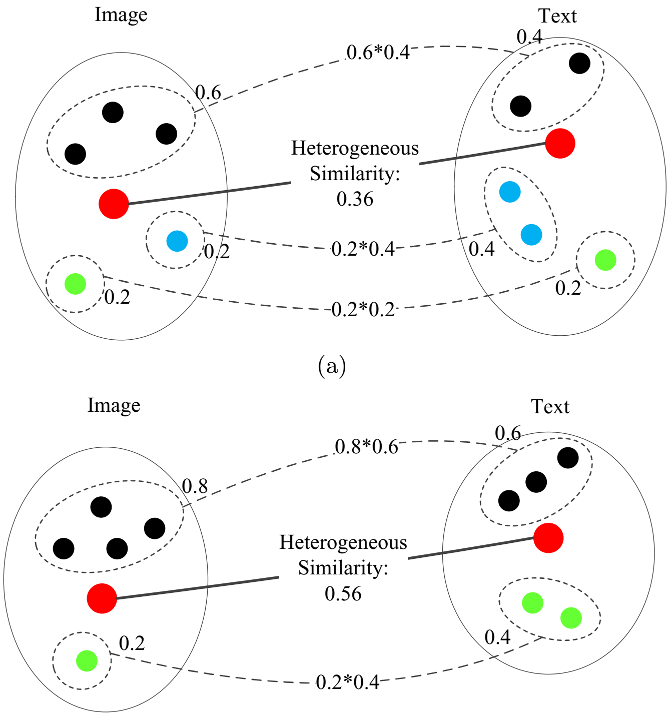
Normally, kNN classification is used to classify data point I_i (T_i) based on closest training examples in the feature space $\mathcal{R}^I$ $(\mathcal{R}^T)$. I_i (T_i) is classified by a majority vote of its nearest neighbors, with the object assigned to the class most common amongst its k nearest neighbors. As for cross-media retrieval, given two heterogeneous media objects I_i and T_j, we have to predict whether they belong to the same semantic category, regardless of which category it is. As a toy example, the proposed similarity measure is explicitly shown in Figure 1. To achieve this goal we compute the marginal probability that we assign I_i and T_j to the same semantic category with nearest neighbors, which equals:

$$P(l_i = l_j | I_i, T_j) = \sum_c p(l_i = c | I_i) p(l_j = c | T_j) \tag{1}$$

where l_i (l_j) stands for the label of I_i (T_j), $p(l_i = c|I_i)$ stands for the probability of I_i belonging to category c. $p(l_i = c|I_i)$ is defined as follows:

$$p(l_i = c | I_i) = \frac{\sum\limits_{I_k \in kNN(I_i) \wedge l_k = c} \sigma(sim(I_i, I_k))}{\sum\limits_{I_k \in kNN(I_i)} \sigma(sim(I_i, I_k))} \tag{2}$$

$kNN(I_i)$ stands for the k-nearest neighbors of image I_i in training set, and l_k stands for the label of image I_k. The definition of $p(l_j = c|T_j)$ is similar to



(a)

(b)

Fig. 1. The illustration of our HSNN: (a) nearest neighbors belong to three semantic categories; (b) nearest neighbors belong to two semantic categories. The left side shows the images and right side shows the texts, the final heterogeneous similarity between image and text is achieved as the probability of jointly belonging to the same category.

$p(l_i = c|I_i)$. $\sigma(z) = (1 + exp(-z))^{-1}$ is the sigmoid function, and $sim(I_i, I_k)$ is traditional similarity measure between two homogeneous data points.

The range of heterogeneous similarity of Equation 1 is $[0, 1]$, where 0 is achieved when none of the nearest neighbors of two media objects belong to the same category and 1 is achieved when all of the nearest neighbors belong to only one category.

3 Learning the Ranking Model through AdaRank

Since the similarity measure in Equation 2 can be any kind of similarity measure such as normalized correlation, histogram intersection, Chi square distance and so on. Each similarity measure could measure a certain aspect of relationship between two media objects and could be regarded as a weak ranker. Learning to combine multiple weak rankers is also a key component in retrieval task. In this paper, the ranking model for cross-media retrieval is learned through AdaRank[19], which is a kind of listwise approach[18] to learn to rank. Unlike

AdaBoost, AdaRank could train a ranking model that directly optimize the information retrieval performance measures such as MAP (Mean Average Precision) with respect to the training data, while the loss function in AdaBoost is specific for binary classification. Furthermore, AdaRank tries to optimize a loss function based on query lists while other learning algorithms for ranking attempt to minimize a loss function based on instance pairs. So we adopt AdaRank to learn the ranking model for cross-media retrieval task.

In the learning stage, a number of image (text) queries and their corresponding retrieved texts (images) are given. The relevance of the retrieved texts (images) with respect to the image (text) queries are also provided. The training set can be represented as $\mathcal{L}$=$\{(q_i, \mathbf{o}_i, \mathbf{y}_i)\}$, where q_i is the query, $\mathbf{o}_i = o_{i1}, o_{i2}, ..., o_{in(q_i)}$ is the list of objects with different modalities from the query, $\mathbf{y}_i = y_{i1}, y_{i2}, ..., y_{in(q_i)}$ is a list of labels , where $n(q_i)$ denotes the size of object list $\mathbf{o}_i$ and $\mathbf{y}_i$.

The objective of learning is to construct a ranking function which achieves the best results in ranking of the training data. The AdaRank algorithm is summarized in Algorithm 1. AdaRank runs T rounds and at each round it selects a weak ranker $h_t (t = 1, ..., T)$ with the lowest weighted error. Finally, it outputs a ranking model f by linearly combining the weak rankers.

Algorithm 1. Learning the ranking model through AdaRank

```
Require: $\mathcal{L}$=$\{(q_i, \mathbf{o}_i, \mathbf{y}_i)\}$, and parameter T.
Ensure: The ranking model $f$ by linear combining weak rankers.
Initialize weights $w_{1,i} = \frac{1}{m}$.
for t = 1 to T do
    Select weak ranker $h_t$ with minimum weighted error on training data $\mathcal{L}$:
        $$h_t = \underset{t}{\text{argmin}}\ \epsilon(h_t) \quad (3)$$
    Calculate the corresponding $\alpha_t$:
        $$\alpha_t = \frac{1}{2} \ln \frac{\sum_{i=1}^{m} w_{t,i}\{1 + E(\pi(q_i, \mathbf{o}_i, h_t), y_i)\}}{\sum_{i=1}^{m} w_{t,i}\{1 - E(\pi(q_i, o_i, h_t), y_i)\}} \quad (4)$$
    Update weight $w$ for each query:
        $$w_{t+1,i} = \frac{exp(-E(\pi(q_i, \mathbf{o}_i, f_t), y_i))}{\sum_{j=1}^{m} exp(-E(\pi(q_j, \mathbf{o}_j, f_t), y_j))} \quad (5)$$
    where $f_t(x) = \sum_{k=1}^{t} \alpha_k h_k(x)$.
end for
Output ranking model: $f(x) = f_T(x)$.
```

Initially, AdaRank sets equal weights to the training queries. In each round of iteration, training queries are re-weighted accordingly, where the queries that are not ranked well by f_t are emphasized. f_t is the ranking model created in the t^{th} round. As a result, the learning at the next round will be focused on those hard queries.

The effectiveness of a ranking model is usually evaluated with performance measures such as MAP (Mean Average Precision). The loss function to measure the weighted error of weak ranker h_t is defined as follows:

$$\epsilon(h_t) = \sum_{i=1}^{m} -w_{t,i} E(\pi(q_i, \mathbf{o}_i, h_t), y_i) \qquad (6)$$

where $w_{t,i}$ is the weight of query i in t^{th} round, $\pi(q_i, \mathbf{o}_i, h_t)$ is a permutation for query q_i on object list $\mathbf{o}_i$ by weak ranker h_t. E measures the consistency between π and labels y_i, and the performance measure of mean average precision(MAP) is adopted here. For each query, average precision is the average of precisions computed at the ranks where recall changes. Mean average precision is the mean value of a set of queries. It is widely used in the image retrieval literature[11].

4 Experiments

In this section, we compare the proposed approach with the state-of-the-art methods for cross-media retrieval.

4.1 Dataset Description

We evaluate the proposed approach on the Wikipedia dataset[12], which is chosen from the Wikipedia's "featured articles" This is a continually updated collection of 2700 articles that have been selected and reviewed by Wikipedia's editors since 2009. Each article is accompanied with one or more images from Wekimedia Commons. Both the texts and images are assigned a category label by Wekipedia. There are 29 categories in total. Since some of the categories are very scarce, 10 most populated categories are preserved in this dataset. Each article is split into several sections according to its section headings. Then the accompanied images are assigned to the sections respectively according to image position in the article. The final dataset contains a total of 2866 documents, which are text-image pairs and annotated with a label from the vocabulary of 10 semantic categories. The dataset is randomly split into a training set of 2173 documents and a test set of 693 documents.

Two cross-media retrieval tasks are considered: retrieve the texts using an image query and retrieve the images using a text query. For the former, each image in the test set is used as a query and the result is the ranking of all the texts in the test set. For the latter, each text is used as a query and the result is the ranking of all the images in the test set. The precision-recall (PR) curves and mean average precision (MAP) are taken as the performance measures.

Bag-of-words (BOW) model and topic model are utilized to represent the images and texts respectively. Each image is represented using a histogram of a 128-codeword SIFT codebook and each text is represented using a histogram of a 10-topic LDA text model. All of the compared cross-media retrieval methods in the experiment section adopt the same features and training data for fair comparison purpose. We set $k = 100$ for k nearest neighbors of HSNN. The homogeneous similarities to a media object were made to be zero-mean and unit-variance.

4.2 Contribution of Similarity Measure with Nearest Neighbors

Our first set of experiments examine the performance of heterogeneous similarity measure with nearest neighbors (HSNN) in Table 1. In this table, three methods [12] are compared, which are learning a homogeneous subspace for the modalities through canonical correlation analysis (CCA), learning a high level semantic (SMN) for each object and the combination of the two above algorithms (CCA + SMN) which firstly maps the original heterogeneous features into a homogeneous subspace and then learns the high level semantic feature for each object. Both CCA and SMN seek to find a homogeneous feature representation for media objects with different modalities. After the media objects are represented as homogeneous feature vectors, comparison could be performed by commonly used homogeneous similarity measures. Since our proposed HSNN could incorporate any kind of similarity measure, we evaluate objectively the MAP scores with multiple similarity measures, including the Normalized Correlation, Histogram Intersection and Chi square. The contributions of HSNN are clearly seen.

Table 1. Contribution of HSNN

Similarity measure	Experiment	Image Query	Text Query	Average
Normalized Correlation	CCA[12]	0.249	0.196	0.223
	SMN[12]	0.225	0.223	0.224
	CCA+SMN[12]	0.277	0.226	0.252
	HSNN	**0.311**	**0.246**	**0.278**
Histogram Intersection	CCA[12]	0.168	0.126	0.147
	SMN[12]	0.234	0.212	0.223
	CCA+SMN[12]	0.206	0.206	0.206
	HSNN	**0.318**	**0.242**	**0.280**
Chi square	CCA[12]	0.178	0.117	0.148
	SMN[12]	0.238	0.213	0.225
	CCA+SMN[12]	0.208	0.206	0.207
	HSNN	**0.309**	**0.243**	**0.276**

4.3 Contribution of Learning the Ranking Model through AdaRank

Next, we examine the performance of the ranking model trained through AdaRank in Table 2. Each similarity measure is regarded as a weak ranker, here 3 weak rankers are combined in total. It can be seen that AdaRank further improves the performance by linearly combining multiple weak rankers.

Table 2. Contribution of learning the ranking model through AdaRank

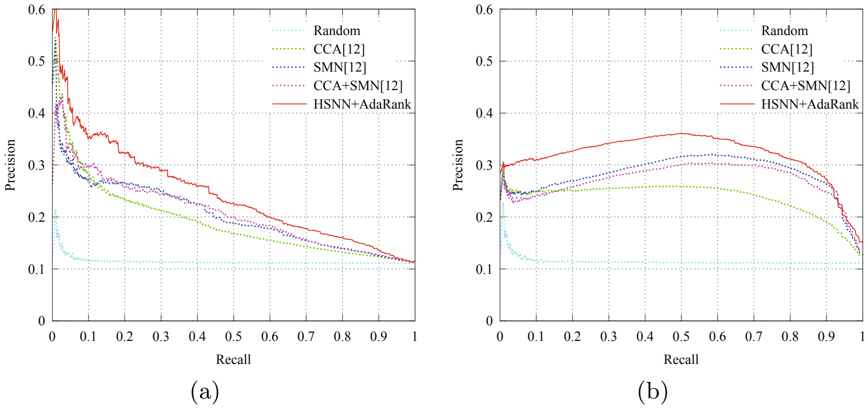
Experiment	Image Query	Text Query	Average
Normalized Correlation	0.311	0.246	0.278
Histogram Intersection	0.318	0.242	0.280
Chi square	0.309	0.243	0.276
AdaRank	**0.321**	**0.251**	**0.286**

4.4 Comparison with State-of-the-Art Methods

Table 3 shows the performance of our proposed HSNN + AdaRnnk , compared with the state-of-the-art methods as discussed in Section 4.2. Here Random means using randomly ranking images/texts as result. Figure 2 shows the PR curve of all of the above methods. It can be seen that HSNN attains higher precision at most levels of recall.

Table 3. Retrieval Performance(MAP Scores)

Experiment	Image Query	Text Query	Average
Random	0.118	0.118	0.118
CCA[12]	0.249	0.196	0.223
SMN[12]	0.225	0.223	0.224
CCA+SMN[12]	0.277	0.226	0.252
HSNN + AdaRank	**0.321**	**0.251**	**0.286**



(a) (b)

Fig. 2. Precision recall curves

4.5 Comparison with Image Retrieval System

In this section, we compare the cross-media retrieval with the unimedia image retrieval, where both query and retrieved result are images. Similar to [12], we adapt the cross-media retrieval to unimedia image retrieval in two ways. For the first method, a query image is complemented with a text article, then the text article is served as a proxy to retrieval the images in dataset; For the second method, the images in the dataset are also complemented by text articles, then the text articles in the database are retrieved and served as a proxy for the images in dataset.

Table 4. Content based image retrieval

Experiment	MAP Score
HSNN + AdaRank (Proxy Text Ranking)	**0.321**
HSNN + AdaRank (Proxy Text Query)	**0.251**
CCA + SMN [12] (Proxy Text Ranking)	0.277
CCA + SMN [12] (Proxy Text Query)	0.226
Image SMN [11]	0.161
Image SIFT Features [16]	0.135
Random	0.117

Table 4 shows the comparison of the cross-media retrieval approaches with a number of unimedia image retrieval methods. The method of [16] represents images as distributions of SIFT features and the method of [11] project the images to a semantic space. The MAP score of unimedia retrieval methods has shown the difficulty of image retrieval on this Wikipedia dataset. The cross-media retrieval methods significantly improve the performance. This indicates that by fully exploiting the rich information of multiple media types, we could understand the content of multimedia more accurately from a cross-media point of view.

5 Conclusion

In this paper, we have proposed a novel heterogeneous similarity measure with nearest neighbors (HSNN), which could compute the similarity between media objects with different media types. The heterogeneous similarity is obtained by computing the probability of two media objects belonging to the same class. Moreover, multiple similarity measures are regarded as the weak rankers. An effective ranking model for cross-media retrieval is learned from multiple weak rankers through AdaRank.

In the future, on one hand, we will jointly model other media types such as audio and video; on the other hand, the correlation between categories could also be explored.

Acknowledgements. The work described in this paper was fully supported by the National Natural Science Foundation of China under Grant Nos. 60873154 and 61073084, and the National Development and Reform Commission High-tech Program of China under Grant No. [2010]3044.

References

1. Carneiro, G., Chan, A., Moreno, P., Vasconcelos, N.: Supervised learning of semantic classes for image annotation and retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(3), 394–410 (2007)
2. Escalante, H., Hérnadez, C., Sucar, L., Montes, M.: Late fusion of heterogeneous methods for multimedia image retrieval. In: Proceeding of the 1st ACM International Conference on Multimedia Information Retrieval (2008)
3. Greenspan, H., Goldberger, J., Mayer, A.: Probabilistic space-time video modeling via piecewise gmm. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(3), 384–396 (2004)
4. He, X., Ma, W.Y., Zhang, H.J.: Learning an image manifold for retrieval. In: ACM international Conference on Multimedia (2004)
5. Jeon, J., Lavrenko, V., Manmatha, R.: Automatic image annotation and retrieval using cross-media relevance models. In: Proceedings of the 26th Annual International ACM SIGIR Conference (2003)
6. Pan, J., Yang, H., Faloutsos, C., Duygulu, P.: Automatic multimedia cross-modal correlation discovery. In: ACM International Conference on Knowledge Discovery and Data Mining, SIGKDD (2004)
7. Paramita, M., Sanderson, M., Clough, P.: Diversity in photo retrieval: overview of the imageclefphoto task 2009. CLEF Working Notes (2009)
8. Peng, Y., Ngo, C.W.: Clip-based similarity measure for query-dependent clip retrieval and video summarization. IEEE Transactions on Circuits and Systems for Video Technology 16(5), 612–627 (2006)
9. Peng, Y., Yang, Z., Xiao, J.: Audio retrieval by segment-based manifold-ranking. In: IEEE International Conference on Multimedia and Expo. (2009)
10. Qian, G., Sural, S., Gu, Y., Pramanik, S.: Similarity between euclidean and cosine angle distance for nearest neighbor queries. In: ACM SIG Symposium on Applied Computing (2004)
11. Rasiwasia, N., Moreno, P., Vasconcelos, N.: Bridging the gap: Query by semantic example. IEEE Transactions on Multimedia 9(5), 923–938 (2007)
12. Rasiwasia, N., Costa Pereira, J., Coviello, E., Doyle, G., Lanckriet, G.R.G., Levy, R., Vasconcelos, N.: A new approach to cross-modal multimedia retrieval. In: ACM International Conference on Multimedia (2010)
13. Smeaton, A.F., Over, P., Kraaij, W.: Evaluation campaigns and trecvid. In: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval (2006)
14. Tsikrika, T., Kludas, J.: Overview of the wikipediamm task at imageclef 2009. In: Working Notes for the CLEF 2009 Workshop (2009)
15. Typke, R., Wiering, F., Veltkamp, R.C.: A survey of music information retrieval systems. In: Proceedings of ISMIR (2005)
16. Vasconcelos, N.: Minimum probability of error image retrieval. IEEE Transactions on Signal Processing 52(8), 2322–2336 (2004)

17. Wu, Y., Zhuang, Y., Pan, Y.: Contentbased video similarity model. In: Proceedings of the Eighth ACM International Conference on Multimedia, pp. 465–467 (2000)
18. Xia, F., Liu, T., Wang, J., Zhang, W., Li, H.: Listwise approach to learning to rank - theory and algorithm. In: Proceedings of the 25th International Conference on Machine Learning (2008)
19. Xu, J., Li, H.: A boosting algorithm for information retrieval. In: The 30th Annual International ACM SIGIR Conference (2007)
20. Yang, Y., Xu, D., Nie, F., Luo, J., Zhuang, Y.: Ranking with local regression and global alignment for cross media retrieval. In: ACM International Conference on Multimedia, pp. 175–184 (2008)
21. Yang, Y., Zhuang, Y., Wu, F., Pan, Y.: Harmonizing hierarchical manifolds for multimedia document semantics understanding and cross-media retrieval. IEEE Transactions on Multimedia 10(3), 437–446 (2008)
22. Zhuang, Y., Yang, Y., Wu, F.: Mining semantic correlation of heterogeneous multimedia data for cross-media retrieval. IEEE Transactions on Multimedia 10(2), 221–229 (2008)

TempoM^2: A Multi Feature Index Structure for Temporal Video Search*

Mario Döller, Florian Stegmaier, Simone Jans, and Harald Kosch

Department of Distributed Information Technology
University of Passau, Germany
{forename.surname@uni-passau.de}

Abstract. Efficient temporal video search will play an important role in the future related to the vast growth of video data in the Web. Here, index structures are one way to ensure efficient retrieval over a large amount of data. However, index structures targeting on indexing video data are rare. In this context, the paper introduces the $TempoM^2 - tree$ framework, which features a two-level index structure supporting the retrieval of similar video segments in combination with temporal relations.

Keywords: Temporal Video Index, Multimedia retrieval, MPEG-7.

1 Introduction

Due to the massive increase of audiovisual data in the Web[1], multimedia search in general and video search in particular is a very hot topic in the scientific literature. For instance in the domain of soccer videos, search requests such as *"Give me all video segments where a free kick is followed by a goal"* are of particular interest. To evaluate such type of requests, two essential steps are necessary. On the one side, video segments have to be detected representing the desired concepts [14] and on the other side the temporal relation between those video segments have to be evaluated. Research according to the first step fall into the domain of content based image/video search, where low/mid level features are used for similarity search. Research in evaluating temporal relations is often expressed by string matching algorithms [9].

Another important factor for efficient querying is the use of index structures coping with the tremendous amount of data in an effective way. In the past, many index structures for high-dimensional data have been introduced in order to manage indexing of features. As most of those only allow addressing one certain feature (e.g., color) at a time, special cases targeting also on multi-feature indexing have been invented. However, those index structures are only useable for detecting similar video segments but do not support the evaluation of temporal relations between segments.

[*] Many thanks to Marco Patella for providing us the source code of the M2 index structure.

[1] http://royal.pingdom.com/2011/01/12/internet-2010-in-numbers/

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 323–333, 2012.

In this context, the paper presents a novel indexing framework (called $TempoM^2 - tree$) for temporal search over similar video segments. The framework builds on a two level architecture, where the first level operates on a multilevel search tree (we used the M^2tree) for detecting similar video segments. Then, the second level provides a structure which represents the temporal relationships among the annotated video segments.

The remainder of this paper is organized as follows: Section 2 introduces related work in the area of multi-feature index structures and methods for temporal search. This is followed by the presentation of our $TempoM^2$-Index framework in Section 3. The evaluation of the novel index framework is conducted in Section 4 and finally, the article is concluded in Section 5.

2 Related Work

Fast search over multimedia data is very often realized by utilizing index structures [16], which are tuned for the retrieval in high dimensional feature vectors. Besides, the well known k-NN (Nearest-Neighbor) or epsilon similarity, temporal requests are another important multimedia search paradigm.

Due to the fact that our work bases on multi-feature index structures, subsection 2.1 will survey recent works. This is followed by a literature survey on temporal search methods in 2.2.

2.1 Multi Feature Access Methods

One differentiating factor of access methods is the amount of semantically different feature vectors a method is capable of indexing at a certain time. Let us consider a content-based image retrieval (CBIR) scenario where similar images are identified by multiple low-level features (e.g., color and edges). Most of the existing techniques index each low-level feature separately. However, this demands a further ranking of possible result items at a later stage which requires additional processing overhead. The second possibility consider multi-feature indexing such as the work of Ngu et al. [12] that rely on a single M-tree [4] for all features. Due to the identified weak points such as problematic performance for high dimensional vectors (dim>20) or the use of a neural network training process, the authors in [8] proposed a new ASAP (Adaptive Searching by Aggressive Partial-distance) algorithm. Their method introduced a novel representation for f-feature points into a 2D vector and a bit signature. Based on this representation the classic access method B^+-tree is used to operate on the 2D vectors. Then, the bit signature filters the identified possible result items by the dimensional level.

In addition to the modification of the data item itself (e.g., by finding a novel representation for the feature vector) further work deal with structural modifications at the tree level. One example is the MOSAIC-tree [6] where multiple features are indexed by manipulating the leaf nodes. Here, for every level (representing a feature) a tree based multi-dimensional access method is used and the leaf nodes of the upper level point to the root of the successor tree indexing the next level. Another example basing on the M-tree family is the M^2-tree [3] which supports retrieval over a multi-feature representation by a modified internal structure and searching algorithms.

Recently, the MFI-tree [7] (acronym for MultiFeature index tree) has been introduced. MFI is a hierarchical tree holding two kinds of nodes (leaf and cluster nodes). The cluster nodes support pruning and should avoid the *curse of dimensionality* (which states that for high dimensions a sequential scan outperforms the use of the index tree) effect by being optimized for browsing search.

2.2 Access Methods for Temporal Search

Indexing data in terms of temporal relationships has a long tradition and became according to video search use cases (events, moving objects, etc.) new importance. In this context, two comprehensive surveys [11,13] have been published recently covering the period from 1980 to 2010. Most of the investigated access methods in the surveys are used for time stamp or time interval requests. For video databases those kind of requests are fairly rare as a user would need to know the temporal chain of the videos. So, the user would need to give a concrete time point or time interval where interesting events may occur. In video search, user are more interested on the content of some video scenes and then their temporal relations.

In this context, a similar approach related to our work has been introduced in [2]. The MINDEX index structure is a multilevel access method for the search of salient objects in video data. The authors combine hash tables holding the *ID*s of salient objects with a B^+-tree representing the temporal relationships among those *ID*s. However, to the authors best knowledge, none of the found works combine multi-feature access methods for temporal search as proposed in our work.

3 $TempoM^2$-Index Framework

3.1 Data Model

The data model of the $TempoM^2$ tree for video search considers video segments VS_x according to the following common characteristics (relied on the MPEG-7 standard [10]):

- VS_x is defined by a time interval $[VS_x.start, VS_x.end]$, which describes the covered time period within the whole video V.
- Every video segment VS_x has a content description (e.g., multiple low level features, textual description, etc.).
- A video segment VS_x can contain further video segments, whereas the time period of every internal video segment must be covered by VS_x.

A recursive view on the given definition of video segments results in a tree structure, which holds the following conditions: The nodes in the tree can have arbitrary child nodes. The nodes at the same level can have overlapping time periods or cause temporal holes. The set of video segments does not degrate to a graph.

3.2 $TempoM^2$-Tree

The structure of the $TempoM^2$-tree consists of two levels (see figure 1). The first level bases on the multi-feature M^2-tree [3], which is used for evaluating content based search of the source and/or target node of the temporal relation. In our tests the content description of a video segment belongs to visual features such as *ScalableColor* or *EdgeHistogram*. However, the framework is independent in terms of the used features. In order to evaluate the example query of the introduction, semantic concepts (e.g., goal) have to be derived, for example by the use of low level features [15]. The leaf nodes of the M^2-tree hold the combined feature vectors of the video segments ($VS_i, i \in \mathbb{N}$) and point to their respective representation at the second level of the $TempoM^2$-tree structure. The intermediate nodes (N_i, SN_i) of the M^2-tree consist of the combined features vectors and a covering radii $r_i^{[N]}$ (see [3] for detailed information).

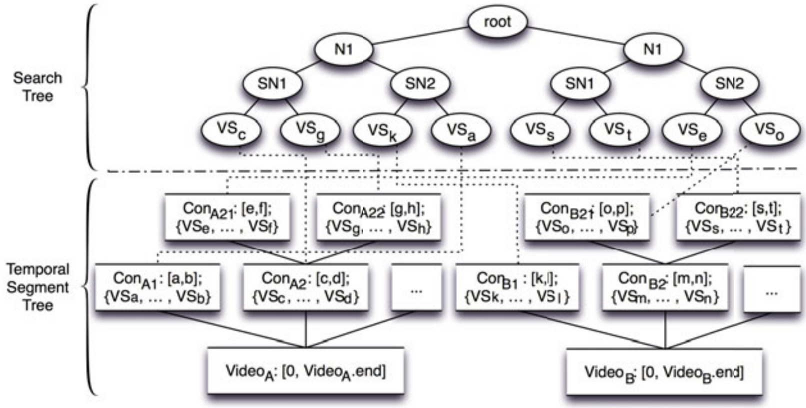

Search Tree
Temporal Segment Tree
root
N1
N1
SN1
SN2
SN1
SN2
VSc
VSg
VSk
VSa
VSs
VSt
VSe
VSo
ConA21: [e,f]; {VSe, ... , VSf}
ConA22: [g,h]; {VSg, ... , VSh}
ConB21: [o,p]; {VSo, ... , VSp}
ConB22: [s,t]; {VSs, ... , VSt}
ConA1: [a,b]; {VSa, ... , VSb}
ConA2: [c,d]; {VSc, ... , VSd}
...
ConB1 : [k,l]; {VSk, ... , VSl}
ConB2: [m,n]; {VSm, ... , VSn}
...
VideoA: [0, VideoA.end]
VideoB: [0, VideoB.end]

Fig. 1. Structure of the $TempoM^2$-tree

The second level represents the temporal relations in a modified video segment tree. Based on the given data model the video segments of a video $V = \{VS_1, VS_2, ..., VS_n\}, n \in \mathbb{N}$ feature due to overlapping and temporal wholes a partial order $(VS_x, VS_y \in V \wedge VS_x \neq VS_y | VS_x \preceq VS_y)$. See figure 2 for an example.

In order to support fast evaluation of temporal relations a total order of video segments is beneficial. For instance, in a video segment tree with partial order, the overlapping of video segments leads to the necessity of traversing the tree in multiple directions (e.g., right or left and up or down). In order to avoid such circumstances, a new type of node is introduced, named container node,

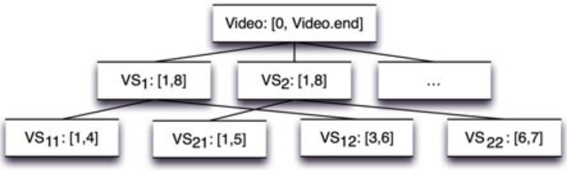


Fig. 2. Example partial order video segment tree

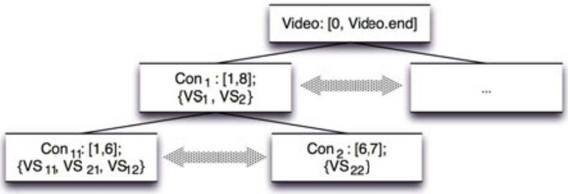


Fig. 3. Example total order video segment tree

which contains all neighbored video segments that overlap at a certain level (see figure 3 which represents the modified segment tree based on figure 2).

The definition of a container node is as follows: A container node also has a time interval $[Con_j.start, Con_j.end]$, where for all video segments covered by Con_j the following holds: $\forall VS_i \in Con_j : VS_i.start \geq Con_j.start \wedge VS_i.end \leq Con_j.end$. Furthermore, every container node provides information (by the variables *meetMark* and *meetByMark*), whether there is a temporal gap between the left or right neighbors. Besides, the following conditions are introduced between neighboring container nodes:

- Let Con_N be a parent node and Con_X its child node, then the following must hold: $Con_N.start \leq Con_X.start \wedge Con_N.end \geq Con_X.end$. This implies that the time interval of the child node must be completely spanned by the parent node
- Let Con_N be the left neighbor node at the same level then the following holds: $Con_N.end \leq Con_X.start$. This means there are no time-related overlaps. The same condition holds for the right neighbors.

Based on the introduced structure the temporal relations defined by Allen [1] are now implicitly expressed (by navigating in only one direction, either horizontal or vertical in the tree structure), which makes temporal retrieval operations more efficient.

3.3 Search in the $TempoM^2$-Tree

Search within the $TempoM^2$-tree is triggered by three input parameters, namely *targetResource, sourceResource* and *relationType*. The connection between those three parameters is highlighted in figure 4. By elaborating the example request in section [1], the *sourceResource* relies to video segments showing *free kicks*, the *targetResource* relies to *goal scenes* and the temporal relation would be expressed by the semantics of *precedes*.

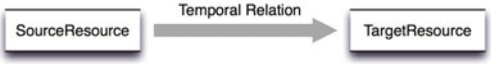


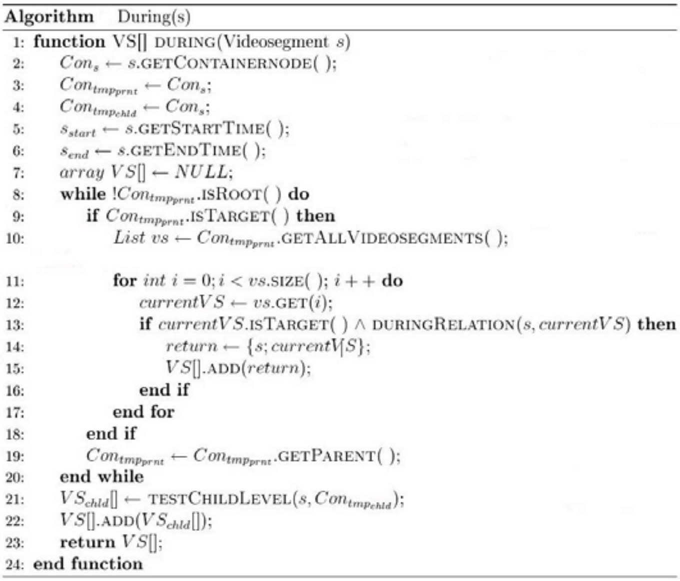
Fig. 4. Connection of input parameters for the $TempoM^2$-tree

To identify similar video segments, the *targetResource* and *sourceResource* parameters hold descriptions of video features (e.g., the color distribution in a video segment), whereas only the *sourceResource* is mandatory. We used the MPEG-7 standard for the representation of low level features. The *relationType* holds the desired temporal relationship and relies on Allen's definitions [1]. The input parameters are leaned against retrieval requests that base on the MPEG Query Format [5] to formulate temporal search. However, the index environment is not stuck together and can be used separately. In the following, the process of search is highlighted in detail. Note, the following description is elaborated for the case when both parameters (*targetResource* and *sourceResource*) are given. If only the *sourceResource* is apparent then the search is simplified as all video segments are returned that follow, depending on the temporal relation, the respective search direction in the modified segment tree.

Similarity Search. The first operation in the evaluation of our temporal search request is a similarity search in the $M^2 - tree$ for the given *sourceResource* and *targetResource* parameters, respectively. This results in two sets of video segments $S = \{VS_i, ..., VS_m\}$ for the *sourceResource* and $T = \{VS_j, ..., VS_n\}$ for the *targetResource* with $i, j, n, m \in \mathbb{N}$, which directly point to the nodes in the modified segment tree. Note, the two sets may not disjunct and T is only computed iff the *targetResource* parameter is given.

Temporal Search. After receiving the sets of video segments (S, T) the process of navigation through the modified segment tree is performed. The direction of navigation is determined by the temporal relation given in *relationType*. Due to the semantic (total order) of our modified segment tree the search direction can be either have a horizontal (in case of *follows, meets, precedes, metBy*) or vertical (in case of *starts, finishes, overlaps, contains, startedBy, finishedBy, overlappedBy* and *during*) orientation.

As an example, this paper presents the generic algorithm for the vertical cases in more detail. For all elements in set S the algorithm `During` is executed. After initializing internal variables (line 2 to 7), the search traverses from the given video segment to the root of the tree (line 8 to 20). In case, the traversal meets a container node that is in the target set T (see line 9) all stored video segments are investigated, whether they are marked and hold the temporal relation condition (line 10 to 18). Marked means in this case, that the video segment is in the list



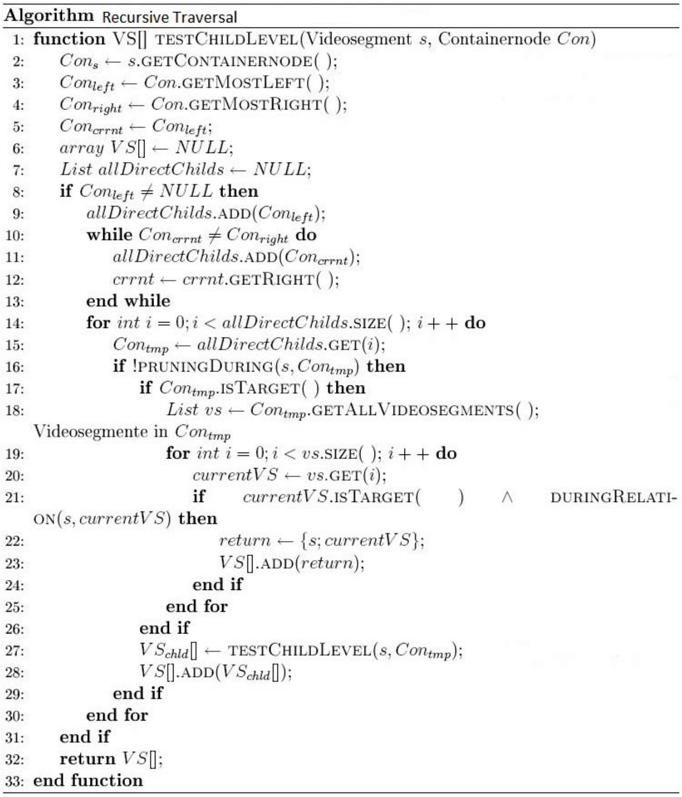
```
Algorithm During(s)
1: function VS[] DURING(Videosegment $s$)
2:     $Con_s \leftarrow s$.GETCONTAINERNODE( );
3:     $Con_{tmpprnt} \leftarrow Con_s$;
4:     $Con_{tmpchld} \leftarrow Con_s$;
5:     $s_{start} \leftarrow s$.GETSTARTTIME( );
6:     $s_{end} \leftarrow s$.GETENDTIME( );
7:     $array\ VS[] \leftarrow NULL$;
8:     while !$Con_{tmpprnt}$.ISROOT( ) do
9:         if $Con_{tmpprnt}$.ISTARGET( ) then
10:            $List\ vs \leftarrow Con_{tmpprnt}$.GETALLVIDEOSEGMENTS( );

11:            for $int\ i = 0; i < vs$.SIZE( ); $i++$ do
12:                $currentVS \leftarrow vs$.GET($i$);
13:                if $currentVS$.ISTARGET( ) $\wedge$ DURINGRELATION($s, currentVS$) then
14:                    $return \leftarrow \{s; currentV[S\}$;
15:                    $VS[]$.ADD($return$);
16:                end if
17:            end for
18:        end if
19:        $Con_{tmpprnt} \leftarrow Con_{tmpprnt}$.GETPARENT( );
20:    end while
21:    $VS_{chld}[] \leftarrow$ TESTCHILDLEVEL($s, Con_{tmpchld}$);
22:    $VS[]$.ADD($VS_{chld}[]$);
23:    return $VS[]$;
24: end function
```

of set T. The combination of video segments that fulfill both requirements are added to the result list.

Afterwards, the successors of the current node are tested recursively (see line 21) which leads to a call to the `RecursiveTraversal` algorithm. Here, starting from the outermost left and right nodes of the given container node a search is started (line 8 to 13), which leads to a test of all extracted video segments (after a pruning phase in line 16). The algorithms presented can be used for all other vertical related temporal relations by simply exchanging the pruning method as well as the temporal test condition. See the listing below for a set of pruning criteria for the temporal relationships targeting on vertical search direction.

Pruning Criteria. Depending on the elaborated temporal direction, different pruning rules can be applied.

- *during:* all container nodes Con_N and their subtrees can be ignored for a video segment s iff the following holds: $Con_N.end \leq s.start \vee Con_N.start \geq s.end$.
- *overlaps:* all container nodes Con_N and their subtrees can be ignored for a video segment s iff the following holds: $s.end \geq Con_N.end$.
- *overlappedBy:* all container nodes Con_N and their subtrees can be ignored for a video segment s iff the following holds: $s.start \leq Con_N.start$.
- *starts:* all container nodes Con_N and their subtrees can be ignored for a video segment s iff the following holds: $s.end \geq Con_N.end \vee s.start \leq Cond_N.start$
- *startedBy:* all container nodes Con_N and their subtrees can be ignored for a video segment s iff the following holds: $s.start \leq Cond_N.start$
- *finishes and finishesBy:* all container nodes Con_N and their subtrees can be ignored for a video segment s iff the following holds: $s.end \geq Cond_N.end$



```
Algorithm Recursive Traversal
function VS[] TESTCHILDLEVEL(Videosegment $s$, Containernode $Con$)
    $Con_s \leftarrow s$.GETCONTAINERNODE( );
    $Con_{left} \leftarrow Con$.GETMOSTLEFT( );
    $Con_{right} \leftarrow Con$.GETMOSTRIGHT( );
    $Con_{crrnt} \leftarrow Con_{left}$;
    $array\ VS[] \leftarrow NULL$;
    $List\ allDirectChilds \leftarrow NULL$;
    if $Con_{left} \neq NULL$ then
        $allDirectChilds$.ADD($Con_{left}$);
        while $Con_{crrnt} \neq Con_{right}$ do
            $allDirectChilds$.ADD($Con_{crrnt}$);
            $crrnt \leftarrow crrnt$.GETRIGHT( );
        end while
        for $int\ i = 0; i < allDirectChilds$.SIZE( ); $i++$ do
            $Con_{tmp} \leftarrow allDirectChilds$.GET($i$);
            if !PRUNINGDURING($s, Con_{tmp}$) then
                if $Con_{tmp}$.ISTARGET( ) then
                    $List\ vs \leftarrow Con_{tmp}$.GETALLVIDEOSEGMENTS( );
Videosegmente in $Con_{tmp}$
                    for $int\ i = 0; i < vs$.SIZE( ); $i++$ do
                        $currentVS \leftarrow vs$.GET($i$);
                        if $currentVS$.ISTARGET( ) $\wedge$ DURINGRELATION($s, currentVS$) then
                            $return \leftarrow \{s; currentVS\}$;
                            $VS[]$.ADD($return$);
                        end if
                    end for
                end if
                $VS_{chld}[] \leftarrow$ TESTCHILDLEVEL($s, Con_{tmp}$);
                $VS[]$.ADD($VS_{chld}[]$);
            end if
        end for
    end if
    return $VS[]$;
end function
```

4 Evaluation

The test data has been created by the use of our semi-automatic video annotation tool (VAnalyzer[2]) for soccer videos. The data set consists of 19 videos consisting of more than 6500 video segments which are annotated by the MPEG-7 standard. In order to increase the amount of test data, some of the extracted video segments have been manually duplicated and rearranged to new videos in order to have a final amount of 10000 video segments for testing. Note, the extension of the amount of video segments serve as basis for the time performance evaluation only and is not accurate (and has not performed) in case of evaluating the quality of search results.

For the content based retrieval, we used the MPEG-7 based *ScalableColor* and *EdgeHistogram* descriptors. Unfortunately, we could not evaluate our access method in comparison to the MINDEX tree (which is the one going in a similar direction) as the authors were not able to provide us their source code.

[2] http://www.dimis.fim.uni-passau.de/iris

4.1 Inserting Data

Figure 5 illustrates the average time per video segment for an insertion process. The measured time includes the parsing of the MPEG-7 document, the insertion into the $M^2 - tree$ and finally the insertion into the modified segment tree. One can observe that the performance of the underlying modified segment tree is stable whereas the necessary amount of time per video segment increases in the $M^2 - tree$ by the amount of data.

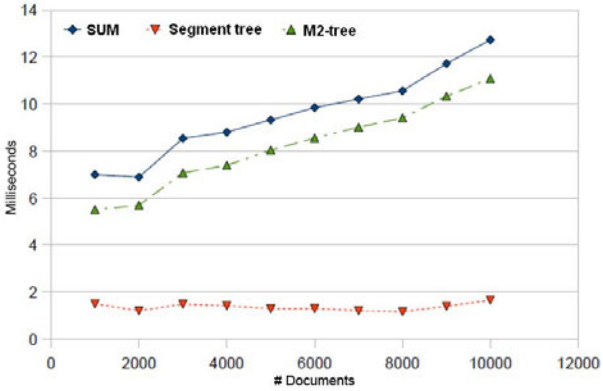


Fig. 5. Average time per video segment for inserting video segments

4.2 Temporal Search

A complete search in the $TempoM^2 - tree$ covers three essential parts. First, the input request (in form of a MPEG Query Format temporal query) is parsed and the individual parameters are extracted. Then, similarity searches for the given source and target resources are executed (in case both are available) and finally

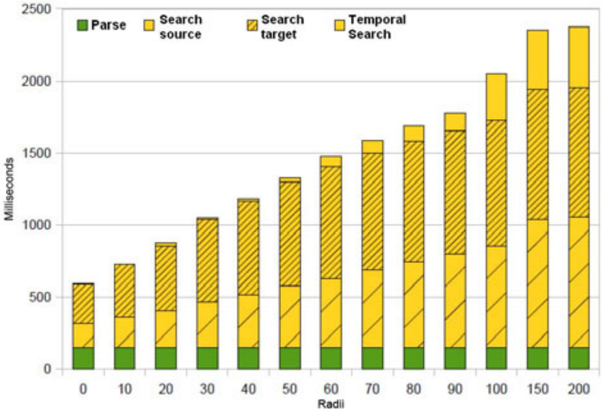


Fig. 6. Average time of a range search for the *follows* operation

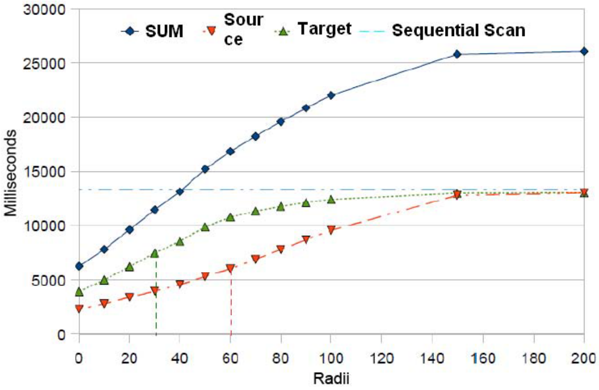


Fig. 7. Distance calculation for the *follows* operation

a temporal search over the two resulting sets in the modified segment tree is performed. Figure 6 demonstrates the average time consumption in milliseconds (y-axis) of those operations for a *follows* temporal request.

The most time is consumed by the similarity search operations but in contrast to that, the navigation within the modified segment tree is fast. Of course with increasing radii the amount of video segments that have to be combined increases as well. However, a too large radius is not sufficient for the consumer as well as the amount of results in the result set would be too high.

A further reason for avoiding too high radii is demonstrated in Figure 7. Here one can see that a sequential scan outperforms the combination of similarity and temporal search at the level of around 40. This figure evaluated the amount of distance calculations that are necessary to find all solutions.

5 Conclusion and Future Work

This article presented a novel access method for supporting efficient temporal search over video segments. One of the main features of the index method is its combination with a multi-feature index tree supporting similarity search over video segments and its further filtering according to temporal relations. The video segments are annotated by means of low level features specified in the MPEG-7 standard. Besides, the access method supports two different search modes depending on the amount of given parameters. The starting video segments are detected by a similarity search over the multi-feature index tree.

Future work will consider on the one side indexing of high-level representations related to semantic concepts or objects in order to increase the accuracy of found video segments like presented in [15]. Moreover, on the other side, the use and the integration of the index environment into multimedia databases and their applications is planned.

References

1. Allen, J.F.: Maintaining Knowledge about Temporal Intervals. Communications of the ACM 26(11), 832–843 (1983)
2. Chen, L., Özsu, M.T., Oria, V.: Mindex: An efficient index structure for salient object-based queries in video database. Multimedia Systems 10(1), 56–71 (2004)
3. Ciaccia, P., Patella, M.: The M2-Tree: Processing Complex Multi-Feature Queries with Just One Index. In: Proceedings of the First DELOS Network of Excellence Workshop on Information Seeking, Searching and Querying in Digital Libraries, Zurich, Switzerland (2000)
4. Ciaccia, P., Patella, M., Zezula, P.: M-tree: An efficient Access Method for Similarity Search in Metric Spaces. In: Proceedings of the 23rd Int. Conf. on Very Large Data Bases (VLDB), Athens, Greece, pp. 426–435. Morgan Kaufmann (1997) ISBN: 1-55860-470-7
5. Döller, M., Tous, R., Gruhne, M., Yoon, K., Sano, M., Burnett, I.S.: The MPEG Query Format: On the way to unify the access to Multimedia Retrieval Systems. IEEE Multimedia 15(4), 82–95 (2008)
6. Goh, S.-T., Tan, K.-L.: MMOSAIC: A Multi-Feature Access Method for Large Image Databases. In: Bench-Capon, T.J.M., Soda, G., Tjoa, A.M. (eds.) DEXA 1999. LNCS, vol. 1677, pp. 862–871. Springer, Heidelberg (1999)
7. He, Y., Junqing, Y.: MFI-tree: An effective multi-feature index structure for weighted query application. Computer Science and Information Systems 7, 139–152 (2010)
8. Jagadish, H.V., Ooi, B.C., Shen, H.T., Tan, K.-L.: Toward efficient multifeature query processing. IEEE Transaction on Knowledge and Data Engineering 18, 350–362 (2006)
9. Lin, C.-H., Chen, A.L.P.: Approximate Video Search Based on Spatio-Temporal Information of Video Objects. Asian Journal of Health and Information Sciences 3(1-4), 52–68 (2008)
10. Martinez, J.M., Koenen, R., Pereira, F.: MPEG-7. IEEE Multimedia 9(2), 78–87 (2002)
11. Mokbel, M.F., Ghanem, T.M., Aref, W.G.: Spatio-temporal access methods. IEEE Data Engineering Bulletin 26(1), 40–49 (2003)
12. Ngu, A.H.H., Sheng, Q.Z., Huynh, D.Q., Lei, R.: Combining multivisual features for efficient indexing in a large image database. VLDB Journal 9, 279–293 (2001)
13. Nguyen-Dinh, L.-V., Aref, W.G., Mokbel, M.F.: Spatio-Temporal Access Methods: Part 2 (2003 - 2010). IEEE Data(base) Engineering Bulletin 33(1), 46–55 (2010)
14. Weng, M.-F., Chuang, Y.-Y.: Multi-cue fusion for semantic video indexing. In: Proceedings of the 16th International Conference on Multimedia 2008, Vancouver, British Columbia, Canada, October 26-31, pp. 71–80 (2008)
15. Zampoglou, M., Papadimitriou, T., Diamantaras, K.I.: From low-level features to semantic classes: Spatial and temporal descriptors for video indexing. Signal Processing Systems 61(1), 75–83 (2010)
16. Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity Search - The Metric Space Approach. Series: Advances in Database Systems, vol. 32, p. 220. Springer, Heidelberg (2006) ISBN: 0-387-29146-6

Towards Automatic Detection of CBIRs Configuration

Christian Vilsmaier, Rolf Karp, Mario Döller,
Harald Kosch, and Lionel Brunie

University of Passau, Chair of Distributed and Multimedia
Information Sytems, Innstrasse 43, 94032 Passau, Germany
and
INSA de Lyon, LIRIS, Campus de la Doua, Batiment Blaise Pascal,
20 Avenue Albert Einstein, 69621 Villeurbanne Cedex, France
{Christain.Vilsmaier,Lionel.Brunie}@insa-lyon.fr,
{Harald.Kosch,Mario.Doeller}@uni-passau.de,
Rolf.Karp@googlemail.com

Abstract. Many *Content Based Image Retrieval systems* (CBIRs) have been invented in the last decade. The general mechanism of the search process is very similar for each of these CBIRs, and the calculation of rankings is determined by the comparison of features (low-, mid-, high-level). Nevertheless, all things being equal, the respective realization leads to different results. Knowledge about the internal configuration (used features, weights and metrics) of these systems would be beneficial in many usage scenarios (e.g., by using a query image content sensitive query forwarding strategy or improved result ranking strategies in meta search engines). In this context, the paper presents an approach that supports an automatic detection of the configuration of CBIR systems. We demonstrate that the problem can be partly traced back to an optimization problem and tested several optimization algorithms. The approach has been evaluated based on the ImageCLEF test set and shows good results.

Keywords: CBIRs configuration, Image Database, Low-level Feature detection.

1 Introduction and Background

Due to the digitalization and miniaturization of cameras as well as their integration into mobile phones, the amount of digitally available images has increased tremendously in the last decade. In order to make those images searchable based on content, CBIR has received a lot of attention in the last several years. The theoretical background which was developed during these years was exhaustively explored in [2] and [11] and various systems implementing CBIR were examined in [14] and [7]. A CBIR system (CBIRs) allows searching of images based on their extracted features (low-, mid-, high-level, see [15] for details) instead of textual descriptions. The retrieval process matches the extracted features of the stored images to those of a query image, thereby calculating its score and rank which subsequently results in a list of best matches. The general mechanism of

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 334–345, 2012.

the search process is very similar for all of these CBIR systems. Nevertheless, they use different combinations of features, feature metrics and feature weights so that their search behavior and results differ.

In the context of meta-search engines [12] that provide access to multiple heterogeneous retrieval systems, the knowledge of their internal configuration would be beneficial. For instance such knowledge could allow the meta-search engines to establish an image content sensitive CBIR selection. Thus, CBIR systems that would not be of additional value to the query response can be ignored altogether. Furthermore, the gathered information can be used to improve the result aggregation process of the different retrieved result sets. Imagine examples of meta-search engines, for instance, in the domain of art galleries that provide search facilities of their works of art.

Related to this, the paper proposes a novel approach for an automatic detection of the configuration of CBIR systems. The detection process is based upon the analysis of a small set of test queries that are executed on the CBIR system in question. The analysis uses an optimization algorithm and filter strategies in order to identify the best feature/weight combination.

The remainder of this paper is organized as follows: In section 2 related work is discussed. In section 3 and 4 the assumptions and the developed approach are explained. In Section 5 the methods of evaluation as well as their results are analyzed. Finally in section 6 results and future work are discussed.

2 Related Work

The search in image repositories is a very active research field and many retrieval techniques and frameworks have been proposed in the past (see exhaustive surveys [14,7]). In this context, several articles focused on the qualitative evaluation of those techniques. For instance, in [4] the authors proposed an objective method for evaluating image content by means of visual content words as basis vectors for similarity calculations. Another important initiative in this domain is the ImageCLEF benchmark [10], which provides an annotated image test set.

Furthermore, in the literature detailed analyses can be found that investigate the individual features [15] and searches for correlations among them. Consequently, recommendations are given as to which features perform well for certain types of data [13].

Very little work related to the topic of our paper is available. For instance, the implementation of [1] presented a peer to peer approach for a self-organized image retrieval network. However, although the feasibility of an image retrieval network has been shown, little attention has been paid to identifying the configuration of the associated CBIR systems. The authors in [9] describe an algorithm for obtaining knowledge about the importance of features by analyzing user log files of the VIPER system. In their approach, features which are frequently present in images marked as positive by users receive a higher weighting. Moreover, their technique was used to improve retrieval quality due to a novel relevance feedback technique. However, their work relies on access to query logs and the internals of the system, which are not available in our case. It should also be noted that our proposed approach does not aim at evaluating the quality of a CBIRs but at automatically detecting the used configuration.

3 Methodology

Current CBIR systems primarily use (low-level) features to compare query images to images that are stored in their system. Formula 1 shows the internal representation of an image i in a CBIRs α:

$$rep_{\alpha}(i) = \{f_{\mathrm{j}}(i) | j \in \{1..|F_{\alpha}|\}\} \tag{1}$$

The image i is represented as a set of feature vectors $f_{\mathrm{j}}()$, which are extracted from i. These features of the feature set F_{α} are digital representations such as color, edge or shape characteristics. Mixtures of such characteristics can also be represented in a feature. The used set of features may be different for every CBIR system. Even two retrieval systems which are using an equal set of features could reply differently to the same query as they might assign different weights or distance functions to their features. Formula 2 formalizes the *configuration* of CBIRs$_{\alpha}$ as a 3-tuple. This tuple consists of a set of features F$_{\alpha}$, a set of feature metrics $\mathit{\Delta}_{\alpha}$ and a set of feature weights W$_{\alpha}$.

$$config(CBIRs_{\alpha}) = (W_{\alpha}, \mathit{\Delta}_{\alpha}, F_{\alpha}) \tag{2}$$

The calculation of CBIRs α's score respective to the query image q and a stored image i' uses the features, weights and metrics of this configuration. It is defined as follows: Let $score_{\alpha,\mathrm{i'}}(q)$ be the score of α respective to query image q and a stored image i' and let $\delta_{\alpha\mathrm{j}} \in \mathit{\Delta}_{\alpha}$ be the distance function of feature $f_{\mathrm{j}}()$. Furthermore, let $w_{\alpha\mathrm{j}} \in W_{\alpha}$ be the weight assigned by the system α to feature $f_{\mathrm{j}}()$. The score of image i' regarding the query image q is then calculated as follows:

$$score_{\alpha,\mathrm{q}}(i') = \sum_{j=1}^{|F_{\alpha}|} w_{\alpha\mathrm{j}} \delta_{\alpha\mathrm{j}}(f_{\mathrm{j}}(i'), f_{\mathrm{j}}(q)) \tag{3}$$

4 Approach Outline

As illustrated in section 1, our aim is an automatic detection of the configuration of a CBIRs. In this context the internal representation and the supported classification cases are presented in subsection 4.1. The overall process is then described in subsection 4.2 whereas the details of the algorithm are shown in subsection 4.3.

4.1 Feature Distribution

The basis of our approach is the exploitation of a rich set of implemented and well known features and feature metrics. However, for reasons of simplification the description of the implementation explained in this article focuses on the features. Nevertheless, a metrics detection mechanism has also been integrated in our implementation and can be used by assigning multiple metrics to a feature. The internal representation of our system is analogous to that of the CBIRs introduced in section 3, methodology. Formula 4 shows the internal representation Rep_ω of an image i:

$$Rep_\omega(i) = \{f_j(i) | j \in \{1..|F_\omega|\}\} \tag{4}$$

Since knowing all features, especially the proprietary ones that could be used in a CBIRs, is not feasible different classification cases have to be considered. Five such cases are possible (see also figure 1):

1) $F_\alpha = F_\omega$: In this case our representation is aware of exactly the same features the CBIRs is using. Here, only the weights for the features have to be found.
2) $F_\alpha \subset F_\omega$: In this case the features which are not used by α have to be identified and the weights for the remaining features have to be found.
3) $F_\alpha \supset F_\omega$: In this case the CBIR system in question uses features that are not present in our internal representation. The current focus of our implementation is the detection of this case in order to avoid false positives. Future research will consider feature classes (e.g. by statistical analysis according to the correlation of specific features [3]) by trying to identify which feature class has been used by the CBIRs.
4) $F_\omega \cap F_\alpha \neq \emptyset$: Not included in this case are constellations that are included in cases 1, 2 or 3. Similar to case 3, the current focus of our implementation is the detection of this case.
5) $F_\omega \cap F_\alpha = \emptyset$: In this case, as in cases 3 and 4, the current focus of our implementation is the detection of this constellation.

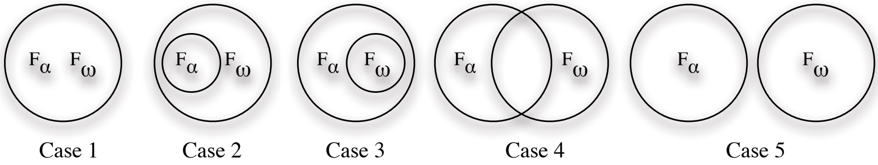


Fig. 1. Distinct cases for the detection of the configurations

4.2 Overall Process

The detection process of our algorithm starts with an enrollment of the CBIR in question.

Then a set of test images, from now on referred to as image test set (ITS), is used for querying the CBIRs (see figure 2). The selection of these images is arbitrary. The necessary size of ITS has been experimentally evaluated. In our tests setting the amount of test images at five showed a reasonable balance between processing speed and the accuracy of our detection approach.

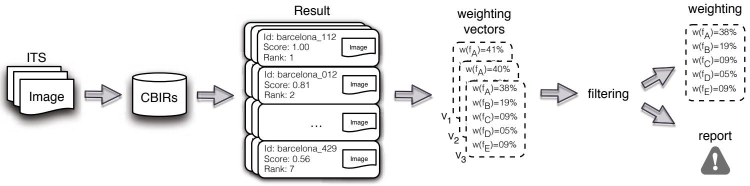


Fig. 2. Approach Outline

As shown in figure 2, for every single query image of the ITS the CBIRs responds with a ranked list of result images and associated scores.

Depending on the search paradigm used the length of those lists may vary. But as long as the returned number of results exceeds five items per list, for at least three returned result lists, those different paradigms do not influence the algorithms performance. The implemented paradigms of the CBIRs can therefore be left out of consideration. For the same reason false positives and negatives do not have to be considered. This is because of the approaches' exclusive interest in the detection of the configuration of the CBIRs. The approaches aim is not to judge the quality of the CBIR, but to detect its configuration.

Using the ranked lists received, we calculate the feature weighting vectors that generate self-computed scores very similar to the result scores. If no score is available, this calculation can also be made by the mere comparison of the ranks, which also omits a necessary normalization step. As this results in a poorer performance of the approach the scoring was preferred. This calculation is achieved by minimizing an objective function which evaluates possible weight vectors for the feature set by comparing our self-calculated scores to the result scores of the CBIR system. This way, the problem can be traced back to a vector optimization problem - an optimization algorithm can be used to find a good weight vector, preferably one very similar to the one the CBIRs uses internally. This optimization is performed for every query to be able to statistically evaluate the calculated optimal weight vectors. Confidence rating values are assigned to every feature according to different criteria, for example a weights' standard deviation across the multiple weight vectors. This rating is used to filter irrelevant features. Once there are no more features to filter, the feature weight configuration of the CBIRs is calculated using the arithmetical average.

4.3 Approach Details

Algorithm 1 presents the cornerstones of the described approach. The algorithm's goal is to find optimal weight values for every feature, in case every feature is known in our representation. Conversely, it's desirable that the algorithm reports if the analyzed CBIR system uses unknown features.

The algorithm is initialized with all features known to our internal representation (line 2). This set of possible candidates is then successively thinned out by removing irrelevant features (line 3). The *while*-loop in line 3 is repeated until no more features can be discarded due to a low rating. The first *for* loop optimizes an objective function *obj* (line 7) for every image in the image test set (*ITS*). The objective function is used to evaluate the fitness of possible feature weight vectors. It is listed in formula 6, with v being the feature weight vector to be tested, n being the number of result images returned by the CBIRs and $score_{\alpha,q}(i'_k)$ being the returned score value for result image i'_k respective to input image q (see formula 3). The function *score′* is listed separately in formula 5. It is very similar to formula 3 and is used to calculate an known score value for result image i'_k, input image q and feature weight vector v. Here, $F_{\omega}{}'$ is the set of currently remaining features inside *PossibleFeatures*, f_{j} is the j-th remaining feature and $\delta_{\omega\mathrm{j}}$ is one of the distance functions used by our internal representation ω. Additionally, the function *scale* normalizes *score′*'s values to the interval $[0, 1]$.

$$score'_{\omega,\mathrm{q},\mathrm{v}}(i') = scale_q[\sum_{j=1}^{|F_{\omega}{}'|} v_{\mathrm{j}}\delta_{\omega\mathrm{j}}(f_{\mathrm{j}}(i'), f_{\mathrm{j}}(q))] \tag{5}$$

$$obj_q(v) = \sqrt{\frac{\sum_{k=1}^{n}(score_{\alpha,q}(i'_k) - score'_{\omega,q,v}(i'_k))^2}{n}} \tag{6}$$

As mentioned above, the objective function *obj* calculates a distance value for the scoring obtained by using feature weight vector v and the actual scoring of the CBIRs. Larger distance values mean that a tested vector results in self-computed scores less similar to those returned by the CBIRs, i.e. there are larger differences between *score* and *score′* for the different result images.

Furthermore, algorithm [16,6,5] to find a feature weight vector which most closely resembles the CBIRs' configuration. Multiple optimization algorithms have been evaluated and the results are presented in section 5. This optimization is performed for every image in *ITS*, so *FeatureWeights* consists of vectors with each vector containing the optimal weight values for one image in *ITS* (line 7).

The reason for calculating an optimal feature weight vector for multiple images is to be able to conduct a statistical analysis of these values afterwards, which is done in the second *for loop* starting in line 9. *FeatureWeights* contains multiple weight values for a single feature, one for every image in *ITS* (for example see figure 2 where for f_A three different score settings have been detected). *rateFeature* assigns a rating between 0 and 1 to every feature, depending on different configurable criteria (line 10). Currently, a higher standard deviation of the weight values of one feature for different images results in a reduction in its rating, as does too small an average weight. The ratings of all features are also lowered if the distance values returned by the objective function are higher, which signifies that a weight vector cannot reproduce the CBIR system's behavior well enough. If any feature has a rating smaller than *minRating* (line 11) it is not used again for future executions of the outer *while* loop since a smaller rating suggests a lower probability of a feature being used by the analyzed CBIRs. The outer while is only repeated if at least one feature has been discarded in the current run (line 13). When the *while* loop finishes, depending on whether there are still remaining features in *PossibleFeatures*, an average of the previously computed

Algorithm 1. analyze(ITS, F_ω, CBIRsResults[])

```
 1: $doContinue \leftarrow true$
 2: $PossibleFeatures \leftarrow F_\omega$
 3: while $doContinue$ do
 4:    $doContinue \leftarrow false$
 5:    for $i = 1$ to $i = |ITS|$ do
 6:       $q \leftarrow ITS[i]$
 7:       $FeatureWeights[i] \leftarrow optimize(obj_q)$
 8:    end for
 9:    for $i = 1$ to $i = |PossibleFeatures|$ do
10:       $Ratings[i] \leftarrow rateFeature(i, FeatureWeights)$
11:       if $(Ratings[i] < minRating)$ then
12:          $PossibleFeatures.remove(i)$
13:          $doContinue \leftarrow true$
14:       end if
15:    end for
16: end while
17: if $PossibleFeatures.isEmpty()$ then
18:    return $null$
19: else
20:    return $average(FeatureWeights)$
21: end if
```

optimal feature weights for the remaining features is returned. Otherwise it is reported that the analyzed CBIRs uses unknown features (lines 17-20).

5 Evaluation

The evaluation section is divided into three parts. The first part addresses the runtime of the implementation of our approach. The second part evaluates the performance of the presented approach respective to the detection of the features that were relevant for the analyzed CBIRs. Finally, the last part focuses on the accuracy of the weight allocation for the detected features.

Our tests used the publicly available image set of the ImageCLEF benchmark [10] which can be obtained at `http://www.imageclef.org/2011`. We used the full set of 20,000 images. As a CBIR system we used Lire [8] which is an open source Java CBIR library containing a good set of implemented features[1]. Furthermore, Lire was chosen as it is an extensible library and could therefore be adapted with little effort.

5.1 Runtime

One possible way of approximating the configuration of a CBIRs is to try a number of feature weight vectors one by one (brute force). Equation 7 represents the number of possible feature weighting allocations, where n stands for the number

[1] `http://www.semanticmetadata.net/lire/`

of features multiplied by the number of feature distances (metrics) and k stands for the granularity of individual weight values. A granularity of 1 would only enable the algorithm to identify whether a feature-metric combination was chosen with a weighting of 100% or 0%. By contrast, a granularity of 100 would make it possible to allocate individual weight values in the weight vector in 1% steps.

$$t(n,k) = \binom{n+k-1}{k} \tag{7}$$

Under realistic circumstances, using a brute force approach would lead to a prohibitively large runtime. The application of different optimization algorithms solved this runtime problem. An implementation of Cuckoo Search [16] as well as an implementation of Particle Swarm Optimization [6], the usage of Multi-Directional Search and the Nelder-Mead Method from [5] all needed approximately 15 seconds for the analysis of one system and delivered very good results. However, about two minutes of additional time was required for extracting the feature vectors from images and pre-calculating feature distance values between images, regardless of which algorithm was used.

5.2 Feature Detection

The evaluation involved the analysis of 500 different configurations (combinations of features and weights) of a Lire-CBIRs. For every one of the cases defined in section 4.1 a test set of 100 different configurations was randomly generated, with each of the five test sets adhering to its respective classification case constraint. Also, for every optimization algorithm 50,000 evaluations of the objective function were performed.

Figure 3 shows the average precision and recall percentages regarding the correct identification of features over the 100 different configurations belonging to case 1. As mentioned in section 5.1 all of the optimization algorithms delivered very good results. The brute force approach using a granularity value of 10 was chosen for comparison as it had a runtime similar to the other optimization algorithms. All of the optimization algorithms returned mostly the same features as the selected features in the CBIRs configuration. This means that for nearly every possible configuration the optimal feature weighting calculated by the algorithms was using all the features known to our internal representation. This is the desired behavior, as our internal representation implemented exactly the same features as the Lire-configurations in this case. False negatives - features which are not marked as detected but are in fact used by the CBIRs - did not occur often. These few false negatives, reflected in the marginal deviation from 100%, were mostly caused by configurations where one of the features used a weight percentage of less or 1%.

Figure 4 shows the average precision and recall percentages over the 100 different configurations of case 2. In this case the internal representation implements more features than the CBIRs provides. Here, false positives - features that are marked as detected but are not in fact used by the CBIRS - can occur in addition to the previously described false negatives. Cuckoo Search as well as

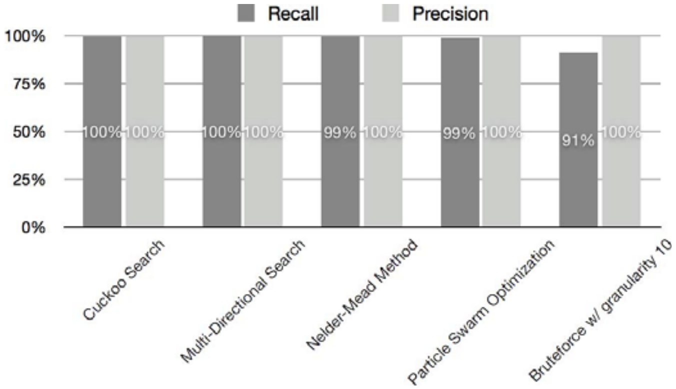


Fig. 3. Precision and Recall: Case 1

Multi-Directional Search also exhibited very good performance. For all the tests every feature used was detected and false positives occurred only in a few cases. The performance of the Nelder-Mead Method and Particle Swarm Optimization were slightly weaker. They both had a small percentage of false negatives, see recall, and a small percentage of false positives, see precision. Those false negatives and positives belonged to tests where only a very small weight value was assigned to a feature. In a real world scenario, though, it would not be very detrimental to the performance of our approach to not be able to correctly identify features with very small weights or to incorrectly assign very small weights to irrelevant features. So these small deviations do not pose a problem.

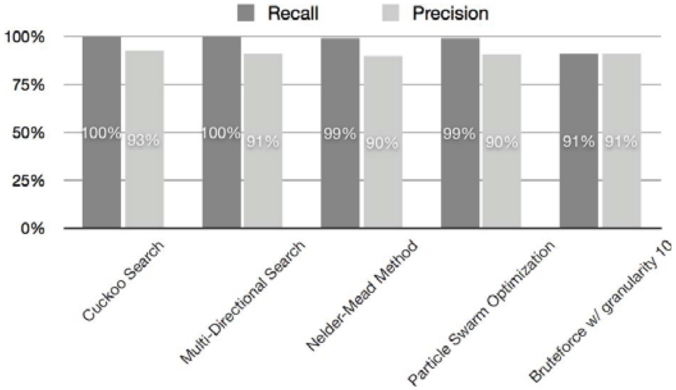


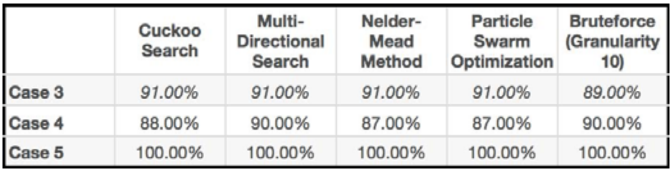
Fig. 4. Precision and Recall: Case 2

Since the focus of the analysis of cases 3 to 5 is not to understand which features were used but rather to discover that the configuration cannot be detected, it would not have been useful to calculate recall and precision values for these cases. That is why in table 1 only the success rates of the different optimization algorithms for the remaining cases are illustrated. A test case was counted as successful if our implementation returned that it was not able to determine the CBIR system's configuration.

In case 3 our internal representation implemented only a fraction of the features of the CBIRs. In line 2 of table 1 the average success rate over the test runs for every algorithm is shown. Here, each of the algorithms detected (in almost all tests with a probability of 91 %) that the internally implemented features were not a superset of or equal to the CBIR system's features.

In case 4 our internal representation implemented a fraction of the features of the CBIRs as well as additional features which were not implemented by the CBIRs. Very similar to case 3 in most tests, our implemented approach detected that it was not able to identify the CBIR system's configuration due to missing features. As the internal representation had a larger amount of features available to approximate the CBIR system's behavior, in marginal cases a CBIRs's configuration was sometimes incorrectly considered to be detected. The brute force method delivered better results in these cases because it was weaker in terms of optimization precision and was thus more likely to identify a configuration as unknown.

Table 1. Sucess rates of Cases 3, 4 and 5



	Cuckoo Search	Multi-Directional Search	Nelder-Mead Method	Particle Swarm Optimization	Bruteforce (Granularity 10)
Case 3	*91.00%*	*91.00%*	*91.00%*	*91.00%*	*89.00%*
Case 4	88.00%	90.00%	87.00%	87.00%	90.00%
Case 5	100.00%	100.00%	100.00%	100.00%	100.00%

In case 5 the internal representation implemented only features which were not implemented by the CBIRs. As illustrated, all optimization algorithms used as well as the brute force approach were able to detect this classification class.

5.3 Weighting Allocation

In this subsection the weighting allocation performance of our approach is evaluated. This is done by computing the deviation of the detected weighting allocation to the weighting allocation of the CBIRs using the euclidian distance between the weight vectors.

Figure 5 shows a visualization of the deviation for test case 1 for every implemented algorithm. All of the algorithms used did have a small weighting deviation. Cuckoo Search had a deviation of 0.0175, whereas the Multi-Directional Search and the Nelder-Mead Method had a deviation of 0.0217 and 0.0271. PSO and the brute force approach had a deviation of 0.0322 and 0.0637, respectively. In Figure 6 the average deviation for test case 2 is visualized. Again, all of the algorithms do have a small weighting deviation, though mostly slightly larger than in case 1. All of these deviation values are relatively small, meaning that all of the algorithms are able to approximate CBIRs configurations well if all used features are known. Cockoo Search was the best overall algorithm in our tests, though the performance of the various optimization algorithms can be very dependent on their configured parameters.

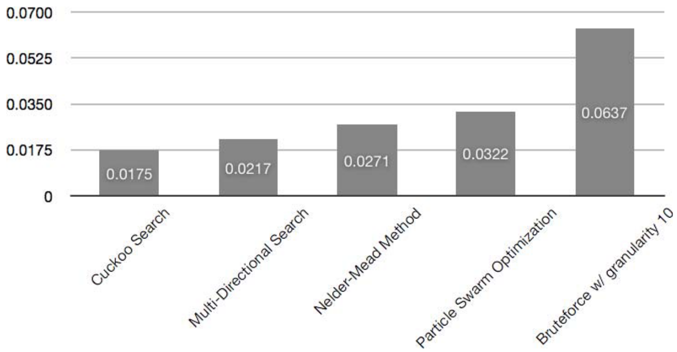


Fig. 5. Averaged weighting deviation in Case 1

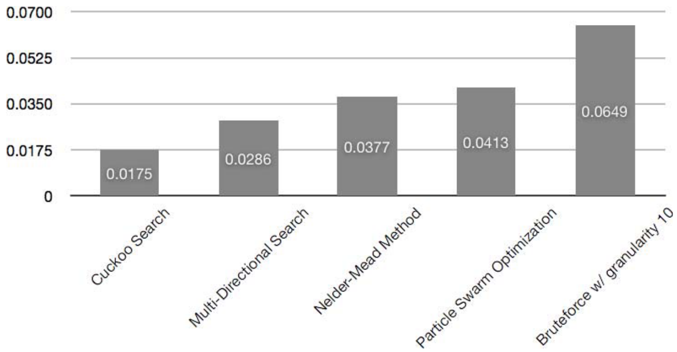


Fig. 6. Averaged weighting deviation in Case 2

All in all, these first test results are already very promising but further improvement and fine tuning of our approach are both necessary.

6 Conclusion

This article presented a novel approach for the detection of the configuration of content based image retrieval (CBIR) systems. The focus of this work was on the correct identification of feature settings and their assigned weights. Related to the combination of a system's configuration and our internal representation, five different classification cases have been highlighted. We demonstrated that our proposed approach is capable of detecting the complete configuration for two cases and is also able to mark the others are as currently not detectable. Moreover, we demonstrated that the problem can be traced back to an optimization problem. The evaluation showed a high accuracy for feature and weight allocation and demonstrated good performance by the use of different optimization algorithms.

Future work will consider the development of feature classes in order to solve the missing classification classes as well. Furthermore, the approach will be adopted in a distributed search scenario for improving query distribution decisions and result ranking strategies.

References

1. Barton, S., Dohnal, V., Sedmidubsky, J., Zezula, P.: Building self-organized image retrieval network. In: Proceeding of the 2008 ACM Workshop on Large-Scale Distributed Systems for Information Retrieval, pp. 51–58 (2008)
2. Datta, R., Joshi, D., Li, J., Wang, J.: Image retrieval: Ideas, influences, and trends of the new age. ACM Computing Surveys (CSUR) (January 2008)
3. Eidenberger, H.: Evaluation of content-based image descriptors by statistical methods. Multimedia Tools and Applications 35(3), 241–258 (2007)
4. Black, J.A., Fahmy, G., Panchanathan, S.: A Method for Evaluating the Performance of Content-Based Image Retrieval Systems Based on Subjectively Determined Similarity between Images. In: Lew, M., Sebe, N., Eakins, J.P. (eds.) CIVR 2002. LNCS, vol. 2383, pp. 356–366. Springer, Heidelberg (2002)
5. John Ashworth Nelder, R.M.: A simplex method for function minimization. Computer Journal 7, 308–313 (1965)
6. Kennedy, J., Eberhart, R.C.: A new optimizer using particle swarm theory. In: Proceedings of the Sixth International Symposium on Micromachine and Human Science, Nagoya, Japan, pp. 39–43 (1995)
7. Kosch, H., Maier, P.: Content-based image retrieval systems - reviewing and benchmarking. 54 Journal of Digital Information Management (8), 1–21 (March 2010)
8. Lux, M., Chatzichristofis, S.: Lire: lucene image retrieval: an extensible java cbir library. In: Proceeding of the 16th ACM International Conference on Multimedia, pp. 1085–1088. LIRE (2008)
9. Müller, H., Müller, W., Marchand-Maillet, S., Pun, T., Squire, D.M.: Learning feature weights from user behavior in content-based image retrieval. In: Proceedings of the International Workshop on Multimedia Data Mining, Boston, USA, pp. 67–72. ACM (2000)
10. Müller, H., Tsikrika, T.: Global pattern recognition: The imageclef benchmark. IAPR Newsletter 32(1), 3–6 (2010)
11. 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), 1349–1380 (2000)
12. Stegmaier, F., Döller, M., Kosch, H., Hutter, A., Riegel, T.: AIR: Architecture for Interoperable Retrieval on distributed and heterogeneous Multimedia Repositories. In: Proceedings of the 11th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2010), Desenzano del Garda, Italy, pp. 1–4. IEEExplore (2010)
13. Thomas Deselaers, D.K., Ney, H.: Features for image retrieval: an experimental comparison. Information Retrieval 11(2), 77–107 (2007)
14. Veltkamp, R., Tanase, M.: A survey of content-based image retrieval systems. In: Content-based Image and Video Retrieval, pp. 47–101 (2002)
15. Kore, S., Kondekar, V.H., Kolkure, V.S.: Image retrieval techniques based on image features: A state of art approach forcb ir. International Journal of Computer Science and Information Security 7(1), 69–76 (2010)
16. Yang, X., Deb, S.: Engineering optimisation by cuckoo search. International Journal of Mathematical Modelling and Numerical Optimisation 1(4), 330–343 (2010)

On Stability of Adaptive Similarity Measures for Content-Based Image Retrieval

Christian Beecks and Thomas Seidl

Data Management and Data Exploration Group
RWTH Aachen University
Germany
{beecks,seidl}@cs.rwth-aachen.de

Abstract. Retrieving similar images is a challenging task for today's content-based retrieval systems. Aiming at high retrieval performance, these systems frequently capture the user's notion of similarity through expressive image models and adaptive similarity measures, which try to approximate the individual user-dependent notion of similarity as close as possible. As image models appearing on the query side can significantly differ in quality compared to those stored in the multimedia database, similarity measures have to be robust against these individual quality changes in order to maintain high retrieval performance. In order to evaluate the robustness of similarity measures, we introduce the general concept of the *stability of a similarity measure with respect to query modifying transformations* describing the change in quality on the query side. In addition, we include a comparison of the stability of the major state-of-the-art adaptive similarity measures based on different benchmark image databases.

Keywords: content-based image retrieval, feature signature, adaptive similarity measure, evaluation measure, average precision stability.

1 Introduction

Modeling image contents for the purpose of content-based image retrieval [4,21,23,13] is a challenging task. While the computational effort spent for extracting and generating expressive image models is nearly unrestricted on the database side, the effort spent on the query side is often limited due to the following reasons: first, users frequently demand the retrieval system to answer their queries as fast as possible, thus including the extraction of complex local feature descriptors is a time consuming task which has to be done quickly or even skipped. Second, users issuing queries in a mobile environment, e.g., by taking a picture with a mobile phone, are often restricted in terms of their devices' energy consumption and bandwidth restrictions. As a consequence, processing images with the aim of generating expressive image models has to be kept short which inevitably leads to a gap of quality between the query side and the database

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 346–357, 2012.

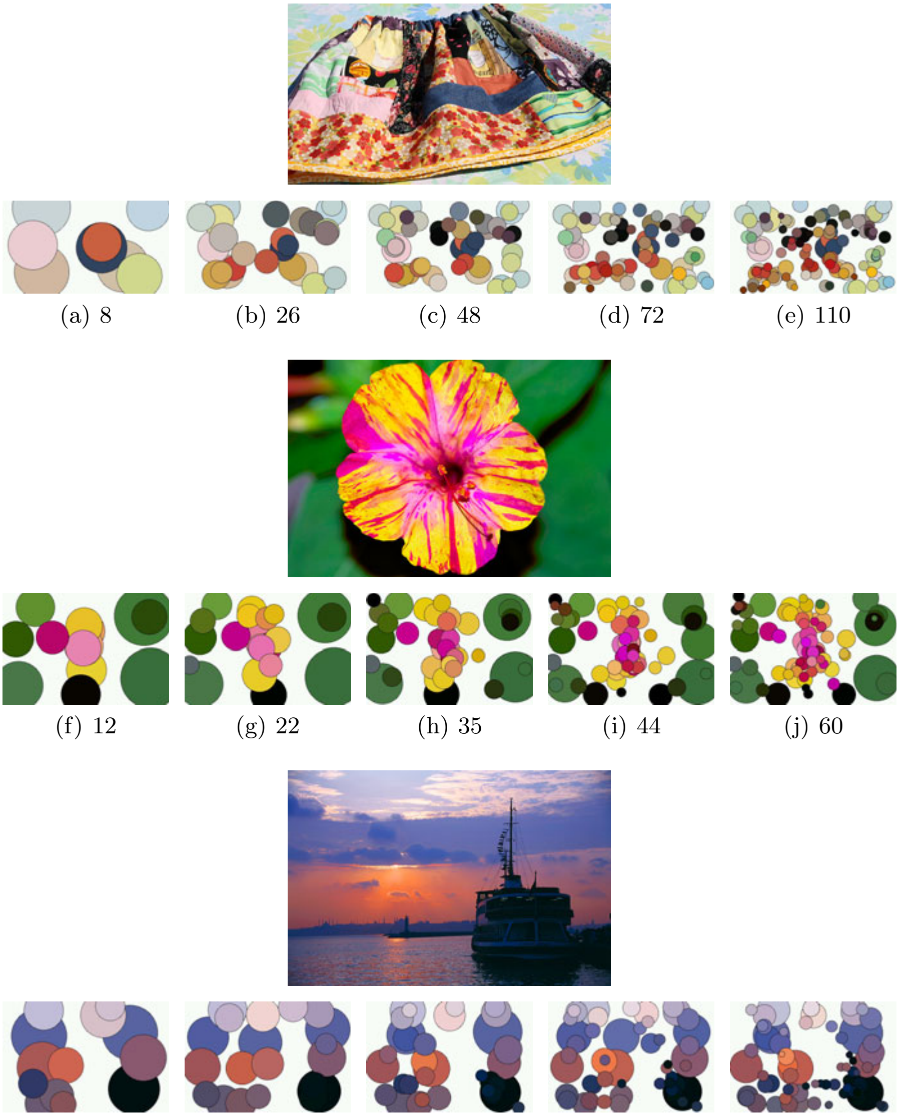
side. Image models appearing on the query side can significantly differ in quality compared to those stored in the multimedia database. Thus, the retrieval system's similarity measure has to be robust against these individual changes.

Although the performance of similarity measures for different types of image models is investigated in various studies [2,7,19], none of them addresses the issue of query-side-dependent quality restrictions. They all assume the quality of the image model on the query side is the same as that on the database side. For this reason, we study the *stability* of adaptive similarity measures, namely the *Hausdorff Distance* [9], *Perceptually Modified Hausdorff Distance* [18], *Earth Mover's Distance* [20], *Weighted Correlation Distance* [12], and *Signature Quadratic Form Distance* [1,3], in the context of content-based image retrieval. To this end, we first introduce the general concept of the *stability of a similarity measure with respect to query modifying transformations*, and we then evaluate this stability on different benchmark image databases by using *Mean Average Precision* [15] as a running example.

The structure of this paper is as follows: in Section 2, we describe feature signatures as flexible image models and list adaptive similarity measures applicable to such models. In Section 3, we outline existing evaluation measures which can be used within our proposed stability measure. In Section 4, we introduce the general concept of the *stability of a similarity measure* and explain the differences to existing evaluation measures. We evaluate the stability of the adaptive similarity measures on different benchmark image databases in Section 5, before we conclude our paper with an outlook on future work in Section 6.

2 Modeling and Comparing Image Contents through Feature Signatures and Adaptive Similarity Measures

Describing the content of an image by its feature distribution over a feature space is a common way to make images accessible. While many similarity models, which cope with visual object recognition tasks such as near-duplicate detection, rely on complex unaggregated local features, similarity models for the purpose of content-based multimedia retrieval frequently aggregate individual feature distributions in order to obtain more compact and robust content representations. In general, modeling image content follows two steps: First, local features are extracted, for instance SIFT [14] descriptors at some salient points [16,24]. Second, these features are aggregated into a more compact representation. One prominent way of aggregating and comparing the extracted local features is the *bag-of-visual-words* [22] approach. Based on a predetermined *visual vocabulary*, the extracted local features are assigned to the *visual words* of that specific visual vocabulary. The similarity between images is then defined through a distance between the visual word frequencies, stored in form of a vector. Although this approach provides high retrieval performance, it is limited in flexibility due to the static visual vocabulary. In fact, all images have to be represented by the same visual words, resulting in a sparse high-dimensional vector representation. Moreover, the availability of the database's visual vocabulary has to be ensured on the query side in order to compute the content representation of an image.



(a) 8 (b) 26 (c) 48 (d) 72 (e) 110

(f) 12 (g) 22 (h) 35 (i) 44 (j) 60

(k) 13 (l) 20 (m) 33 (n) 52 (o) 70

Fig. 1. Three example images from the *MIR Flickr* database [8] and their corresponding feature signatures over a feature space comprising position, color, and texture information. The number of representatives, i.e. centroids, is depicted accordingly.

An alternative of aggregating and comparing the images' local features is by making use of *adaptive similarity measures* [2], which are independent of a visual vocabulary. They allow to compare images whose local features are extracted and aggregated individually. Formally, each image $\mathcal{I}$ is mapped to a set of local features $f_1, \ldots, f_n \in \mathbb{F}$ within a feature space $\mathbb{F}$. Subsequently, these features are partitioned by a partitioning $\mathcal{P} = \{P_1, \ldots, P_k\}$ where each feature f_i is assigned to its nearest partition. As a result, each partition is represented by a representative $r_i \in \mathbb{F}$ and a weight $w_i \in \mathbb{R}^{\geq 0}$ which form the components of a *feature signature* S as follows:

$$S = \{\langle r_i, w_i \rangle | r_i \in \mathbb{F} \wedge w_i \in \mathbb{R}^{\geq 0}\}_{i=1}^{k}.$$

Frequently, the partitioning $\mathcal{P}$ is obtained by the k-means clustering algorithm: the representatives r_i are the centroids of each cluster P_i with weights w_i denoting the relative frequencies, i.e. $r_i = \sum_{f \in P_i} \frac{f}{|P_i|}$ and $w_i = \frac{|P_i|}{\sum_i |P_i|}$.

In Figure 1, we depict three example images and their feature signatures which were generated by mapping randomly selected image pixels into a seven-dimensional feature space $(L, a, b, x, y, \chi, \eta) \in \mathbb{F} = \mathbb{R}^7$ comprising color (L, a, b), position (x, y), contrast χ, and coarseness η information. The extracted seven-dimensional features are clustered by an adaptive variant of the k-means clustering algorithm [12] in order to obtain the feature signatures. Thus, the number of centroids is determined dynamically and controlled by the number of selected image pixels. As can be seen in the figure, the higher the number of centroids, which are depicted as circles in the corresponding color, the better the visual content approximation, and vice versa. While a small number of centroids only provides a coarse approximation of the original image, a large number of centroids may help to assign individual centroids to the corresponding parts in the images. Given these examples, the question arises; which image model, i.e. feature signature, provides the highest retrieval performance? Furthermore, as the quality of a feature signature on the query side is frequently unpredictable, another question arises; which adaptive similarity measure is the most robust one? In particular the evaluation of the latter, the *robustness* or *stability*, is the focus of this paper. Therefore, we introduce the general concept of a *similarity measure's stability with respect to query modifying transformations* in Section 4, after describing existing evaluation measures, which can be used within our proposed stability measure, in the next section.

3 Evaluation Measures

In general, evaluating a similarity measure is done by querying an image collection and analyzing the results. For this purpose, the images are sorted in descending order according to their similarity regarding the query image, i.e. the retrieval system computes a *ranking* of the database, and each image is assigned a class label. The class labels are provided by the *ground truth* of the image collection and define the *relevancy* of each image with respect to the query image. A good overview of measuring the retrieval systems' effectiveness and a broad introduction to several evaluation measures can be found, for instance, in the book of Manning et al. [15].

In fact, many evaluation measures are based on *precision* and *recall* values – first used by Kent et al. [11] – which reflect the fraction of retrieved images that are relevant and the fraction of relevant images that are retrieved [15], respectively. Thus, a high precision value indicates that many relevant images have been retrieved while a high recall value indicates that the complete amount of relevant images is reached by the retrieved images. These values can be computed for each retrieved image within the ranking and can then be visualized by the so-called *precision and recall* curve. A frequently encountered aggregation of multiple precision and recall curves is the *Mean Average Precision* value, which approximates the average area under the curves [15]. Other evaluation measures are the *F-Measure* [25], which is the weighted harmonic mean of precision and recall [15], or the *Normalized Discounted Cumulative Gain* [10], which measures the usefulness of multiple rankings.

To sum up, the aforementioned evaluation measures judge the retrieval performance according to a single ranking or multiple rankings. Although they are frequently used throughout the research area of content-based retrieval, see for instance the performance evaluations for content-based image retrieval [2,7,19], they miss the ability to express the variance of a measured value. For example, measuring the same Mean Average Precision values twice for two different similarity measures does not necessarily mean that both similarity measures show the same retrieval performance. One similarity measure can show a higher variance than the other one, which is, in this example, not reflected within the Mean Average Precision values.

In order to counteract this issue, we propose to include the stability into the evaluation of the retrieval performance. As we are focusing on the retrieval performance of adaptive similarity measures for content-based image retrieval, we show how to evaluate the stability of a similarity measure by making use of conventional Mean Average Precision values in the next section.

4 Stability of a Similarity Measure

As mentioned above, we are interested in evaluating the *stability* of adaptive similarity measures in the context of content-based image retrieval with respect to query modifying transformations, which has not been investigated in previous studies [2,7,19] so far. This will provide further insight into the behavior of adaptive similarity measures and will thus help to guide further research and developments.

In order to generally define the stability of a similarity measure, we combine existing evaluation measures, as described in the previous section, with query modifying transformations. These transformations reflect the general discrepancy between the image models generated on the query side and those stored in the image database. Without loss of generality, we assume that the modifications of the image models are only done on the query side. Further, we make use of Mean Average Precision as evaluation measure in the remainder of this paper. This evaluation measure can be replaced with any other evaluation measure where appropriate. However, by using Mean Average Precision (MAP) as evaluation measure, we denote our resulting stability measure as *Average Precision*

Stability (APS). It is defined for a similarity measure δ over a database $\mathcal{DB}$ storing the images, a set of queries Q, and a set of query modifying transformations Φ as follows.

Definition 1. *Average Precision Stability (*APS*)*
*Given a similarity measure δ, a database $\mathcal{DB}$, a set of queries $Q = \{q_1, \ldots, q_l\}$, and a set of query modifying transformations $\Phi = \{\phi_1, \ldots, \phi_m\}$, the Average Precision Stability (*APS*) is then defined as:*

$$\mathrm{APS}_{\Phi}(Q, \delta, \mathcal{DB}) = \frac{E[\mathbb{M}]}{1 + \sigma_{\mathbb{M}}},$$

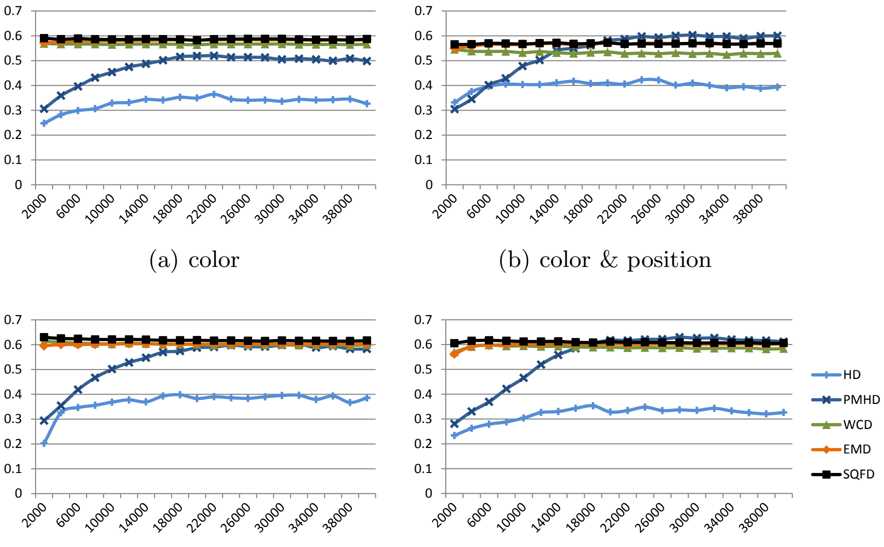
where $\mathbb{M}$ denotes the distribution of Mean Average Precision values with respect to the query modifying transformations $\Phi = \{\phi_1, \ldots, \phi_m\}$ applied to each query contained in the set of queries Q, i.e. $\mathbb{M} = \bigcup_{i=1}^{m} \{\mathrm{MAP}(\{\phi_i(q_1), \ldots, \phi_i(q_l)\}, \delta, \mathcal{DB})\}$. $E[\mathbb{M}]$ and $\sigma_{\mathbb{M}}$ denote the expected value and standard deviation.

According to Definition 1, the *Average Precision Stability* is defined as the expected Mean Average Precision value divided by the standard deviation of those Mean Average Precision values with respect to a set of query modifying transformations. In this way, it reflects the similarity measure's stability as follows: in case the similarity measure is invariant against the query modifying transformations, the *Average Precision Stability* becomes the expected Mean Average Precision value, otherwise the *Average Precision Stability* decreases with varying Mean Average Precision values. As can be seen in the definition, the proposed *Average Precision Stability* generalized the Mean Average Precision measure by including the variance of the Mean Average Precision values. Consequently, it is also bounded between 0 and 1.

In general, this concept of the stability of a similarity measure can be extended to any other evaluation measure, for instance the *F-Measure* or the *Normalized Discounted Cumulative Gain*, by replacing the evaluation measure appropriately. It is thus flexible to fit individual user and system requirements when evaluating the retrieval performance of content-based multimedia retrieval systems. However, as Mean Average Precision is a frequently encountered evaluation measure in the area of content-based multimedia retrieval, we provide an *Average Precision Stability* evaluation study of adaptive similarity measures for the purpose of content-based image retrieval in the following section.

5 Experimental Evaluation

We evaluated the similarity measures' stability on the following benchmark image databases: the *Corel Wang* database [26] comprises 1,000 images which are classified into ten themes. The themes cover a multitude of topics, such as beaches, flowers, buses, food, etc. The *Coil 100* database [17] consists of 7,200 images classified into 100 different classes. Each class depicts one object photographed from 72 different directions. The *MIR Flickr* database [8] contains



(a) color

(b) color & position

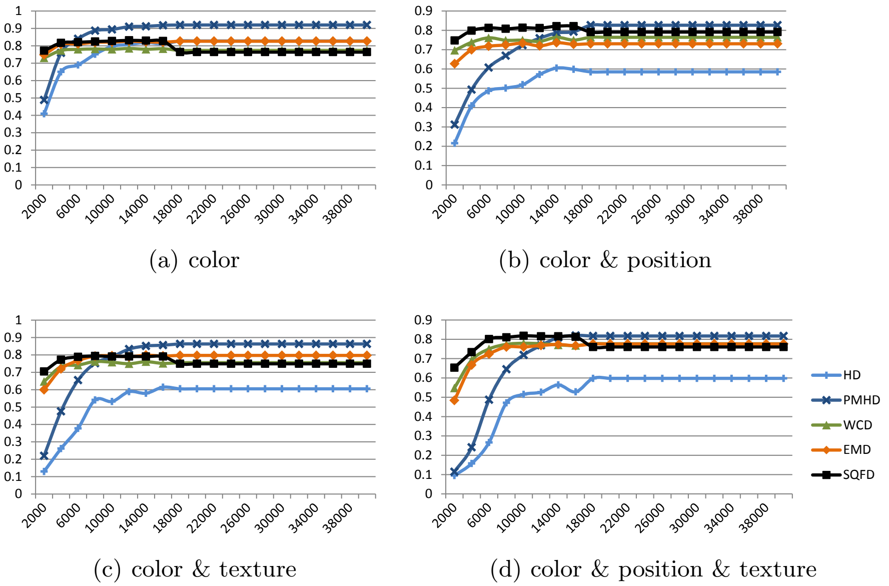
(c) color & texture

(d) color & position & texture

Fig. 2. Mean Average Precision values for the *Corel Wang* database based on different features: (a) color, (b) color & position, (c) color & texture, and (d) color & position & texture. The number of image pixels is varied between 2,000 and 40,000 pixels.

25,000 images downloaded from `http://flickr.com/` including textual annotations. The *101 objects* database [5] contains 9,196 images which are classified into 101 categories. Finally, we include the *ALOI* database [6] which is similar to the *Coil 100* database but comprises 72,000 images. The themes, classes, textual annotations, and categories are used as ground truth to measure precision and recall values [15] after each retrieved image. For the *MIR Flickr* database, we define virtual classes which contain all images sharing at least two common textual annotations and are used as ground truth.

The resulting Mean Average Precision values, which are aggregated over 100 randomly selected queries for each combination of image database and similarity measure, are shown in Figures 2 to 6 where the number of image pixels considered for the extraction of color, position, and texture features is varied between 2,000 and 40,000. Thus the resulting query feature signatures generated by the adaptive k-means clustering algorithm vary in size between 1 and 115 centroids. The image databases always contain the feature signatures based on the clustering of 20,000 image pixels. In this way, the query modifying transformations are given by the change in cardinality of the query feature signatures, which is the most natural modification regardless of any specific local features. It can be seen in the figures, that the depicted mean average precision values depend on the applied feature spaces of the corresponding image database. In general, it turns out that the Hausdorff Distance (HD) and the Perceptually Modified Hausdorff Distance (PMHD) are very sensitive to change in feature signature quality on



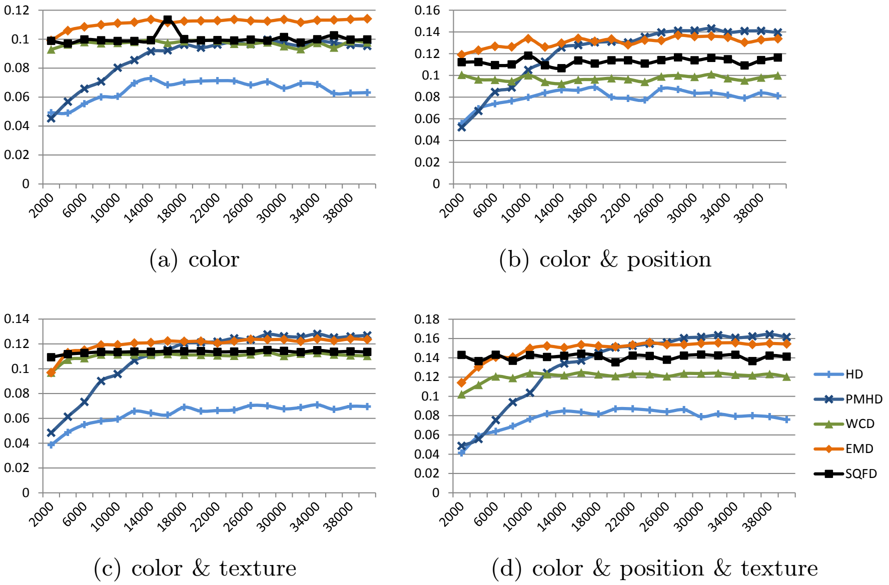
(a) color

(b) color & position

(c) color & texture

(d) color & position & texture

Fig. 3. Mean Average Precision values for the *Coil 100* database based on different features: (a) color, (b) color & position, (c) color & texture, and (d) color & position & texture. The number of image pixels is varied between 2,000 and 40,000 pixels.



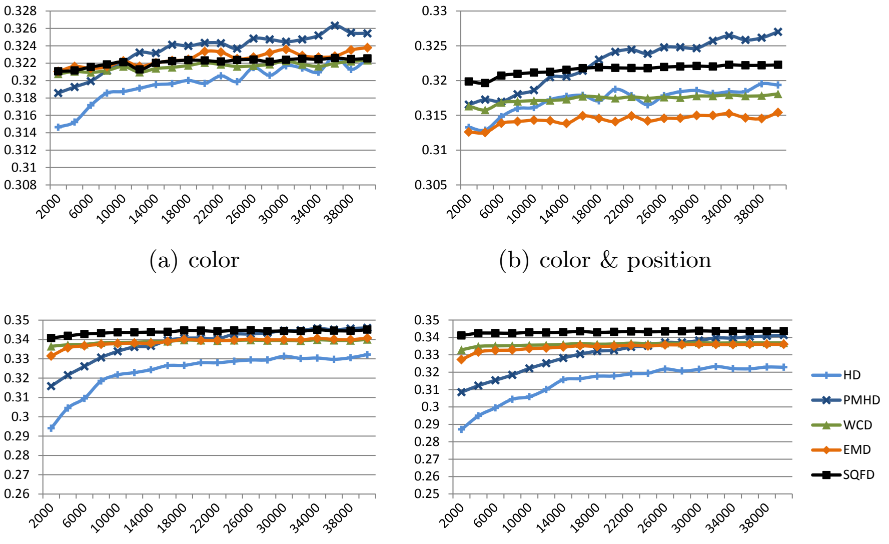
(a) color

(b) color & position

(c) color & texture

(d) color & position & texture

Fig. 4. Mean Average Precision values for the *101objects* database based on different features: (a) color, (b) color & position, (c) color & texture, and (d) color & position & texture. The number of image pixels is varied between 2,000 and 40,000 pixels.



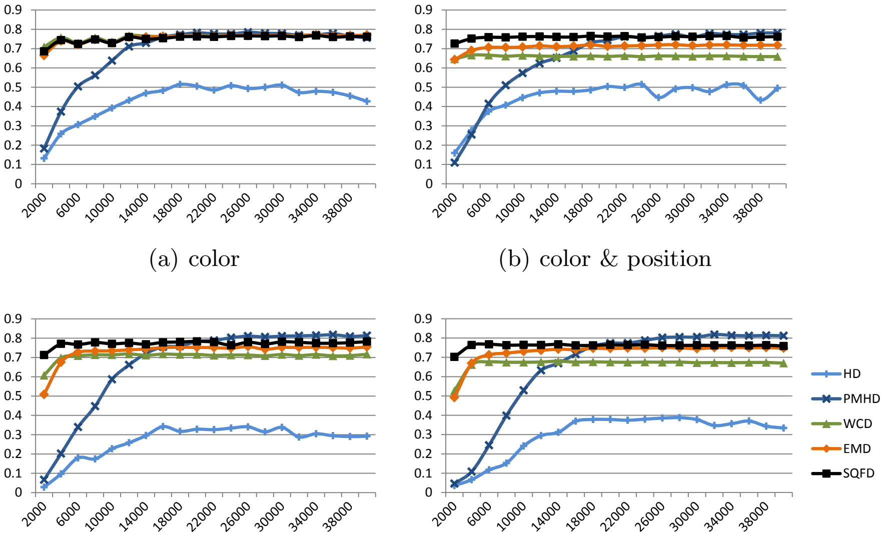
(a) color

(b) color & position

(c) color & texture

(d) color & position & texture

Fig. 5. Mean Average Precision values for the *MIR Flickr* database based on different features: (a) color, (b) color & position, (c) color & texture, and (d) color & position & texture. The number of image pixels is varied between 2,000 and 40,000 pixels.



(a) color

(b) color & position

(c) color & texture

(d) color & position & texture

Fig. 6. Mean Average Precision values for the *ALOI* database based on different features: (a) color, (b) color & position, (c) color & texture, and (d) color & position & texture. The number of image pixels is varied between 2,000 and 40,000 pixels.

Table 1. Average Precision Stability (APS) regarding different features and sizes of the query feature signatures

database	$\mathbb{F}$	HD	PMHD	WCD	EMD	SQFD
Corel Wang	c	0.322	0.450	0.566	0.580	**0.585**
	c,p	0.392	0.488	0.530	0.563	**0.567**
	c,t	0.354	0.494	0.599	0.601	**0.616**
	c,p,t	0.310	0.493	0.584	0.594	**0.607**
Coil 100	c	0.712	0.802	0.765	**0.805**	0.763
	c,p	0.501	0.662	0.744	0.706	**0.784**
	c,t	0.481	0.678	0.730	**0.746**	0.743
	c,p,t	0.515	0.721	0.759	0.751	**0.769**
101 objects	c	0.065	0.086	0.097	**0.111**	0.100
	c,p	0.080	0.118	0.097	**0.130**	0.112
	c,t	0.063	0.107	0.110	**0.119**	0.113
	c,p,t	0.076	0.128	0.120	**0.147**	0.141
MIR Flickr	c	0.319	**0.323**	0.321	0.322	0.322
	c,p	0.317	**0.321**	0.317	0.314	**0.321**
	c,t	0.321	0.335	0.339	0.338	**0.344**
	c,p,t	0.311	0.327	0.336	0.334	**0.343**
ALOI	c	0.393	0.591	**0.747**	0.736	0.738
	c,p	0.411	0.546	0.658	0.698	**0.753**
	c,t	0.247	0.547	0.691	0.693	**0.761**
	c,p,t	0.269	0.517	0.645	0.686	**0.750**
average APS		0.323	0.437	0.488	0.499	**0.512**

the query side, while the Weighted Correlation Distance (WCD), Earth Mover's Distance (EMD), and Signature Quadratic Form Distance (SQFD) show more stable Mean Average Precision values. (A definition of these distance-based similarity measures can be found, for instance, in the work of Beecks et al. [2].)

In order to verify the observations mentioned above, we measured the Average Precision Stability: the results are reported in Table 1 where we highlighted the highest Average Precision Stability values of each row. On average, the Signature Quadratic Form Distance (SQFD) shows the highest Average Precision Stability values followed by the Earth Mover's Distance (EMD) and the Weighted Correlation Distance (WCD). In accordance with Figures 2 to 6, the Hausdorff Distance (HD) and the Perceptually Modified Hausdorff Distance (PMHD) show the lowest Average Precision Stability values, as they are more sensitive to query modifying transformations changing the query feature signatures' cardinalities.

To sum up, the experimental evaluation shows that the stability of the aforementioned similarity measures depends on the quality of the feature signatures appearing on the query side. While complex similarity models, such as the Earth Mover's Distance, Weighted Correlation Distance, and Signature Quadratic Form Distance, which take into account the complete structure of the feature signatures for the similarity value computation, are more robust against varying query signatures, the matching-based Hausdorff Distances suffer from query signatures deviating from the database signatures with respect to the cardinality. Thus the latter are not feasible when the image models appearing on the query side significantly differ in size compared to those stored in the image database. In this case, the Earth Mover's Distance, the Weighted Correlation Distance, and particularly the Signature Quadratic Form Distance should be favored in order to obtain the highest stability.

6 Conclusions

We investigated the stability of the major adaptive similarity measures with respect to query modifying transformations. For this purpose, we defined the *Average Precision Stability* and evaluated the similarity measures' stability regarding the fundamental modification of size of the query feature signatures. As a result, the Signature Quadratic Form Distance shows the highest stability.

As future work, we plan to examine the Average Precision Stability of the similarity measures with respect to the qualities of photometric and geometric transformations.

References

1. Beecks, C., Uysal, M.S., Seidl, T.: Signature quadratic form distances for content-based similarity. In: Proc. ACM International Conference on Multimedia, pp. 697–700 (2009)
2. Beecks, C., Uysal, M.S., Seidl, T.: A comparative study of similarity measures for content-based multimedia retrieval. In: Proc. IEEE International Conference on Multimedia & Expo, pp. 1552–1557 (2010)
3. Beecks, C., Uysal, M.S., Seidl, T.: Signature quadratic form distance. In: Proc. ACM International Conference on Image and Video Retrieval, pp. 438–445 (2010)
4. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. ACM Computing Surveys 40(2), 1–60 (2008)
5. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples an incremental bayesian approach tested on 101 object categories. In: Proc. of the Workshop on Generative-Model Based Vision (2004)
6. Geusebroek, J.-M., Burghouts, G.J., Smeulders, A.W.M.: The Amsterdam Library of Object Images. International Journal of Computer Vision 61(1), 103–112 (2005)
7. Hu, R., Rüger, S., Song, D., Liu, H., Huang, Z.: Dissimilarity measures for content-based image retrieval. In: Proc. IEEE International Conference on Multimedia & Expo, pp. 1365–1368 (2008)
8. Huiskes, M.J., Lew, M.S.: The mir flickr retrieval evaluation. In: Proc. of the 1st ACM International Conference on Multimedia Information Retrieval, pp. 39–43 (2008)
9. Huttenlocher, D.P., Klanderman, G.A., Rucklidge, W.A.: Comparing Images Using the Hausdorff Distance. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(9), 850–863 (1993)
10. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of ir techniques. ACM Trans. Inf. Syst. 20, 422–446 (2002)
11. Kent, A., Berry, M.M., Luehrs, F.U., Perry, J.W.: Machine literature searching viii. operational criteria for designing information retrieval systems. American Documentation 6(2), 93–101 (1955)

12. Leow, W.K., Li, R.: The analysis and applications of adaptive-binning color histograms. Computer Vision and Image Understanding 94(1-3), 67–91 (2004)
13. Lew, M.S., Sebe, N., Djeraba, C., Jain, R.: Content-based multimedia information retrieval: State of the art and challenges. ACM Transactions on Multimedia Computing, Communications, and Applications 2(1), 1–19 (2006)
14. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60, 91–110 (2004)
15. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York (2008)
16. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(10), 1615–1630 (2005)
17. Nene, S., Nayar, S.K., Murase, H.: Columbia Object Image Library (COIL-100). Technical report, Department of Computer Science, Columbia University (1996)
18. Park, B.G., Lee, K.M., Lee, S.U.: Color-based image retrieval using perceptually modified Hausdorff distance. Journal on Image and Video Processing 2008, 1–10 (2008)
19. Rubner, Y., Puzicha, J., Tomasi, C., Buhmann, J.M.: Empirical evaluation of dissimilarity measures for color and texture. Computer Vision and Image Understanding 84(1), 25–43 (2001)
20. 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)
21. Sebe, N., Lew, M.S., Zhou, X., Huang, T.S., Bakker, E.M.: The state of the art in image and video retrieval. In: Proc. ACM International Conference on Image and Video Retrieval, pp. 1–8 (2003)
22. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: IEEE International Conference on Computer Vision, pp. 1470–1477 (2003)
23. Smeulders, A.W.M., 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), 1349–1380 (2000)
24. Tuytelaars, T., Mikolajczyk, K.: Local invariant feature detectors: a survey. Foundations and Trends in Computer Graphics and Vision 3(3), 177–280 (2008)
25. van Rijsbergen, C.J.: Information Retrieval. Butterworth (1979)
26. Wang, J.Z., Li, J., Wiederhold, G.: Simplicity: Semantics-sensitive integrated matching for picture libraries. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(9), 947–963 (2001)

Retrieval of Multiple Instances of Objects in Videos

Andrei Bursuc[1,2], Titus Zaharia[1], and Françoise Prêteux[3]

[1] Institut Télécom, Télécom SudParis, ARTEMIS Department, UMR CNRS 8145 MAP5,
9 rue Charles Fourier, 91011 Evry Cedex, France
{Andrei.Bursuc,Titus.Zaharia}@it-sudparis.eu
[2] Alcatel-Lucent Bell Labs France, route de Villejust 91620 Nozay, France
[3] Mines ParisTech, 60, Boulevard Saint-Michel 75272 Paris Cedex, France
Francoise.Preteux@mines-paristech.fr

Abstract. This paper tackles the issue of retrieving different instances of an object of interest within a given video document or in a video database. The principle consists in considering a semi-global image representation based on an over-segmentation of image frames. An aggregation mechanism is then applied in order to group a set of sub-regions into an object similar to the query, under a global similarity criterion. Two different strategies are proposed. The first one involves a greedy, dynamic region construction method. The second is based on simulated annealing, and aims at determining a global optimum. Experimental results show promising performances, with object detection rates of up to 79%.

Keywords: object-based indexing and retrieval, multiple instance detection, partial matching, MPEG-7 visual descriptors, video indexing.

1 Introduction

While an increasing number of solutions have provided a variety of satisfying results for concept detection in videos [1], retrieving different instances of the same object in video sequences still remains a challenge. The main difficulty is related to the specification of semi-global image representations that need to be considered, together with the elaboration of efficient partial matching strategies. In addition, variations in visual appearance and object's pose have to be taken into account appropriately. This relatively recent topic of research has been considered in the TRECVID [2] 2010 evaluation campaign, under the so-called instance search task, and TRECVID work is currently ongoing for the 2011 edition.

Related work includes two types of approaches, including (possibly dense) interest points as well as local regions.

Currently, interest points are among the most popular tools for object recognition and classification for both images and videos.

Early approaches for object retrieval, using interest points, have been developed by Sivic and Zisserman in their Video Google system [3]. In this case, SIFT descriptors [4] are extracted from video keyframes with the help of two types of

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 358–369, 2012.

overlapping image patches: Harris-Affine [5] regions, based on interest point neighborhoods, and so-called Maximally Stable Extreme Regions (MSER) [6]. The *bag-of-words* technique is used for achieving fast and efficient retrieval of objects interactively selected by the user with the help of a bounding box. Other applications involving interest points include scene classification and image understanding (*e.g.*[7], [8]). Interest points yield a high repeatability, *i.e.* they can be extracted reliably and are often identified in other images where the same object/scene appears.

However, the number of interest points extracted from an image varies a lot with the image content (from a few hundred to several thousands).

Starting from the method proposed in [9], Li *et al.* [10] group points of interest in graphs by using Delaunay triangulations. They take in consideration different geometric constraints with the goal of characterizing the geometric properties of the neighborhood of each node. Moreover each node has to be represented as an "affine" combination of its neighboring nodes. The obtained model is then matched to different scenes in order to determine the object of interest.

Aiming to improve the accuracy of the process by injecting more spatial localization information in the visual representation, a different scheme, based on a dense sampling of the image with a regular grid (possibly defined over a range of scales) is proposed in [11], [12], [13]. Such approaches prove to be particularly useful for stereo matching [12]. On the downside, dense sampling cannot reach the same level of repeatability as obtained with interest points, unless sampling is performed extremely densely, in which case the number of features becomes unacceptably large.

In order to combine the advantages of both schemes, Tuytelaars [14] has recently introduced the dense interest points, starting from densely sampled image patches and then applying for each feature a local optimization of the position and scale within a bounded search area. The outcome of this process is a set of interest points on a semi-regular grid, densely covering the entire image as is the case with dense sampling, but with repeatability properties closer to those of standard interest points.

Browne and Smeaton [15] propose a different approach. In order to perform character retrieval in animated videos, they use a number of templates of each object to be detected and a matching procedure to compare each image against the available templates. In this case, templates are represented by all the yellow parts of the faces of the cartoon characters from "The Simpsons" series.

In [16], authors generate for each keyframe a hierarchy of regions represented by a Binary Partition Tree. Various visual descriptors are extracted from each region and used to create visual codebooks. Another region-based approach is proposed in [17], where frames are divided in rectangular cells forming a grid and the descriptors of each cell are used. Histogram-based descriptors (*e.g.* HSV histogram, MPEG-7 Edge Histogram, Wavelet histogram) are here used in order to cluster the cells into a Bag of Features to be compared with a dictionary.

Gould *et al.* [19] propose to combine appearance-based features computed on superpixels [18] patches with relative location priors in a two stage classification process. Malisiewicz *et al.* [20] have shown that using image segmentation is efficient to improve the spatial support for object detection and recognition.

In [21], the authors construct “region adjacency graphs” of pre-segmented objects and retrieve similar objects with the help of a new graph matching method based on an improvement of relaxation labeling techniques. In [22] image segments resulted from different segmentation algorithms are used as primitives to extract other features (*e.g.* color, texture and interest points) and for training detection models for a predefined set of categories.

Finally, let us mention the approach introduced in [23]. Here, a different, region-based representation is proposed. The idea is to represent the image as a dense map of (overlapping) regions.

The advantage of region-based approaches comes from the possibility of directly exploiting the connectivity information (*i.e.* adjacency between regions), which can be highly useful in the matching stage.

The approach proposed in this paper adopts a region-based representation strategy, which involves an over-segmentation of the image. Let us underline that arbitrary segmentation methods can be considered, since our goal is achieve independency to the adopted segmentation procedure. The obtained representation is described with the help of an extended version of the MPEG-7 dominant color descriptor (DCD) [24]. Finally, two matching strategies are proposed in order to retrieve similar objects of interest.

The rest of the paper is structured as follows. Section 2 recalls the MPEG-7 DCD representation, together with the associated similarity measures. Section 3 introduces the two algorithms proposed for performing partial image matching. Experimental results are presented and discussed in Section 4. Finally, Section 5 concludes the paper and opens perspectives of future work.

2 DCD Representation

The video document is first segmented in shots and for each shot, a set of representative key-frames is determined, using the approach recently proposed in [27]. The object search process is further performed uniquely upon the obtained key-frames. This makes it possible to significantly reduce the computational complexity.

Each key-frame is then segmented by applying standard algorithms. In our case, we have used the Mean Shift technique proposed in [28], but other segmentation methods can be used as well. Each region (or segment) determined is described by a unique, homogeneous color, defined as the mean value of the pixels of the given region. The set of colors, together with their percentage of occupation in the image (*i.e.*, the associated color histogram) are regrouped into a visual representation, which is similar to the MPEG-7 DCD. More precisely, let $C_I = \{c_1^I,\ c_2^I,\ \dots\ c_{N_I}^I\}$ be the set of N_I colors obtained for image I, and $H_I = (p_1^I,\ p_2^I,\ \dots\ p_{N_I}^I)$ the associated color histogram vector. The visual image representation is defined as the couple (C_I, H_I). The difference here is that an arbitrary number of dominant colors is supported, in contrast with the MPEG-7 DCD, where the maximal number of colors is limited to eight. More sophisticated DCD-based approaches [25], [26], can also be considered.

The query is by definition an object of arbitrary shape and is processed in the same manner in order to derive its visual representation.

The advantage of the DCD representation comes from the fact that objects with arbitrary numbers of colors can be efficiently compared by using, for example, the Quadratic Form Distance Measure introduced in [29], which can be re-written for arbitrary length representations as described by the following equation:

$$D_h^2(H_Q, H_I) = \sum_{i=1}^{N_Q} \sum_{k=1}^{N_Q} a\left(c_i^Q,\, c_k^Q\right) p_i^Q p_k^Q \;+\; \sum_{j=1}^{N_I} \sum_{l=1}^{N_I} a\left(c_j^I,\, c_l^I\right) p_j^I p_l^I \;- \sum_{i=1}^{N_Q} \sum_{j=1}^{N_I} a\left(c_i^Q,\, c_j^I\right) p_i^Q p_j^I \quad , \tag{1}$$

where $H_Q = (p_1^Q,\, p_2^Q,\, \dots\, p_{N_Q}^Q)$ and $H_I = (p_1^I,\, p_2^I,\, \dots\, p_{N_I}^I)$ respectively denote the DCD histogram vectors of length N_Q, and N_I respectively associated to the query (Q) and candidate (I) images. The function a, describe the similarity between two colors c_i and c_j and is defined as:

$$a\left(c_i,\, c_j\right) \;= 1 - \frac{d(c_i, c_j)}{d_{max}} \tag{2}$$

where d is the Euclidean distance between colors c_i and c_j and d_{max} is the maximum Euclidean distance between any 2 colors in the considered color space (*e.g.*, for the RGB color space $d_{max} \cong 442$).

Let us note that each color region in a candidate image has a specific contribution to the global distance. Thus, the contribution of color c_j^I in an image I to the global distance between image I and query Q is defined as:

$$C(c_j^I, Q) = \;\sum_{l=1}^{N_I} a\left(c_j^I,\, c_l^I\right) p_j^I p_l^I - \sum_{i=1}^{N_Q} a\left(c_i^Q,\, c_j^I\right) p_i^Q p_j^I \tag{3}$$

The above-defined distance is used as a global criterion in the matching stage. Here, the objective is to determine, in each key-frame of the considered video sequence, candidate regions visually similar with the query.

3 Dynamic Region Construction

First, we eliminate the far-off colors, based on a color similarity criterion. A permissive threshold is here used, the goal being to reject highly improbable colors but in the same time to keep a sufficiently large variety of candidate regions. We then label the rest of the regions in connected components and consider each of them as a candidate for our query.

Next, the objective is to develop a dynamic region construction algorithm. This represents the core of the proposed methodology. In this stage, the candidate object is iteratively refined by removing and adding individual segments until the global matching distance is minimized. To achieve this goal, we have tested a recursive, greedy optimization method as well as the simulated annealing algorithm [30]. An overview of the proposed approach is illustrated in Fig.1.

In order to quickly retrieve the different instances of the selected video inside the current video or from other videos we have first developed a greedy region construction algorithm.

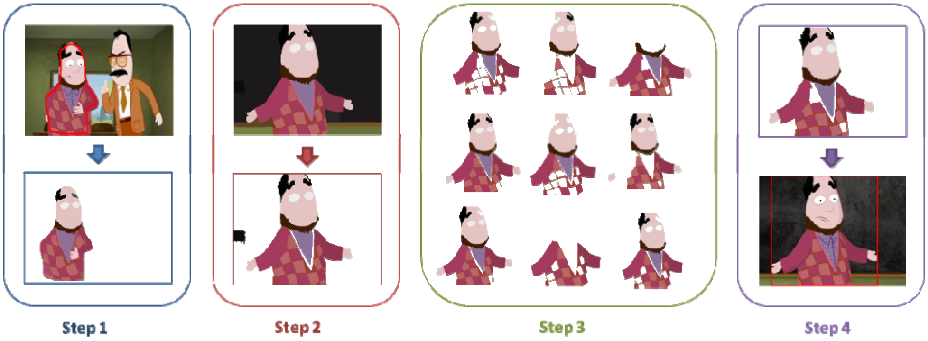


Fig. 1. Overview of the algorithm: (1): The user selects an object and the corresponding segmented region is extracted; (2): Regions with colors highly different from the query are filtered out; (3): Different configurations of candidate objects are generated by adding/removing color segments; (4): The object with the minimal score is selected and displayed in its bounding box.

3.1 Greedy Region Construction

The algorithm starts from the initial set of regions obtained after the filtering stage described in the previous section. At each stage, we consider the current candidate object in image *I* and attempt to improve the current similarity measure between query and candidate objects. More precisely, we recursively eliminate the color segment which provides the highest contribution to the global distance (equation 3). We then check if the global distance is decreasing or not. If yes, we eliminate the corresponding region, update the color frequency vector H_I, and re-iterate the algorithm on the new candidate object obtained. If not, the region is maintained and the algorithm successively tries to eliminate the following regions (sorted by decreasing order of their contribution to the global distance).

Let us note that each time an attempt to eliminate a segment is performed, the region connectivity needs to be re-calculated in order to determine the eventually newly created connected components. Each connected component is then treated separately.

The algorithm stops when no improvement to the current global distance is obtained, whatever the region under investigation. We then return to the previously obtained best score configuration and stop the algorithm.

The strategy of recursively eliminating the highest contributor to the global score increases the speed of the algorithm, by pruning the search space. However, the risk is to remain blocked in a local minimum, because whenever the distance is increasing, the algorithm stops.

For this reason we investigated a second approach, based on simulated annealing global optimization, described in the next section.

3.2 Simulated Annealing Optimization

The Simulated Annealing (SA) algorithm is a well-known stochastic optimization technique inspired from the behavior of condensed matter at low temperatures. The procedure employs methods that originated from statistical mechanics to find global minima of systems with large numbers degrees of freedom. The correspondence between combinatorial optimization problems and the way natural systems search for the ground state (lowest energy state) was first realized by Kirkpatrick *et al.* [30] who applied Monte Carlo methods to the solution of global optimization problems. Furthermore, authors generalized the Metropolis algorithm [31] by using an approach with successively decreasing temperatures. At each stage, the system is simulated by the Metropolis procedure until the system reaches equilibrium.

At the beginning of the procedure, the system starts with a high temperature *T*. Then, a cooling (or annealing) scheme is applied by slowly decreasing T according to some given procedure. At each T a series of random new states are generated and the states that improve the cost function are accepted. Instead of always rejecting states that do not improve the cost function, such states can be accepted with some probability depending both on the amount of energy increase and of the current temperature *T*. This process randomizes the iterative improvement phase and also allows occasional uphill moves (*i.e*., moves that do not improve the solution) in an attempt to reduce the probability of blocking the algorithm into a local minimum.

As temperature *T* decreases, configurations that increase the cost function are more likely to be rejected. It has been demonstrated that the SA procedure is asymptotically optimal, *i.e.* leads to a solution that is arbitrary close to the global minimum [32].

We employ the SA algorithm in order to take advantage of the higher number of possible region configurations and, thus, find the global optimum score for all candidate regions.

The energy E considered here is defined as the global matching score between query and candidate objects (equation 1). To each color segment c, a binary state S = $S(c)$ is associated with. If the color state S is equal to 0 than color segment c is considered as not belonging to the current candidate object. On the contrary, if the state *S* has a value 1, then c is considered as part of the current object.

At each step, the algorithm attempts to change the stage of the current color segment, by investigating the variation of the global energy $\Delta E = E(S') - E(S)$ when the state *S* is set to its complementary value *S*'.

If the variation $\Delta E \leq 0$, then the current state *S* is replaced by its complementary value S'.

If the energy variation ΔE has a positive value, then a random variable α, $0 \leq \alpha \leq 1$ is generated. The current state S is replaced by S' if the following condition is satisfied:

$$\alpha \leq e^{(-\Delta E/T)} \text{ and } (S') < E(S) \; , \tag{4}$$

where *T* denotes the value of the temperature at the current state.

A multiple number of iterations, denote by n_{it} is performed at a given temperature. Then, the temperature of the system is then iteratively lowered, according to a given freezing scheme (or annealing schedule). In our work, we have adopted the following temperature variation:

$$T_{n+1} = \tau T_n \tag{5}$$

where τ is the (constant) cooling rate with value between 0 and 1, and T_n is the temperature at the n^{th} iteration. Let us note that as the temperature is decreased by τT_n, the probability of accepting a large decrease decays exponentially towards zero.

The algorithm starts at an initial temperature T_0 and stops when a freezing temperature T_f is reached.

Let us observe that, in contrast with the greedy-based approaches (where the only operation supported is the removal a given segment), here a color segment can be both removed and added to the current candidate object.

A particular attention has to be paid to the parameters involved in the considered freezing scheme.

Keeping the same temperature for a long period of time will guarantee finding the best solutions since the SA algorithm is asymptotically optimal. This means that the longer the algorithm runs the better is the quality of the solution obtained. However, from a practical point of view, this is not acceptable because of computational issues.

In order to overcome such limitations, we introduce some additional conditions for stopping the algorithm earlier. Thus, if after a certain number of consecutive decreases of temperature and state perturbations, no new solution has been accepted, we stop the annealing process. In our experiments, we have considered the maximum number of rejected consecutive states as twice the number of color segments, corresponding to a test of the two states of each color segment.

Concerning the other parameter values, we consider the initial temperature T_0 between 0.5 and 0.9 and the freezing temperature T_f in the range 10^{-2} – 10^{-4}. Finally, the typical values of the cooling factor τ are in the range from 0.9 to 0.99.

Let us now describe the experimental results obtained.

4 Experimental Results

We have tested both two proposed method on the Raymond corpus[1] consisting of 13 cartoon videos, each with the duration of about 7 minutes. Such videos have of course the particularity of presenting relatively flat colors, which facilitates the segmentation task. However, they present an impressive number of characters displaying the same palette of colors. Moreover these characters are often located in crowded scenes (up to 20 characters in certain scenes), which is quite a challenge for object detection purposes.

The videos have been first segmented in shots and for each shot, a representative key-frame has been determined with the help of the methods recently proposed in [27]. We thus constructed a database of about 1630 video shots with 1630 corresponding key-frames with approximately 80-100 regions per frame.

[1] Corpus established by the “2minutes” company (www.2minutes.fr) within the framework of HD3D^2 French research project.

Experiments have been run on a PC with Intel Xeon CPU W3530 at 2.80 GHz and 12 Gb RAM.

On this data set, concerning the time of response to the queries, when searching for an object within a given video, we achieved less than 2 seconds with the greedy algorithm and 10 to 30 seconds for the SA-based approach.

Both methods have been integrated in the OVIDIUS platform [33], which is an online video indexing system integrating several video search modules. With the help of HTML5 and JavaScript tools, the users can select an arbitrary shape region corresponding to an object of interest directly and query the system in order to retrieve other instances of the same object throughout the video.

Fig. 2 shows an example of query performed with OVIDIUS.

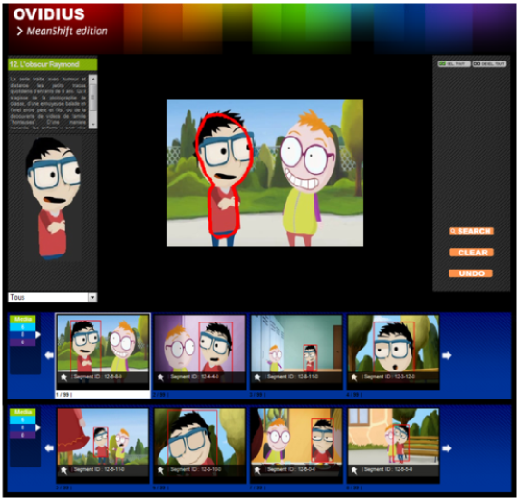


Fig. 2. The results of a query performed by the user with the OVIDIUS patform. The retrieved results are ordered in decreasing similarity to the query for the characters selected in the left side of the figure.

The upper row shows the greedy results, while the lower one those obtained with the SA approach. We observe that in both cases the selected character has been successfully retrieved, whatever the background of the scene, the object's size and position.

In order to objectively evaluate the performances of our approach, we have established a ground truth query set. Thus, we have considered a number of 12 objects, with 2 different queries per object (corresponding to two manual user-selections on two different instances of the same object). This yields a total number of 24 items in the ground truth data set. Some of the considered query objects are illustrated in Fig. 3.

Next, we have computed the recognition rate for each query by counting the positive results in the first N_i results (where N_i represents the total number of positive results possible for a query object i). The average recognition rate obtained when performing queries inside a given video with the greedy algorithm is of 77%. Concerning the SA method, we obtained a slightly better recognition rate of 79%.

In order to investigate the scalability issues, we have also performed the queries within the entire data set of 1630 key-frames. In this case, the performances naturally degrade with detection rates of 61% for the greedy approach and of 71% for the SA technique. By providing a more global solution, the SA approach shows a more scalable behavior. However, the price to pay is the time of response to the queries: about 160-180 seconds for the SA approach, with respect to less than 20 seconds for the greedy algorithm.

We also noticed that there are some differences between the two methods in the order of similarity of the retrieved frames, with the SA better returning nearly identical objects (Fig. 4).

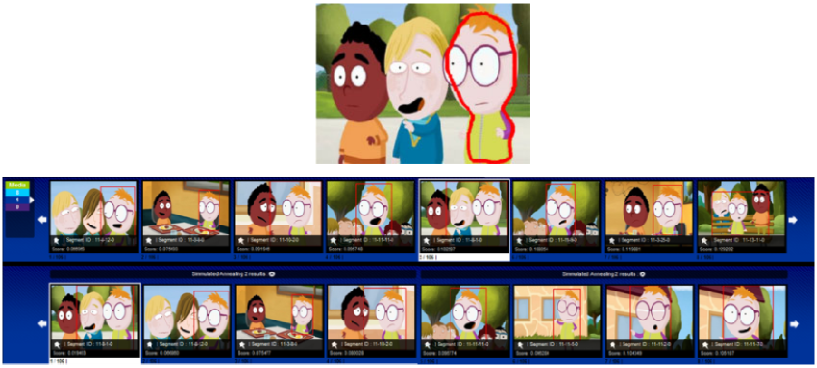


Fig. 3. First retrieved results with both the greedy region construction (upper row) and the SA (lower row) methods. The order is improved with the SA method and the obtained global scores are lower.

Moreover, when performing a finer analysis, we observe that the SA method improves the value of the score for the frames containing objects similar with the query. In order to quantify this refinement we have measured the distances obtained between a query object and the image out of which it has been selected. In average, the SA distances prove to be 20.28% inferior to those provided by the greedy algorithm. This result proves the SA method obtains more global optima, which offers interesting perspectives in terms of scalability, in the case where larger datasets have to be considered.

We have then extended our experiments to a collection of natural videos. We have used the Sound and Vision dataset used for the TRECVID 2010 Instance Search Task. We have considered 34 videos summing up to approx. 15 hours of video content and 5580 keyframes, with a single representative keyframe per shot. In this case, we have constrained the segmentation algorithm to generate a higher number of segmented regions obtaining an average of 200 regions per image.

In the case of natural videos, because of computational issues, we have considered only the greedy method. The set of the 11 queries considered is illustrated in Fig. 3.

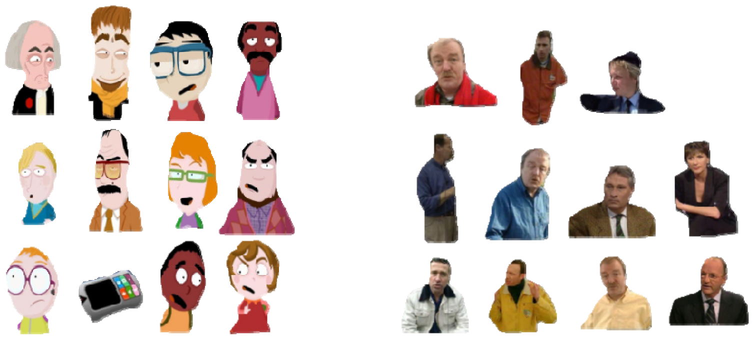


Fig. 4. Examples of queries considered in our experiments on the two datasets

The computation time for a video consisting of 200-220 shots ranged from 5 to 16 seconds, depending on the details in the model, while for a query within all the shots in the database it varied between 180 and 500 seconds.

We have computed the recognition rate in the same manner as described above. The recognition rate obtained for intra-video queries is of 51.80%, However, the performances degrade in the case of a search in the whole database of keyframes, with a global recognition rate of 24%. In our future work, we plan to integrate some additional spatial information within the DCD representation in order to increase the discriminative power of the visual representation and thus to increase the scalability.

Fig. 5 illustrates two queries with natural videos.

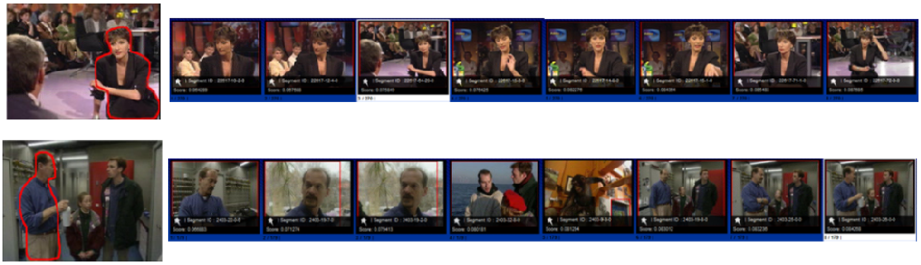


Fig. 5. Results obtained for queries on natural videos. The user selections are displayed on the left column and the results obtained are displayed in the right column in the decreasing order of similarity with the selected object.

We can observe that similar characters are obtained in both cases, despites strong variations in the appearance of the considered characters (in-door and out-door scenes, different lightening conditions and poses).

5 Conclusions and Perspectives

In this paper, we have presented two interactive, region-based, object retrieval methods which make it possible to detect throughout the video multiple instances of an object selected by the user. The first one is based on a greedy region construction, while the second involves a simulated annealing approach.

Promising retrieval results are obtained on a database of cartoon characters involving various characters in visually different scenes. We have performed some preliminary experiments on a dataset of natural videos as well.

The perspectives of future work concern the improvement of the algorithm with additional descriptors (from simple ones such as aspect ratio, physical distance and connectivity of segments to more complex ones embedding interest points descriptors) in order to adapt it on real-life videos. We are also planning to improve the speed and accuracy of our SA algorithm by adding an adaptive neighborhood functionality. Finally, we intend to subscribe our solutions to the TRECVID 2011 Instance Search Task.

Acknowledgments. The current work has been developed within the framework of the UBIMEDIA Common Laboratory established between Institut TELECOM and Alcatel-Lucent Bell Labs France.

References

1. Snoek, C.G.M., Worring, M.: Concept-Based Video Retrieval. Foundation and Trend in Information Retrieval 2(4), 215–322 (2008)
2. Smeaton, A.F., Over, P., Kraaij, W.: Evaluation campaigns and TRECVid. In: Proc. 8th ACM International Workshop on Multimedia Information Retrieval, MIR 2006, USA, October 26 - 27, pp. 321–330. ACM Press, New York (2006)
3. Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: IEEE International Conf. on Computer Vision, ICCV 2003 (2003)
4. Lowe, D.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (IJCV) 2(60), 91–110 (2004)
5. Mikolajczyk, K., Schmid, C.: An Affine Invariant Interest Point Detector. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part I. LNCS, vol. 2350, pp. 128–142. Springer, Heidelberg (2002)
6. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide baseline stereo from maximally stable extremal regions. In: British Machine Vision Conference (BMVC 2002), pp. 384–393 (2002)
7. Fergus, R., Perona, P., Zisserman, A.: Weakly supervised scale-invariant learning of models for visual recognition. Int. Journal of Computer Vision 71(3), 273–303 (2007)
8. Leibe, B., Leonardis, A., Schiele, B.: Robust object detection with interleaved categorization and segmentation. IJCV 77(1-3), 259–289 (2008)
9. Jiang, H., Drew, M.S., Li, Z.: Matching by linear programming and successive convexification. IEEE Trans. PAMI 29, 959–975 (2007)
10. Li, H., Kim, E., Huang, X., He, L.: Object matching with a locally affine-invariant constraint. In: IEEE International Conf. on Computer Vision and Pattern Recognition (CVPR 2010), pp. 1641–1648 (2010)
11. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: IEEE Conference on Computer Vision and Pattern Recognition (2005)
12. Tola, E., Lepetit, V., Fua, P.: A fast local descriptor for dense matching. In: IEEE International Conf. on Computer Vision and Pattern Recognition, CVPR 2008 (2008)

13. Tuytelaars, T., Schmid, C.: Vector quantizing feature space with a regular lattice. In: IEEE International Conf. on Computer Vision, ICCV 2007 (2007)
14. Tuytelaars, T.: Dense Interest Points. In: IEEE International Conf. on Computer Vision and Pattern Recognition (CVPR 2010), pp. 2281–2288 (2010)
15. Browne, P., Smeaton, A.F.: Video retrieval using dialogue, keyframe similarity and video objects. In: IEEE International Conf. on Image Processing (ICIP 2005), September 11-14, pp. III-1208- III-1211 (2005)
16. Foley, C., et al.: TRECVID 2010 Experiments at Dublin City University. TRECVid 2010 - Text REtrieval Conference TRECVid Workshop, Gaithersburg, MD (November 2010)
17. Gorisse, D., et al.: IRIM at TRECVID 2010: Semantic Indexing and Instance Search. TRECVid 2010 - Text REtrieval Conference TRECVid Workshop (November 2010)
18. Ren, X., Malik, J.: Learning a classification model for segmentation. In: IEEE International Conf. on Computer Vision (ICCV 2003), vol. 1, pp. 10–17 (2003)
19. Gould, S., Rodgers, J., Cohen, D., Elidan, G., Koller, D.: Multi-class segmentation with relative location prior. International Journal on Computer Vision (2008)
20. Malisiewicz, T., Efros, A.: Improving spatial support for objects via multiple segmentations. In: British Machine Vision Conference, BMVC 2007 (2007)
21. Chevalier, F., Domenger, J.P., Benois-Pineau, J., Delest, M.: Retrieval of objects in video by similarity based on graph matching. Pattern Recognition Letters 28(8), 939–949 (2007)
22. Vieux, R., Benois-Pineau, J., Domenger, J.-P., Braquelaire, A.: Segmentation-based multi-class semantic object detection. In: Multimedia Tools and Applications, pp. 1–22 (2010)
23. Kim, K., Grauman, K.: Boundary Preserving Dense Local Regions. In: IEEE International Conf. on Computer Vision and Pattern Recognition (2010)
24. Manjunath, B.S., Ohm, J.R., Vasudevan, V.V., Yamada, A.: Color and Texture Descriptors. IEEE Transactions on Circuits and Systems for Video Technology 11(6), 703–715 (2001)
25. Yang, N.C., Chang, W.H., Kuo, C.M., Li, T.H.: A fast MPEG-7 dominant color extraction with new similarity measure for image retrieval. Journal of Visual Communication and Image Representation 19(2), 92–105 (2008)
26. Zin, T.T., Tin, P., Toriu, T., Hama, H.: Dominant Color Embedded Markov Chain Model for Object Image Retrieval. In: 5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, September 12-14, pp. 186–189 (2009)
27. Tapu, R., Zaharia, T.: A complete framework for temporal video segmentation. In: Proc. IEEE Int. Conf. on Consumer Electronics Berlin (ICCE-Berlin), Germany (September 2011)
28. Comaniciu, D., Meer, P.: Mean Shift: A Robust Approach Toward Feature Space Analysis. IEEE Tran. on Pattern Analysis and Machine Intelligence, 603–619 (May 2002)
29. Hafner, J., Sawhney, H.S., Equitz, W., Flickner, M., Niblack, W.: Efficient color histogram indexing for quadratic form distance functions. IEEE Trans. Pattern Anal. Machine Intell. 17, 729–736 (1995)
30. Kirkpatrick, S., Gelatt, C.D., Vechi, M.P.: Optimization by simulated annealing. Science, 220 (1983)
31. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equations of state calculation by fast computing machines. Journal of Chemical Physics 21(6), 1087–1092 (1953)
32. Lundy, M., Mees, A.: Convergence of an annealing algorithm. Mathematical Programming 34, 111–124 (1986)
33. Bursuc, A., Zaharia, T., Prêteux, F.: Mobile Video Browsing and Retrieval with the OVIDIUS Platform. In: Proc. ACM Multimedia 2010 International Conference, Florence, Italy (October 2010)

Video Summarization Based on Balanced AV-MMR

Yingbo Li and Bernard Merialdo

EURECOM, Sophia Antipolis, France
{Yingbo.Li,Bernard.Merialdo}@eurecom.fr

Abstract. Among the techniques of video processing, video summarization is a promising approach to process the multimedia content. In this paper we present a novel summarization algorithm, Balanced Audio Video Maximal Marginal Relevance (Balanced AV-MMR or BAV-MMR), for multi-video summarization based on both audio and visual information. Balanced AV-MMR exploits the balance between audio information and visual information, and the balance of temporal information in different videos. Furthermore, audio genres and human face of each frame are analyzed in order to be exploited in Balanced AV-MMR. Compared with its predecessors, Video Maximal Marginal Relevance (Video-MMR) and Audio Video Maximal Marginal Relevance (AV-MMR), we design a novel mechanism to combine these indispensible features from video track and audio track and achieve better summaries.

Keywords: Multi-video summarization, MMR, Video-MMR, AV-MMR, Balanced AV-MMR.

1 Introduction

Nowadays video is ubiquitous in mobile phone, TV, Internet and so on. The amount of available videos is much more than the needs of a person, not mentioning that many videos are duplicates. Therefore the research on automatic video processing and retrieval has become a focused topic. Among the techniques for video processing, video summarization has been recognized as an important measure. For example, TRECVID [14] for rushes summarization has been an event in multimedia domain. Video summarization produces the summaries by analyzing the content of a source video stream, and condenses this content into an abbreviated descriptive form.

Currently many approaches of video summarization are proposed to process a single video [1] [2] [13], while there are many instances where multi-video data appears. For example, the YouTube website presents multiple related videos on the same webpage. Consequently, it is useful to discover the underlying relations inside a set of video. This need has been focused by some researchers and several successful algorithms [3] [4] have been developed. Many existing algorithms only consider the features from the video track, and neglect the audio track because of the difficulty of combining the information of audio and video. Several existing algorithms [5] [6] consider both the audio and visual information in the summarization, but they are

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 370–382, 2012.

domain-specific. In [5] the authors consider that the video segments with silent audio are useless, but it is not always like that in the real videos. In [6], an algorithm has been proposed to summarize music videos: the chorus in audio track and the repeated shots in video track are detected. But the algorithms like [5] [6] only focus on a small domain, because in a domain-specific algorithm it is easier to utilize some special features or characteristics. For example, in sports video a loud ambient noise is a strong indication that the current visual information is likely to be important. However, in a generic algorithm we cannot rely on these specific characteristics. So there are not many generic algorithms exploiting both audio and visual information until now.

In this paper we propose a generic summarization algorithm by using both audio and visual information. Our algorithm is inspired by the previous algorithms, Video-MMR [4] by only using visual information, and AV-MMR in [12] by exploiting both audio and visual information. However, AV-MMR is a simple extension of Video-MMR, and does not consider the characteristics of audio and video track. Therefore, in this paper we propose a novel algorithm, Balanced AV-MMR, to improve AV-MMR by:

- Considering the balance between audio information and visual information in a short time
- Analyzing and using the influence of audio genres
- Exploiting audio changes from one genre to another
- Analyzing and utilizing the information brought by the face
- Using the temporal distance of video frames in a set
- Finally designing a novel mechanism to combine these features

The rest of the paper is organized as follows: Section 2 reviews the principles of Video-MMR and AV-MMR. Section 3 and Section 4 discuss the property of audio track and the importance of human face. And then the theory of Balanced AV-MMR is proposed to use the features in Section 5. After that in Section 6 we present the experimental results of Balanced AV-MMR. Our conclusion is in Section 7.

2 Review of Basic Systems

The series of Maximal Marginal Relevance (MMR) algorithms in video summarization originate from MMR algorithm in text summarization [11]. The first version is Video-MMR [4] based on pure visual information, which is extended to AV-MMR [12] by simply adding the audio part into the formula of Video-MMR. Before explaining our proposed algorithm, we need to first review these two MMR algorithms in video summarization.

2.1 Video-MMR

The goal of video summarization is to select the most important instants in a video or a set of videos. Because of the similarity between text summarization and video summarization, MMR [11] is easily extended to the video domain as Video-MMR in [4]. When iteratively selecting keyframes to construct a summary, Video-MMR selects a keyframe whose visual content is similar to the content of the videos, but at the same time different from the frames already selected in the summary. Video Marginal Relevance (Video-MR) is defined as:

$$\textit{Video-MR}(f) = \lambda\, Sim_1(f, V\backslash S) - (1-\lambda) \max_{g\in S} Sim_2(f,g) \tag{1}$$

where $Sim_1(f, V\backslash S) = \frac{1}{|V\backslash(S\cup f_i)|}\sum_{f_j\in V\backslash(S\cup f_i)} sim(f_i, f_j)$, V is the set of all frames in all videos, S is the current set of selected frames, g is a frame in S and f is a candidate frame for selection. Based on this measure, a summary S_{k+1} can be constructed by iteratively selecting the keyframe with Video-MMR:

$$S_{k+1} = S_k \cup \underset{f\in V\backslash S_k}{argmax}\left(\lambda\, Sim_1(f, V\backslash S_k) - (1-\lambda) \max_{g\in S_k} Sim_2(f,g)\right) \tag{2}$$

2.2 AV-MMR

In [12] the authors have proposed AV-MMR, an algorithm that exploits the information from both audio and video. Eq. 2 of Video-MMR is extended into Eq. 3.

$$S_{k+1} = S_k \cup \underset{f\in V\backslash S_k}{argmax}[\lambda\, Sim_{I1}(f, V\backslash S_k) - (1-\lambda)\max_{g\in S_k} Sim_{I2}(f,g) +$$
$$\mu\, Sim_{A1}(f, A\backslash S_k) - (1-\mu)\max_{g\in S_k} Sim_{A2}(f,g)] \tag{3}$$

where Sim_{I1} and Sim_{I2} are the same measures as Sim_1 and Sim_2 in Eq. 2. Sim_{A1} and Sim_{A2} play roles similar to Sim_{I1} and Sim_{I2}. A and V are the collections of audio and video frames. Eq. 3 combines visual and audio similarities corresponding to the same frame, and it is called Synchronous AV-MMR.

3 Analysis of Audio Genres

Before exploiting audio information in Balanced AV-MMR it is necessary to analyze the characteristics of audio track. In this paper we analyze the audio through the genres. The audio can be classified into several genres: speech, music, speech&music, noise, silence and so on. The property of each genre is obviously different.

We consider one second as an atom, which we call “audio frame”. Besides audio frame, contiguous audio frames with the same genre (silence, music or speech) are considered as an “audio segment”. In the community of audio processing and speech recognition, Hidden Markov Model (HMM) is agreed to be a promising method. We use a successful toolkit, The Hidden Markov Model Toolkit (HTK) [9], to construct a recognition system of audio genres for audio frames. In this paper we restrict audio genres to silence, music and speech because of the limitation of training data that we could annotate. Speech&music including singing is regarded as speech here. The test data is the audio tracks of 549 videos in 89 sets from 7 categories: *Document, News, Music*, *Advertisement*, *Cartoon*, *Movie*, and *Sports*. With these various audio files, we can guarantee the diversity of our audio files.

To analyze the audio frames and segments of each video category, we compute the percentages of silence, music and speech frames or segments in all the frames or segments of each video category. The percentages of audio frames of three genres are shown in Table 1, and Table 2 represents the percentages of audio segments. Since each category of video is analyzed separately, the sum of each row in Table 1 and Table 2 is 1.

Table 1. The percentages of audio frames of each audio genre

Percentages of audio frames	Silence	Music	Speech
Document	0.0294	0.2732	0.6974
News	0.0298	0.2821	0.6881
Music	0.0259	0.2150	0.7591
Advertisement	0.0356	0.5726	0.3918
Cartoon	0.0205	0.4300	0.5495
Movie	0.0153	0.3933	0.5915
Sports	0.0508	0.4492	0.5000

Table 2. The percentages of audio segments of each audio genre

Percentages of audio segments	Silence	Music	Speech
Document	0.0554	0.4905	0.4541
News	0.0317	0.4869	0.4814
Music	0.0557	0.4629	0.4814
Advertisement	0.0979	0.4756	0.4265
Cartoon	0.0561	0.4619	0.4819
Movie	0.0530	0.4899	0.4571
Sports	0.1074	0.4862	0.4065

In Table 1 and Table 2:

- *Sports* category obviously has the largest percentages of silence frames and segments compared with the other video categories. And the ratio of music segments to speech segments in *Sports* is high compared with the other categories.
- Compared with *Sports*, *Advertisement* category has a high percentage of silence segments but low percentage of silence frames, which means that silence segments are usually very short segments in *Advertisement*. And *Advertisement* contains short music segments and long speech segments.
- *Advertisement* is definitely different from *Sports* according to above analysis.
- Refer to *Music* and *News*, they have similar percentages of three kinds of segments, but the percentages of the frames are different. The ratio of speech to music in *Music* category is larger compared to this ratio in *News*, which is caused by the singing, even with music, being regarded as speech. Except above reason, another minor reason is that a few frames of speech are recognized as music because of our limited training data.

Limited to the space of paper, it is impossible to provide a complete description of the characteristics of every video category. Nevertheless, through the above analysis we can see that different categories of videos own obvious and different audio characteristics. The genres of audio frame and segments are indispensible features of the video.

Audio frames with the same genres seem to be more similar at the semantic level because of their same genre. Furthermore, the boundaries between audio segments, defined as "audio transition" in this paper, are important because of the possible significant changes of visual information and audio information. For example in *News* the transition from music to speech genre is probably the beginning of the speech of an anchorman or journalist. Consequently, we will exploit audio genres and audio transitions in Balanced AV-MMR to improve the existing AV-MMR.

4 Analysis of Human Face

Human face is particularly important in video track, because most of current videos are human oriented. Moreover, the video track is relevant to the audio track and the appearance of the face in the video cannot be isolated with the audio track, so we carry out the analysis of the face in different audio genres.

We exploit the toolkit provided by Mikael Nilsson in [10] to detect the face in 89 video sets mentioned in Section 3. The percentage of frames with faces of each audio genre in all the frames is shown in Table 3 for each video category. Because there are possibly several faces in a frame, we also present the percentages of the number of faces of each audio genre in the total amount of faces in Table 4 for each video category. The sum of each row is equal to 1. As well, we make the analysis of large faces. The large face here is defined as the face with both the width and height larger than 90 pixels (video size is 320 by 240 pixels). The percentages of large faces in faces of each audio genre are shown in Table 5.

Table 3. The percentage of frames with faces in the frames of each audio genre

Number of frames	Silence	Music	Speech
Document	0.0090	0.2003	0.7907
News	0.0027	0.2333	0.7640
Music	0.0039	0.1141	0.8820
Advertisement	0.0220	0.5291	0.4489
Cartoon	0.0053	0.3686	0.6262
Movie	0.0119	0.4685	0.5196
Sports	0.0313	0.3697	0.5990

Table 4. The percentages of faces of each audio genre in all the face

Number of faces	Silence	Music	Speech
Document	0.0094	0.2078	0.7828
News	0.0028	0.2330	0.7642
Music	0.0038	0.1104	0.8858
Advertisement	0.0240	0.5257	0.4502
Cartoon	0.0051	0.3784	0.6165
Movie	0.0090	0.3969	0.5941
Sports	0.0298	0.3761	0.5940

Table 5. The percentages of large faces in faces of each audio genre

Number of faces	Silence	Music	Speech
Document	0	0.2000	0.3151
News	0	0.1711	0.2921
Music	0.1000	0.0876	0.1874
Advertisement	0.2491	0.2352	0.1960
Cartoon	0.1667	0.1584	0.2269
Movie	0	0.0701	0.1543
Sports	0.2051	0.1362	0.1570

Comparing Table 3 to Table 4, the difference is little so the influence of the frames comprising multiple faces is little. And

- In video categories of *Document*, *News* and *Music*, most faces appear in speech audio frames. This is consistent with the characteristics of these videos, where the singer and reporter speak a lot.
- In *Advertisement*, speech audio frames and music audio frames almost averagely share the number of faces because faces in *Advertisement* are uniformly distributed.
- In *Sports* up to around 3% faces are in silence audio frames as there are many human actions in silence while the other videos do not have similar characteristic.

In Table 5:

- 68% faces in *Advertisement* are large faces, indicating the characteristic of extremely human orientation. It is the same for *Cartoon* and *Sports* because of the existence of a large number of large faces. So a big spatial part of video frames is the face in *Cartoon*, *Sports* and *Advertisement.*
- In *News*, *Document* and *Movie*, there is not any large face in silence audio frames, caused by the silent prologue and epilogue containing few large faces.

The viewers of video summary - human beings favor the summary covering the significant frames with the face in the video. Moreover, we have only analyzed several characteristics of the face in some categories of videos, but it is obvious that the appearance of the face in a video is consistent with the category and property of this video. So the face is important feature to improve our Balanced AV-MMR, and has strong relation with the audio track according to our analysis. Furthermore, a frame containing the face should be more similar to another frame with face than the frame without face in the semantic level.

5 Balanced AV-MMR

Assume that in a short time the audio attracted more attention from the user, the user would pay less attention to video content and vice versa, because the attention of a person in a short time is limited. In an audio segment, the duration is usually short. Therefore, there is a balance between audio information and visual information in an audio segment. Consequently we give our novel algorithm the name “balance”. Balanced AV-MMR exploits the information from audio genre, the face and the time to improve the balance information and similarities of frames in semantic level.

According to the analysis of Section 3 and 4, audio genre and the face are important features in the video, which can influence the balance between audio and video in an audio segment. When audio transition happens, there is a significant change in the audio. At that time the user would pay more attention to the audio and the audio becomes more important than usual in the balance. Similarly, when the face appears in the video track of an audio segment, the video content becomes more important in the balance.

Moreover, the face and audio genre can influence the similarities between frames in the semantic level. For video track the similarity of two frames both containing the face is larger than the similarity between one frame with the face and another frame without the face. For audio track two frames from the same audio genre, for example the speech, are more similar.

In a video two closer frames according to time seem to be redundant. Two frames in a video seem less different from two frames from two individual and non-duplicated videos, even if they have the same similarities according to low-level features. Therefore it is necessary to consider the influence of temporal information on our summarization.

In this section, we will introduce several factors of audio, face and time to AV-MMR and propose the variants of Balanced AV-MMR.

5.1 Fundamental Balanced AV-MMR

From the formula of AV-MMR and the analysis of the balance between audio and video information in a segment, we introduce the balance factor between visual and audio information and generalize the fundamental formula of Balanced AV-MMR as:

$$\begin{aligned} f_{k+1} = arg\ max_{f \in V \backslash S_k} \{ & \rho(f) [\lambda\, Sim_{I1}(f, V \backslash S_k) \ - \ (1-\lambda) \max_{g \in S_k} Sim_{I2}(f, g)] \\ & + (1-\rho(f)) [\mu\, Sim_{A1}(f, A \backslash S_k) - (1-\mu) \max_{g \in S_k} Sim_{A2}(f, g)] \} \end{aligned} \quad (4)$$

In [12], it indicates $\lambda = 0.7$ and $\mu = 0.5$. Through bringing ρ into Eq. 4, Balanced AV-MMR considers the balance between audio and video. When ρ increases, the visual information takes a more important role in Balanced AV-MMR, and vice versa. Eq. 4 is our fundamental formula for the following variants. When ρ is equal to 0.5, Eq. 4 degenerates into AV-MMR.

5.2 Balanced AV-MMR V1: Using Audio Genre

Through the audio analysis in Section 3, we have known that audio genre is an important feature and can reflect the characteristics of the videos. It is obvious that the audio frames with the same genre are more similar than the audio frames with different genres, even if they own the same similarity according to the audio features like Mel-frequency cepstral coefficients (MFCC). MFCC is used to compute the similarity of the short-term power property of two audio frames, but their similarity of semantic level cannot be reflected. Consequently, we can introduce an augment factor for audio genres to adjust the similarity of MFCC vectors. Here we use τ to denote this factor. $Sim_{A1}(f, A\backslash S_k)$ and $Sim_{A2}(f, g)$ in Eq. 4 become:

$$sim'_{A1}(f_i, A\backslash S_k) = \frac{1}{|A\backslash(S_k \cup f_i)|}\sum_{f_j \in A\backslash(S_k \cup f_i)} \tau(f_i, f_j) sim(f_i, f_j)$$
$$sim'_{A2}(f, g) = \tau(f, g) sim(f, g) \tag{5}$$

where $sim(f_i, f_j)$ and $sim(f, g)$ are original similarities by MFCC, same with the definitions in Eq. 3 and Eq. 4. And $\tau(f_i, f_g) = 1 + \theta_\tau \cdot (\theta_P - |P(f_i) - P(f_g)|)$. θ_τ is a weight to adjust the influence of the audio genre. $\theta_P = 0.2$. $P(f_i)$ and $P(f_g) = 0, 0.1, \text{or } 0.2$ when the audio frame f_i is silence, music or speech genres.

Audio transitions indicate significant audio changes. In *Music* category, the transition from silence or music audio to speech audio indicates the possible appearance of the singer, beginning singing at that time. In *News* category, the transition from silence audio to speech audio usually indicates the start of the news by a journalist or an anchorperson.

Around audio transition the user would pay more attention to the audio and less attention to the video track, according to our balance principle. Consequently we modify the balance ratio ρ to ρ' by considering the transition factor φ_{tr}:

$$\rho'(f) = \frac{\rho(f)}{\rho(f) + (1-\rho(f)) \cdot (1+\varphi_{tr}(f))} = \frac{\rho(f)}{1+\varphi_{tr}(f) - \rho(f) \cdot \varphi_{tr}(f)} \tag{6}$$

Because of φ_{tr} and $\tau(f_i, f_j)$, the fundamental formula of Balanced AV-MMR, Eq. 4, transforms into the following formula:

$$f_{k+1} = arg\, max_{f \in V\backslash S_k}\{\rho'(f)[\lambda\, Sim_{I1}(f, V\backslash S_k) - (1-\lambda)\max_{g \in S_k} Sim_{I2}(f, g)$$
$$+(1-\rho'(f))[\mu\, Sim'_{A1}(f, A\backslash S_k) - (1-\mu)\max_{g \in S_k} Sim'_{A2}(f, g)] \tag{7}$$

5.3 Balanced AV-MMR V2: Using Face Detection

According to the analysis in Section 4, the face is extremely important in visual information, so the video frame becomes more important when the face appears in a video frame. Since our balance principle favors one hand and dislikes the other hand in audio and visual information, the balance factor ρ' should increase in this case. After introducing the face factor β_{face} to $\rho'(f)$ in Subsection 5.2, it becomes:

$$\rho''(f) = \frac{\rho(f) \cdot (1+\beta_{face}(f))}{\rho(f) \cdot (1+\beta_{face}(f)) + (1-\rho(f)) \cdot (1+\varphi_{tr}(f))} = \frac{\rho(f) \cdot (1+\beta_{face}(f))}{1+\varphi_{tr}(f) + \rho(f) \cdot (\beta_{face}(f) - \varphi_{tr}(f))} \tag{8}$$

where $\beta_{face}(f) = 1 + facenumber(f) * \theta_{face}$. θ_{face} is a weight for adjusting the influence of the face.

Besides the balance factor $\rho''(f)$, the appearance of face also influences the similarity of two video frames. In semantic level, a frame comprising faces are more similar to another frame with faces than the frame without face. Also, two frames with the face often reveal the relevant content of the video, such as the different or same journalists in *News* and actors in *Movie*. Therefore the similarities Sim_{I1} and Sim_{I2} in Eq. 4 evolve into:

$$Sim'_{I1}(f_i, V\backslash S_k) = \frac{1}{|V\backslash(S_k \cup f_i)|} \cdot \sum_{f_j \in V\backslash(S_k \cup f_i)} \beta'_{face}(f_i, f_j) sim(f_i, f_j)$$
$$Sim'_{I2}(f, g) = \beta'_{face}(f, g) \cdot sim(f, g) \tag{9}$$

where $\beta'_{face}(f_i, f_j) = 1 + (facenumber(f_i) + facenumber(f_j))/2 * \theta_{face}$.

Based on above development, Eq. 7 of Balanced AV-MMR V1 can be reformulated as:

$$f_{k+1} = arg\, max_{f \in V\backslash S_k} \{\rho''(f)[\lambda\, Sim'_{I1}(f, V\backslash S_k) - (1-\lambda) \max_{g \in S_k} Sim'_{I1}(f, g)]$$
$$+ (1 - \rho''(f))[\mu\, Sim'_{A1}(f, A\backslash S_k) - (1-\mu) \max_{g \in S_k} Sim'_{A2}(f, g)]\} \tag{10}$$

5.4 Balanced AV-MMR V3: Adding Temporal Distance Factor

At last, we prefer considering the influence of temporal distance of two frames f_i and f_j, from the same video or not, on the visual and audio similarities:

- Closer frames according to time in a video commonly represent more relevant content, so two closer frames in a video are regarded more similar than two further frames.
- For multiple videos, a frame is more similar to another frame in the same video than a frame from another non-duplicated video.

Then we can consider temporal information for selecting frames from multiple videos to the summary. This balance is called “temporal balance”. The temporal factor is named as α_{time} and

$$\alpha_{time}(f_i, f_j) = \begin{cases} 1 & , if\ f_i\ and\ f_j\ are\ from\ two\ videos; \\ 1 + \theta_{time} \cdot \left(1 - \frac{|t(f_i) - t(f_j)|}{10 * D_M}\right) & , if\ f_i\ and\ f_j\ are\ from\ the\ same\ video. \end{cases} \tag{11}$$

where $t(f_i)$ and $t(f_j)$ are the frame times of f_i and f_j in video M. D_M is the total duration of video M. θ_{time} is a weight to adjust the influence of the temporal distance. Then the similarities of the frames in Balanced AV-MMR become:

$$Sim''_{I1}(f_i, V\backslash S_k) = \frac{1}{|V\backslash(S_k \cup f_i)|} \cdot \sum_{f_j \in V\backslash(S_k \cup f_i)} \beta'_{face}(f_i, f_j) \alpha_{time}(f_i, f_j) sim(f_i, f_j)$$
$$Sim''_{A1}(f_i, A\backslash S_k) = \frac{1}{|A\backslash(S_k \cup f_i)|} \cdot \sum_{f_j \in A\backslash(S_k \cup f_i)} \tau(f_i, f_j) \alpha_{time}(f_i, f_j) sim(f_i, f_j) \tag{12}$$

$Sim'_{I2}$and Sim'_{A2} are similarly multiplied by α_{time} and become $Sim''_{I2}$and Sim''_{A2}. Consequently, the formula of Balanced AV-MMR V3 is similar to Eq. 10 of Balanced AV-MMR V2 and generalized as

$$f_{k+1} = arg\, max_{f \in V\backslash S_k} \{\rho''(f)[\lambda\, Sim''_{I1}(f, V\backslash S_k) - (1-\lambda) \max_{g \in S_k} Sim''_{I1}(f, g)]$$
$$+ (1 - \rho''(f))[\mu\, Sim''_{A1}(f, A\backslash S_k) - (1-\mu) \max_{g \in S_k} Sim''_{A2}(f, g)]\} \tag{13}$$

5.5 The Procedure of Balanced AV-MMR

In above subsections, we have explained the formulas of fundamental BAV-MMR, BAV-MMR V1, BAV-MMR V2 and BAV-MMR V3. We need to generalize the procedure of Balanced AV-MMR:

1) Detect the audio genres of the frames by HTK audio system described in Section 3, and the face by the toolkit in Section 4;
2) Compute importance ratio ρ, $\rho^{'}$, or $\rho^{''}$ for each audio segment;
3) The initial video summary S_1 is initialized with one frame, defined as:

$$S_1 = arg \max_{f_i, f_i \neq f_j} [\prod_{j=1}^{n} Sim_I(f_i, f_j) \prod_{j=1}^{n} Sim_A(f_i, f_j)]^{\frac{1}{n}} \tag{14}$$

where f_i and f_j are frames in video set V, and n is the total number of frames except f_i. Sim_I computes similarity of visual information between f_i and f_j; while Sim_A is the similarity of audio;

4) Select the frame f_{k+1} by the formula of a variant of Balanced AV-MMR;
5) Set $S_{k+1} = S_k \cup \{f_{k+1}\}$.
6) Iterate to step 5) until S has reached the predefined size.

6 Experimental Results

Our experimental videos are 36 video sets from 7 categories mentioned in Section 3, comprising 194 videos. Each video set contains 3-15 videos, each of which has the duration of 10 seconds to more than 10 minutes. The diversity of our experimental videos ensures the generic property of the summary produced by BAV-MMR.

The visual content of a keyframe is represented by the Bag-Of-Word (BOW) feature. BOW feature vector of a keyframe is the histogram of the number of visual words that appear in the keyframe. The similarity between two keyframes $sim(f_i, f_j)$ is computed as $sim_I(f_i, f_j) = \cos\left(H_{f_i}, H_{f_j}\right) = \frac{H_{f_i} \cdot H_{f_j}}{\|H_{f_i}\| \|H_{f_j}\|}$, where H_{f_i} and H_{f_j} are the visual word histograms of keyframes f_i and f_j. Audio feature uses MFCC obtained by SPro Toolkit [8]. The similarity of two averaged MFCC vectors is computed and normalized as $sim_A(a_i, a_j) = 1 - \frac{|a_i - a_j|}{\max_{a_m, a_n \in S_{MFCC}}(|a_m - a_n|)}$, where a_i, a_j, $a_m$and a_n are averaged MFCC vectors.

To verify the effect of BAV-MMR, we use Audio Video Distance (AVD) and Video Distance (VD) of the summary with the original videos. AVD is defined as $d_{AVD}(S, V) = \frac{1}{n} \sum_{j=1}^{n} \min_{f_j \in V, g \in S} \left[1 - (sim_I(f_j, g) + sim_A(f_j, g))/2\right]$, where n is the number of frames in V. g and f_j are frames respectively from video summary S and V. And similarly VD is defined as $d_{VD}(S, V) = \frac{1}{n} \sum_{j=1}^{n} \min_{f_j \in V, g \in S} (1 - sim_I(f_j, g))$.

In the fundamental formula of BAV-MMR, Eq. 4, we need to decide the value of parameter ρ. In this paper we consider ρ as a constant value for different frames. To remain consistent with Video-MMR, we use the same method, Summary Reference Comparison (SRC), comparing the summary qualities from different weights, to decide ρ. ρ varies from 0.0 to 1.0, with each step of 0.1. The results of SRC are shown in Fig. 1. SRC here uses $d_{AVD}(S,V)$.

From Fig. 1 when $\rho = 0$, d_{AVD} is large when the summary size is small and vice versa. Since we want a commonly small d_{AVD} for different summary sizes, at last we select $\rho = 0.5$.

By trial and error, the various parameters in the variants of BAV-MMR are set to the following values:

- In Eq. 6 $\varphi_{tr}(f) = 0.1$ when the audio transits from silence to music at f and vice versa, or from speech to music and vice versa; $\varphi_{tr}(f) = 0.2$ when the audio transits from silence to speech and vice versa; and $\varphi_{tr}(f) = 0$ if there is not any audio transition in frame f.
- The weights θ_τ, θ_{time} and θ_{face} are chosen as 0.3, 0.3 and 0.2.

The means of AVDs and VDs of 36 experimental video sets from Video-MMR, AV-MMR and the variants of BAV-MMR are shown in Fig. 2 and Fig. 3. We have not drawn the curve of the fundamental BAV-MMR with $\rho = 0.5$, which is the same with AV-MMR in Fig. 1. It is clear that the variants of BAV-MMR are better than Video-MMR and AV-MMR because of the smaller distances with the original videos. Among the variants of BAV-MMR, BAV-MMR V1 is better than AV-MMR, and BAV-MMR V2 is better than BAV-MMR V1. While BAV-MMR V3 outperforms BAV-MMR V2 a lot because BAV-MMR V3 improves the algorithm in both audio and video track, but BAV-MMR V1 and BAV-MMR V2 separately improves audio track and video track in the summarization.

When the summary size increases, the improvements of BAV-MMR V1 and BAV-MMR V2 is not as good as the smaller summary size, which is caused by more

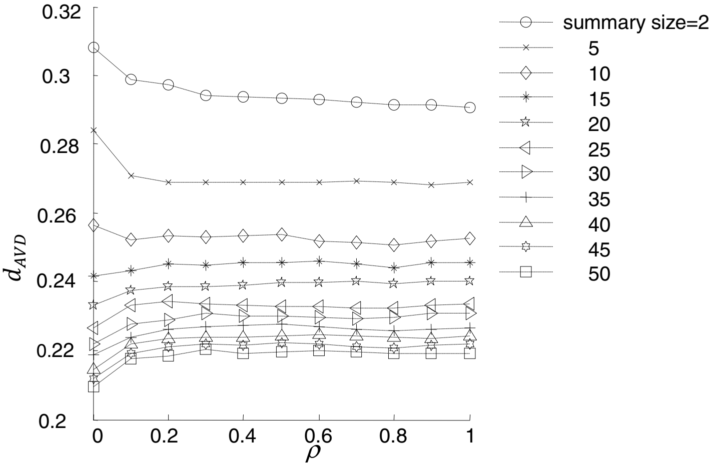


Fig. 1. SRC of ρ_T

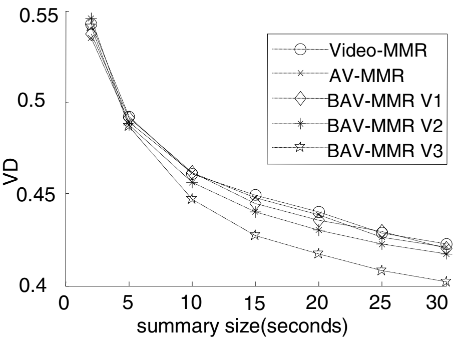


Fig. 2. VDs of different measures

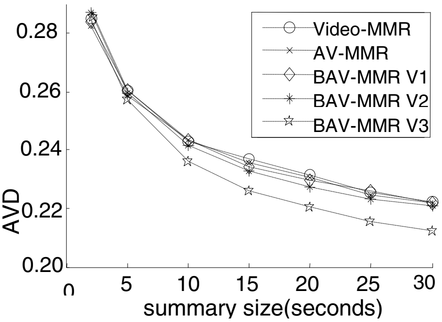


Fig. 3. AVDs of different measures

various audio genres and more face types in the summaries. However, the temporal information is not influenced a lot by the selected frames in the summaries, so BAV-MMR V3 keeps its curve trend when the summary size increases.

The limitation of BAV-MMR is the manual decisions of the weights φ_{tr}, θ_{τ}, θ_{time} *and* θ_{face}. So it is necessary to automatically and optimally tune these weights for a generic summarization algorithm. A particular set of optimized weights for each category of video is favorable. Furthermore, BAV-MMR may benefit from a variable ρ according to the property of frame or segment.

7 Conclusion

In this paper, we have proposed a novel summarization algorithm, Balanced AV-MMR by considering the balance between audio and visual information in a segment, and temporal balance of inter- and intra- video. Besides, we use audio genre and the face to adjust the similarities of the frames. Balanced AV-MMR is a new improvement of the series of MMR algorithms in video summarization. And several variants of BAV-MMR have been proposed and proved better than previous algorithms. However the weights in Balanced AV-MMR are manually decided, so it is necessary to automatically optimize the weights to the category of the video, summary size, and so on in the future.

References

[1] Yahiaoui, I., Merialdo, B., Huet, B.: Automatic Video Summarization. In: Multimedia Content-based Indexing and Retrieval, Rocquencourt, France (September 2001)

[2] Money, A.G., Agius, H.: Video summarisation: A conceptual framework and survey of the state of the art. Journal on Visual Communication & Image Representation (2008)

[3] Wang, F., Merialdo, B.: Multi-document Video Summarization. In: ICME, New York City, USA (2009)

[4] Li, Y., Merialdo, B.: Multi-Video Summarization Based on Video-MMR. In: International Workshop on Image Analysis for Multimedia Interactive Services, Italy (2010)

[5] Furini, M., Ghini, V.: An Audio-Video Summarization Scheme Based on Audio and Video Analysis. In: IEEE CCNC Proceedings (2006)

[6] Xu, C., Shao, X., Maddags, N.C., Kankanhalli, M.S.: Automatic Music Video Summarization Based on Audio-Visual-Text Analysis and Alignment. In: ACM SIGIR, Salvador, Brazil (2005)

[7] Das, D., Martins, A.F.T.: A Survey on Automatic Text Summarization. Literature Survey for the Language and Statistics II course at CMU. (November 2007)

[8] SPro Toolkit, http://www.irisa.fr/metiss/guig/spro

[9] University of Cambridge, http://htk.eng.cam.ac.uk

[10] Nilsson, M., Nordberg, J., Claesson, I.: Face Detection using Local SMQT Features and Split Up SNoW Classifier. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (2007)

[11] Carbonell, J., Goldstein, J.: The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In: SIGIR, Melbourne Australia (1998)

[12] Li, Y., Merialdo, B.: Multi-Video summarization based on AV-MMR. In: International Workshop on Content-based Multimedia Indexing, France (2010)

[13] Truong, B., Venkatesh, S.: Video abstraction: A systematic review and classification. ACM Transactions on Multimedia Computing, Communications and Applications 3(1) (January 2007)

[14] TRECVID, http://trecvid.nist.gov/

Sequence Kernels for Clustering and Visualizing Near Duplicate Video Segments

Werner Bailer

DIGITAL – Institute for Information and Communication Technologies
JOANNEUM RESEARCH Forschungsgesellschaft mbH, Graz, Austria
werner.bailer@joanneum.at

Abstract. Organizing and visualizing video collections containing a high number of near duplicates is an important problem in film and video post-production. While kernels for matching sequences of feature vectors have been used e.g. for classification of video segments, kernel-based methods have not yet been applied to matching near duplicate video segments. In this paper we survey the application of six sequence-based kernels to clustering near duplicate video segments using kernel k-means and hierarchical clustering, and the application of kernel PCA for generating content visualizations for browsing. Evaluation on the TRECVID 2007 BBC rushes data set shows that the results of the kernel based methods are comparable to other approaches for matching near duplicates, eliminating differences between dynamic time warping and string matching. These results show that hierarchical clustering outperforms kernel k-means. We also show that well-arranged visualizations of both single- and multi-view content sets can be obtained using kernel PCA.

1 Introduction

In this paper we consider the problem of organizing and visualizing video collections containing a high number of near duplicates. Such collections exist for example in the film and video production process, where a large amount of raw material is shot, and a small fraction of it is selected for use in post-production. The material is highly redundant, containing often many takes of the same scene, which are similar, but differ in small details. A substantial amount of literature on the problem of matching and detecting near duplicate video segments exists (for an overview see e.g. [2,16], also fostered by two iterations of the TRECVID [23] rushes summarization task.

Kernels for matching sequences of feature vectors have been proposed and applied to feature sequences from videos for problems such as classifying events or person trajectories. As several of the works on near duplicate detection use sequence-based similarity measures, it seems promising to apply such kernels to collections of near duplicate video segments. Although this seems a logical step, a recent paper [13] seems to be the only work that mentions the use of a sequence-based kernel in a video summarization system.

The rest of this paper is organized as follows. In the remainder of this section we briefly discuss related work on sequence-based kernels and the application of

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 383–394, 2012.

kernel k-means and kernel PCA to video content. Section 2 discusses several kernels for sequences of feature vectors and describe their application to clustering video segments using kernel k-means and hierarchical clustering as well as using kernel PCA for projection to a 2-dimensional space for visualization. Section 3 reports experimental results and Section 4 concludes the paper.

Several approaches for sequence matching based on the idea of the pyramid match kernel have been proposed. The original pyramid match kernel [9,11] partitions the feature space in each of the dimensions of the input feature vector. Its efficiency advantage is based on avoiding explicit distance calculations, but only counting elements that end up in the same bin of the pyramid. This assumes that the L_1 distance can be applied to the feature vectors, and no specific distance functions can be used. The vocabulary guided pyramid matching approach proposed in [10] addresses this problem, as it uses a clustering step to construct the pyramid, supporting arbitrary distance measures. The approach has been extended to spatio-temporal matching in [4], using sets of clustered SIFT and optical flow features as local descriptors. Their approach is similar to spatial pyramid matching proposed in [12], which applies the pyramid matching only to the image space (i.e., subdividing an image into a spatial pyramid, and counting features of the same type in each of the bins), but uses clustering in the feature space (i.e., the common bag of words approach).

Another temporal matching method based on the pyramid match kernel is described in [24,25]. Temporally constrained hierarchical agglomerative clustering is performed to build a structure of temporal segments. The pyramid match approach is applied to the decision values of different SVMs instead of the features. The similarity between segments is determined using the earth mover's distance and the pyramid match kernel is applied to the similarities on the different hierarchy levels. This approach explicitly assumes that the temporal order of the individual subclips is irrelevant (as is e.g. the case for news stories). Then the temporal order within the clips is aligned using linear programming.

Kernels for sequences based on dynamic time warping (DTW) [14] have been proposed. The dynamic time alignment kernel (DTAK) proposed in [22] is one of them. Instead of only considering the kernel values along the optimal DTW alignment, the time series alignment kernel proposed in [5] considers the values along all possible paths in DTW alignment.

The authors of [26] use the Levenshtein distance between sequences of clustered local descriptors for classification of still images. Recently, a kernel for matching sequences of histograms of visual words has been proposed [3]. The authors consider different similarity measures between the histograms and use them instead of symbol equality in the Needleman-Wunsch distance. The result of sequence matching is then plugged into a Gaussian kernel. In [1] a kernel based on longest common subsequence (LCSS) matching of sequences has been proposed. An arbitrary kernel can be plugged in to determine the similarity between two elements of the sequences, and the kernel value is determined as the normalized sum of the similarities along the backtracked longest common sequences.

While methods such as kernel k-means and kernel PCA have been used for features derived from video sequences (e.g., pedestrian trajectories [18]), there is little work applying these methods to matching and organizing near duplicate video content. An approach for unsupervised summarization of rushes video is proposed in [13]. It uses a technique called constrained aligned cluster analysis, for both segmentation of the input video and clustering, which is based on kernel k-means and the dynamic time alignment kernel (DTAK) [22]. Unfortunately the authors do not provide objective evaluation results for clustering repeated takes, but only an example for one video.

The contributions of this paper are the following. As only the DTAK kernel has been applied the clustering near duplicate video content, we consider also other types of sequence-based kernels in order to compare their applicability to this problem. As the distance function based on string matching clearly outperforms the one based on DTW in the experiments reported in [2], we are interested whether the same holds for the kernels based on each of these approaches. In addition to using kernel k-means for clustering, we also investigate the use of hierarchical clustering based on the kernel matrix of the sequences. Finally, we use kernel PCA to project a collection of video segments to a 2-dimensional space for visualization purposes based on the similarity of the sequences over time. To the best of our knowledge, this has not been proposed before.

2 Kernel-Based Clustering and Visualization

In this work we aim at finding similar video segments S from a set originating from a single video or a collection of videos. A segment is represented by samples s_i (e.g., frames, key frames). Each of these samples is described by a feature vector x_i, consisting of arbitrary features of this sample (or a temporal segment around this sample). In order to represent the video segment, we concatenate the individual feature vectors to form a feature vector $X = (x_1, \ldots, x_m)$ of the segment. Clearly, not every segment has the same length and/or consists of the same number of samples, thus the lengths of the feature vectors of different segments will differ. We thus need to be able to determine the similarity between such feature vectors having different lengths.

In this section, we first analyze some kernels, which can be applied to the problem of matching such feature vectors. We then discuss how these kernels can be applied to clustering near duplicate video segments using kernel k-means and hierarchical clustering, as well as to projecting video sequences to a 2-dimensional space for visualization.

2.1 Candidate Sequence Kernels

In the following, we review six kernels for sequences of feature vectors with varying lengths and harmonize the formulations of the kernel functions. We denote as $X = (x_1, \ldots, x_m)$ and $Y = (y_1, \ldots, y_n)$ the sequences of feature vectors of two segments. The term sequence denotes a possibly non-contiguous subsequence. As we intend to support arbitrary ground distances between the feature vectors of the input samples, we use a kernel for matching the feature vectors of elements of the feature sequences, denoted as $\kappa_f(x_i, y_j)$.

Earth Mover's Distance. An advantage of the EMD is that it can applied to different ground distances [19]. We use EMD in a similar way as applied in [24], but we do not use the proposed temporal alignment (TPAM), as it actually does sequence alignment, which is similar to the methods discussed below. We define a kernel using the EMD, replacing the ground distance d_{ij} with $\kappa_f(x_i, y_j)$, as

$$\kappa_{\mathrm{EMD}}(X, Y) = -\frac{\sum_{i=1}^{m} \sum_{j=1}^{n} \widehat{f_{ij}}(-\kappa_f(x_i, y_j))}{\sum_{i=1}^{m} \sum_{j=1}^{n} \widehat{f_{ij}}}, \tag{1}$$

where $\widehat{f_{ij}}$ is the optimal flow determined as

$$\begin{gathered} \widehat{f_{ij}} = -\arg\min_{f_{ij}} \sum_{i=1}^{m} \sum_{j=1}^{n} (-\kappa_f(x_i, y_j)) f_{ij}, \\ \sum_{j=1}^{n} f_{ij} \leq w_{x_i}, 1 \leq i \leq m, \text{ and} \\ \sum_{i=1}^{m} f_{ij} \leq w_{y_j}, 1 \leq j \leq n. \end{gathered} \tag{2}$$

The the weights of samples x_i and y_j are chosen as $w_{x_i} = 1/m$ and $w_{y_j} = 1/n$ respectively. Under this condition the formulation is equivalent to the Earth Mover's Similarity proposed in [17].

Temporal Pyramid Match. In order not to constrain the choice of distances in the feature space, pyramid matching can only be applied to the temporal domain, in a similar way as proposed for spatial [12] or spatio-temporal [4] pyramid matching. As we do not perform clustering of the feature vectors in advance, we define a threshold θ to determine whether two feature vectors match or not. The temporal pyramid match kernel is then defined as

$$\kappa_{\mathrm{TPM}}(X, Y) = \sum_{l=1}^{L} \tfrac{1}{2^{L-l+1}} \Gamma^l + \tfrac{1}{2^L} \Gamma^0, \tag{3}$$

where L is the number of pyramid levels ($L = \lceil \log_2 \max(|X|, |Y|) \rceil$) and Γ^l is the number of elements matching on level l, i.e., the number of elements falling into the same temporal bin on level l for which $\kappa_f(x_i, y_j) \geq \theta$.

Dynamic Time Alignment. The dynamic time alignment kernel (DTAK) proposed in [22] is based on the dynamic time warping (DTW) approach for sequence alignment [14]. DTW tries to align the samples of the sequences so that the temporal order is kept, but the distance (i.e., the sum of the distances of aligned elements) is globally minimized. Each sample of one sequence is aligned with one or more samples from the other sequence. Let $\psi_x(k)$ be the alignment function, with $1 \leq \psi_x(k) \leq \psi_x(k+1) \leq |X|$. In addition, a local continuity constraint γ can be defined, s.t. $\psi_x(k+1) - \psi_x(k) \leq \gamma$. Then DTAK is defined as

$$\kappa_{\mathrm{DTAK}}(X,Y) = \max_{\psi_x,\psi_y} \frac{1}{\sum_{k=1}^{N} m(k)} m(k)\kappa_f(x_{\psi_x(k)}, y_{\psi_y(k)}), \tag{4}$$

where $N = \max(|X|,|Y|)$ and $m(k)$ is a weighting coefficient. The kernel can be defined recursively and efficiently implemented using dynamic programming.

Weighted All Subsequences. The all subsequences kernel [21] is defined as $\kappa_{\mathrm{ASS}}(X,Y) = \sum_{\sigma \in \Sigma^*} \phi_\sigma(X)\phi_\sigma(Y)$, where σ denotes a sequence from the possible set of sequences Σ^*, and $\phi_\sigma(X)$ counts the number of times σ occurs as a subsequence of X. Clearly, $\phi_\sigma(X)\phi_\sigma(Y)$ is only non-zero, if σ is a subsequence of both X and Y. Thus a dynamic programming approach can be applied to determine the set of *common* subsequences. The approach is based on the observation that the kernel can be defined recursively. This kernel assumes sequences of discrete values, which does not generally hold for feature vectors. Thus we introduce a threshold θ and consider elements in the sequence as matching, iff $\kappa_f(x_i, y_j) \geq \theta$. The kernel is then defined as

$$\begin{aligned}&\kappa_{\mathrm{ASS}}(X,\emptyset) = 1,\\ &\kappa_{\mathrm{ASS}}((x_1,\ldots,x_{n-1}),Y) = \kappa_{\mathrm{ASS}}((x_1,\ldots,x_{n-2}),Y)+\\ &\textstyle\sum_{k:\kappa_f(x_{n-1},y_k)\geq\theta} \kappa_{\mathrm{ASS}}((x_1,\ldots,x_{n-2}),(y_1,\ldots,y_{k-1})),\end{aligned} \tag{5}$$

where $\emptyset$ denotes the empty sequence. The kernel value is normalized by the possible maximum number of common sequences of X and Y. In addition, we want to weight the result by the similarities of the matching elements. This can be done by summing $\kappa_f(x_i, y_j)$ for all elements for which $\kappa_f(x_i, y_j) \geq \theta$ and normalizing.

Longest Common Subsequence. Kernels based on the longest common subsequence (LCSS) algorithm have been proposed in [3,1]. The kernel described in [1] already allows plugging in any kernel for measuring the distance between the feature vectors of the samples of the two sequences, and includes the similarities in the result of the kernel. The kernel uses a recursive definition of LCSS and a threshold θ to decide if two feature vectors are considered as matching.

$$\mathrm{LCSS}(X,Y) = \begin{cases} 0, & \text{if } |X| = 0 \vee |Y| = 0,\\ \kappa_f(x_{|X|}, y_{|Y|})+ & \\ \;\mathrm{LCSS}(\mathrm{Head}(X),\mathrm{Head}(Y)), & \text{if } \kappa_f(x_{|X|}, y_{|Y|}) \geq \theta,\\ \max(\mathrm{LCSS}(\mathrm{Head}(X),Y), & \\ \;\mathrm{LCSS}(X,\mathrm{Head}(Y))) & \text{otherwise,}\end{cases} \tag{6}$$

where θ is a threshold to consider two feature vectors as matching and $\mathrm{Head}(X) = (x_1,\ldots,x_{|X|-1})$. The kernel function to determine the length of the single longest common subsequence is given as $\kappa_{\mathrm{LCSS}} = \mathrm{LCSS}(X,Y)$. Similarity weighting can be achieved by performing backtracking of the longest sequence, summing the values of $\kappa_f(\cdot)$ of the matches and normalizing.

All Longest Common Subsequences. In [1] the authors propose to consider all subsequences ending in the last element of either of the two sequences:

$$\kappa_{\text{ALCSS}}(X,Y) = \begin{aligned}&\textstyle\sum_{i=m}^{1} \text{LCSS}((x_1,\dots,x_i),Y)+\\ &\textstyle\sum_{j=n-1}^{1} \text{LCSS}(X,(y_1,\dots,y_j)).\end{aligned} \tag{7}$$

This requires backtracking of all sequences ending in the last element of either X or Y. The result of the kernel function is normalized to account for sequences of different lengths.

2.2 Kernel-Based Clustering

In this section we discuss the application of the kernels reviewed above for clustering collections of near duplicate video segments.

Kernel *k*-Means. The basic idea of kernel k-means is to apply the well-known k-means algorithm to data points mapped into a high-dimensional feature space. As with other kernel methods, the kernel trick allows performing the required calculations (distance to cluster center, update of cluster center) only by dot products of the mapped data points, thus avoiding the explicit construction of the high-dimensional feature space. In each iteration, the updated cluster index j' of a feature vector X (assuming equal weights for all feature vectors) is given as [6]

$$j'(X) = \text{argmin}_j \left(-2 \sum_{Y \in C_j} \kappa(X,Y) + \sum_{Y,Z \in C_j} \kappa(Y,Z) \right), \tag{8}$$

where C_j is the set of feature vectors in cluster j. An approach based on kernel k-means using the DTAK kernel has been proposed in [13]. Here we generalize this approach and plug in the different types of sequence kernels discussed above.

An issue with k-means is of course the question of the optimal number of clusters. As this question is independent of the use of sequence-based kernels, we do not discuss it here, but refer the reader to the literature.

Hierarchical Clustering. Hierarchical clustering is a common technique to build a cluster structure out of a similarity matrix. Here we use the kernel matrix K of the video segments of the collection as input, i.e., the elements of K are $k_{ij} = \kappa(X_i, Y_j)$. We use the clustering algorithm proposed in [2] for clustering different takes of the scene. It is based on single-linkage clustering, but has an additional constraint to first cluster takes or assign them to scenes before merging scenes. Instead of the number of clusters, this algorithm has a minimum distance parameter which determines when to stop clustering. As several of the kernels use a similarity threshold, we use this threshold θ as the cutoff distance for clustering. This means, that feature vectors can be clustered, if they contain at least one element for which $\kappa_f(\cdot) > \theta$.

2.3 Kernel-Based Visualization

Principal component analysis (PCA) is a well-known method to apply an orthogonal linear transform which projects data into a coordinate system spanned by the principal components. The dimensions of the coordinate system are ordered by decreasing variance of the data. Thus a small number of principal components often approximates the data quite well. Projection of data using PCA to a plane is commonly used for visualization purposes.

In [20] the kernel PCA is introduced, which applies the idea of the PCA to data transformed to a high-dimensional space, using the kernel trick to avoid explicit construction of this space. Instead, the projection to the space spanned by the k first principal components can be determined as

$$P_k(X) = \left(\sum_{i=1}^{l} \alpha_i^j \kappa(X, Y_i) \right)_{j=1}^{k} , \tag{9}$$

where l is the size of the kernel matrix (i.e., in our case, the number of video segments in the collection) and $\alpha^j = (1/\lambda_j) v_j$ is defined from the eigenvectors v_j and eigenvalues λ_j of the kernel matrix. The kernel matrix K contains the mutual kernel values between the video segments of the collection as input, i.e. the elements of K are $k_{ij} = \kappa(X_i, Y_j)$. As the PCA is defined on centered data, a similar step is required for the kernel matrix: $\tilde{K} = K - \frac{1}{l}\mathbf{1}K - \frac{1}{l}K\mathbf{1} + \frac{1}{l^2}\mathbf{1}K\mathbf{1}$, where $\mathbf{1}$ is a matrix of size $l \times l$ with all elements 1.

We aim at using this approach for projecting a collection of video sequences to a low-dimensional space (as an alternative to applying multidimensional scaling to a similarity matrix between the segments). Using the sequence kernels discussed above, the kernel PCA is expected to yield a projection of the data, in which near duplicates are close in the projected space. Such a representation is useful for video browsing and interactive search in video collections containing near duplicate segments.

3 Results

In the following we present results of experiments for clustering repeated takes and visualizing collections of unedited video material.

3.1 Clustering Repeated Takes

The proposed clustering algorithms using the different kernels have been evaluated on a subset of the TRECVID 2007 BBC rushes test data set (the same subset as used e.g. in [2,8]). The subset consists of six randomly selected videos out of this data set (in total 3 hours, for more details see [2]), using the ground truth provided by NHK [15]. In order to avoid side effects from different shot segmentations, the results are based on the ground truth shots. Every 10th frame

Table 1. Mean and median of the frame based F1 measure of clustering results with hierarchical clustering and kernel k-means (with different choices of k) for the six sequence kernels

clustering	**hierarchical**		**kernel k-means**		**kernel k-means**		**kernel k-means**	
methods			k ground truth		k as hierarch.		best $k \in [3;15]$	
	mean	**median**	**mean**	**median**	**mean**	**median**	**mean**	**median**
ALCS	0.50696	0.48305	0.40378	0.41629	0.39730	0.39898	0.49800	0.51472
LCS	0.58775	0.57874	0.45652	0.38854	0.38106	0.37901	0.62065	0.65442
ASS	0.43178	0.41910	0.42903	0.40174	0.37684	0.36764	0.49262	0.47499
EMD	0.49595	0.51785	0.43934	0.44547	0.43277	0.43945	0.51365	0.50151
TPM	0.55394	0.54737	0.40873	0.37693	0.38878	0.37401	0.48079	0.49547
DTAK	0.60153	0.58055	0.48300	0.44685	0.48173	0.47998	0.59541	0.60995

of the videos is used in the feature sequence, and from each frame we extract a feature vector consisting of the MPEG-7 ColorLayout (cl) descriptor (DC and the first two AC coefficients of each channel), the MPEG-7 EdgeHistogram (eh) descriptor and a scalar visual activity (va) value. The kernel function between individual feature vectors is defined as $\kappa_f((dc_X, eh_X, va_X)^T, (dc_Y, eh_Y, va_Y)^T) = \kappa_{MPEG-7}((dc_X, eh_X)^T, (dc_Y, eh_Y)^T)\ \kappa_{RBF}(va_X, va_Y)$, where κ_{MPEG-7} is the MPEG-7 kernel proposed in [7] (with equal weighting of both MPEG-7 features). For the sequence kernels that need a similarity threshold, we set $\theta = 0.03$.

In Table 1 we report the mean and median F1 measure for take clustering. The F1 values are calculated from the frame precision/recall measure proposed for evaluating clustering of repeated takes in [8]. In general, the results are comparable to those of other clustering approaches. An interesting result is that precision and recall are more balanced than in clustering results reported in the literature for clustering with other distance functions. In contrast to the results reported for string matching and DTW based distance functions [2], the LCS or ALCS kernels do not outperform the DTAK kernel. The reason seems to lie in the properties of the kernel $\kappa_f(\cdot)$ between individual feature vectors, which is in the form of $\exp(-\text{distance}(\cdot))$, and thus better distinguishes well matching subsequences than a linear distance measure.

From the string matching kernels, LCS performs better than those considering more than one subsequence (ALCS, ASS). Also EMD and TPM, which do not enforce an ordered sequence, perform similarly well. For TPM, it seems that the temporal tolerance introduced by the pyramid match is sufficient to cover the timing differences and insertions between different takes. The optimal sequence found by EMD often contains many samples in the correct temporal order.

From the clustering methods, hierarchical clustering seems to be the better choice. It yields better results than kernel k-means with the same number of clusters or the number of clusters from the ground truth. We conclude that the reason for this is that the hierarchical clustering algorithm used includes a specific constraint for the take clustering problem. The best kernel k-means results slightly outperform hierarchical clustering result in terms of median, but not in terms of mean.

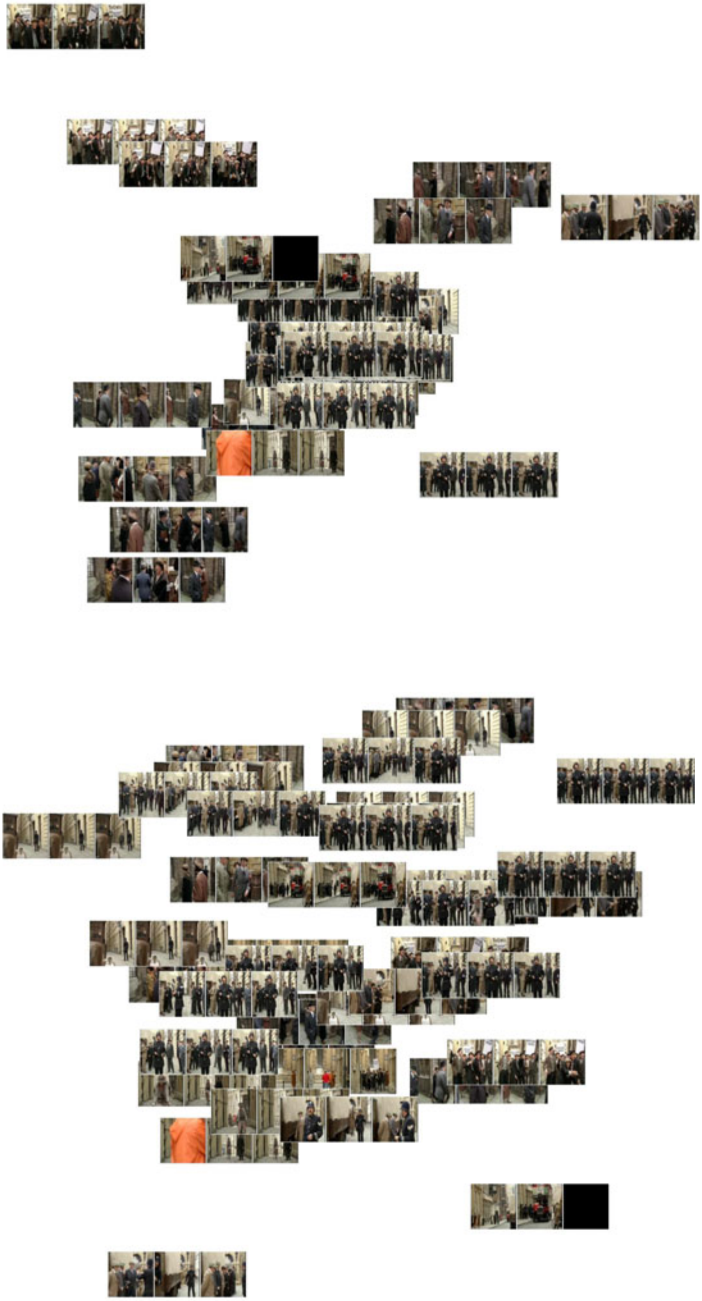


Fig. 1. Projection of segments from video MRS025913 using kernel PCA (2 principal components) with ALCS kernel (top) and DTAK kernel (bottom)

3.2 Kernel PCA for Visualization

We perform visualization experiments on two data sets: on the videos from the TRECVID 2007 BBC rushes set used for the clustering experiments and on a set of multi-view test material from the 2020 3D Media project[1] (3 views, about 6 minutes). Kernel PCA has been applied and the data has been projected to the plane spanned by the first two principal components. Each video segment is visualized by its first, center and last key frame.

Figure 1 shows the results for one video from the BBC rushes set, using the ALCS and the DTAK kernel. In both visualizations, the proximity is related to the similarity of the sequences. It is difficult to find objective criteria for assessing the quality of the visualizations, especially as the difference in clustering performance between the two kernels is quite small. However, the data seems to be organized more clearly in the visualization produced using the ALCS kernel.

Figure 2 shows the visualization of the multi-view content set using the LCS kernel. The different views are rather spread along the horizontal axis, while the different takes are on the vertical axis (note e.g. the shot with the calibration board being close to the bottom in both cases). Only takes from the one shot with structured light experiments (the dark frames in the middle) are outliers and not well fit into the projection space.

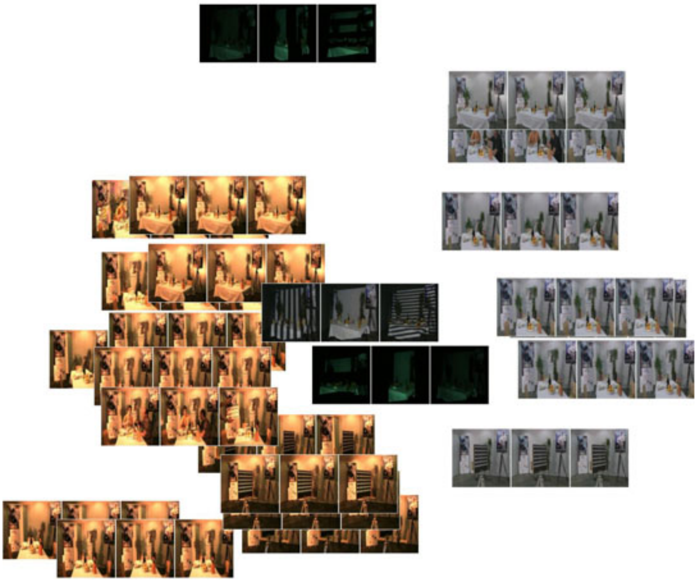


Fig. 2. Projection of segments from the 2020 3D Media multi-view set using kernel PCA (2 principal components) with LCS kernel

[1] http://www.20203dmedia.eu/

4 Conclusion

In this paper we have analyzed the application of six sequence-based kernels for clustering and visualizing collections of near duplicate video segments.

In contrast to previous work using kernel k-means clustering in summarization, we have compared the performance of different kernels and have also used hierarchical clustering on the kernel matrix. No strong differences in the performance of the different kernels have been observed. However, we see that kernels that determine a single best matching sequence perform slightly better than those that weight the results from several matching sequences. Differences in clustering results observed between string matching and dynamic time warping distances are not evident between kernels based on these paradigms. Our results show that hierarchical clustering outperforms kernel k-means in most cases.

We have also shown that meaningful visualizations for interactive browsing and presentation of summaries can be generated using kernel PCA to project the data into a plane. Once the kernel matrix has been calculated, both clustering and visualization can be performed very efficiently. Despite the similar performance of the kernels in clustering, the string matching based kernels (e.g., LCS, ALCS) produce visualizations with a more comprehensible organization of the data. The quality of the obtained visualization needs to be further evaluated in a user study.

Acknowledgments. The author would like to thank Felix Lee for the key frame plotting script, and Hannes Fassold for his support in implementing the kernel PCA. The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 215475, "2020 3D Media – Spatial Sound and Vision".

References

1. Bailer, W.: A Feature Sequence Kernel for Video Concept Classification. In: Lee, K.-T., Tsai, W.-H., Liao, H.-Y.M., Chen, T., Hsieh, J.-W., Tseng, C.-C. (eds.) MMM 2011 Part I. LNCS, vol. 6523, pp. 359–369. Springer, Heidelberg (2011)
2. Bailer, W., Lee, F., Thallinger, G.: A distance measure for repeated takes of one scene. The Visual Computer 25(1), 53–68 (2009)
3. Ballan, L., Bertini, M., Del Bimbo, A., Serra, G.: Video event classification using string kernels. Multimedia Tools Appl. 48(1), 69–87 (2010)
4. Choi, J., Jeon, W.J., Lee, S.-C.: Spatio-temporal pyramid matching for sports videos. In: Proc. 1st ACM International Conference on Multimedia Information Retrieval, pp. 291–297. ACM, New York (2008)
5. Cuturi, M., Vert, J.-P., Birkenes, O., Matsui, T.: A kernel for time series based on global alignments. Computing Research Repository, abs/cs/0610033 (2006)
6. Dhillon, I.S., Guan, Y., Kulis, B.: Kernel k-means: spectral clustering and normalized cuts. In: KDD, pp. 551–556 (2004)
7. Djordjevic, D., Izquierdo, E.: Relevance feedback for image retrieval in structured multi-feature spaces. In: Proc. MobiCom (2006)

8. Dumont, E., Mérialdo, B.: Rushes video parsing using video sequence alignment. In: Proc. CBMI 2009 (June 2009)
9. Grauman, K., Darrell, T.: The pyramid match kernel: Discriminative classification with sets of image features. In: IEEE ICCV, vol. 2 (2005)
10. Grauman, K., Darrell, T.: Approximate correspondences in high dimensions. In: NIPS, pp. 505–512 (2006)
11. Grauman, K., Darrell, T.: The pyramid match kernel: Efficient learning with sets of features. J. Mach. Learn. Res. 8, 725–760 (2007)
12. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR (2006)
13. Liu, Y., Zhou, F., Liu, W., De La Torre, F., Liu, Y.: Unsupervised summarization of rushes videos. In: Proc. ACM Multimedia, pp. 751–754 (2010)
14. Myers, C.S., Rabiner, L.R.: A comparative study of several dynamic time-warping algorithms for connected word recognition. The Bell System Technical Journal 60(7), 1389–1409 (1981)
15. NHK Science & Technical Research Laboratories. Test modules for TRECVID activity. Use case scenario. Ver.1.2.0E (April 2008)
16. Over, P., Smeaton, A.F., Awad, G.: The TRECVID 2008 BBC rushes summarization evaluation. In: Proceedings of the 2nd ACM TRECVid Video Summarization Workshop, TVS 2008, pp. 1–20. ACM, New York (2008)
17. Rahimi, A., Kiran, R.: How earth mover's distance comprares two bags. Technical report, Intel Labs Berkeley (2007)
18. Ricci, E., Tobia, F., Zen, G.: Learning pedestrian trajectories with kernels. In: ICPR, pp. 149–152 (2010)
19. Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover's distance as a metric for image retrieval. Int. J. of Computer Vision 40(2), 99–121 (2000)
20. Schölkopf, B., Smola, A., Müller, K.-R.: Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation 10(5) (1998)
21. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge Univ. Press (2004)
22. Shimodaira, H., Noma, K.-I., Nakai, M., Sagayama, S.: Dynamic time-alignment kernel in support vector machine. In: NIPS (2001)
23. Smeaton, A.F., Over, P., Kraaij, W.: Evaluation campaigns and TRECVid. In: Proc. 8th ACM International Workshop on Multimedia Information Retrieval, pp. 321–330 (2006)
24. Xu, D., Chang, S.-F.: Visual event recognition in news video using kernel methods with multi-level temporal alignment. In: IEEE CVPR (2007)
25. Xu, D., Chang, S.-F.: Video event recognition using kernel methods with multilevel temporal alignment. IEEE Trans. Pattern Anal. Mach. Intell. 30 (2008)
26. Yeh, M.-C., Cheng, K.-T.: A string matching approach for visual retrieval and classification. In: Proc. 1st ACM International Conference on Multimedia Information Retrieval, pp. 52–58. ACM, New York (2008)

Investigating Gesture and Pressure Interaction with a 3D Display

Martin Halvey, David Hannah, Graham Wilson, and Stephen A. Brewster

School of Computing Science, University of Glasgow, Glasgow, G12 8QQ, UK
{first.last}@glasgow.ac.uk, gawilson@dcs.gla.ac.uk

Abstract. We examine the use of a mobile device to provide multifunctional input and output for a stereoscopic 3D television (TV) display. Through a number of example applications, we demonstrate how a combination of gestural and haptic input (touch and pressure) can be successfully deployed to allow the user to navigate a complex information space (multimedia and TV content), while at the same time visual feedback can be used to provide additional information to the user enriching the experience. In order to investigate the usefulness of our example applications a user evaluation was conducted, where our prototypes were compared with more traditional devices for multimedia interaction. The results of the user evaluations highlight the benefits of our approach and also provide some design guidelines.

Keywords: Mobile devices, 3D, TV, pressure, gestures.

1 Introduction

Multimedia content is now an important part of everyday life and the TV is fast becoming a hub for interacting with much of this content, including increasingly complex media such as 3D video, video/photo collections, social media through the internet etc. It is important that we provide users with the best options for browsing, searching and consuming this growing and complex content. In many cases, the TV remote control is itself a limiting factor, as normally it only provides simple fixed buttons for interaction, indeed many households also have several media devices each with a different remote. As such mobile devices may provide a suitable alternative to the traditional remote, especially as a high number of phone users already interact with their phone while watching TV[1]. Initial research into using a mobile phone as a universal interaction device [1] highlighted that perhaps universal control over all appliances might not be ideal but that control over particular appliances might be beneficial. Therefore the aim of this work is to focus on the specific use of mobile phones to provide richer interactions with media on a TV screen. Currently, there are applications that run on mobile phones which provide access to content, but many of these applications simply replicate the 'look and feel' of a regular TV remote. As such, these applications do not take full advantage of all of the features that a mobile phone can offer which a normal TV remote cannot offer. These include, but and are

[1] Nielsen/Yahoo: "Mobile Shopping Framework", November 2010.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 395–405, 2012.

not limited to: (1) a local display for additional information, be it visual, audio or haptic, (2) potential for other types of input e.g. gestures on or with the device, audio input, etc. and (3) multi-user interaction, as many users in the same space would have devices that could be used to control the TV or to share information, etc.

In this paper we investigate the first two points. We examine the use of gestures and pressure input as novel input modes to control the display, the ability of users to use the visual display on the phone while interacting in this fashion and the use of different locations on the phone as locations for input. In this work we use a 3D display as it represents the current state of the art in televisions and, although it represents a more complex information visualisation to navigate, the use of 3D spatial layouts may better illustrate content structure/relationships.

2 Related Work

In this section we briefly describe research in a number of areas related to our research. Gestures are increasingly being used for interaction with mobile devices. These gestures can be loosely classified into 'discrete action' gestures and 'continuous control' gestures. The more traditional style of gesturing, discrete action control, involves the user performing an action that, once completed, the system attempts to recognise. These gestures are often used to replace one or multiple physical button clicks. Successful examples of these techniques in commercial devices often involve short movements that are fast and easy to perform such as a single stroke of a touch screen to change the view, or double tapping a phone to silence it [2]. Recently, continuous control gestures have become more prevalent. With these forms of gesture the interactions between the user and the device are closely coupled. The user provides a continuously changing stream of input, and the device adjusts the feedback constantly to respond to the user's input. Both types of gestures can be performed with a device as well as on a touchscreen e.g. by shaking or tilting the device [3]. Pressure based interaction is a relatively new area of research. Some initial work by Ramos *et al.* [4] used a pressure-sensitive stylus as a means of controlling interaction widgets. They concluded that an interaction that requires both positioning/movement and the application of pressure (specifically through the same device) should separate the two actions as much as possible so as not to interfere with either. Wilson *et al.* [5] demonstrated that eyes free pressure interaction while squeezing a mobile device was feasible and almost as accurate as when using visual feedback. Brewster and Hughes [6] asked users to input text on a pressure-sensitive screen, using light touches for lower-case letters and harder presses for uppercase letters. They found that the pressure-augmented keyboard resulted in faster but more error-prone text entry. Clarkson *et al.* [7] suggest further uses for pressure-augmented keypads such as preview zooming, 3D navigation or "affective input" where emotional state is derived from the degree of force used in an interaction.

3 Hardware

We simulated a 3D TV using a PC with an Nvidia 3D Vision graphics card and active shutter glasses. Users interact with the 3D display using a Nexus One mobile phone connected to the PC via Bluetooth, providing a tangible remote controller for the 3D display. Users can use its touchscreen as proxy for interacting with the data on the TV. The touchscreen can also act as an additional display. Previous research [8] has identified a number of potential benefits to having secondary screen available via a phone interface for multimedia interaction, namely additional control, methods to enrich content, additional ways to share content, and finally the ability to transfer television content. Users can also use the Nexus device to perform device movement gestures, as it includes an accelerometer. Dachselt and Buchholz [9] have previously studied throw and tilt interactions with remote displays, they investigated continuous and discrete tilt gestures in two media interaction environments including a 3D map for Google Earth. However, these interactions did not include the TV or multiple types of media. In addition to the standard input and output modalities currently available on a mobile device, we added pressure input, which can add a z-dimension to typically 2D, x-y GUIs. Pressure input has been shown to be effective at improving interaction [5] and could be included in future mobile devices. Pressure input was provided by using two standard Force Sensing Resistors (FSR's) and a linearising amplifier [5]. This allowed users to push at two different points on the back of the device. In this way we have taken advantage of the additional space on the back of the device for controlling the television which allows interaction without obscuring the touchscreen, as outlined by Baudisch and Chu [10]. More detailed descriptions of the hardware used can be found in Hannah *et al.* [11].

4 Interfaces for Media Interaction

4.1 Browsing Image Collections

Fig. 1 (A) shows an example of a visualisation which allows the user to view a collection of images in a similar way to the iTunes Cover Flow, but with the content stretching back into the screen in 3D. Users can navigate through the image stream by tilting the device to move backwards and forwards through the image queue. As the user rotates the phone accelerometer data is filtered and commands sent to the TV to move forwards or backwards. One drawback of the visualisation outlined in Fig. 1(A) is that while it takes advantage of the 3D space it can be difficult to view images further back in the queue. Pressure input could be used to overcome this problem, as Fig. 1(B) shows a version of one of Ramos *et al.*'s [4] pressure widgets in use. Here, the stream of photos on the TV dynamically kinks as pressure is applied on the FSRs; images further down the queue can then be more easily seen. This exploits both the easy interaction with the mobile device and the additional visual space allowed by the 3D display to show more information. Again it is possible for users to see different views of the information on the local display on their mobile device. Fig. 1(C) shows an alternative solution to the occlusion problem outlined above. The same collection of images is arranged in spiral visualization with the current image in front and the tail of the spiral going into the screen in 3D. As the user rotates the phone accelerometer data is used to rotate the spiral forwards or backwards, with different photos being brought into focus at the top of the spiral. Rate of rotation is proportional to the angle of tilt of the phone. The use of the mobile phone also means that the user can see a replication of the TV display on the phone screen in 2D; such as a replication of the spiral/Cover Flow view or, alternatively (as shown in Fig. 1(D)) users can see the image at the top of the data queue on the mobile device; this allows the user to see a local version of the image as well as some metadata. This interaction example demonstrates some of the flexibility of using a phone as a remote controller; it can duplicate the TV screen, or individual users can have individual displays of information, allowing multiple views of the same data through multiple devices, realising some of the possibilities outlined by Cesar *et al.* [8].

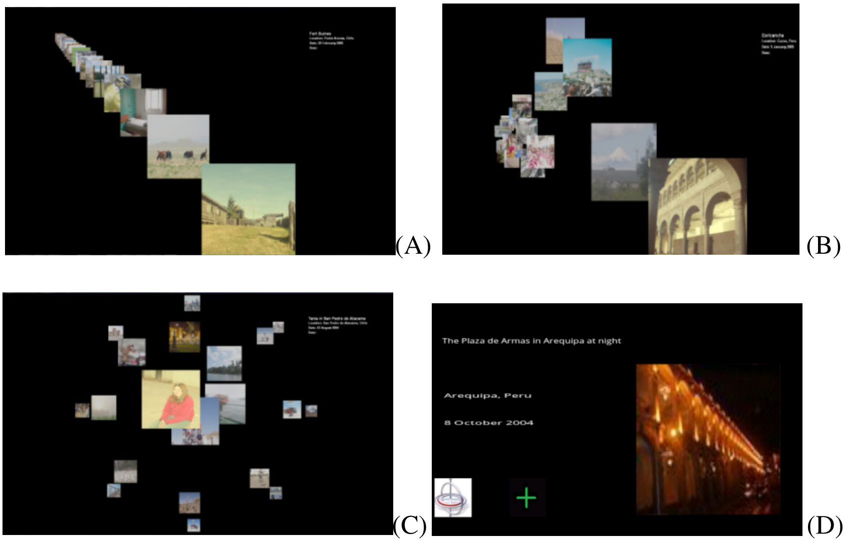


Fig. 1. 3D visualization of a collection of images (A), pressure input on the mobile device causes images on the 3D display to kink out to allow images at the back to be viewed more clearly (B), 3D visualization of a collection of images in a spiral (C) and visualisation of image at top of the queue as displayed on phone (D)

4.2 EPG Browsing

The input methods described above can also be used to navigate through more complex structures. Richer interactions can also be used to control an Electronic Programme Guide (EPG). EPGs contain a lot of dense information and, when many TV channels are available, can often be difficult and slow to navigate using a regular TV remote. There are already applications to allow users to browse EPGs on additional devices, e.g. tablets or phone; however these are not always connected to the TV. Fig. 2(A) shows a possible 3D representation of an EPG. A fisheye view in 3D gives the user an idea of programmes close to the time being viewed; users can see and compare content easily and efficiently as the visualization makes full use of the TV screen. The program information is modelled as a spiral viewed from the side, where a 24 hour period is displayed on a full 360° rotation of the spiral, with channels on the y axis. The spiral view means that there are no discontinuities between the InvINdays; the next day joins smoothly to the previous one. In many EPGs different days are presented on different screens, making the transition from one day to the next more awkward. The viewer can preview what is on earlier or later by rotating the phone side-to-side to rotate the spiral. Tilting the phone forward or back moves up or down through the channels. By squeezing on the FSRs on the back of the phone the user can control the scaling of the view, for example zooming in to a particular time/channel area. Pressing harder causes the view to “pop” through to the following day at the same time, allowing rapid skipping through different days. A particular programme can be selected by tapping on the phone screen, either through a button on the remote (Fig 2(C)) or the EPG (Fig. 2(B)). This brings up a page about the show on the phone and allows the it to be played on the 3D TV. In order to provide a comparison with more traditional EPG browsing a flat EPG (Fig. 2(B)) and a basic remote control were also implemented on the mobile device (Fig. 2(C)). Both the 3D and 2D displays could be controlled via gestures with the mobile device or by buttons on the remote control on the mobile device.

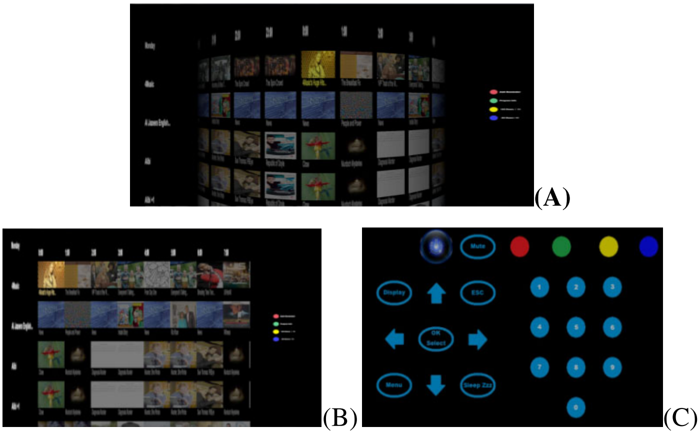


Fig. 2. An EPG displayed as a fisheye on a cylinder in 3D (A), a flat 2D EPG with the same information (B) and an example TV remote control app on a mobile phone (C)

5 User Evaluations

5.1 System Configuration

Image Browsing

For the user evaluation six different system configurations were used, based on the TV visualisation (linear or spiral), phone screen content (replication of TV content or local content) and the use of pressure (kinking or none).

- S1: Linear visualisation on 3D display (Fig. 1A), entire queue visualised on phone screen in 2D, no pressure input

- S2: Linear visualisation on 3D display (Fig. 1A), image at top of queue displayed on phone screen (Fig. 1D), no pressure input
- S3: Linear visualisation 3D display with pressure input for kinking (Fig. 1B), entire queue visualised on phone screen in 2D
- S4: Linear visualisation 3D display with pressure input for kinking (Fig. 1B), image at top of queue displayed on phone screen (Fig. 1D)
- S5: Spiral visualisation on 3D display (Fig. 1C), entire queue visualised on phone screen in 2D, no pressure input
- S6: Spiral visualisation on 3D display (Fig. 1C), image at top of queue displayed on phone screen (Fig. 1D), no pressure input.

EPG Browsing

For the EPG evaluation four different configurations were used, based on visualisation (3D or flat/2D) and input type (gestures or touchscreen remote):

- S1: 3D Spiral with gestures (Fig. 2A and gestures)
- S2: 3D Spiral with remote (Fig. 2A and Fig. 2C)
- S3: Flat with gestures (Fig. 2B with gestures)
- S4: Flat with remote (Fig. 2B and Fig. 2C)

5.2 Collection and Tasks

Image Browsing

For the experiments reported in this paper the CLEF 2007 image collection was chosen [12]. CLEF 2007 is a set of 20,000 images, 60 search topics, and associated relevance judgments. The topics were categorised into a number of different categories, including: easy/hard, semantic/visual, and geographic/general. In order to choose tasks which were as similar as possible, 6 tasks from the medium and visual categories were chosen. For each of the tasks, 25 relevant images were chosen as well as 100 irrelevant images from the remainder of the collection. For each topic the users were presented with these 125 images in a random order and were asked to navigate through the stream and mark images as being relevant.

EPG Browsing

For the EPG browsing tasks, one week of an online TV guide was downloaded. The guide contained 255 channels. The content was normalised so that each TV program corresponded to one hour in the TV guide. Twelve navigation tasks were created. For 8 of these the participant had to navigate to a particular channel on a particular day at a particular time and set a reminder for the content (by pressing a button on the mobile device), this marked the end of the task. 4 of the 8 tasks required the participants to navigate through the collection in more of a horizontal than a vertical (time based) direction. The other 4 of the 8 tasks required the participants to navigate through the collection in more of a vertical than a horizontal (channel based) direction. All of the tasks were designed so that the participants had to navigate through an equidistant amount of the EPG. The final four tasks were free browsing tasks where the participants were given a broad time period e.g. Tuesday morning, and they were asked to set a reminder for any program of their choice in that time period. For each system configuration the participants were given 1 horizontal navigation task, 1 vertical navigation task and 1free browsing task.

5.3 Experimental Design

The experiment lasted around 90 minutes and the participants received compensation of £10. Following a training period the participants carried out six image browsing tasks using all 6 configurations (giving 36 tasks), in a within subject design. The order of system configuration was rotated and the topic order was randomised. There was a maximum time limit for each task which was 5 minutes; participants could finish early if they wished. The participants then carried out the 12 EPG browsing tasks. Again system configuration was rotated and search tasks were randomised. Participant interactions with the display and mobile device were logged. They also filled out a number of questionnaires at different stages of the experiment. Throughout the evaluation the participants were encouraged to comment on the system and interactions, and notes were taken of any comments made. We were able to calculate precision and recall values for all of the image browsing tasks. We also counted the number of gestures required to complete each task for each system.

6 Results

17 participants took part in the evaluation. They were mostly staff and students of the University. The participants consisted of 11 males and 6 females, with an average age of 29 years old. In the entry questionnaire participants indicated that they regularly interacted with images and also regularly watched television.

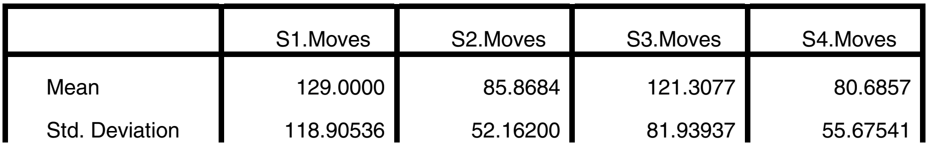
6.1 Task Performance: Image Browsing

Participants successfully completed the tasks, with the average precision being quite high (Max=98.33% (S2), Min= 95.77% (S6), Avg= 96.9%). A one factor ANOVA showed no significant effect of system on precision (p=0.231) and pairwise comparisons showed no differences. The recall of the systems was slightly lower than the precision, but still high (Max= 91.17% (S5), Min= 77.05% (S2), Avg= 84.85%). A one factor ANOVA showed no significant effect of system on recall (p=0.175) and pairwise comparisons showed no differences. In terms of the average time to complete each task, the systems where users had the images on the mobile phone instead of the replicated stream took slightly more time to complete the task (Max= 255.84sec (S6), Min= 201.53 sec (S2), Avg= 233.74 sec). Pairwise comparisons shows significant differences between S6 and both S3 (p=0.03) and S4 (p=0.004). The number of gesture rotations to find items was approximately the same across all systems (Max= 171.11 (S5), Min= 142 (S2), Avg= 153.42), with no statistically significant results.

6.2 Task Performance: EPG Browsing

Due to a technical error, the interactions for the first 4 participants were not logged correctly so only the logs for the remaining 13 were used for analysis (for user preference in the next section the responses from all 17 were used). An analysis of user performance in terms of setting a reminder showed that there was little difference between systems. Out of two direct tasks (i.e. vertical reminder task and horizontal reminder task) the average number completed successfully was 1.38 (std. dev=0.63), 1.84(std. dev=0.8), 1.31(std. dev=0.38) and 1.15(std. dev=0.76) for S1, S2, S3 and S4 respectively. Surprisingly S4 the system that is most close to current EPG browsing was the worst performing. None of the differences were statistically significant. An analysis of the effort involved in terms of average time to set a reminder and number of gestures (either rotations or button presses where appropriate) revealed some differences between the systems. With respect to time to set a reminder, the average times in seconds were 81.33 (std. dev=34.43), 80.97(std. dev=56.59), 76.68(std. dev=41.84), 83.95(std. dev=54.39), and for S1, S2, S3 and S4.

Table 1. Rotation and button moves per system



	S1.Moves	S2.Moves	S3.Moves	S4.Moves
Mean	129.0000	85.8684	121.3077	80.6857
Std. Deviation	118.90536	52.16200	81.93937	55.67541

In terms of actions, it required more gestures than button presses to set reminders. The results of pair wise comparisons between the systems showed that the differences between S1 (3D with gestures) and S2 (3D with remote) was statistically significant (p=0.033). The difference in terms of time and interactions between the systems using buttons and gestures can be explained by the feedback given by users during the experiment. Many users commented that the gestures were good for moving large distances through the EPG easily, but more difficult for fine control i.e. moving one or two time slots or programs. In contrast they commented that the buttons were easy for fine control but not so good to use for moving large distances in the EPG.

6.3 User Preferences

Gestures

In post-search task questionnaires we solicited subjects' opinions on the use of gestural interaction to navigate both the images collections and the EPG. The following Likert 5-point scales and semantic differentials were used; some of the scales were inverted to reduce bias. The scales used were: "How easy was it to use the system" (Use), "How easy was it to learn to use the system" (Learn to use), "When interacting with the system I felt in control/not in control, comfortable/uncomfortable, confident/ unconfident". The following Semantic differentials were used: The videos I have received through the searches were: "wonderful/terrible", "satisfying/frustrating", "stimulating/dull", "easy/difficult", "flexible/rigid", "efficient/inefficient", "novel/standard", "effective/ineffective. Many of these scales and semantic differentials are used for different aspects of the system. The participants generally gave positive responses with respect to the use of gestures for all differentials and questions. It was noticeable that the participants were generally more positive for image browsing than EPG browsing. This is not totally surprising, as for the image browsing the users only had one degree of freedom for navigation, whereas the EPG browsing had two degrees of freedom, meaning that it was a more complex form of navigation. None of the differences between replies for gestures vs. buttons were statistically significant. To gain more insight into user preference for buttons or gestures, the participants were asked to judge directly between the two approaches for EPG browsing. The participants were asked, "Which of the systems did you…": "find best overall" (Best), "find easier to learn to use" (Learn), "find easier to use" (Easier), "prefer" (Prefer), and "find more effective for the tasks you performed" (Effective). These questions were also used for other direct comparisons where we compare different aspects of the systems. The users had a preference for buttons over gestures for browsing the EPG. In particular, the participants found buttons easier to both use and learn to use. When asked about their preference in more detail, many participants stated that they were familiar with using buttons, at the same time many of these participants complained that using buttons was boring.

Use of Screens

The participants had a slight preference for the 2D visualisations. When asked about this, the users stated that they found the 2D visualisation more familiar than the 3D, but at the same time they complained that they found 2D boring and the 3D 'exciting' and 'cool'. For the image browsing task two different visualisations were used on the phone screen, one showing a 2D visualisation of the 3D visualisation on the TV and the other showing the image at the top of the queue on the phone screen (see Fig. 1D). In terms of user preference the users had a preference for having the image at the top of the current queue of images on the phone instead of the visualisation of the data stream. This result is encouraging as it could have been the case that users found the different views of the same information confusing, instead many users found it beneficial. Some commented that it was useful to use the phone to validate the image at the top of the visualisation on the screen, indeed many users noted that when the entire visualisation was on the phone that many of the images were too small. It should be noted that some users stated that they did not use the phone screen at all, just looking at the TV screen; many of these users are responsible for the no difference responses. The participants were also asked about the three different visualisations used in the 3D interface. Most users had a preference for the linear or the spiral layouts. For the spiral layout, participants stated that they liked to see more images but that it was disorienting at times. For the linear layout participants stated that they liked its simplicity but that they were able to see fewer images than with the other visualisations. With respect to the use of pressure to kink the linear layout, many users found it disorienting and said they would have liked to have been able to switch focus to other parts of the visualisation than the front, this is similar feedback to that for the spiral.

Interaction with Back of Device

As the back of the device was used in both the image and EPG browsing it was possible to compare using the back of the device for both types of task. It was found that, in general, the participants were more positive about using the back of the device for EPG browsing. This is an encouraging result, as for the EPG there were two sensors on the back of the device instead of one for the image browsing. It appears that as they become more familiar with the application that the users become more positive despite the increase in complexity. Perhaps with more training users will become more familiar with this type of interaction. It should also be noted that for some users the ease with which they could interact with the back of the device was affected by the way in which they gripped or held the mobile device. Some users adjusted their grip as the tasks proceeded, this may have affected the more positive perception for the EPG browsing.

Use of Pressure

In contrast with the relative increase in positivity in feedback of users towards using the back of the device, users are generally slightly less positive about using pressure when browsing the EPG than the image collection. Again this could be because the interaction is more complex, also coupled with this, for the image browsing it was not necessary to use pressure input whereas to complete the EPG browsing tasks quickly it was essential. It should be noted that while the positivity decreases, users were still generally positive or neutral about the use of pressure. In a direct comparison between the uses of pressure input versus the use of buttons for skipping forward and back 24 hours in the EPG the feedback was split. Again when asked, users stated that they found buttons more familiar and this was the reason that some users had a preference for using buttons.

7 Conclusions and Discussion

We have presented some novel prototypes for navigating media on a 3D TV-like display using a mobile device. The use of a mobile device for browsing content on a television has a number of benefits, including additional displays, gestural input and the possibility of additional input and output capabilities. Our user evaluation demonstrated that users were able to navigate through media using gestural and pressure input easily and successfully. There was very little difference in terms of performance between our prototypes and more traditional interaction methodologies, although the benefits outlined above are not available through traditional interaction methodologies. There are a number of design guidelines that can be made based on our findings:

- Users can adapt quite quickly to using positions on the back of the device to interact. However, care must be taken with the positioning so that users can grip and use the remainder of the device easily.
- Despite concerns that users would not be able to use the screen when tilting the device to gesture, users were able to use the screen if they wished. Thus, the screen on a mobile device can be used as an additional display. However, more research is needed on the impact of divided attention between two displays for

media browsing, as navigation time was impacted slightly by using the screen to display additional information.

- In general, users found gestures useful for navigating large distances quickly; however, they found fine control more difficult resulting in over- and undershooting selections. Considerations for fine control should be taken into account.

As can be seen, a number of interesting problems and future research questions have been highlighted. The work presented in this paper is an important step to allowing more dynamic, social and natural control and navigation of multimedia which could possibly be deployed in home and public settings.

Acknowledgments. This research has been funded by the Industrial Members of MobileVCE (www.mobilevce.com), with additional financial support from EPSRC grant EP/G063427/1.

References

1. Roduner, C., Langheinrich, M., Floerkemeier, C., Schwarzentrub, B.: Operating Appliances with Mobile Phones - Strengths and Limits of a Universal Interaction Device. In: Proceedings of the International Conference on Pervasive Computing (2007)
2. Ronkainen, S., Hakkila, J., Kaleva, S., Colley, A., Linjama, J.: Tap input as an embedded interaction method for mobile devices. In: Proceedings of the 1st International Conference on Tangible and Embedded Interaction, TEI 2007 (2007)
3. Eslambolchilar, P., Williamson, J., Murray-Smith, R.: Multimodal feedback for tilt controlled speed dependent automatic zooming. In: Proceedings of ACM UIST (2004)
4. Ramos, G., Boulos, M., Balakrishnan, R.: Pressure Widgets. In: Proceedings of the International Conference on Human Factors in Computing Systems, CHI 2004 (2004)
5. Wilson, G., Stewart, C., Brewster, S.: Pressure-based menu selection for mobile devices. In: Proceedings of MobileHCI (2010)
6. Brewster, S., Hughes, M.: Pressure-Based Text Entry for Mobile Devices. In: Proceedings of MobileHCI (2009)
7. Clarkson, E.C., Patel, S.N., Jeffrey, S.P., Abowd, G.D.: Exploring Continuous Pressure Input for Mobile Phones. In: Proceedings of ACM UIST (2005)
8. Cesar, P., Bulterman, D.C.A., Jansen, A.J.: Usages of the Secondary Screen in an Interactive Television Environment: Control, Enrich, Share, and Transfer Television Content. In: Tscheligi, M., Obrist, M., Lugmayr, A. (eds.) EuroITV 2008. LNCS, vol. 5066, pp. 168–177. Springer, Heidelberg (2008)
9. Dachselt, R., Buchholz, R.: Natural Throw and Tilt Interaction between Mobile Phones and Distant Displays. In: Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems, CHI 2009 (2009)
10. Baudisch, P., Chu, G.: Back-of-device interaction allows creating very small touch devices. In: Proceedings of the 27th International Conference on Human Factors in Computing Systems, CHI 2009 (2009)
11. Hannah, D., Halvey, M., Wilson, G., Brewster, S.: Using Multimodal Interactions for 3D TV and Multimedia Browsing. In: Proceedings of EuroITV (2011)
12. Grubinger, M., Clough, P.: On the creation of query topics for ImageCLEFPhoto. In: Proceedings of the 3rd Workshop on Image and Video Retrieval Evaluation (2007)

Film Comic Reflecting Camera-Works

Masahiro Toyoura, Mamoru Kunihiro, and Xiaoyang Mao

University of Yamanashi, Takeda 4-3-11, Kofu, Yamanashi, Japan
mtoyoura@yamanashi.ac.jp

Abstract. We propose a novel technique for automatically creating film comics reflecting the camera-works of an original movie. Camera-works are one of the most important effects contributing to the mise en scene of the movie. A skilled director can use the camera-works dexterously for drawing the attention of audiences, representing sentiments, and give a change of pace in the movie. When creating film comics, camera-works are detected from the original movie, and mapped to panels and layouts of special comic styles. The technique is called as *the grammar of manga*. Our new algorithm is presented for automatically tiling the stylized panels into comic pages based on the grammar of manga. The results of our subject study show that reflecting camera-works in film comics enables the stories being presented in a more readable, vivid and immersive way.

Keywords: Film comic, grammar of film language, manga representation, and camera-work.

1 Introduction

A film comic is a kind of comic book created by selecting images from a movie and arranging them into a book of comic style. By turning a movie into a comic, film comic enables a wide range of readers, from little babies to comic manias, to easily browse or enjoy movie stories yet through another form of art media. Most of popular cartoon animations have been formed into film comics. We can buy printed film comics at book stores or enjoy digital film comics online. The process for creating a film comic includes the tasks for selecting images from the movie, editing the images into the comic pages, and inserting the balloons for representing the lines of characters. Even now, these tasks are mainly done manually. It is usually very time-consuming to select a compact set of images well depicting the story from the huge number of images in original movie. The most difficult thing, however, is how to well convey the mise en scene of original movie in comic style, so as to make the resulting comic as immersive and enjoyable as the original movie. Recently several researches on automatic generation of film comic have been reported [1,2,3,4,5]. Although each of those researches succeeded in developing some effective methods or tools for the automation of a part or the whole of the creation process, none of them has addressed, in particular, the automatic mapping of mise en scene from movie to comic. In this paper, we challenge the issue through reflecting camera-works in comic. Typical camera-works and their effects are well-known as a part of *the grammar of film language* [6]. A skilled director can use the camera-works dexterously for drawing the attention of audiences, representing sentiments, and give a change of pace in the movie. On the other hand, camera-works are also well-known

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 406–417, 2012.

as a part of *the grammar of manga* [7]. Modern Japanese manga, which is said to have been established by Osamu Tezuka, is actually featured with the introduction of the style of movies [8]. In manga, camera-works are represented as panels and layouts of special styles commonly accepted by creators and readers. We adopt those manga styles in representing the camera-works in film comic.

Our major contributions can be summarized as follows:

1. A new method for mapping the camera-works in a movie to the special panel styles and layouts of based on the grammar of manga.
2. A novel panel algorithm for automatically tiling the panels to form comic pages with camera-works representations.
3. A subject study to compare the film comics with and without camera-works.

Our experimental results show that reflecting camera-works in film comic can not only improve the readability of stories, but also enables viewers to better experience the mise en scene of original movies in a more immersive way.

The new mapping method and layout algorithm are implemented as a part of our fully automatic film comic generation system. Since the other parts of the system, including the selection of representative images, the detection of camera-works from movies and the balloon representation of lines of characters, are built upon existing technologies, we will mention them briefly in the related works and focus our discussion only on the major contributions, due to the length limit of manuscript. The remainder of the paper is organized as follows. Section 2 briefly reviews the related works. Section 3 describes the mapping of camera-works between the movie and comic. Section 4 presents the new algorithm for creating the page layouts of film comic. Section 5 discusses the results of the subject studies and Section 6 concludes the paper by showing several future research directions.

2 Related Works

A pilot work on film comic generation was done by W.I. Hwang et al. [1,3]. Although their system supports some stylized effects such as speed line and rotational trajectory which contribute to the mise en scene of movie, those effects need to be added by users manually. Preuß et al. [5] proposed an automatic system for converting movies to comics, but the method requires the screenplay of the movie to be given and hence limits its applications to a very special case. More recently, R. Hong et al. [2] presented fully automatic system which employs face detector, lip motion analysis, and motion analysis to realize the automatic script-face mapping and key-scene extraction. As another unique approach, Kunihiro et al. [4] used gaze information of viewers for positioning the balloons and trimming the images. They assume that the lines of characters can be obtained from the captions of the movie. Gaze information enables the system to identify the speaker even if the speaker does not have skin colored face, which occurs frequently in case of cartoon animation. The regions not attended by viewers would be trimmed to fit into a panel. Since film comics are usually created from cartoon animations, our system adopt the gaze-based approach for the trimming of images and the positioning of balloon.

Another research field closely related to film comic is video summary. While a film comic usually serves as an alternative art media of storytelling, video summary aims to enable users to quickly review videos and use it as the index to access to the detailed information when necessary [9]. Despite of such differences, they share many elemental technologies. As new styles of video summaries, storyboards [10], video tapestries [11,12] and comic-like layouts [13] are developed in recent years with which all important information are represented in a single image. Although they are effective for quickly capturing the summary of a video, film comic is more effective in getting readers immersed in a story by presenting a series of camera-works in a way temporally synchronized with the original video. Yoshitaka et al. have proposed to structuralize a movie based on the grammar of film language for making a video summary [14]. They also proposed to recognize the feelings of tension, liberty and loneliness from camera-works in another research [15].

Our system employs Porter's color histogram based approach [16] to segment a movie into shots and choose the image with the average color histogram of a shot as the representative image of the shot. The spatio-temporal image proposed in [17] is used for extracting camera-works from movies.

3 Mapping Camera-Works from Movie to Comic

In this section we explain typical camera-works and their corresponding comic representation. Typical camera-works include fading in/out, panning, tilting and zoom in/out.

Fading out is a camera-works that makes a scene disappear by gradually decreasing or increasing the brightness of the images. A shot ending with fading out gives a mild impression about the disappearing of the scene. Following the style of manga, we represent fading out with a stylized panel as shown in Figure 1(a). The fading out panel is positioned at the end of a page.

Fading in is a camera-work that brings a scene out by gradually increasing or decreasing the brightness of images. Same as fading out, the shot starting with fading in gives a mild impression on the appearing of the scene. We represent fading in with a panel as shown in Figure 1(b). A fading in panel should be positioned at the beginning of a page. A fading in is usually used right after fading out for achieving a smooth transition between two shots. Since a fading out panel is located at the end of a page, the fading in panel will be naturally positioned at the beginning of next page. Therefore, no special consideration is required in laying out the fading in panel.

Panning is a camera-work that changes the point of view along horizontal direction. It can be used to attract viewer's attention from right to left or left to right, enabling the viewer to catch the whole image of a wide object or impressing the viewer with the breadth of a landscape. It is also used for giving a sense of motion or fluidity to otherwise static shots. Inspired by some representations in manga, we represent panning with a stylized set of panels as shown in the middle row of Figure 1(c). Multiple panels corresponding to the images selected at successive positions during the panning are placed in the same row to convey the passage of time, continuousness, perspective or relationship between objects.

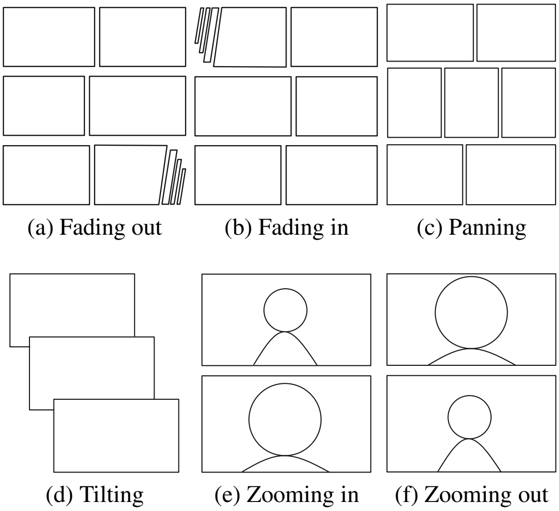


Fig. 1. Stylized panels and layouts for representing typical camera-works in film comic

Tilting is a camera-work that changes the point of view vertically up to down, or down to up. The former is also called as tilting down and the latter tilting up. The height or hugeness of objects can be emphasized with tilting. We represent the tilting with a page consisting of 3 panels corresponding to the images from the top, middle and bottom point of view, respectively (Figure 1(d)). Two adjacent panels are partially overlapped in such a way that the lower panel comes on top of the higher panel for tilting down, and vice versa for tiling up.

Zooming in is a camera-work that brings objects closer without physically moving a camera. It can attract viewer's attention to the focusing objects, by weakening the impression of background and showing the closed up views of the objects. Since zooming in is unnatural changing of point of view for human, it also has the effect of heightening the tension. Following the style of manga, we present zooming in with a two panel page as shown in Figure 1(e). The two panels correspond to the frame before and after the zooming in, respectively.

Zooming out is a camera-work pushing objects further away without physically moving a camera. It gives an overall image of the scene and makes it easy to grab the relationship between objects. The feeling of liberation is also expected with zooming out, since the focus is released from the objects. Similar to zooming in, we use a two panel page as shown in Figure 1(f) to represent zooming out, with the two panels corresponding to the frames before and after the zooming out, respectively.

4 Panel Layout

In this section, we discuss how to arrange panels to create the pages of film comic, taking into consideration the stylized panels and layouts described in the previous section.

The panel layout algorithm should follow the rules below:

1. Assign one page for the shot of tilting, zooming in, and zooming out.
2. Assign the last panel of a page for the shot of fading out.
3. Assign one row for the shot of panning.
4. Tile the other images in such a way that each page contains m rows and n columns as possible. The remaining space is filled by expanding the widths of some panels.

As shown in Figure 2, let p denote the number of panning shots, and $r_i (i = 1, 2, \cdots)$ the number of other shots between two panning shots. First, we decide the number of pages in a way that each page contains m rows and n columns as possible. With n panels in one row, the number of total rows s is $s = p + \Sigma_i \lceil r_i/n \rceil$. Note that $\lceil x \rceil$ is the ceiling of x. If each page consists of m rows, the number of pages t is $t = \lceil s/m \rceil$.

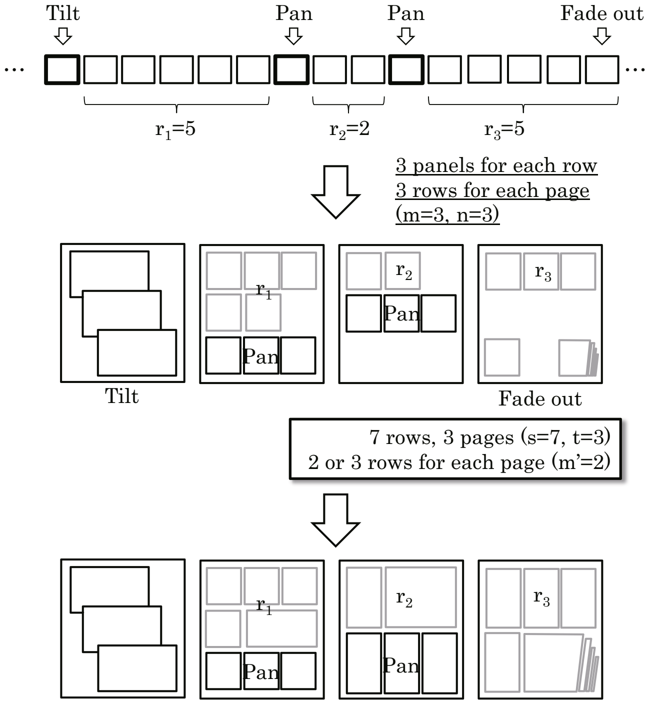


Fig. 2. Lay out panels in pages and rows

We cannot ensure that all pages have m rows exactly, except for the case that s is dividable by m. Let m' be the quotient of dividing the number of rows s by the number of pages t. We set the number of rows in each page to either m' or $m' + 1$, so as to keep the number of rows balanced among pages with a difference of 1 at most. In case m' is much less than n, a layout with narrow long panels will be created. When $|m' - n| > m_{th}$, we change n to $n-1$ and repeat this to decide new m' to improve the ratio of panels in row and column. m_{th} is a user given threshold to control the aspect ratio of panels.

As shown with the upper left image of Figure 3, a row right before a panning may not be fully filled with panels. The remaining space can be filled by expanding the width of a necessary number of panels. If those panels are not correctly chosen, however, we may get a layout with some panels across two rows. Such an example is shown in the upper right of Figure 3. This occurs when the third image is selected to expand. To avoid generating such a layout, we should select the last image to expand for instance.

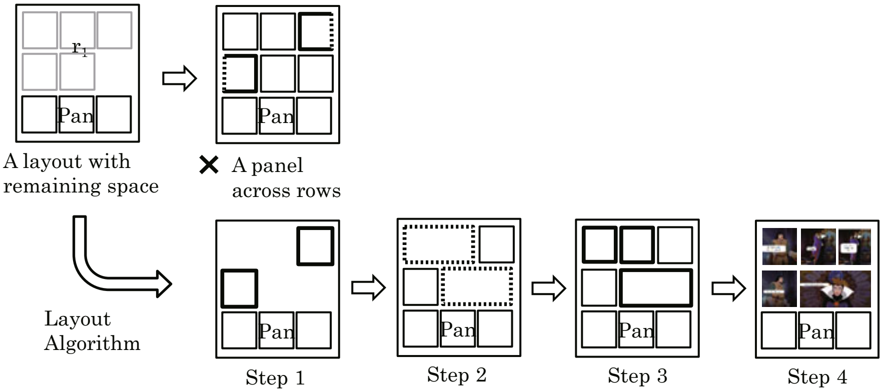


Fig. 3. Layout with/without a panel across rows

For any n and r_i, we can get a correct layout with the following algorithm.

Layout algorithm for filling the remaining space (Figure 3):

Step 1. If n is odd, assign a panel of width 1 at either the leftmost or the rightmost of each row.
Step 2. Divide the remaining space into the slots of width 2.
Step 3. Let $v = r \bmod n$, the remainder of dividing r by n. Randomly choose v slots and assign 1 panel of width 2 to each of them. The remaining slots are assigned with 2 panels of width 1.
Step 4. Fill all shots with the r images sequentially. The images are resized according to the width of the panels.

The above algorithm has been proven to be able to generate correct layouts always without panels across two rows. We include the formal mathematical proof in Appendix A.

5 Experimental Results

We conducted a subject study to investigate whether introducing camera-works improves the readability of stories and contributes to the representation of mise en scene of original movies. The subjects were 20 university students. They were asked to watch both the comic without stylized representation for camera-works, called as A, and the comic with the stylized panels and layouts reflecting the camera-works, called as B. B includes panels representing fading in/out, panning, tilting and zooming in/out. The movies used for the test were “Snow White”, “Pinocchio” and “Shrek 3”. The lines of characters were extracted from captions, and the gaze information of pre-viewers was used for positioning the balloons and trimming the images. The examples of generated comic pages are shown in Figures 4 and 5[1].

Our questionnaire includes the following questions together with a free feedback description. We presented A in first to a half of subjects, and B in first to the other half to eliminate the order effect.

Questions:

1. Is the story easy to understand?
2. Is the comic presented in a vivid style?
3. Are the scenes immersive?
4. Can you feel sentiments of characters?
5. Can you easily focus on the subjects in each scene?
6. Did you enjoy reading?

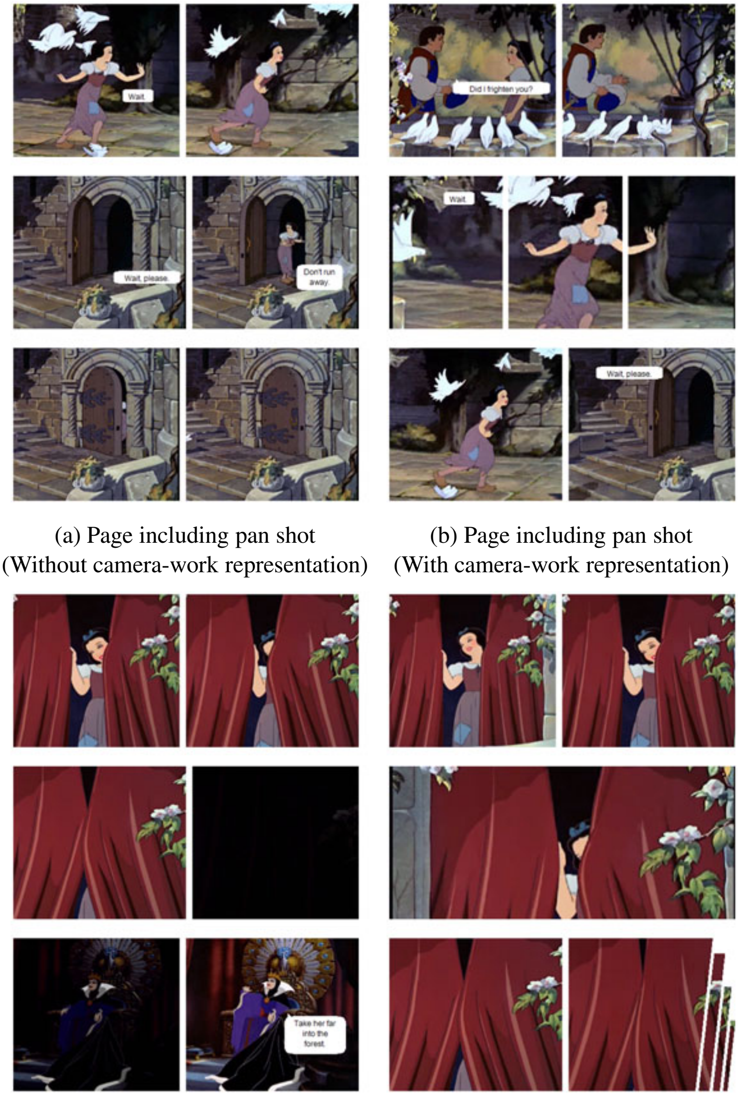
The results are shown in Table 1. The evaluation results are summed up in the nominal scale with 3 values for adopting χ^2-test. When $\chi^2(2)$ is larger than 9.2, 6.0 and 4.6, p is smaller than 0.01, 0.05 and 0.10, respectively.

Table 1. Results of the subject study. *: $p \leq 0.05$, **: $p \leq 0.01$, **bold values** are the largest ones.

	Snow White				Pinocchio				Shrek 3			
Q	$\chi^2(2)$	−	0	+	$\chi^2(2)$	−	0	+	$\chi^2(2)$	−	0	+
1	1.6	4	**8**	**8**	9.1^{*}	3	4	**13**	9.1^{*}	3	4	**13**
2	24.1^{**}	2	1	**17**	19.9^{**}	1	3	**16**	19.9^{**}	1	3	**16**
3	19.6^{**}	2	2	**16**	12.4^{**}	2	4	**14**	12.4^{**}	2	4	**14**
4	9.7^{**}	2	5	**13**	9.1^{*}	3	4	**13**	9.1^{*}	3	4	**13**
5	7.6^{*}	2	6	**12**	4.3	4	5	**11**	4.3	4	5	**11**
6	15.7^{**}	3	2	**15**	13.3^{**}	1	5	**14**	13.3^{**}	1	5	**14**

For Q1 of “Pinocchio” and “Shrek 3”, more subjects preferred B, the comic pages with stylized panels and layouts reflecting camera-works. We can conclude that reflecting camera-works improves the readability of stories. However, B was not preferred for Q1 of “Snow White”. In the free feedback description, a subject answered that there are too many panels being emphasized in the comic pages reflecting camera-works. Another subject answered that he got tired since it was too long. There are many stylized panels and layouts in case of “Snow White” in fact, since the original movie had many camera-works. It seems that the excessive presentment had prevented subjects from understanding the story of “Snow White” well. To narrow down to the panels that really

[1] The intellectual property of “Snow White” shown in Figure 4 and Figure 5 has expired.



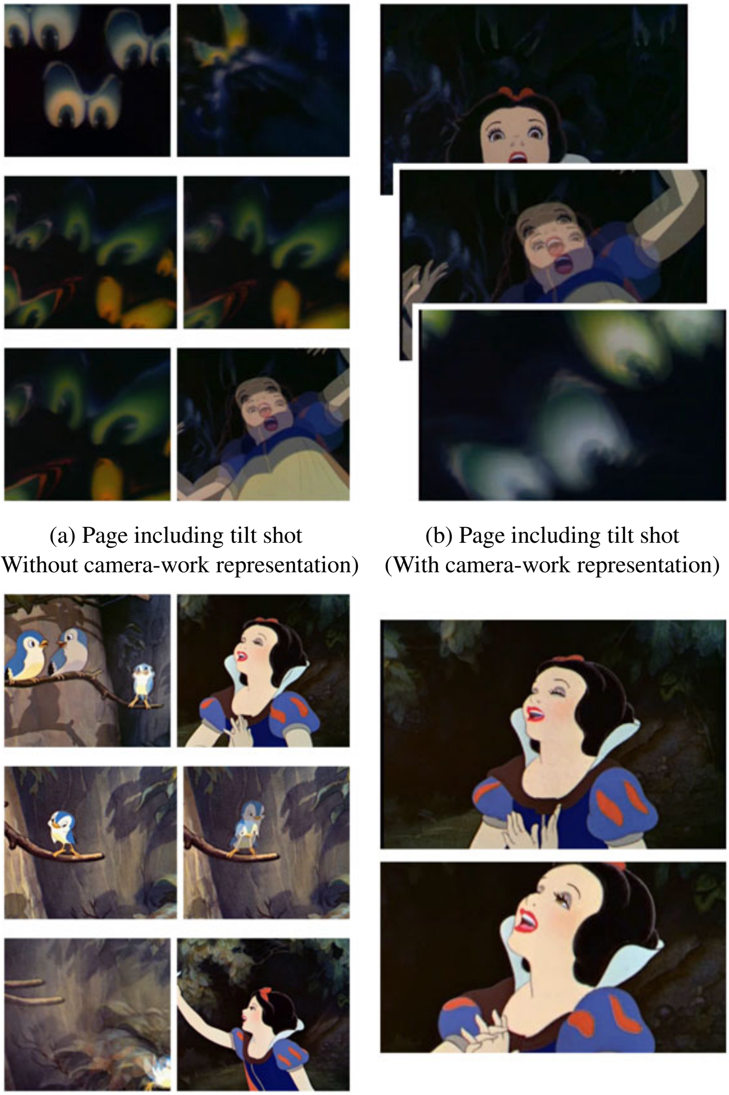
(a) Page including pan shot
(Without camera-work representation)

(b) Page including pan shot
(With camera-work representation)

(c) Page including fade out shot
(Without camera-work representation)

(d) Page including fade out shot
(With camera-work representation)

Fig. 4. Generated comic pages with/without stylized panels and layouts. (a) and (b) are the pages including panning shots, and (c) and (d) including fading out shots. (a), (c) are comic pages without reflecting camera-works, while (b), (d) are those with the stylized panels and layouts reflecting camera-works. Read from top left to bottom right.



(a) Page including tilt shot
(Without camera-work representation)

(b) Page including tilt shot
(With camera-work representation)

(c) Page including zoom-in shot
(Without camera-work representation)

(d) Page including zoom-in shot
(With camera-work representation)

Fig. 5. Generated comic pages with/without stylized panels and layouts in manga. (a) and (b) are the pages including tilting shots, and (c) and (d) including zooming in shots. (a), (c) are comic pages without reflecting camera-works, while (b), (d) are those with the stylized panels and layouts reflecting camera-works. Read from top left to bottom right.

need to be emphasized, more advanced video understanding technologies is required. We are going to address the problem as a future work.

For Q2, Q3 and Q4 of all movies, B was preferred by more subjects. Since camera-works usually can help giving a vivid impression or rhythmic pace to a movie, we can expect that a comic with camera-works reflected can also achieve the same effects. A subject answered that he could not concentrate on reading A compared with B, since A had uniform panels. The result of Q4 showed that effects of camera-works can also facilitate viewers to better feel the sentiments of characters with our proposed method. For Q5, B was better evaluated for "Snow White" only. In "Pinocchio" and "Shrek 3", there were the shots including two and more people talking together, which might be a reason for preventing viewers from focusing on the subject. For Q6, B was preferred by most users for all movies. We can conclude that a film comic generated with our proposed method enable viewers to view the story in a more enjoyable and immersive way.

Subjects also answered that echoic and mimetic words would improve the quality of the comic and inserting additional panels without balloons may make the comic more rhythmic. Those feedbacks are important clues to the further improvement of the technology in our future work.

6 Conclusions and Future Works

We proposed a new method for automatically generating film comics with stylized panels and layouts reflecting the camera-works of the original movie. We have confirmed that the comic generated with the proposed technique can better convey the mise en scene of the original movie.

One of our future works is to develop a technique for narrowing down to the most important panels that really need to be emphasized in the comics. A subject of the evaluation experiment pointed out that the comics were over accentuated when too many camera-works were used in a movie. We will seek more powerful image understanding and motion analysis technologies for estimating the importance of camera-works. We would like to provide an interactive editing tool enabling users to easily reflect their personal styles, while continuing exploring the potential of automatic approach in the mapping of mise en scene between movie and comic.

References

1. Chun, B.-K., Ryu, D.-S., Hwang, W.-I., Cho, H.-G.: An Automated Procedure for Word Balloon Placement in Cinema Comics. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Nefian, A., Meenakshisundaram, G., Pascucci, V., Zara, J., Molineros, J., Theisel, H., Malzbender, T. (eds.) ISVC 2006, Part II. LNCS, vol. 4292, pp. 576–585. Springer, Heidelberg (2006)
2. Hong, R., Yuan, X.T., Xu, M., Wang, M., Yan, S., Chua, T.S.: Movie2comics: a feast of multimedia artwork. In: Proceedings of the International Conference on Multimedia, pp. 611–614 (2010)

3. Hwang, W.I., Lee, P.J., Chun, B.K., Ryu, D.S., Cho, H.G.: Cinema comics: Cartoon generation from video stream. In: International Conference on Computer Graphics Theory and Applications, pp. 299–304 (2006)
4. Kunihiro, M., Mao, X.: The automatic generation of the film comics from the animation with eyes information. In: Symposium on Visual Computing, Graphics and CAD (2008) Article 13 (in Japanese)
5. Preuß, J., Loviscach, J.: From movie to comics, informed by the screenplay. In: ACM SIGGRAPH (Poster) (2007)
6. Arijon, D.: Grammar of the Film Language. Silman-James Press (1991)
7. Tsukamoto, H.: Manga Bible (2007)
8. Ohtsuka, E.: Guide for Movie Style Manga. Ascii Books (2010) (in Japanese)
9. Boreczky, J.S., Girgensohn, A., Golovchinsky, G., Uchihashi, S.: An interactive comic book presentation for exploring video. In: ACM Conference on Computer-Human Interaction (CHI), pp. 185–192 (2000)
10. Goldman, D.B., Curless, B., Salesin, D., Seitz, S.M.: Schematic storyboarding for video visualization and editing. ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH) 25(3), 862–871 (2006)
11. Barnes, C., Goldman, D.B., Shechtman, E., Finkelstein, A.: Video tapestries with continuous temporal zoom. ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH) 25(3) (2010)
12. Correa, C.D., Ma, K.L.: Dynamic video narratives. ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH) 29(3) (2010)
13. Myodo, E., Takagi, K., Yoneyama, A.: Generation of efficient comic-like layout for video summaries. Forum on Information Technology (FIT) 3, 291–292 (2008) (in Japanese)
14. Yoshitaka, A., Deguchi, Y.: Video summarization based on film grammar. In: IEEE Workshop on Multimedia Signal Processing, pp. 333–336 (2005)
15. Yoshitaka, A., Matsui, R., Hirashima, T.: Extracting movie rendition based on camera work detection and classification. Transactions of Information Processing Society of Japan 47(6), 1696–1707 (2006) (in Japanese)
16. Porter, S.V.: Video Segmentation and Indexing using Motion Estimation. PhD thesis, University of Bristol (2004)
17. Ngo, C.W., Pong, T.C., Zhang, H.J.: Motion analysis and segmentation through spatio-temporal slices processing. IEEE Transactions on Image Processing 12, 341–355 (2003)

Appendix A. The Validity of Scheduling Algorithm Described in Section 4

Let n be the number of panels in a row, r_i the number of images. The number of rows k required for arranging r_i images is calculated as $k = \lceil r_i/n \rceil$, which is an integer and satisfies the following equation.

$$(k-1)n < r_i \leq kn, \quad k \geq 1. \tag{1}$$

In addition, let l and b denote the quotient and remainder of dividing kn by r_i, respectively. We have

$$kn = lr_i + b, \quad 0 \leq b < r_i. \tag{2}$$

To proof that a way of scheduling panels without panels across two rows exists, we first give Lemma 1 and Lemma 2.

Lemma 1. If $k \geq 2$, $l = 1$.

Proof. By Eq. (1), $r_i \leq kn$, therefore, $l \geq 1$. If $l \geq 2$, $kn \geq 2r_i$ is satisfied by Eq. (2). By Eq. (1),

$$kn \geq 2r_i > 2(k-1)n. \tag{3}$$

Hence,

$$k > 2k-2. \tag{4}$$

The above inequation is not true when $k \geq 2$. Thus, $l < 2$ is proved by contradiction. Since l is an integer by definition, $l = 1$. □

Lemma 2. $b \leq n-1$.

Proof. Assume $b \geq n$. By Eq. (1),

$$kn = r_i + b \geq r_i + n. \tag{5}$$

Then,

$$(k-1)n \geq r_i. \tag{6}$$

It conflicts with Eq. (1). Hence, $b < n$. Since b is an integer by definition, $b \leq n-1$. □

Based on Lemma 1 and Lemma 2, we can derive Proposition as follows.

Proposition. For any n and any r_i, a way of scheduling panels without panels across two rows exists.

Proof. When $k = 1$, no panel that acrosses two rows exists, since panels are set in one row. When $n = 1$, no panel that acrosses two rows exists, since one panel is set in each row. When $k \geq 2, n \geq 2$, the remining space of b panels can be space is filled by doubling the width of b images selected from r_i images. $r_i - b$ images are remining panels of width 1. As $r_i - b$ equals to $kn - 2b$ by Eq. (2) and Lemma 2, b panels of width 2 and $kn - 2b$ panels of width 1 should be set in kn. Now, by Lemma 2,

$$kn - 2b \geq kn - 2(n-1). \tag{7}$$

Since $n \geq 2$,

$$kn - 2(n-1) = (k-2)n + 2 \geq 2k - 2. \tag{8}$$

Furthermore, since $k \geq 2$,

$$2k - 2 \geq k.$$

Hence, $kn - 2b$, or the number of panels of width 1, is larger than k, the number of rows.

b panels of width 2 can be arranged in the following ways. When n is odd, one panel of width 1 is set at either the leftmost or the rightmost of each row. The remining space can be devided into the slots of width 2, which can be filled with panels of width 2 or pairs of panels of width 1. When n is even, the whole space can be divided into the slots of width 2, which can be filled with panel of width 2 or pairs of panels of width 1. □

Large-Scale Similarity-Based Join Processing in Multimedia Databases

Harald Kosch and Andreas Wölfl

University of Passau, Germany
Distributed Information Systems, Faculty of Informatics and Mathematics
{Harald.Kosch,Andreas.Woelfl}@uni-passau.de

Abstract. This paper presents efficient parallelization strategies for processing large-scale multimedia database operations. These strategies are implemented by extending and parallelizing the GiST (Generalized Search Tree)-framework. Both data and pipeline parallelism strategies are used to execute multi join operations. We integrate the parallelized framework into an Oracle 11g Multimedia Database using its extension mechanisms. Our strategies and their implementations are tested and validated with real and random data sets consisting of up-to 10 millions of image objects.

Keywords: Multimedia Databases, Similarity-based Operations, Parallel Processing.

1 Motivation

In multimedia databases, the set of multimedia objects are described by a collection of features. In the case of images, examples of features include color histograms, color moments, textures, shape descriptions and so on [1]. Common multimedia database operations are *similarity-based selection queries* [2]. Two main types of selections are considered. First, one looks for objects whose feature vectors are within a given range (range queries) to the feature vector of a given query object. Second, one finds objects whose feature vectors have the most similar values to the feature vector of a given query object (nearest-neighbor queries). In addition, *similarity-based join queries* are considered [3,4,5]. The similarity-based join is useful to merge two sets of multimedia objects based on their pairwise similarity. It can be employed within a single database, for instance in social media applications to compute pairwise similarity of images posted by different interest groups. Similarity-based joins may also be employed to merge sets of image objects stemming from different repositories in order to find out pairwise similarity and thus create interlinking among them. In this scope, our paper studies methods for efficiently joining large sets of image objects. Our solution strategies include several data and pipeline parallelization methods.

2 Multimedia Database Operations

Let M_1 and M_2 be two image tables in a multimedia database. The image tables contain the image objects together with their feature vectors extracted

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 418–428, 2012.

beforehand from the images. For the following join example, we suppose that we have one feature vector for an image object. A similarity-based join performs for each image object of the left input table M_1 a similarity search in the right input table M_2. Let M_1 have the following two objects: $(x_1, (12, 15, 10))$ and $(x_2, (22, 32, 5))$ and M_2 has the objects $(y_1, (10, 14, 10))$ and $(y_2, (14, 16, 12))$. The similarity search is a range query with a range of 4 using the Euclidean distance. y_1, y_2 are the result objects of the similarity search of x_1 in M_2. For x_2 we do not have any similar objects in M_2. Thus, two result tuples are output: $(x_1, y_1, (12, 15, 10), (10, 14, 10))$ and $(x_1, y_2, (12, 15, 10), (14, 16, 12))$.

Cascading the joins allows one to merge more than two multimedia object sets (referred hereafter as multi join operation). Consider now a third image table M_3 with the following tuples $(z_1, (8, 15, 10))$ and $(z_2, (16, 20, 5))$. The result of the former join between M_1 and M_2 shall be joined to M_3 using a range query (range of 5 using the Manhattan distance). This subsequent join works as follows. First, we determine which feature vector of the join result M_1 with M_2 is used for the subsequent join, then we perform for each join result tuple a similarity search in M_3. We choose the feature vector of M_2 for the subsequent join. For the first tuple, $(x_1, y_1, (12, 15, 10), (10, 14, 10))$ we find z_1 to be similar, while z_2 is not similar enough. For the second tuple, $(x_1, y_2, (12, 15, 10), (14, 16, 12))$ we don't find any similar image object in M_3. Thus, $(x_1, y_1, z_1, (12, 15, 10), (10, 14, 10), (8, 15, 10))$ is the only result tuple of the multi join operation.

Let $M_1(p, fv_p, a_p)$ and $M_2(q, fv_q, a_q)$ be two *base* image tables, p, q are the image objects. $fv_p = f_1, f_2, ...$ is the set of feature vectors representing the low-level features of the object p (respectively for q). a_p and a_q are object types containing the attribute components that may be used to describe the objects. Formally, the **Similarity-based Image Join** is defined as:

(a) $join(M_1, M_2, f_p, f_q, \varepsilon) = X \Leftrightarrow X \in M_1 \times M_2 \wedge \forall((p, fv_p, a_p), (q, fv_q, a_q)) \in X \bullet q \in \varepsilon - similarity(M_2, p, f_p, f_q, \varepsilon)$.
(b) $join(M_1, M_2, f_p, f_q, k) = X \Leftrightarrow X \in M_1 \times M_2 \wedge \forall((p, fv_p, a_p), (q, fv_q, a_q)) \in X \bullet q \in k\text{-}NN\text{-}similarity(M_2, p, f_p, f_q, k)$.

$\varepsilon - similarity(M_2, p, f_p, f_q, \varepsilon)$ is the ε-similarity in M_2 computing all neighbors whose distance to the query image p is below a threshold ε. $k\text{-}NN\text{-}similarity(M_2, p, f_p, f_q, k)$ is the k-NN-similarity computing the k-nearest neighbors to the query image p in M_2.

The result of a similarity-based join is a new intermediate image table M which contains the components of both joined image tables, thus the table schema expresses as: $M(p, q, fv_p, fv_q, a_p, a_q)$. The definition above can simply be extended to the join of one base table with an intermediate image table and to a join of two intermediate tables.

The **Multi Similarity-based Join** specifies $n \geq 1$ similarity-based joins. It is represented by a *linear processing tree* with the following syntax: $PT ::= M_i$ or $PT ::= join(PT_1, M_i, f_1, f_2, \varepsilon)$ or $PT ::= join(PT_1, M_i, f_1, f_2, k)$ where PT_1 is a linear processing tree.

For simplifying the notations and without loosing the generality of the optimization consideration, each image table M_i is used exactly once.

3 Related Work

This paper concentrates on methods for *efficient* processing of large-scale multi similarity-based join operations. To compute the similarity-based operations efficiently, the feature space is usually indexed using a multidimensional index data structure [6].

The similarity search in *large databases* may reveal time-consuming, if many index nodes have to be examined in a range search. To cope with this, recent works proposed a compact representation of the similarity join result [7]. While this limits the problem for a similarity-based join, it cannot solve the problem for multi join operations. If a nearest-neighbor search is employed, the computational complexity is in the many distance computations and in the size of the priority queues used for pruning. It becomes specially time- and space-critical for high-dimensional database spaces (see [8]). Recent works focused on improving the nearest-neighbor algorithm by using distance estimators, in order to reduce the storage requirements [9]. It has been proven that this can prune the priority queue without altering the output of the query. But the time complexity is still present. As a consequence, several authors considered the parallelization of similarity-based selection queries [10,11,12].

These works consider either *data distribution*, *space partitioning*, or *multiplexed index structures. Data distribution* assigns data rectangles/spheres to different computing nodes in a round robin manner or by a hash function. *Space partitioning* divides the space into partitions which are assigned to separate computing nodes. *Multiplexed index structures* are distributed over nodes with pointers across the nodes. Data distribution balances the load, while space partitioning activates few nodes. Multiplexed index structures are more flexible, they can balance the number of activated nodes vs load balancing among nodes. These strategies are well-researched. Coming to the processing of similarity-based join operations, Kosch et al. [3] considered simultaneously processing of two index structures, and Yu et al. [4] focused on dimension reductions and distance index structures. Both work did not consider parallelization strategies. In spatial databases, parallelization of joins have been considered by several authors (see Section 5 of Jacox et al. [13]). These works consider mainly the parallelization of the filter step of a spatial join which is special to spatial databases and cannot be directly transferred to multimedia databases. A recent work concentrated on the optimal placement of similarity-based multimedia operations in complex multimedia queries, a parallelization strategy is however only considered for the selection operators [14]. Distributed processing in IR systems considered also data distributions. Content replications and distributions for cluster-based architectures is for instance considered by Klampanos et al. [15].

Extending the parallelization of similarity-based selections to multi similarity-based joins is a challenging task. First, the index look-ups of the left input table image objects to the right image table must be parallelized. Second, the processing of the different join levels must be also parallelized in order to fully exploit the computing power of a parallel machine.

4 Large-Scale Processing

We want to effectively process large-scale *multimedia queries* specifying $n \geq 1$ similarity-based joins. For each $M_i(o_i, fv_i, a_i)$ $(1 \leq i \leq n)$ with $fv_i = f_{i1}, f_{i2}, ...$ being the set of feature vectors representing the low-level features of the object o_i, we assume that multidimensional index structures are available for each different $f \in fv_i$ to efficiently carry out the ε or k-NN-similarity. These multidimensional index structures shall be able to hold a large number of data points in possibly high-dimensional data spaces. From related work [6], *Data Partitioning* index structures are good candidates. We concentrate in this work on hierarchical index structures (e.g., X-,SS-, SR- and TV-trees), as they are the mostly used and tuned [16] index structures in multimedia database products.

Nested-Loop Index Join. The *method* for performing a similarity-based join is to apply the ε or k-NN-similarity for each object of the left input table M_1 as a query object o looking for its similar objects from the right input table M_2. The latter operation is implemented as an index look-up in the index structure of the right input table M_2.

The research problem we pose is how to process effectively large-scale operations. Our two main methods are: **Data Parallelism** of a single similarity-based operation and and **Pipeline Parallelism** of multiple similarity-based operations.

Data Parallelism. We suppose that we are disposing a cluster system with a high-speed interconnected computing nodes in a distributed memory architecture. Each node has b cores which access the shared memory on the node. We assume a shared disk architecture.

The processing is done in two phases. In the **index build phase** the feature vectors of all available image tables are distributed in round robin manner over the computing nodes. At each node, a local index structure is built. Thus, each distribution is complete and not overlapping. We have chosen the data distribution over the space partitioning parallelization (see Section 3), because it is load balancing for multi join processing. In a space partitioning approach, each level of the multi join processing introduces a load imbalance which could easily sum up to an important overall imbalance and thus leading to much higher-response times.

The processing of the similarity-based join is done in a master-worker manner (**processing phase**). The multimedia database server acts as master. It forwards the common feature vector for each object of the left input table M_1 to all computing nodes (workers), where an index look-up in the common feature vector index structure of the right input table M_2 is performed. The processing of each local join (on each node) can be done independently from the others. This scales very well. The results are sent back to the master. It prepares then for the subsequent join.

Example. Consider two image tables, M_1, M_2 and $a = 4$ computing nodes. The feature vectors of M_1 and M_2 are:

$fv_1 = \{dominant\_color, color\_histogram\}$ and
$fv_2 = \{dominant\_color, edge\_histogram\}$.

In the *index build phase*, the four available feature vectors (2 for M_1 and 2 for M_2) are distributed uniformly over the 4 nodes, thus each node holds 1/4 of the complete index structures. In the *processing phase*, we like to perform a *3-NN* similarity-based join of the two image tables. The common feature vector for M_1 and M_2 shall be the *dominant_color*. The master initiates the join by scanning M_1. For each tuple, it extracts the *dominant_color* feature vector and forwards the query information (vector and *3-NN*) to all nodes, where an index look-up on the local *dominant_color* index of the M_2 table is performed. The result (the tuple id) is returned to the master.

Pipeline Parallelism. We propose to process multiple similarity-based joins in pipelined fashion. The principle idea is taken from right-deep processing of join operations in parallel relational databases. The way of processing traditional joins works with exact matching, while similarity-based join processing works on possibly multiple feature vectors based on similarity matching. Thus, the distributed implementation of the pipelined similarity-based join cannot directly use right-deep processing implementations. We therefore originally designed a pipelined processing strategy for similarity-based joins with a fully threaded realization exploiting multi-core parallelism on different index structures (and feature vectors). The pipeline parallelism works as follows, once the result of a join arrives at the master node, the feature vector for the subsequent join is looked up the corresponding base table. It is immediately send to all nodes to probe against the next right input operator. Thus, the former join and the subsequent join execute on each node in parallel. In order to scale, the processing of the former join and the subsequent one is done by different threads allocated to different cores on each node. For instance, if each node has $b = 4$ cores, one may execute 4 join levels in parallel. The access to the shared memory could be the limiting factor for this parallelization. But, as the joins are performed on different right image tables, the access are not concurrently to the same data. This strategy scales well.

5 Efficient Parallel Implementation

The implementation of the multidimensional index structure and the similarity-based join operation is done in two main parts, the **Database Extension (SBMJLibrary)** and the **External Index Structure (GiSTServer)**.

Database Extension (SBMJLibrary): An Oracle 11g Database with the *Oracle Data Cartridge Interface Technology (ODCI)* offering extensibility interfaces is used. We can extend the query processing, type system and data indexing by calling external C,C++ or Java routines. We implemented the similarity-based

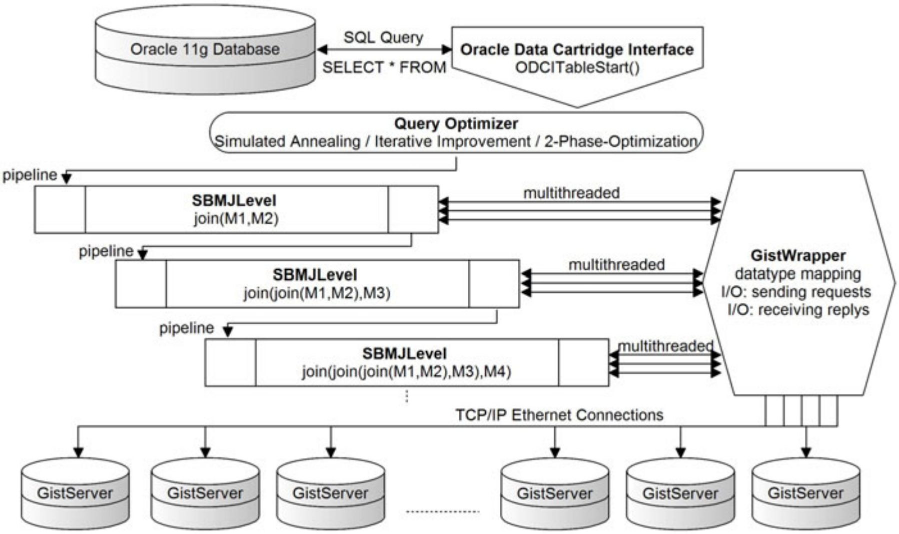


Fig. 1. Framework for the Similarity-based Join Execution

join operation as an external table function in C++. Figure 1 shows the framework for the similarity-based join execution. In particular one sees the components involved, these are the Oracle 11g Database, the ODCI Interface, the Query Optimizer, the SBMJLevel and the GiST-Framework. The ODCI Interfaces include external routines that are combined to a shared library. Each subsequent level of a similarity-based join is represented by an instance of a SBMJLevel. In each SBMJLevel, the specific join parameters (e.g., involved table indices, join type, etc.) and a reference to the GiSTWrapper are saved. New intermediate result tuples are computed by requesting index scans via GiSTWrapper calls. The GiSTWrapper is a singleton Object, designed to communicate with the GiSTServers.

Now, let's take a look on our realization. The join processing starts with the **Query Optimization**. Its task is to find a good ordering of the join operations, where few intermediate join result tuples are produced. In a previous paper [5], we introduced a cost model to compute the number of intermediate result tuples. This paper uses the cost model and implements an Iterative Improvement search strategy to compute a good ordering. The Iterative Improvement algorithm makes a specified number of local optimization. The algorithm starts at a random state and improves the solution by repeatedly performing swap- or 3-cycle-transformations until a local minimum or the maximal number of iterations is reached. The query optimization is the first part of the similarity-based join processing.

The **execution of the joins**, thus generating the result tuples is done in a *parallel and multithreaded manner* (**processing phase**). Each level of a similarity-based join (SBMJLevel) runs in a separate thread and is connected to an own input queue. An input queue contains intermediate result tuples from the upper level and acts as a data pipeline (*pipeline parallelism*). The input queue of the top level is initialized with feature vector / row-id pairs of the left input table of the deepest join. With each feature vector in the input queue, a new index scan request to the GiSTWrapper is started. The result of this query is a set of matching pairs, which are merged to a new intermediate result tuple, that is directly piped out to the input queue of the lower level. Considering that one level will always produce new tuples faster (or slower) than another, the level processing is also multithreaded to avoid a bottleneck. If there are more than one intermediate tuple in the input queue, the level runs a predefined number of worker threads for parallel index scan requests.

Now let us concentrate on the *GiSTWrapper*. GiST stands for *Generalized Search Tree* and is an extensible data structure framework, developed at the University of Berkeley [17]. The GiST-Library (libgist2.0) contains implementations for R-,R*,SS-,SR-,B- and NP-Trees and is designed for storing multi-dimensional index structures. Our extension, the GiSTWrapper is an interface between the database and the external index structure, which is based on the GiST-framework. The two main tasks of the GiSTWrapper are first to map the Oracle ODCI-datatypes to native C-datatypes and second to forward a query (e.g., index scan request) to the GiSTServers. Considering that the external index structure is uniformly distributed over multiple GiSTServers in the network, one query has to be sent to all GiSTServers by using a TCP/IP connection. For this purpose and to satisfy the criteria of a parallel and multithreaded execution, the GiSTWrapper runs a new worker thread for each connection. This thread holds the connection until the GiSTServer completes the computing of the query. The result of all threads is merged into a set of tuples, which are returned to the level thread. Note that in a *k-NN*-query, the result set contains *k * #GiSTServers* tuples and must be filtered before it is returned.

External Index Structure (GiSTServer). A GiSTServer communicates over our multithreaded **TCP/IP-Connection-Pool** which runs a configurable number of threads at start-up. When a new request arrives and a free thread is available, the execution of the request is allocated to that thread. If not, the request has to wait until a new thread is available. After a new request is received, it is forwarded to a **request handler**. The request handler parses each component, checks for errors and classifies the request into data manipulating and data querying requests. This determination is important for thread synchronization. In order to avoid that the threads must be fully synchronized when calling the GiST-framework, we preload multiple GiST-framework instances into the memory at start-up. This allows truly parallel query execution of threads on each node. This start-up handling is done by the **GiSTHolder-Pool** and the **GiSTHolder**. A GistHolder holds the entire GiST-framework in protected memory dynamically. Thus, a GiSTHolder can process a request on his own, in an independent memory area. The GiSTHolder-Pool is responsible for the scheduling of the GiSTHolder. In this manner, multiple data querying requests can be executed in parallel. On the completion of a request, either a status code (data manipulating) or the result tuples (data querying) are returned to the database.

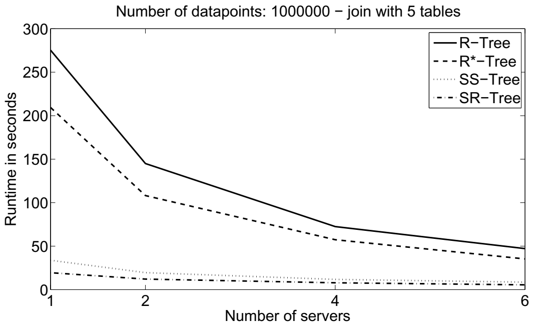


Fig. 2. Similarity-based Join Response Times with Real Data (1 Million), up to 6 Processors

As the TCP/IP Connection-Pool only supports synchronous communication, the reply is sent with the same connection as the request was received. Furthermore, we added several updates and bug fixes to the original GiST-framework, e.g., large file support on Linux platforms, so the GiSTServer can handle larger files than 2GB both on 64bit and 32bit compilates.

6 Experimental Results

We compiled and installed our implementation on a network cluster system. Our cluster system consists of one master node and 16 worker nodes, interconnected by a reliable Gigabit Ethernet network. To store the external index structure, we used a shared disk with 2TB disk space, accessible by each node. All nodes have the following hardware specification:

```
Operating System: openSUSE 10.3 (X86-64)
Kernel: 2.6.22.19-0.4-default x86_64
CPU(s): 1 Quad Core Intel Xeon E5405s
CPU Clock: 2000.069 MHz (each core)
Memory: 16078.3MB
```

We installed the Oracle 11g Database on the master node, the GiSTServers on the worker nodes.

For the following series of tests, we used two different data sets: first, *randomly generated data* (up to 10 millions) and second, *real feature vectors*, extracted from images of the Flickr Database (up to 1 million). The main purpose of the tests is to measure the response time improvements of the *processing phase* while increasing the number of computing GiSTServers.

We executed similarity-based joins with varying image data sets (random data, real data), four index structure types (R-Tree, R*-Tree, SS-Tree, SR-Tree), different table sizes (10^3 to 10^7) and join depths (1-4 cascading joins). We first found out that on every node, at least 10000 datapoints must be inserted to achieve an improvement. From that, the greater the table size the clearer the response times enhancements. In general, the greater the join depth, the better

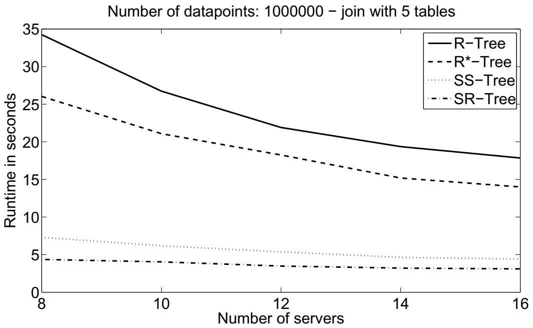


Fig. 3. Similarity-based Join Response Times with Real Data (1 Million), from 8 Processors

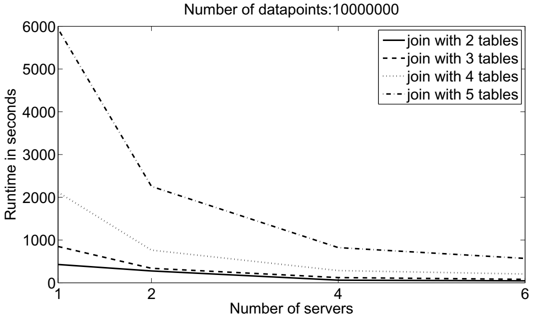


Fig. 4. Similarity-based Join Response Times with Random Data (10 Millions), up to 6 Processors

the response time enhancements. The response time highly depends on which tree family is used: Sphere-based trees deliver (compared to rectangle-based trees) the best performance. In this paper, due to the page size limit, the figures will be shown for the maximal amount of treated image objects: 10 millions in the case of randomly generated data and 1 million in the case of real data and for the maximal join depths of four.

The result of a cascading similarity-based join with 4 levels on the real data is shown in figures 2 and 3: $join(join(join(join(M_1, M_2), M_3), M_4), M_5)$. The feature vector in the real data is a 64-dimensional MPEG-7 *Scalable Color*. In order to regularize the number of result tuples, we set the size of the left input table of the deepest join M_1 to 10 datapoints. Each look-up table contained 1 million datapoints. We used a *1-NN* similarity search on the deepest level and a *3-NN* similarity search on the other levels. With this setup, 810 result tuples are produced. The Database Extension and the GiSTServers were configured to exploit the entire hardware capacity. Each of the 4 SBMJLevel were running 4 worker threads, so $16 * \#nodes$ threads were requesting the external index structures in parallel. Regarding the memory limitation on one node, the GiSTServers' thread-pool was set to 8 threads. The scaling is very good.

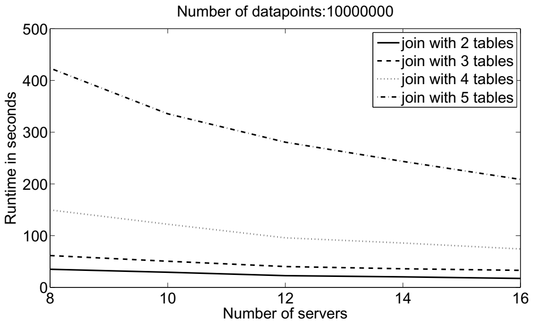


Fig. 5. Similarity-based Join Response Times with Random Data (10 Millions), from 8 Processors

Doubling the number of GiSTServers leads to just above *halving the response time*. The slightly lower values on a larger number of computing nodes results from a communication overhead and the bottleneck of the shared disk access.

Figures 4 and 5 show the results for the similarity-based joins on 5 randomly generated multimedia tables with 10 millions datapoints on an R*-Tree based index structure. We used the same thread- and similarity search configuration as in the previous tests. The response time improvements are similar to figures 2 and 3, except for nodes=1,2. We had to reduce the thread-pool of the nodes to 1 thread on 1 node and 3 threads on 2 nodes. This was necessary to avoid memory overflow.

In general, we observed that the running times for the real data sets are appreciable faster, which is due to the higher density areas in the real data sets. The difference decreases with a higher number of processors involved. Our parallelization strategy clearly scales with the size of the data sets joined and the number of processors used. It brings down the response times of complex queries to reasonable waiting periods with 10 millions data points involved.

7 Conclusion

This paper presented efficient parallelization methods for processing large-scale multimedia database operations. In special, similarity-based image join operations were efficiently been carried out by using data and pipeline parallelization strategies. In future works, we will extend the parallelization framework of GiST with further index structures, e.g., from the Windsurf framework[1]. We also intend to integrate so called ”distance joins” between two image input sets, e.g., to compute the K-closest pairs of the two image input sets, ordered by the distance of objects in each pair.

[1] http://www-db.deis.unibo.it/Windsurf/

References

1. Lew, M.S., Sebe, N., Djerba, C., Jain, R.: Content-based multimedia information retrieval: State-of-the-art and challenges. ACM Transactions on Multimedia Computing, Communications, and Applications 2(1), 1–19 (2006)
2. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. ACM Computing Surveys 40(2), 1–60 (2008)
3. Kosch, H., Atnafu, S.: Processing a multimedia join through the method of nearest neighbor search. Inf. Process. Lett. 82(5), 269–276 (2002)
4. Yu, C., Cui, B., Wang, S., Su, J.: Efficient index-based knn join processing for high-dimensional data. Information and Software Technology 49, 332–344 (2007)
5. Kosch, H.: Optimizing similarity-based image joins in a multimedia database. In: Proceedings of the ACM International Workshop on Very-Large-Scale Multimedia Corpus, Mining and Retrieval, VLS-MCMR 2010, pp. 37–42. ACM (2010)
6. Samet, H.: Foundations of Multidimensional and Metric Data Structures. Morgan Kaufmann (2006)
7. Bryan, B., Eberhardt, F., Faloutsos, C.: Compact similarity joins. In: Proceedings of the IEEE International Conference on Data Engineering, ICDE 2008, pp. 346–355. IEEE (2008)
8. Samet, H.: Techniques for similarity searching in multimedia databases. PVLDB 3(2), 1649–1650 (2010)
9. Bustos, B., Navarro, G.: Improving the space cost of k-NN search in metric spaces by using distance estimators. Multimedia Tools and Appl. 41(2), 215–233 (2009)
10. Berchtold, S., Böhm, C., Braunmüller, B., Keim, D.A., Kriegel, H.-P.: Fast parallel similarity search in multimedia databases. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 1–12. ACM (1997)
11. Alpkocak, A., Danisman, T., Ulker, T.: A Parallel Similarity Search in High Dimensional Metric Space Using M-Tree. In: Grigoras, D., Nicolau, A., Toursel, B., Folliot, B. (eds.) IWCC 2001. LNCS, vol. 2326, pp. 166–171. Springer, Heidelberg (2002)
12. Manjarrez-Sanchez, J., Martinez, J., Valduriez, P.: Efficient Processing of Nearest Neighbor Queries in Parallel Multimedia Databases. In: Bhowmick, S.S., Küng, J., Wagner, R. (eds.) DEXA 2008. LNCS, vol. 5181, pp. 326–339. Springer, Heidelberg (2008)
13. Jacox, E.H., Samet, H.: Spatial join techniques. ACM Transactions on Database Systems 32(1) (2007)
14. Wu, Z., Cao, Z., Wang, Y.: Multimedia selection operation placement. Multimedia Tools and Appl. 54(1), 69–96 (2011)
15. Klampanos, I.A., Jose, J.M.: An Evaluation of a Cluster-Based Architecture for Peer-to-Peer Information Retrieval. In: Wagner, R., Revell, N., Pernul, G. (eds.) DEXA 2007. LNCS, vol. 4653, pp. 380–391. Springer, Heidelberg (2007)
16. Shen, H.T., Huang, Z., Cao, J., Zhou, X.: High-dimensional indexing with oriented cluster representation for multimedia databases. In: Proceedings of the IEEE International Conference on Multimedia and Expo, ICME 2009, pp. 1628–1631. IEEE (2009)
17. Hellerstein, J.M.: Generalized Search Tree. In: Encyclopedia of Database Systems, pp. 1222–1224. Springer, US (2009)

Client-Driven Price Selection for Scalable Video Streaming with Advertisements*

Musab S. Al-Hadrusi and Nabil J. Sarhan

Department of Electrical and Computer Engineering, Wayne State University
5050 Anthony Wayne Drive, Detroit, MI 48202, USA
{hadrusi,nabil}@wayne.edu

Abstract. This paper considers the scalable delivery framework of streaming video content with advertisements. In this framework, the revenues generated from the ads are used to subsidize the cost and thus attract more clients. We analyze a predictive scheme that provides clients with multiple price options, each with a certain number of expected viewed ads. The price depends on the royalty fee of the requested video, its delivery cost based on the current system state, the applied scheduling policy, and the number of viewed ads. The price is lower when the number of viewed ads is larger.

Keywords: Media streaming, periodic broadcasting, pricing, scheduling, stream merging, waiting-time prediction.

1 Introduction

Multicast-based delivery of video streams can be achieved by stream merging [15,13,19,8] (and references within) or periodic broadcasting [20,9,21] (and references within). Stream merging reduces the delivery cost by aggregating clients into larger groups that share the same multicast streams while periodic broadcasting divides each media file into multiple segments and broadcasts each segment periodically on dedicated server channels. Successful video streaming services require incorporating effective business models. A plurality of business models are currently being utilized [14]. In this paper, we use advertisements for subsidizing the cost of premium content, such as new movies.

Incorporating advertisements in the multicast-based approach is challenging because requests (and/or streams) are aggregated to reduce the overall delivery costs. This paper utilizes the recently proposed framework for scalable delivery of media content with advertisements [2], which combines the benefits of stream merging and periodic broadcasting. Video ads are delivered on dedicated broadcasting channels, whereas stream merging is used for the primary video content. A client starts by joining an ads' broadcast channel for some time and then receives the requested video by stream merging. In [17], a waiting-time prediction algorithm has been proposed to estimate the ads' viewing period based on the requested video and the system's current state. Thus, a client is presented with the expected ads' viewing time and the associated price for streaming

* This work is supported in part by NSF grant CNS-0834537.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 429–439, 2012.

the requested video. The revenues generated from the ads are used to subsidize the price and thus attract more clients. Clients with larger ads' viewing time get lower prices. The algorithm considers the dynamic and complex natures of stream merging and request scheduling. The main limitation of that work is that the client is presented with only one choice of price and associated ads' viewing time.

This paper analyzes a solution that provides clients with the opportunity for selecting from multiple price options, thereby providing a better customer-centric and profit-making solution. The proposed *Client-Driven Price Selection* (CPS) approach provides clients with multiple price options, each with a certain number of expected viewed ads. The price depends on the royalty fee of the requested video, the video delivery cost (to be computed dynamically based on the current system state because of request aggregation), and the number of viewed ads (and thus the revenue generated from these ads). We modify the waiting-time prediction algorithm in [17] and request scheduling in [2] to allow for multiple options and then study the effectiveness of the overall solution in detail. As in prior work, we assume the system uses a mechanism to ensure that the clients do actually watch the ads (such as requiring their input to advance to the next ad, using interactive ads, etc.).

We study the effectiveness of the CPS approach and various scheduling policies through extensive simulation in terms of numerous performance metrics. These metrics include *waiting-time prediction accuracy*, *percentage of clients receiving waiting times*, *client defection* (i.e., turn-away) probability, *average waiting time*, *price*, *arrival rate*, and *profit*. The defection probability is the probability that customers defect because of waiting times exceeding their tolerance. The average deviation between the expected and actual times of service is used as a measure of accuracy. The accuracy decreases with the deviation. The reported average waiting time includes the time spent viewing ads and the initial waiting time to receive the ads. We consider the impact of various design and workload parameters. We combine the equation-based [2] and willingness-based [17] arrival rate models to assess the impacts of the price, purchasing capacity, and defection probability on the effective arrival rate. To account for the different ways the clients may select from the available options that meet their purchasing capacity (based on purchasing capacity and willingness model), we experiment with a variety of price selection criteria: *Random Price*, *Lowest Price*, *Median Price*, and *Highest Price*.

The rest of the paper is organized as follows. Section 2 discusses background information and related work. Section 3 presents the proposed approach. Section 4 discusses the performance evaluation methodology. Section 5 presents and analyzes the main results. Finally, conclusions are drawn.

2 Background Information

Resource sharing techniques [3,4,7,10,16,11,15] face the scalability challenge of multimedia streaming systems by utilizing the multicast facility. Stream merging techniques are resource sharing techniques that combine streams when possible to reduce the delivery cost. *Earliest Reachable Merge Target* (ERMT) [5,6] is a near optimal hierarchical stream merging technique, which allows stream to merge multiple times. A new client joins the closest reachable stream (target) and receives the missing portion by a new stream. After the merger stream finishes and merges into the target, the latter can get extended to satisfy the playback requirement of the new client(s), and this extension can affect its own merge target.

For stream merging, a scheduling policy is used to select a video to service when a *channel* becomes available. A channel is a set of resources needed to deliver a video stream. All requests in the selected video can be serviced using only one channel. The number of channels is called *server capacity*.

The main scheduling policies include *First Come First Serve* (FCFS) [4], *Maximum Queue Length* (MQL) [4], *Maximum Factored Queue Length* (MFQL) [1], and *Minimum Cost First* (MCF) [19]. MCF achieves the best overall performance by capturing the significant variation in stream lengths caused by stream merging. *MCF-P*, which is the preferred implementation, selects the video with the least cost per request.

For supporting video streaming with advertisements, a scalable delivery framework was proposed in [2]. This framework has the following main characteristics. (1) Clients start by joining an ads' broadcast channel for some time and then receive the requested video by stream merging. (In this paper, "requested video", "actual video", or "video", unless otherwise indicated, refer to one of a primary videos and not an ad.) (2) Ads are combined and broadcast on dedicated server channels. Hence, when beginning to listen to an ads' channel, the client views different ads until streaming of the desired video finishes. Multiple channels can be used with time-shifted versions of the combined ads, as shown in Figure 1, to reduce the waiting time for reaching the beginning of an ad. With N_{adCh} channels, the maximum value of this time is ad_len/N_{adCh}, where ad_len is one ad length. (3) Ads are only viewed prior to watching the actual video to allow for uninterrupted viewing and more enjoyable playback experience.

For the scalable delivery framework, two modified versions of MCF-P scheduling policy were proposed in [2] to ensure that ads are viewed by a large number of clients: *Any N* and *Each N*. *Each N* considers a video for scheduling only if each waiting request for it has viewed at least N ads, whereas *Any N* considers a video for service only if any one of its waiting requests has viewed at least N ads.

3 Client-Driven Price Selection

This paper explores the idea of presenting each client with multiple price options. Providing clients with multiple price options enhances customer satisfaction and system profitability. The proposed *C̲lient-Driven P̲rice S̲election* (CPS) approach is targeted for the scalable video delivery framework with advertisements [2]. This approach performs waiting-time prediction and provides the client with the list of expected waiting times (or times of service) and their associated prices. The waiting-time prediction is done by modifying the prediction algorithm in [17]. The waiting-time is essentially the total time between the request arrival and the commencement of streaming of the requested video.

The idea of the CPS algorithm can be described as follows. When a new request is made, it is mapped to the ads' channel with the closest ad start (or end) time. The algorithm examines the ad start times on that channel in the order of closeness to the current time for possible assignment as the expected time for that request. Because only multiple ads can be viewed by clients, the later ad start times represent the different possible service times. At each ad start time, the server estimates the number of available channels and predicts the videos that can be serviced at that time. When the requested video is the expected video to be serviced, the corresponding ads' viewing time and associated price is added to the list of choices to present to the client. The process continues until, the *prediction window* (Wp) is exceeded. This window provides a tradeoff between the implementation complexity and the number of price options the client receives as well a tradeoff between the prediction accuracy and the percentage of clients receiving expected times. If the video prediction does not return any expected time, the average waiting time for the requested video is used instead.

Figure 1 illustrates how CPS generally works with $W_p = 3$ ad lengths. When a new client makes a request at time T_{Now}, the algorithm examines the ad start times within the prediction window in the order T_0, T_1, T_2, and T_3. For each one of these times, if the requested video is selected in the prediction, the corresponding ads' viewing time and price is added to the list of client's choices. The list for example may include (1 ad, $\$2.2$) and (3 ads, $\$1.8$), assuming that the video is selected at times T_1 and T_3. The server has to make sure that the client will view at minimum the selected number of ads by placing an additional constraint on the scheduling policy.

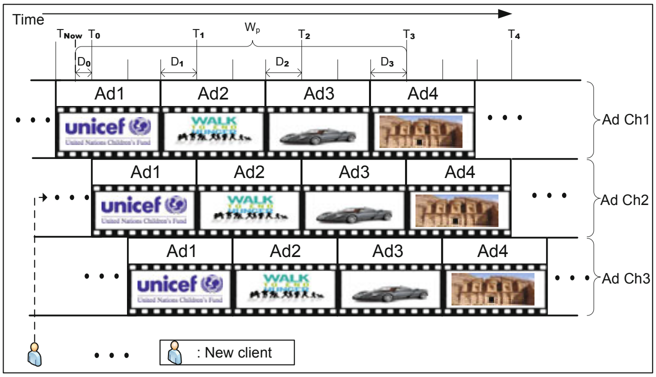


Fig. 1. Illustration of the CPS Algorithm

Figure 2 shows a simplified version of the algorithm, which is performed upon the arrival of request R_i to video v_j. This algorithm extends the algorithm proposed in [17] to allow multiple price options. Providing multiple price options is enabled by continuing to provide expected ads' viewing times and associated prices to the client even after the prediction algorithm determines an expected time for service for that request. Thus, if the request is expected to be serviced at time T_i, the CPS algorithm returns an expected number of ads and a matching price to the client and continues to find other expected service times until the prediction window is exceeded. When determining the latter expected numbers of ads and associated prices, the CPS algorithm acts as if the request has not been serviced earlier. As discussed earlier, the implementation of the scheduling policy (not shown in the Figure) has also to be changed in order to enforce that the client views the number of ads that he/she selected earlier. To make the paper as

```
for $(v = 0; v < N_v; v++)$ // Initialize
  $assigned\_time[v] = -1;$ // Not assigned expected time
$adChNo$ = Get label # of the ads' channel with the closest start time;
$T$ = Get next ad's start time; $T_0 = T$; $examined\_times = 0;$
// Find number of available channels at time T
$N_c = available\_channels + will\_be\_available(T_{Now}, T);$
while $(T < T_{Now} + W_p)$ { // Loop till prediction window is exceeded
  for $(v = 0; v < N_v; v++)\{$
    if $(isQualified(v, T, adChNo))$ {
      if $(assigne\_time[v] == -1)$
        $expected\_qlen = qlen(v, adChNo) + \lambda[v]\times$
        $((T - T_{Now}) + examined\_times \times ad\_len/N_{adCh});$
      else // Video $v$ has been assigned an expected time
        $expected\_qlen = \lambda[v] \times (T - assigned\_time[v])/N_{adCh};$
      $objective[v]$ = find scheduling objective for video v;
    } // end if $(isQualified(v, T, adChNo))$
    else $objective[v] = -1;$ // v is not qualified
  } // end for $(v = 0; v < N_v; v++)$
  while $(c = 0; c \leq N_c; c++)\{$ //for every available channel
    // Find the expected video to serve at time T
    $expected\_video$ = find video with maximum nonzero objective;
    if$(expected\_video == v_j)$ {
      Push $T$ into expected time que $Que_{ET}$ of request $R_i$;
      TempPrice = CalculatePrice(T,$v_j$);
      Push TempPrice into price que $Que_{price}$ of request $R_i$;
      break; }
    else { $assigned\_time[expected\_video] = T;$
        $objective[v] = -1;$ } // -1 means can't be selected again
  } // end while $(c = 0; c \leq N_c; c++)$
  $T = T + ad\_len;$ //Proceed to the next edge
  // Find number of available channels at time T
  $N_c = left\_over + will\_be\_available(T - ad\_len/N_{adCh}, T);$
  $examined\_times++;$
} // end while $(T < T_{Now} + W_p)$
Present $Que_{Price}$ to client and then clear it.
```

Fig. 2. Simplified Algorithm for CPS [performed upon arrival of request R_i to Video v_j]

self-contained as possible, let us briefly discuss how to predict the videos to be served at any particular ad start time T. First, the algorithm determines the videos that will be qualified at that time based on any minimum ads' viewing constraints, imposed by Any N or Each N. For each video that qualifies, the algorithm estimates its waiting queue length at time T, based on the video arrival rates, which are to be computed periodically but not frequently. As in [17], the expected queue length for video v at time T can be found as follows:

$$expected\_qlen[v] = qlen(v, adChNo) + \lambda[v] \times ((T_0 - T_{Now}) + examined\_starts \times ad\_len/N_{adCh}), \quad (1)$$

where $qlen(v, adChNo)$ is the queue length of video v on channel $adChNo$ at the current time (T_{Now}), $\lambda[v]$ is the arrival rate for video v, ad_len is the ad length, N_{adCh} is the number of ads' channels, T_0 is the nearest ad start time, and $examined\_starts$ is the number of examined ad start times so far during the current run of the prediction algorithm. Equation (1) assumes that video v has not been identified before as the expected video to be serviced at an earlier ad start time. Otherwise, the expected arrivals will have to be found during the time interval between T and the latest assigned time ($assigned\_time[v]$) at which video v is expected to be served [17]. In Figure 2, N_c represents the number of available server channels at ad start time T. This number is equal to the number of channels that will be available prior to T plus a left-over value for the number of unused channels at prior ad start times [17].

The pricing depends on the royalty fee of the requested video, the video delivery cost (to be computed dynamically based on the current system state because of request aggregation), and the number of viewed ads (and thus the revenue generated from these ads). Note that the delivery cost varies with the current access rate. Although the prediction algorithm is highly accurate as will be shown later, clients who view more ads than expected should be provided with compensation credits, which can be used to reduce the number of ads to be viewed when accessing other primary media later on.

4 Performance Evaluation Methodology

Table 1 summarizes the workload characteristics. Like most prior studies, we generally assume that the arrival of the requests to the server follows a Poisson Process with an average arrival rate λ. Hence, the inter-arrival time is exponentially distributed with a mean $T_{avg} = 1/\lambda$. Additionally, we assume that the access to videos is highly localized and follows a Zipf-like distribution. With this distribution, the probability of choosing the n^{th} most popular video is $C/n^{1-\theta}$ with a parameter θ and a normalized constant C. The parameter θ controls the skew of video access. Note that the skew reaches its peak when $\theta = 0$, and that the access becomes uniformly distributed when $\theta = 1$. We assume a value of 0.271 [18].

The waiting tolerance of clients is characterized as follows: a client who chooses a certain expected time of service (and thus an associated number of expected ads and a price) waits for service until that expected times plus an added tolerance value Wad (a variable from 0 to 9 ad lengths with default value of 2). The waiting tolerance of all other clients follows an exponential distribution with mean μ_{tol}.

Estimating the overall revenue and profit is challenging because the price influences the arrival rate and number of streams delivered. Subsidizing the price can attract more clients and can eventually increase the overall revenue and profit. By increasing the arrival rate, the delivery costs also decrease because of the higher degrees of stream merging. We combine *equation-based* and *willingness-based* arrival rate models. In the first, the arrival rate changes dynamically based on the defection probability [2]. The defection probability is the probability that clients leave without being serviced because of waiting times exceeding their tolerance. We experiment with the following function:

$$\lambda = \frac{c_1(1-d)}{c_2 + c_3 d^2} \quad (2)$$

Table 1. Summary of Workload Characteristics

Parameter	**Model/Value(s)**
Request Arrival	Poisson Process
Request Arrival Rate	Variable, Default = 40 Requests/min
Server Capacity	200-550
Video Access	Zipf-Like, Skew Parameter $\theta = 0.271$
Movie-Related Characteristics	80 120-min movies
Waiting Tolerance Model for clients without expected times of service	Poisson, min= 3 ads, mean= 5 ads, max= 8 ads
Waiting Tolerance Model for clients with expected times of service	Expected Service Time + Wad, Wad: Variable, Default= 2 Ad lengths
Ad-Related Characteristics	Ad Length= 30 sec, # different ads= 8, # ads channels= 3
Minimum Ads Constraint (N)	Variable, Default = 2
Prediction Window (W_p)	Variable, Default = 9
Qualification Threshold (Q_{Th})	Variable, Default = 0.25
Scale (b), Shape (α), Elasticity (δ)	Variables, Defaults= 1.0, 1, 7, resp.
Equation-Based Model Constants	$c1 = 60$, $c2 = 0.5$, $c3 = 1$

where d is the defection probability, and c_1, c_2, and c_3 are constants. In the second model, we utilize client purchasing capacity and willingness models. The capacity of a client to spend for a particular service or product can be modeled using Pareto distribution [12]. The Pareto probability density function can be given as follows in equation (3):

$$f_p(x) = \alpha \times b \times x^{-(\alpha+1)} \quad for \quad x \geq b, \tag{3}$$

where b (also called *scale*) represents the minimum value of x, and α represents the shape of the distribution. Most clients have capacities close to b. The distribution is more skewed for larger values of α. Hence, as α increases, fewer clients can pay much more than b. Clients with larger capacities are more likely to spend more. The willingness probability of a client with capacity y to pay for a product or service with price p can be given by

$$Prob(willingness) = \begin{cases} 1 - (\frac{p}{y})^{\delta} & 0 \leq p \leq y \\ 0 & p > y, \end{cases} \tag{4}$$

where δ is the *elasticity* [12]. As δ increases, more clients are willing to spend.

In prior work, the equation-based model [2] and the willingness-based model [17] were used separately. In this paper, we combine these two models as they are not exclusive. The equation-based model captures the impact of the current defection probability on the future arrival rate, whereas the equation-based model captures the impacts of the price and purchasing capacity on the willingness of the client to purchase the streaming service. Therefore, each one of these two models impacts the arrival rate in a different way. Figure 3 illustrates how the two models are combined.

We consider a commercial *Movie-on-Demand* system. Without loss of generality, we assume a *cost-plus* pricing. The price in the considered system covers the movie royalty fee, delivery fee, and operational cost minus subsidization credit. In the discussed

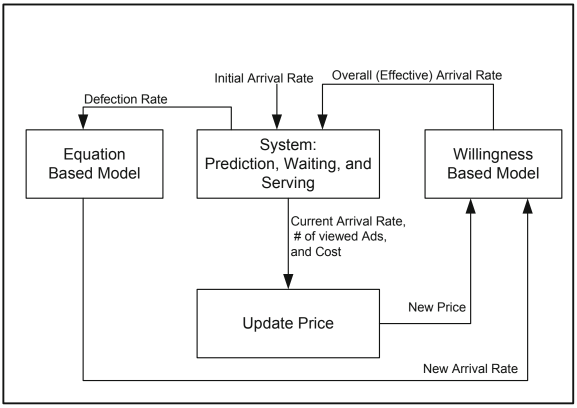

Initial Arrival Rate
Overall (Effective) Arrival Rate
Defection Rate
Equation Based Model
System: Prediction, Waiting, and Serving
Willingness Based Model
Current Arrival Rate, # of viewed Ads, and Cost
Update Price
New Price
New Arrival Rate

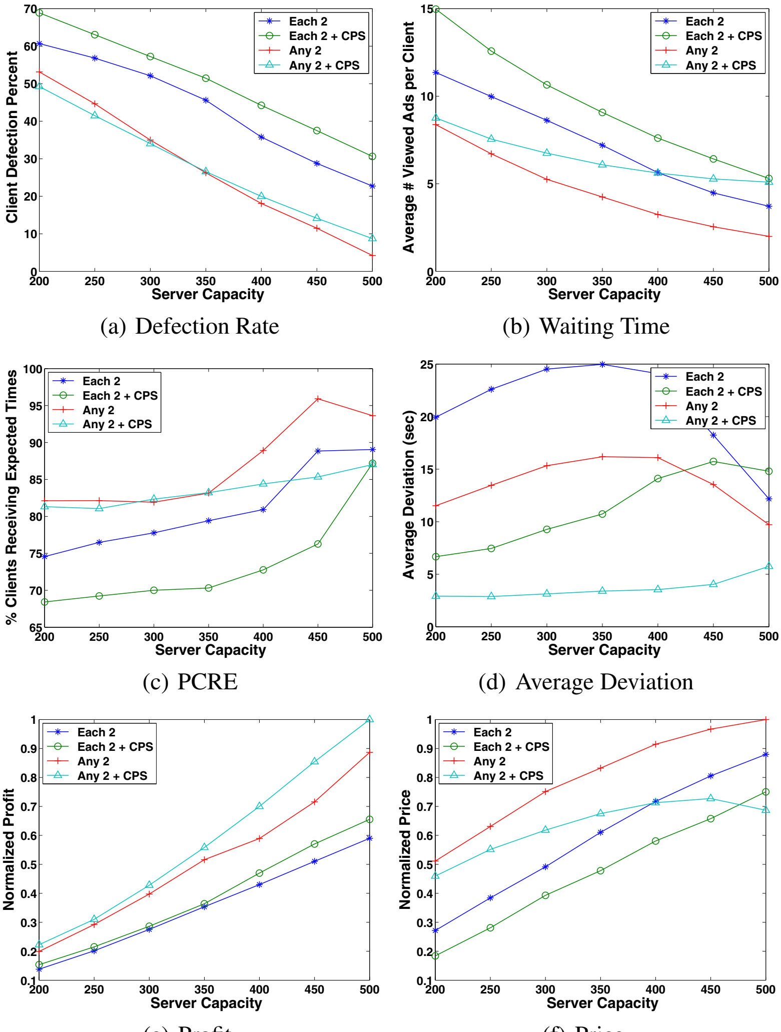
Fig. 3. Combining the Equation-Based and Willingness-Based Models

example, the revenue per ad per client is 10 cents, the movie royalty fee is 70 cents, and the delivery cost per GB is 50 cents. Based on service positioning analysis, the service provider seeks to get 70 cents per movie request to cover their operational cost and attain the sought profit. A fixed fraction of the 70 cents is used as a profit.

5 Result Presentation and Analysis

We conducted extensive simulation using many different workload and system parameters. For space limitation, we only show the main results. Only the results for ERMT are shown because it is the most efficient stream merging policy. We assume that the client selects the lowest price unless otherwise indicated. We also consider the impact of the used price selection scheme. The analyzed **performance metrics** include: *waiting-time prediction accuracy*, *percentage of clients receiving waiting times* (PCRE), *client defection* (i.e., turn-away) probability, *average number of viewed ads*, *price*, *arrival rate*, and *profit*. The average deviation between the expected and actual times of service is used as a measure of accuracy. The accuracy decreases with the deviation. The waiting time includes the time spent viewing ads and the initial waiting time to receive the ads. Profit is considered as the most important metric. All other metrics influence the profit. The average number of viewed ads, prediction accuracy, and PCRE are important measures of Quality-of-Service (QoS).

Figure 4 illustrates the effectiveness of CPS for "Any 2" and "Each 2" Scheduling with different server capacities. The value 2 is selected optimally based on extensive analysis. Note that increasing this value increases the ads' viewing time but leads to higher underutilization of resources. The results show that CPS may slightly increase the defection rate, but this is due to the higher arrival rate achieved by accepting more client requests, as shown in Figure 5(a). Figure 5(b) shows the same but with different prediction windows. CPS increases the profit for both scheduling polices. "Any 2" Scheduling with CPS achieves the best overall performance. The average deviation between the predicted and actual waiting times is always less than one ad length and is



(a) Defection Rate

(b) Waiting Time

(c) PCRE

(d) Average Deviation

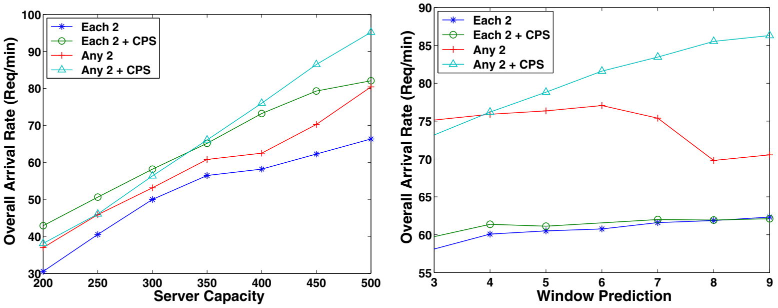
(e) Profit

(f) Price

Fig. 4. Effectiveness of the Proposed CPS Scheme [ERMT, Least Price Selection]

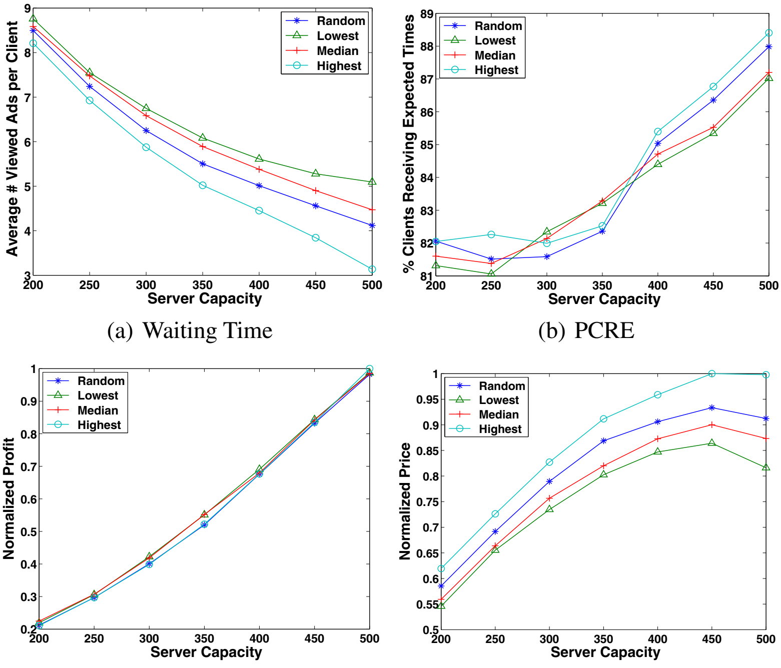
within 6 seconds with the best scheduling policy. The percentage of clients receiving expected times (PCRE) is always larger than 67%.

Figure 6 demonstrates the impact of various price selection schemes: random, lowest, median, and highest. Note that there is a tradeoff between price and waiting time. The price is lower with a larger waiting time. In reality, different clients will have different criteria, and this motivates investigating random selection. Interestingly, although these schemes achieve significant variation in waiting time and price, they perform closely in terms of the profit.



(a) Server Capacity (b) Window Prediction

Fig. 5. Overall Arrival Rate [ERMT, Any 2, CPS, Least Price Selection]



(a) Waiting Time (b) PCRE

(c) Profit (d) Price

Fig. 6. Impact of Various Price Selection Schemes

6 Conclusions

We have analyzed a predictive scheme, called *Client-Driven Price Selection* (CPS), which provides clients with multiple price options, each with a certain number of expected viewed ads. The main results can be summarized as follows. (1) The proposed CPS approach enhances the revenue and profit by giving multiple price choices to the client, thereby attracting more clients. (2) CPS is best when combined with Any N scheduling and ERMT. (3) The achieved waiting time prediction accuracy with this combination is within 6 seconds (20% of an ad length).

References

1. Aggarwal, C.C., Wolf, J.L., Yu, P.S.: The maximum factor queue length batching scheme for Video-on-Demand systems. IEEE Trans. on Computers 50(2), 97–110 (2001)
2. Al-Hadrusi, M., Sarhan, N.J.: A scalable delivery framework and a pricing model for streaming media with advertisements. In: Proc. of SPIE/ACM Multimedia Computing and Networking Conference, MMCN (Januray 2008)
3. Cai, Y., Hua, K.A.: An efficient bandwidth-sharing technique for true video on demand systems. In: Proc. of ACM Multimedia, pp. 211–214 (October 1999)
4. Dan, A., Sitaram, D., Shahabuddin, P.: Scheduling policies for an on-demand video server with batching. In: Proc. of ACM Multimedia, pp. 391–398 (October 1994)
5. Eager, D.L., Vernon, M.K., Zahorjan, J.: Optimal and efficient merging schedules for Video-on-Demand servers. In: Proc. of ACM Multimedia, pp. 199–202 (October 1999)
6. Eager, D.L., Vernon, M.K., Zahorjan, J.: Minimizing bandwidth requirements for on-demand data delivery. IEEE Trans. on Knowledge and Data Engineering 13(5), 742–757 (2001)
7. Gao, L., Kurose, J., Towsley, D.: Efficient schemes for broadcasting popular videos. In: Proc. of the Int'l Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV) (July 1998)
8. Gill, P., Shi, L., Mahanti, A., Li, Z., Eager, D.: Scalable on-demand media streaming for heterogeneous clients. ACM Transactions on Multimedia Computing, Communications, and Applications (ACM TOMCCAP) 5(1), 1–24 (2008)
9. Hu, A.: Video-on-Demand broadcasting protocols: A comprehensive study. In: Proc. of IEEE INFOCOM (April 2001)
10. Hua, K.A., Cai, Y., Sheu, S.: Patching: A multicast technique for true Video-on-Demand services. In: Proc. of ACM Multimedia, pp. 191–200 (1998)
11. Huang, C., Janakiraman, R., Xu, L.: Loss-resilient on-demand media streaming using priority encoding. In: Proc. of ACM Multimedia, pp. 152–159 (October 2004)
12. Jagannathan, S., Almeroth, K.C.: The dynamics of price, revenue, and system utilization. In: Proc. of the IFIP/IEEE International Conference on Management of Multimedia Networks and Services, pp. 329–344 (2001)
13. Ma, H., Shin, G.K., Wu, W.: Best-effort patching for multicast true VoD service. Multimedia Tools Appl. 26(1), 101–122 (2005)
14. Rayburn, D.: Streaming and Digital Media: Understanding the Business and Technology. Focal Press (2007)
15. Rocha, M., Maia, M., Cunha, I., Almeida, J., Campos, S.: Scalable media streaming to interactive users. In: Proc. of ACM Multimedia, pp. 966–975 (November 2005)
16. Sarhan, N.J., Das, C.R.: Caching and scheduling in NAD-based multimedia servers. IEEE Trans. on Parallel and Distributed Systems 15(10), 921–933 (2004)
17. Sarhan, N.J., Al-Hadrusi, M.: Waiting-time prediction and QoS-based pricing for video streaming with advertisements. In: Proc. of IEEE International Symposium on Multimedia, ISM (December 2010)
18. Sarhan, N.J., Das, C.R.: A New Class of Scheduling Policies for Providing Time of Service Guarantees in Video-on-Demand Servers. In: Vicente, J.B., Hutchison, D. (eds.) MMNS 2004. LNCS, vol. 3271, pp. 127–139. Springer, Heidelberg (2004)
19. Sarhan, N.J., Qudah, B.: Efficient cost-based scheduling for scalable media streaming. In: Proc. of Multimedia Computing and Networking Conf., MMCN (January 2007)
20. Shi, L., Sessini, P., Mahanti, A., Li, Z., Eager, D.L.: Scalable streaming for heterogeneous clients. In: Proc. of ACM Multimedia, pp. 337–346 (October 2006)
21. Tantaoui, M.A., Hua, K.A., Do, T.T.: Broadcatch: A periodic broadcast technique for heterogeneous Video-on-Demand. IEEE Trans. on Broadcasting 50(3) (September 2004)

Efficient Storage and Decoding of SURF Feature Points

Kevin McGuinness, Kealan McCusker, Neil O'Hare, and Noel E. O'Connor

CLARITY: Center for Sensor Web Technologies, Dublin City University

Abstract. Practical use of SURF feature points in large-scale indexing and retrieval engines requires an efficient means for storing and decoding these features. This paper investigates several methods for compression and storage of SURF feature points, considering both storage consumption and disk-read efficiency. We compare each scheme with a baseline plain-text encoding scheme as used by many existing SURF implementations. Our final proposed scheme significantly reduces both the time required to load and decode feature points, and the space required to store them on disk.

Keywords: Image features, interest points, quantization, coding.

1 Introduction

Speeded up robust features (SURF) based descriptors [1] have become very popular in modern computer vision and multimedia information retrieval engines, primarily due to their exceptional performance and a fast method for detection and extraction. For large datasets, SURF features can be computed offline and stored alongside the images they represent, allowing matching to be performed without the need to extract features each time an image is visited.

A linear search for an object in a database of feature points requires, for each image, three operations: 1) loading all feature points for the image from the disk; 2) decoding these feature points; and 3) computing the distance between all feature points in the query object to all feature points in the target image. Euclidean distance is generally used in practice, and can be implemented using a fixed number of primitive (add, subtract, square) machine operations. As such, the time required to search a large dataset can be dominated by the time spent loading and decoding feature points if the amount of memory required to represent these feature points exceeds the memory capacity of the hardware.

Storage space for feature points also requires consideration. Typical image files, efficiently compressed, require only a few tens or hundreds of kilobytes of disk storage. Depending on the image and parameters specified for the feature extraction algorithm, each image can produce hundreds or even thousands of feature points. A naïve scheme for storing these feature points (e.g. XML) could require significantly more storage space than the image file itself. Indeed, one XML based storage scheme we investigated required over 13GB to store the

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 440–451, 2012.

feature points for 10,000 images; the images required only a single gigabyte of storage.

Visual feature vector compression has previously been considered by Takacs et al. [15], who used entropy coding of quantized feature vectors to reduce the size of feature vectors to approximately 5 bits per coefficient, in their mobile augmented reality system. They subsequently proposed transform coding of SURF and SIFT [11] components using the Karhunen-Loeve transform, followed by quantization and entropy coding, achieving very high compression rates of approximately 2 bits per coefficient [3]. They later achieved even higher compression rates using a compressed histogram of gradients descriptor [2].

While other authors have proposed very effective schemes for compressing SURF feature vectors, these works were focused on low-bitrate descriptors; the objective of this paper is to devise a storage scheme for SURF feature points that is efficient in terms of both storage space and time required to read and decode the features from a disk. This compression scheme, like the scheme used by Takacs et al. [15], is based on quantization and entropy coding of the coefficients. While Takacs et al. simply noted the use of entropy coded SURF features as part of a larger system, we present a more in-depth discussion on how to implement such an encoding scheme, and an analysis of the effect of quantization on the distance between pairs of descriptors. We also present an empirical analysis of decode time, storage requirements and accuracy using the BelgaLogos dataset [9].

It is worth mentioning that exhaustive search is usually impractical for large-scale online image search systems where the query images are not known a-priori. Visual bag of words (BoW) and bag-of-features (BoF) based techniques (e.g. [13,12,6]) are generally superior for these systems, in terms of both query time and storage requirements, since only vectors of codewords need to be retained for each image, rather than the individual descriptors. Such systems can, in addition, benefit from BoF based compression techniques such that proposed by Jègou et al. [8] to reduce the dimensionality of the codeword vectors, allowing for very compact image descriptors. Nevertheless, storing the individual descriptors is still important for a variety of applications. Search and indexing systems may benefit from storing the descriptors during an intermediate step in the indexing chain, to be processed at a later stage by other modules like visual word extraction or offline indexing. Mobile applications may be required to transmit descriptors over low-bandwidth connections for cloud based matching and detection. Retaining individual descriptors can also be useful for establishing the precise location of matched objects in images.

The remainder of the paper is organized as follows. Section 2 gives an overview of the composition of a SURF feature point and introduces notation for the components. Section 3 describes the dataset we used for analysis and experimentation. Section 4 examines the effect of quantizing SURF coefficients on the square distance between pairs of coefficients. Section 5 discusses quantization strategies, and Section 6 discusses entropy coding of the quantized coefficients. Section 7 evaluates the proposed encoding schemes, and Section 8 concludes the paper.

2 SURF Feature Points

A SURF feature point f is composed of the (x, y) location of the point, a scale value s, an orientation value θ, a Laplacian value $l \in \{-1, +1\}$, and 64 coefficients that make up the descriptor. The coefficients are summations over the horizontal and vertical Haar wavelet filter responses in the $4 \times 4 = 16$ blocks surrounding the feature point. There are four coefficients extracted for each block: $f_1 = \sum d_x$, $f_2 = \sum d_y$, $f_3 = \sum |d_x|$, and $f_4 = \sum |d_y|$, where d_x and d_y are the horizontal and vertical filter responses at each pixel inside a block, and the summations apply over all pixels in the block.

3 BelgaLogos Dataset

The BelgaLogos dataset [9] is composed of 10,000 images manually annotated for 26 logos and trademarks, with 55 images from the dataset provided as query images for these logos. This makes it ideal for evaluating image and feature point matching techniques. We extracted a total of 5,966,580 feature points from these images[1], a mean of 596.7 per image. The range, interquartile range, mean, median, and standard deviation of the number of feature points per image are:

min.	Q.1	median	mean	Q.3	max.	sd.
0	333	517	596.7	756	3938	377.8

The minimum of zero in the above is an outlier – only one image has zero feature points associated with it.

To perform a statistical analysis of the feature points, we randomly sampled 100,000 feature points from the 5,966,580 in the dataset. Figure 1 shows the distribution of feature point components $f_1 \dots f_4$ for all feature points in the sample set. The f_1 and f_2 components for the sample are in the range $[-0.5, 0.5]$; the f_3 and f_4 are in the range $[0, 1]$. The observed ranges and mean values of the coefficients are:

	f_1	f_2	f_3	f_4
mean	-0.0000235	0.0213	0.109	0.154
min	-0.3709430	-0.408639	0.0	0.0
max	0.4143080	0.460317	0.64189	0.74343

The orientation value are in the interval $[0, 2\pi]$, with modes $\{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi\}$. The scale values in the sample set are in the interval $[1.601, 22.680]$.

[1] The number of features extracted is governed by the Hessian threshold, which we set to 0.001.

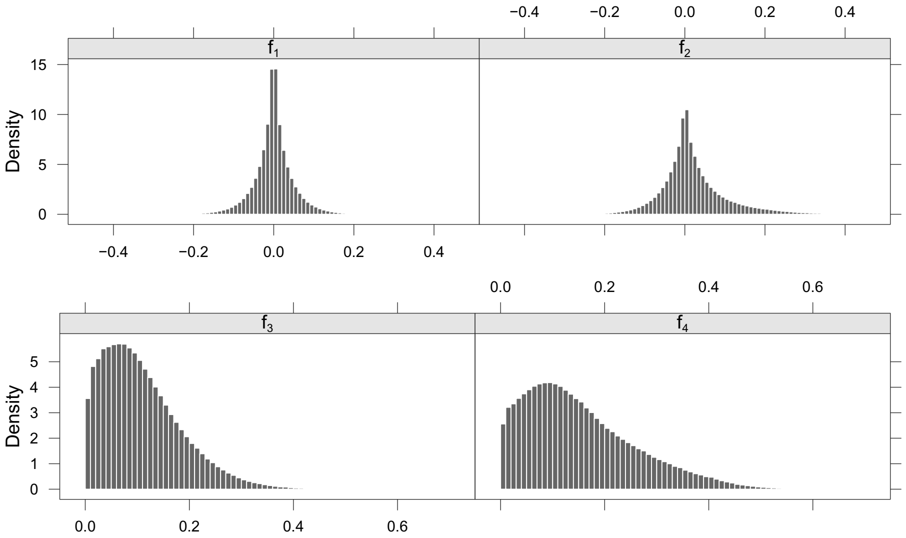


Fig. 1. Histograms showing the distribution of feature point descriptor coefficients (f_1, f_2, f_3, f_4)

4 Coefficient Precision

Matching comparable SURF features (i.e. those with similar scales and Laplacian values) requires computing the Euclidean distance between the coefficient vectors for pairs of descriptors. Common criteria for deciding if two feature vectors match are [14]: 1) comparing against a simple threshold, 2) finding the k nearest neighbors feature vectors and comparing their distance against a threshold, and 3) comparing the the ratio between the nearest and second nearest neighbors (the *nearest neighbor distance ratio*) with a threshold. Reducing the precision of the SURF coefficients allows them to be stored using less bits. Ideally, we would like to reduce the precision so that it does not have a significant affect on the distance between any two vectors. Here we look at the effect of reducing the precision of the coefficients on the distance computation.

Let $\mathbf{u}$ and $\mathbf{v}$ be the coefficient vectors for two SURF feature points, with $\mathbf{v} = (v_1, v_2, \ldots, v_{64})$, and $\mathbf{u} = (u_1, u_2, \ldots, u_{64})$. The square distance between these vectors is:

$$\delta^2(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^{64} (u_i - v_i)^2. \tag{1}$$

Any loss in precision can introduce a small error component at each value in $\mathbf{u}$ and $\mathbf{v}$. Let the maximum error introduced by encoding the coefficients u_i and

v_i be equal to ϵ. In the worst case, this error value can affect each term in the distance formula by 2ϵ, giving a square distance of:

$$\delta^2(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^{64} (u_i - v_i + 2\epsilon)^2. \tag{2}$$

Substituting $\mathit{\Delta}_i = u_i - v_i$ in the above and multiplying out gives:

$$\delta^2(\mathbf{u}, \mathbf{v}) = \sum_{i=1}^{64} \mathit{\Delta}_i^2 + \left[4\epsilon \sum_{i=1}^{64} \mathit{\Delta}_i + 256\epsilon^2 \right]. \tag{3}$$

The first part is just the square distance, so the worst-case error in the square distance computation that results from an error ϵ in each component is:

$$E_{\text{worst-case}}(\epsilon, \mathit{\Delta}) = 4\epsilon \sum_{i=1}^{64} \mathit{\Delta}_i + 256\epsilon^2. \tag{4}$$

Recall that the SURF coefficients f_1 and f_2 are in the interval $[-0.5, 0.5]$ and the coefficients f_3 and f_4 are in the range $[0, 1]$. The maximum distance between any two SURF coefficients is therefore 1, and so the largest value of the summation in the above is 64 (since, $\max \sum_i^{64} \mathit{\Delta}_i = 64$). Substituting this into Eq. (4) gives a general worst-case error:

$$E_{\text{worst-case}}(\epsilon) = 256 \times \epsilon(\epsilon + 1). \tag{5}$$

To keep the square distance error δ_ϵ^2 below some value, we simply choose sufficient digits of precision so that ϵ satisfies:

$$256\epsilon^2 + 256\epsilon < \delta_\epsilon^2. \tag{6}$$

The maximum allowable value of ϵ for a given δ_ϵ^2 can be found by solving the above quadratic inequality and inferring the required precision. The worst case square error in terms of the number of bits of binary precision of each component can be derived by substituting the machine unit roundoff error $\epsilon = 2^{-p}/2$ for p bits of precision in Eq. (5).

$$E_{\text{worst-case}}(p) = 256 \times \frac{2^{-p}}{2} \left(\frac{2^{-p}}{2} + 1 \right) \tag{7}$$

$$= 2^{6-2p} + 2^{7-p}. \tag{8}$$

Usually the value of $k = \sum_{i=1}^{64} \mathit{\Delta}_i$ from Eq. (4) will be significantly less than its worst-case value of 64, which only occurs when the distance between each of the individual components is maximal. It is clear from Figure 1 that the coefficient values are usually close to zero, and so the expected value of k is small. Moreover, the threshold on the Euclidean distance $\delta(\mathbf{u}, \mathbf{v})$ used to match SURF features is usually low. Therefore, although in the worst-case $k = 64$, in the average case – and for cases that will affect the outcome of a whether two are judged to match – $k \ll 64$.

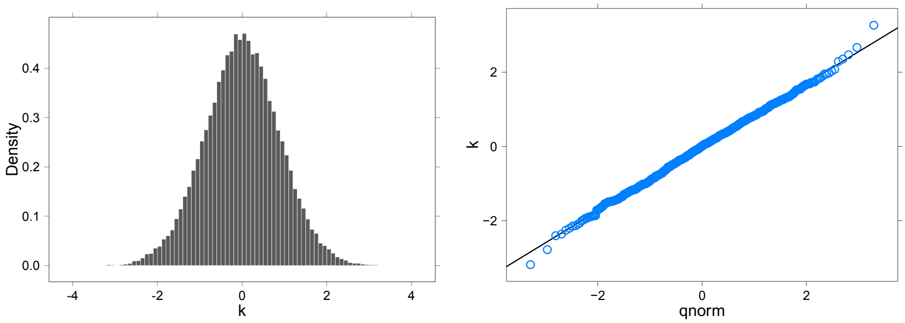


Fig. 2. Histogram (left) showing the distribution of $k = \sum_{i=1}^{64} \Delta_i$ for a sample of 50,000 pairs of feature points. Q-Q plot (right) showing that it is reasonable to assume that k is normally distributed

Figure 2 (left) shows the distribution of k for a random sample of 50,000 pairs of feature points, and Fig. 2 (right) shows the normal Q-Q plot of 5,000 of these values. It is clear from these figures that the distribution of the sample is very close to normal, and it is reasonable to assume that the value of k in general is also normally distributed. The mean and standard deviation were found to be zero (± 0.0077 with 95% confidence) and 0.88 (c.i. $(0.875, 0.887)$ with 95% confidence). Assuming that the sample is representative, this implies that 75% of all values of k are in the $(-1, +1)$ interval, 97.6% are in the $(-2, +2)$ interval, and 99.9% are in the $(-3, +3)$ interval. The upper bound for error given by $k = 64$ in Eq. 8 is therefore unrealistically high in practice; setting $k = 4$ gives a practical upper bound on the error, while setting $k = 1$ gives the square distance error in the majority of cases. The general formula for square distance error in terms of k given p bits of precision is:

$$E(p, k) = 2^{6-2p} + 2^{1-p} k. \tag{9}$$

Table 1 shows the error values for varying amounts of machine precision and values of k. The first column is the number of bits p of machine precision; the second is the roundoff error for p bits of precision; the remaining columns show the effect of this error on the square distance in the worst and typical cases. The last row, $p = 23$, corresponds to IEEE-754 single precision floating point numbers. The table shows that the error incurred by encoding each coefficient using 16 bits of precision is likely to be acceptable. The error incurred by using 8 bits of precision may also be acceptable, and is worth investigating.

5 Quantizing Coefficients

The discussion in the previous section on the effect of rounding error on the outcome of a distance calculation indicates that encoding the coefficients with 16 or possibly even 8 bits of precision should not significantly affect the outcome.

Table 1. Worst-case and typical error values for varying amounts of machine precision

p	$\epsilon = 2^{-p}/2$	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$E_{\text{worst-case}}$
7	0.00390625	0.01953125	0.03515625	0.05078125	0.06640625	1.00390625
8	0.00195312	0.00878906	0.01660156	0.02441406	0.03222656	0.50097656
9	0.00097656	0.00415039	0.00805664	0.01196289	0.01586914	0.25024414
10	0.00048828	0.00201416	0.00396729	0.00592041	0.00787354	0.12506104
12	0.00012207	0.00049210	0.00098038	0.00146866	0.00195694	0.03125381
14	0.00003052	0.00012231	0.00024438	0.00036645	0.00048852	0.00781274
16	0.00000763	0.00003053	0.00006105	0.00009157	0.00012209	0.00195314
18	0.00000191	0.00000763	0.00001526	0.00002289	0.00003052	0.00048828
20	0.00000048	0.00000191	0.00000381	0.00000572	0.00000763	0.00012207
22	0.00000012	0.00000048	0.00000095	0.00000143	0.00000191	0.00003052
23	0.00000006	0.00000024	0.00000048	0.00000072	0.00000095	0.00001526

Since all the coefficients f have a range $r = |\max f - \min f| \leq 1$, they can be remapped so that they lie in the interval $[0, 1]$. We therefore only need to encode the significand (mantissa) of the floating point number – all the bits can be used for precision.

A rigorous description of the quantization scheme requires us to define rounding and quantization functions. The rounding function $r\colon \mathbf{R} \to \mathbf{Z}$ returns the nearest integer to x, rounding ties away from zero:

$$r(x) = \begin{cases} \lfloor x + 0.5 \rfloor & x \geq 0 \\ \lceil x - 0.5 \rceil & x < 0. \end{cases} \tag{10}$$

The clamp function $c\colon \mathbf{R} \to \mathbf{R}$ limits the value of x to a given interval $[x_1, x_2]$:

$$c(x; x_1, x_2) = \max\{\min\{x, x_2\}, x_1\}. \tag{11}$$

The linear remapping function $m\colon \mathbf{R} \to \mathbf{R}$ maps values that lie in the interval $[x_1, x_2]$ to values in the interval $[y_1, y_2]$ as follows:

$$m(x; x_1, x_2, y_1, y_2) = y_1 + \frac{(y_2 - y_1)(x - x_1)}{x_2 - x_1}. \tag{12}$$

Using the above definitions, we can define the p bit quantization function $q_p\colon \mathbf{R} \to \mathbf{Z}$, which maps a real value $x \in [x_1, x_2]$ to a p bit integer representation:

$$q_p(x; x_1, x_2) = r(m(c(x; x_1, x_2); x_1, x_2, 0, 2^p - 1)). \tag{13}$$

The corresponding p bit de-quantization function $d_p\colon \mathbf{Z} \to \mathbf{R}$ returns a real number that approximates $x \in [x_1, x_2]$:

$$d_p(y; x_1, x_2) = c(m(y; 0, 2^p - 1, x_1, x_2); x_1, x_2). \tag{14}$$

5.1 8 and 16-Bit Encoding

We can define a 16-bit encoded point using the following C structure (a similar structure can be defined for 8-bit coding, using instead 8 bits per coefficient):

```
struct encoded_point {
  uint16_t x, y, scale;
  uint8_t orientation, laplacian;
  uint16_t coefficients[64];
};
```

Assuming images are never larger than 65535 × 65535, the values x and y in the above can be assigned from the unencoded values by simple rounding:

$$x' = r(x) \tag{15}$$

$$y' = r(y). \tag{16}$$

Assuming that scale values greater than 30 are sufficiently rare that they can be safely approximated by a value of exactly 30, then we can assign the scale value in the above structure as:

$$s' = q_{16}(s; 0, 30). \tag{17}$$

The orientation parameter can be encoded to 8 bits, assuming that an error of 7/10 of a degree can be tolerated in practice.

$$\theta' = q_8(\theta; 0, 2\pi) \tag{18}$$

The Laplacian parameter could be encoded in a single bit, but we use 8 bits so that the structure is aligned to 16-bit boundaries in memory. The 16 blocks of coefficients $f_1 \ldots f_4$ are encoded as:

$$f_1' = q_{16}(f_1; -0.5, +0.5) \tag{19}$$

$$f_2' = q_{16}(f_2; -0.5, +0.5) \tag{20}$$

$$f_3' = q_{16}(f_3; 0, 1) \tag{21}$$

$$f_4' = q_{16}(f_4; 0, 1). \tag{22}$$

The final 16-bit encoding requires 136 bytes per SURF point, before applying any compression algorithms. For the 8-bit quantization, the q_8 quantizer is used above. The 8-bit encoding requires 72 bytes per SURF point.

6 Compression

The symbol entropy values $H(f)$ for each of the four Haar coefficients $f_1 \ldots f_4$ for the BelgaLogos dataset sample were found to be:

	f_1	f_2	f_3	f_4	mean
8-bit	5.64	6.32	6.09	6.60	6.16
16-bit	13.62	14.30	14.07	14.59	14.14

Shannon's source coding theorem for symbol codes bounds the compression rate that can be achieved using an optimal symbol coding algorithm between $H(f)$ and $H(f)+1$, so based on the mean entropies above we can expect compressing the 8-bit encoded coefficients using an optimal symbol coding algorithm to reduce the number of bits to required to encode a coefficient by between 0.84 and 1.84 bits per symbol. This translates reducing amount of space required to store the coefficients by between 10 and 23%.

To encode 8-bit coefficients, we first computed four empirical probability distributions for the quantized coefficients $f_1 \dots f_4$ by sampling N = 100,000 points from the BelgaLogos dataset. These four sample distributions were then smoothed by convolution with a width 5 Hamming window to give the empirical distributions $P_e(x|n), n \in \{1 \dots 4\}, x \in \{0 \dots 255\}$. The empirical distributions were then used to generate prefix codes $C_n(x)$ for each the Haar coefficients using the Huffman algorithm [5]. To minimize any wasted bits due to a block of Huffman codes not being a multiple of 8 bits, we encoded all Haar coefficients of the same type (e.g. f_1) together in large blocks for all SURF points in a file.

7 Evaluation

To evaluate the effect of quantization on the outcome of a matching experiment, we ran a linear search using the internal query set (55 query images) in the BelgaLogos set for the different encoding strategies. Table 2 summarizes the results of the evaluation. In the table, the column marked 'text/xml' refers to an XML-based encoding scheme; 'text/plain' refers to the text encoding scheme used by the OpenSURF[2] library; 'float/32' refers to storing the coefficients as 32-bit IEEE-754 floating point numbers; 'int/16' and 'int/8' refer to simple quantization of the coefficients to 16 and 8 bits as described above; 'huffman/8' refers to entropy coded 8-bit coefficients. The total query times shown are for all 55 images in the internal query set[3].

There is no change in mean average precision (MAP) when using 16-bit precision to encode the coefficients. There is a slight increase in MAP when using 8-bit coefficients. This increase can be attributed to the clustering effect of quantizing the features, which makes them comparable with visual words.

The raw binary encoding schemes gave the fastest decode times, all producing a roughly equal query time of ∼34 seconds per query on our test machine (34 seconds to match a set of query feature points against all feature points in the BelgaLogos set). Huffman encoding of the coefficients more than doubles the decode time to 75 seconds. Both raw and encoded coefficients are significantly faster than using XML encoded features, giving speedups of 97.6% and 94.7%. Compared with the plain text format generated by OpenSURF, the speedups are 88% for raw features and 74.2% for encoded features.

[2] http://www.chrisevansdev.com/computer-vision-opensurf.html

[3] The test machine used was an Intel® quad core Xeon® 1.86GHz CPU (E5320) with 4GB RAM running Linux kernel 2.6.31-19 and using an ext4 filesystem.

Table 2. Summary of encoding method performance

	text/xml	text/plain	float/32	int/16	int/8	huffman/8
Dataset size (MB)	13172	3683	1590	795	430	346
Mean file size (KB)	1349	377	163	81	44	35
Total query time (min)	1305	267	32	32	32	69
Mean query time (sec)	1422	291	34	34	34	75
Mean average precision (%)	8.53	8.53	8.53	8.53	8.74	8.74

Entropy coding of the 8-bit coefficients gives the greatest space efficiency, reducing the dataset size by 97.4% compared to XML encoded features, by 90.6% compared to plain text encoded features, and by 78.2% compared to binary coding with 32-bit floating point numbers.

7.1 Comparison with PCA Compression

Correlation among the Haar coefficients within descriptors implies a more compact representation may be obtained using principal component analysis (PCA) to reduce the dimensionality of the feature vectors without loss of important information. This approach is similar to the PCA-SIFT technique described by Ke and Sukthankar [10], except that here we focus on matching using the reconstructed descriptors rather than in the transform domain. The following compares several PCA-based compression schemes with the simple quantization and entropy coding schemes outlined above in terms of storage size, decoding speed, and matching accuracy on the BelgaLogos dataset.

We used 100,000 randomly sampled feature points to derive the PCA eigenvector transform matrix Φ. We used the same sample to derive the empirical means $\mu = (\mu_1, \ldots, \mu_{64})$ of the 64 coefficients, and the PCA transform of the sample to derive the empirical range $[\alpha_i, \beta_i]$ of the transformed coefficients. Dimensionality reduction of a coefficient vector $\mathbf{v}$ is performed by centering around the mean and multiplying by the transform matrix: $\Phi^{\mathrm{T}}(\mathbf{v} - \mu)$, then truncating to the desired number of coefficients. Each retained coefficient v_i is then quantized to 8 bits using Eq.(13) with the empirical minimum α_i and maximum β_i. Decoding a quantized coefficient vector $\mathbf{u}$ is performed similarly: by de-quantizing the stored coefficients and performing the inverse transform $\Phi\mathbf{u} + \mu$.

Table 3 compares the performance of PCA compressed descriptors with 8-bit quantization and entropy coding based descriptors on the BelgaLogos dataset. The column heading ‘PCA/50’ denotes a 50 element PCA compressed descriptor quantized to use 8-bits per element. It is clear from the table that the 50 element PCA descriptor performs on par with 8-bit entropy coded coefficients in terms of descriptor size and decode time, but suffers a small loss in matching precision. The 40 and 30 element PCA descriptors are both smaller and faster than entropy coded coefficients, but the sacrificed precision reduces matching accuracy.

Table 3. Comparison of 8-bit PCA compression with quantization and entropy coding

	PCA/50	PCA/40	PCA/30	PCA/20	int/8	huffman/8
Dataset size (MB)	351	294	237	180	430	346
Mean file size (KB)	36	30	24	18	44	35
Total query time (min)	61	53	47	41	32	69
Mean query time (sec)	66	58	51	44	34	75
Mean average precision (%)	8.18	6.94	5.47	2.35	8.74	8.74

The 20 element descriptor is almost half the size of the entropy coded descriptor, but suffers an acute reduction in accuracy. Despite their smaller size, none of the PCA-based descriptors match the decode time of the simple 8-bit scheme.

8 Conclusion

We have presented a simple, easy to implement, scheme for quantization and entropy coding of SURF features that significantly reduces both the space required to store these features and the time required to load and decode them. The experiments show that careful quantization of the feature points can be performed without significantly effecting matching performance. Quantization of coefficients to 8 bits also opens the door to an integer only implementation of feature point distances, which may improve performance, particularly for embedded applications. Entropy coding reduces the storage requirements by 20% over raw 8-bit coefficients, but doubles the decoding time. This modest space improvement rarely justifies the additional complexity and decode time; we anticipate that simple 8-bit quantization may give the best balance between decode time and storage requirements for many applications. If space is at a premium, transform coding can be applied to de-correlate the coefficients prior to quantization at the cost of reduced matching accuracy and increased decode times.

More powerful compression techniques like product quantization [7] are capable of achieving an even more compact representation. For example, [4] used this technique to compress SURF feature vectors to around half the size of the Huffman encoded vectors. This approach requires precomputing a codebook using a VQ algorithm (e.g. Lloyd's algorithm), which may take significant time and can potentially produce suboptimal results. The simple quantization approach presented here can be combined with product quantization to reduce the memory footprint of the codebook by a factor of four without significant effect on reconstruction. Codebooks can therefore be designed with four times more cluster centroids and still fit in the same amount of memory, which may be important for performance when the codebook is required to fit in cache memory [7].

An implementation of our SURF encoding schemes is available online and can be downloaded from: `http://kspace.cdvp.dcu.ie/public/surfenc`.

Acknowledgements. This work is supported by Science Foundation Ireland under Grant 07/CE/I1147 (CLARITY: Center for Sensor Web Technologies). Part of this work is supported by the EU Project FP7 AXES ICT-269980.

References

1. Speeded up robust features (SURF). Computer Vision and Image Understanding 110(3), 346–359 (2008)
2. Chandrasekhar, V., Takacs, G., Chen, D., Tsai, S., Grzeszczuk, R., Girod, B.: Chog: Compressed histogram of gradients a low bit-rate feature descriptor. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2504–2511 (2009)
3. Chandrasekhar, V., Takacs, G., Chen, D., Tsai, S.S., Singh, J., Girod, B.: Transform coding of feature descriptors. In: Visual Communication and Image Processing (2009)
4. Gammeter, S., Gassmann, A., Bossard, L., Quack, T., Van Gool, L.: Server-side object recognition and client-side object tracking for mobile augmented reality. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW 2010), pp. 1–8 (2010)
5. Huffman, D.: A method for the construction of minimum-redundancy codes. Proceedings of the IRE 40(9), 1098–1101 (1952)
6. Jègou, H., Douze, M., Schmid, C.: Improving bag-of-features for large scale image search. International Journal of Computer Vision 87, 316–336 (2010)
7. Jègou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. IEEE Transactions on Pattern Analysis and Machine Intelligence 33, 117–128 (2011)
8. Jègou, H., Douze, M., Schmid, C., Perez, P.: Aggregating local descriptors into a compact image representation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3304–3311 (2010)
9. Joly, A., Buisson, O.: Logo retrieval with a contrario visual query expansion. In: ACM International Conference on Multimedia, pp. 581–584 (2009)
10. Ke, Y., Sukthankar, R.: PCA-SIFT: A more distinctive representation for local image descriptors. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 506–513 (2004)
11. Lowe, D.: Object recognition from local scale-invariant features. In: IEEE International Conference on Computer Vision, pp. 1150–1157 (1999)
12. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
13. Sivic, J., Russell, B., Efros, A., Zisserman, A., Freeman, W.: Discovering objects and their location in images. In: IEEE International Conference on Computer Vision, pp. 370–377 (2005)
14. Szeliski, R.: Computer Vision: Algorithms and Applications, 1st edn. Texts in Computer Science. Springer, Heidelberg (2010)
15. Takacs, G., Chandrasekhar, V., Gelfand, N., Xiong, Y., Chen, W.C., Bismpigiannis, T., Grzeszczuk, R., Pulli, K., Girod, B.: Outdoors augmented reality on mobile phone using loxel-based visual feature organization. In: ACM International Conference on Multimedia Information Retrieval, pp. 427–434 (2008)

How to Select and Customize Object Recognition Approaches for an Application?

Robert Sorschag

Vienna University of Technology
Institute of Software Technology and Interactive Systems
Favoritenstrasse 9-11, A-1040 Vienna, Austria
sorschag@ims.tuwien.ac.at

Abstract. Recently, object recognition has been successfully implemented in a couple of multimedia content annotation and retrieval applications. The employed recognition approaches are carefully selected and adapted to the specific needs of their tasks. In this work, we propose a framework to automate the simultaneous selection and customization of the entire recognition process. This framework only requires an annotated set of sample images or videos and precisely specified task requirements to select an appropriate setup among thousands of possibilities. We use an efficient recognition infrastructure and iterative analysis strategies to make this approach practicable for real-world applications. A case study for face recognition from a single image per person demonstrates the capabilities of this holistic approach.

Keywords: Multimedia content annotation and retrieval, Object recognition, Feature selection, Automatic evaluation.

1 Introduction

Humans use their visual system to recognize objects for the interpretation of scenes and images. Computer vision systems try to imitate this process with approaches to recognize individual objects and object classes. Many of these approaches already exist and it requires experience and a fairly advanced level of computer vision skills to select an appropriate approach for an application. Furthermore, recent studies [1-3] show that the performance of most recognition approaches can be significantly improved when they are adapted to certain tasks, domains, or datasets. Thus, the research question of this work is: How to automate the selection and customization of recognition approaches for a given task?

This auto-selection and customization depends on both, the task and the investigated recognition approaches. In the example of Fig. 1, an appropriate approach has to be selected for a cow recognition system that operates quite fast and accurate. On the one hand, recognition tasks mainly differ by the objects-of-interest to recognize and their appearance in the content. Thus, different recognition approaches might be well suited for tasks where all objects are similarly shown from one viewpoint and for tasks with a higher variability. Moreover, requirements about the recognition accuracy, quality,

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 452–462, 2012.

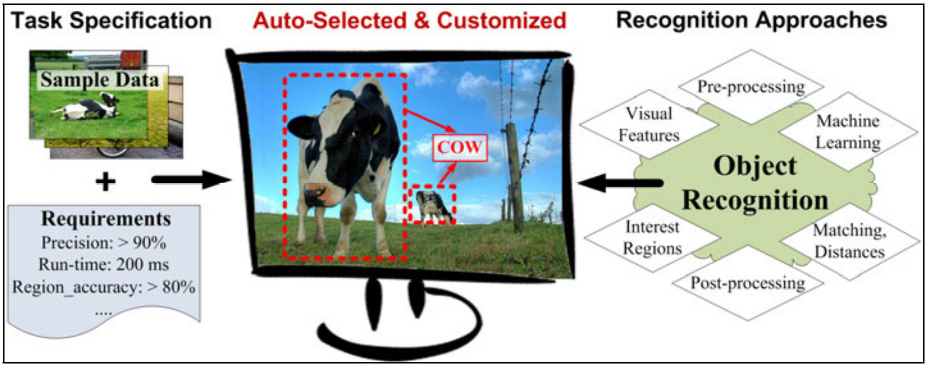


Fig. 1. Auto-selection of a cow recognition system from precise specifications

and speed have to be considered in the selection process. On the other hand, recognition approaches mainly differ by the used visual features, matching and machine learning strategies, as well as their settings.

Nowadays, researchers and practitioners usually perform the approach selection and customization in a tedious and time-consuming process by manual evaluation of different setups. In this process, they collect sample data and specify the task requirements before a prototype is developed using the best practice of related tasks. At last, the prototype is adapted to the sample data by replacement of single processing steps or the adjustment of parameter settings. We automate this process by an extensive framework that operates on example-based data and that includes precise task specifications, an efficient recognition infrastructure, and an evaluation tool. The strength of this approach is demonstrated on a case study for face recognition applications with different task requirements.

2 Related Work

Recently, researchers have achieved quite good results for the recognition of individual objects and object classes, for example, in the Pascal VOC challenge [4]. Best practice approaches are based on visual features that are generated from local region detector-descriptor chains [5-6] and from object models that are suited for recognition with feature matching or machine learning approaches [7]. In the context of object recognition, good visual features should generally compute the same values when they are applied to the same objects and distinct values for different objects. It is well established that feature types mainly differ by the trade-off that they achieve between their discriminative power and invariance. Furthermore, different recognition tasks require different trade-offs, and thus no single visual feature is optimal in all situations [8]. In addition to these findings, [9] has shown that all components of a recognition approach can have strong influences on the achieved results.

The authors of [2] pointed out that the manual process of choosing appropriate algorithms and tuning them for a given task is more an art than a science. A couple of works investigated the automatic selection and customization of recognition approaches from different directions: Attempts to optimize the parameters of specific visual features, like SIFT and HOG, are given in [1], [3], and [10]. Varma et al. [8] proposes a kernel-learning approach to select the best feature combination for a task using a SVM framework that works with all types of features. [11] uses a convolutional neural network to learn new features for each task instead of using manually designed, hand-crafted features while [2] proposed a trainable local feature matching approach that uses a boosting framework. In contrast to these works, we try to automate the simultaneous selection and customization of the entire recognition process with all kinds of visual features, matching strategies, and so on. To the best knowledge of the authors, no work exists that presents such a holistic approach.

Similar to the mentioned works, we select and customize recognition approaches with the help of training data. Although it might be possible to tweak algorithms manually for specific tasks, it is much easier and more intuitive to provide appropriate training data instead [2]. According to [1], these data-driven approaches further increase the probability to learn invariances accurately for specific applications. A potential drawback of data-driven systems is the requirement of training data that is usually human labeled [12-13] or synthetically generated by artificial image transformations [11]. The manual annotation of training data is time-consuming and synthetic approaches often do not capture all the invariances of real data. Thus, [1] and [14] used optical flow tracking as a data collection step to customize the recognition of moving objects without human intervention. We propose a framework that can operate on all kinds of training data and with all of these data collection approaches.

Moreover, vision prototyping tools like Papier-Mache [15] and Eyepatch [16] share some similarities with our work. These tools allow users to create, test and refine recognition strategies with a visual, example-based approach in order to use cameras as additional input devices. However, the users have to select and customize the recognition approaches manually by an examination of the achieved results in a trial-and-error fashion. Furthermore, object recognition infrastructures, like REIN [17] and CORI [18], present another aspect of this work that is rooted in the literature. We employ the latter infrastructure for the proposed auto-selection and customization process.

3 Framework

In this work, we propose a framework for the automatic selection and customization of object recognition approaches for a given task, domain, or dataset. On the one hand, this framework enforces application engineers to specify their tasks in a precise and machine-readable form. On the other hand, every previously added recognition approach can be investigated for automatic selection and those approaches that offer a certain level of flexibility can be customized as well. In the simplest case, a few parameters are adjusted for a chosen recognition approach, but the selection and customization of the entire processing chain is also possible. In the following, we describe the required task specifications and the recognition setups before details about the auto-selection and customization strategy are provided.

3.1 Task Specification

The presented framework operates on example-based specifications of a recognition task where objects-of-interest are annotated in a sample dataset and where the requirements are defined as precisely as possible. These specifications are generated in three steps. First, a dataset has to be collected that represents the task and its objects well with appropriate difficulties and levels of abstraction. If it is, for instance, the task to recognize all kinds of cars from different views, then the dataset should not only include blue SUVs shown from a frontal view. One way to collect a dataset for rapid prototyping is the manual selection of labeled images from Flickr or Google image search. However, preliminary evaluations showed that the best performance is achieved when the sample images or videos are collected from a realistic application environment.

In a second step, the dataset has to be annotated to specify *which* objects are situated *where*. One or more free text labels are thereby used to specify the object's identity and an arbitrary shaped polygon specifies the object's region. A couple of tools exist to annotate images and videos in such a way, like LabelMe [12] and Viper [13], and they generate annotations of different (XML-based) formats. In this work, we neither propose another annotation tool nor define a new annotation format. Instead, we present the used class schema that can be applied for all kinds of object annotations. As shown in Fig. 2 and Fig. 3, this schema contains *media objects* and *real-world objects* that are both derived from the base class *object.* They contain a unique identifier in combination with the *media reference* and one or more *labels*, respectively. The labels can be ambiguous as instances of the same object might be annotated differently when no common vocabulary is shared by all annotators. The *object relation* 'same-as' dissolves such ambiguities while the relations 'part-of' and 'child-of' are used to model hierarchical structures. Annotated *object instances* use the *ground truth* class in contrast to *recognition hypotheses* that present the output of recognition systems. Both classes capture the object-of-interest that is shown in a media object as well as its region, difficulty level, pose and recognition probability.

Thirdly, some *selection criteria* have to be defined to set the desired recognition quality, accuracy, and speed of a task. The recognition *quality* is defined by the recall and precision that should be achieved. The *region accuracy* specifies the required polygon overlap of recognition hypotheses and ground-truth objects to generate a positive match. Another selection criterion considers the analysis *run-time*. Furthermore, it is possible to specify selection criteria for certain objects and their appearance (pose, difficulty, and size).

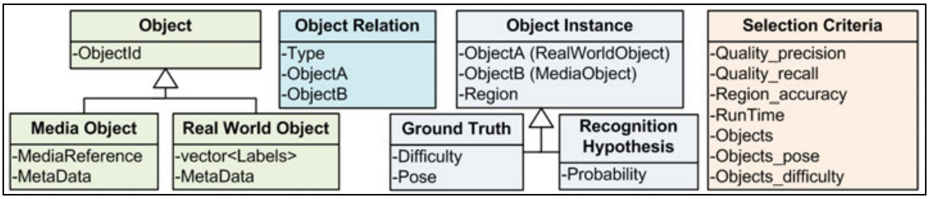


Fig. 2. Class schema for object annotation and task specification

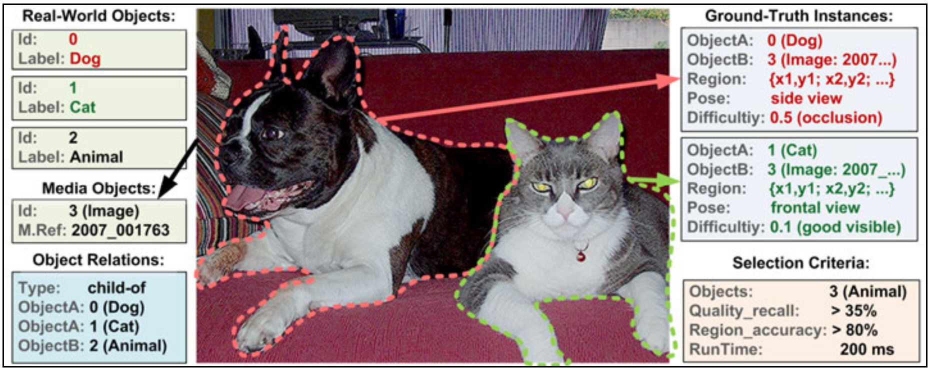
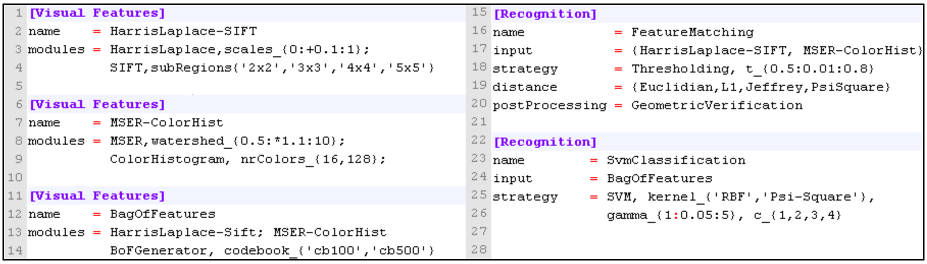


Fig. 3. Task specification for an animal recognition application

3.2 Recognition Infrastructure and Setup

State-of-the-art object recognition systems are composed of heterogeneous components and they are often tailor-made in order to meet the needs of certain applications. Usually, these systems are deeply integrated to the workflow and processing chains of their applications. As a consequence, it is very difficult to adapt such recognition systems to new tasks, and it is a common practice to develop new recognition systems from scratch instead. For this reason, we propose a novel, holistic approach. Recognition components are thereby developed independently from certain tasks in a reusable way. The configurable object recognition infrastructure CORI [18] handles the integration and execution of selected components for a new task. With this infrastructure many different setups can be efficiently executed in parallel. CORI was developed to recognize objects with state-of-the-art approaches. Changes of a running system (parameters as well as the entire processing chain) can be performed without recompilation and deployment, simply by adapting its configuration. We use this capability to compare different recognition setups against each other with a minimal development effort. All kinds of algorithms can be integrated into this infrastructure as components with well defined parameters and IO data structures. CORI is written in C++ but it provides an interface for external processes, and new components can be developed with computer vision toolboxes like OpenCV [19].

A detailed description of CORI's architecture and the integration of new approaches can be found in [18]. In this work, we have extended CORI to generate multiple



```
[Visual Features]
name     = HarrisLaplace-SIFT
modules = HarrisLaplace,scales_{0:+0.1:1};
          SIFT,subRegions{'2x2','3x3','4x4','5x5'}

[Visual Features]
name     = MSER-ColorHist
modules = MSER,watershed_{0.5:*1.1:10};
          ColorHistogram, nrColors_{16,128};

[Visual Features]
name     = BagOfFeatures
modules = HarrisLaplace-Sift; MSER-ColorHist
          BoFGenerator, codebook_{'cb100','cb500'}
```

```
[Recognition]
name           = FeatureMatching
input          = {HarrisLaplace-SIFT, MSER-ColorHist}
strategy       = Thresholding, t_{0.5:0.01:0.8}
distance       = {Euclidian,L1,Jeffrey,PsiSquare}
postProcessing = GeometricVerification

[Recognition]
name         = SvmClassification
input        = BagOfFeatures
strategy     = SVM, kernel_{'RBF','Psi-Square'},
               gamma_{1:0.05:5}, c_{1,2,3,4}
```

Fig. 4. Recognition setups: visual feature extraction (left) and recognition (right)

setups from a simple configuration file, like the one of Fig. 4. On the left side, this configuration specifies two visual feature types (Harris-Laplace points [5] with SIFT descriptors [20], and MSER regions [5] with color histograms) and a combined bag-of-features approach [9]. On the right side, we specify that the first two visual features are used for feature matching with a thresholding strategy while the bag-of-features are classified by SVMs [7]. The configuration format is similar to the original format of CORI, but parameters can be defined as intervals or lists of concrete values. Intervals use a '*{ start value : arithmetic expression : end value }*' syntax (Line 3 of Fig. 4) and lists are comma separated (Line 4). The corresponding analysis graph of Fig. 5 shows many instances of each component with different parameter settings and that they are all executed in parallel. The output of a component is used as input for several succeeding components instead of executing these components again and again in separated graphs for each setup.

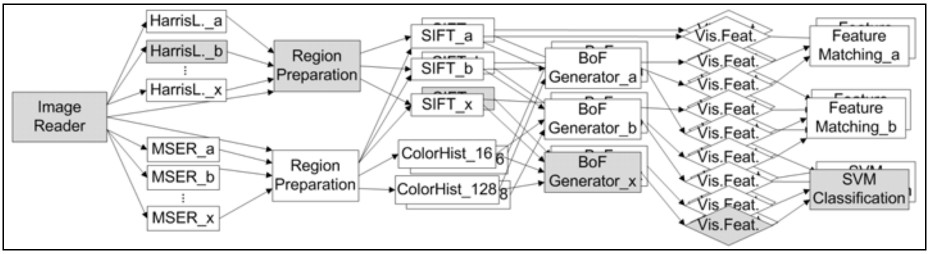


Fig. 5. Analysis graph with the selected setup for a given task (shown in gray)

3.3 Auto-selection and Customization

After a task was specified and the recognition setups have been defined, the proposed framework starts to select the best path through the analysis graph. As shown in Fig. 6, the four steps (1) dataset selection, (2) analysis, (3) evaluation, and (4) setup selection are performed. In this process, the annotated images and videos are used to compare different recognition setups against each other and to select the one that fits best to the specified requirements.

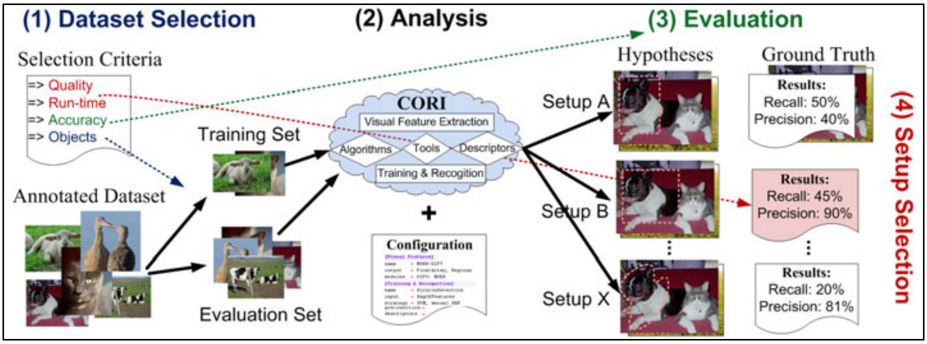


Fig. 6. Auto-selection and customization process

Data Selection: In the first step, two sets are selected from the sample data, one for training and one for evaluation. The size and composition of the first set depend on the investigated recognition approaches. On the one hand, machine learning approaches usually train their object classifiers from the same amount of positive and negative examples [7]. Feature matching strategies, on the other hand, can be trained from a few object examples without the need of any negative examples [20]. When the annotated sample data is large enough, an appropriate training set is straight forwardly selected. Otherwise, we support the manual selection of additional examples from Flickr. The selection of the evaluation dataset is a semi-automatic process that starts with those images from the sample data that are not contained in the training set. Users can then manipulate this initial set to specify the used amount of object instances. In general, the evaluation set should contain the same percentage of object instances as the real application data. For instance, if an object is shown in every 10th image in the application, the same should be true for the evaluation set.

Analysis: During the analysis step, the visual features of each setup are extracted according to the specified recognition setup. CORI makes it possible the extract many different visual features with different parameterizations in parallel and to match them efficiently on distributed multiprocessor architectures [18]. However, usually it is not possible to extract all setups in parallel because of memory and run-time issues. For this reason, we start with the analysis of a few sample images. The memory load and the run-time are captured thereby and it is estimated how long a brute force analysis of all setups would take. If the memory load is too high or if the estimated run-time exceeds a few hours, several analysis runs are individually performed. Therefore, different recognition approaches are analyzed separately and a grid search approach [7] is applied to investigate rough parameterization steps prior to finer ones. Moreover, we exclude all configured recognition approaches from analysis that exceed the specified run-time requirements on the first few images.

Evaluation: The evaluation process starts by sorting the generated recognition hypotheses according to their system setups. For each hypothesis of a setup, we then select all ground-truth instances that are annotated in the same image or video frame and that stem from the same object or from an object with an appropriate relation. If no region information is given in the recognition hypothesis, a positive match is

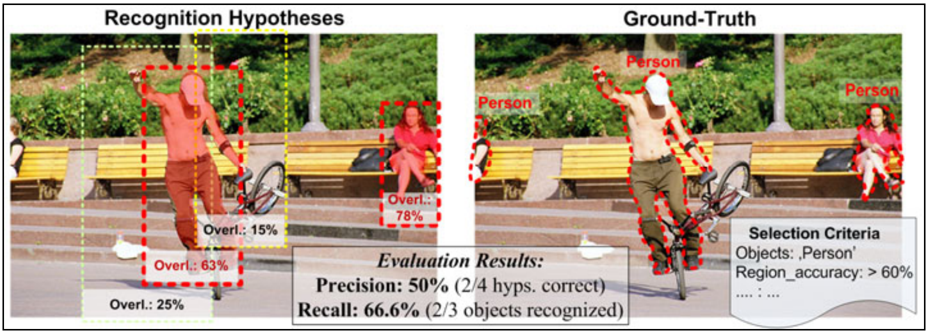


Fig. 7. Evaluation process: The bold bounding boxes in the left image are true positives

generated when at least one of these ground-truth instances exist. Otherwise, all selected ground-truth instances are individually compared against the investigated hypothesis by computation of the region overlap using a polygon intersection. Fig. 7 shows this evaluation for a task where the ground truth (right image) has exact object boundaries while the recognition hypotheses (left image) are given as bounding boxes. The specified region accuracy (> 60%) is then used to classify hypotheses as true positives (bold bounding boxes) or as false positives (thin bounding boxes).

Setup Selection: Finally, the recognition setup that fits best to the selection criteria is chosen for the application. In this process, we compare the specified recall and precision values against the achieved evaluation results from the investigated recognition setups. If only one measure is specified (recall or precision), we select the setup that meets this condition and achieves the highest value of the other measure. If both measures are specified, we select the setup that meets both conditions and that achieves the highest F-measure. When no setup was found that meets the conditions, we present the setups with the highest recall, the highest precision, and the highest F-measure to the user for manual selection. Please note that the specified quality requirements are only used for the auto-selection and customization process. There is no guarantee that these values are later achieved in the application with real data. However, experiments have shown that similar results are achieved when the correlation between the evaluation dataset and real data is high.

4 Case Study

This section demonstrates the capabilities of the proposed auto-selection and customization framework for a couple of different applications in the area of face recognition from a single image per person. In these applications, it has to be decided which trained person is shown in a query image. As pointed out in [21], this task is nontrivial and different recognition approaches (including global features, local features, different matching strategies and different settings) might achieve good results. Face recognition is usually performed on image regions that have been identified by face detection approaches like the popular Viola & Jones AdaBoosting [22]. For the sake of simplicity, we use the FERET face database [23] to avoid annotation issues and to make the experiments comparable to other works. However, exactly the same experiments can be done for other tasks like face recognition in surveillance videos.

Dataset Selection: The used face database [23] consists of 1702 gray-level images of 256 different persons. From each person at least 4 different portray photos are given with different facial expressions, with and without glasses, from slightly different views, and with different lighting conditions, see Fig. 8. In the experiments of this work, we divide these images into three datasets. The first one is the training set. It consists of 128 randomly selected images from different persons. The remaining images are divided into two equally large sets for auto-selection and customization, on the one hand, and to test the performance of the selected approaches, on the other hand. These sets consist of 355 positive examples (trained persons) and 432 negative examples (persons that are not trained).

Fig. 8. Examples of the sample data from FERET database for two persons

Task Specifications: We use following selection criteria to evaluate different task requirements. (1) Recall > 90%, (2) precision > 90%, and (3) best F-measure whereby one evaluation was done without run-time requirements and another one with the requirement that recognition should not exceed 200 ms per query image. The high-precision scenarios assume that a trained person is only returned when the system is quite sure that the query image belongs to this person. In the high-recall scenarios, more than one person is usually returned for each query image. This might be appropriate for applications where the user makes the final decision.

Recognition Setup: We employ a set of visual features that are globally extracted from the face images and locally extracted around interest regions. Each feature is used for different matching strategies with several dissimilarity measures. Furthermore, optional pre-processing, post-processing, and filtering steps are investigated. Table 1 gives an overview of the used components and references to more detailed descriptions of them. Each component has one or more parameters for customization.

Results: Table 2 shows the achieved results for the test set. Please note that the set was *not* used for auto-selection and customization, and thus some results are below the specified selection criteria. The numbers in brackets give the differences to the results of the selection set. For all three scenarios without run-time requirements (left columns), local SIFT features have been selected from dense sampled interest regions. However, the parameterization of each component (e.g. the used sampling scales and step sizes) as well as the matching strategies and dissimilarity measures are different. In the first scenario, features are matched with a nearest neighbor strategy with L1-distance, and a feature voting component generates recognition hypotheses for persons with at least 17 votes. In the second and third scenario, a K-NN feature matching (k = 3 and 5) was used with Euclidian distance combined with an alternative feature voting where the percentage between the highest entry and the second highest entry has to exceed a certain value (22% and 31%). Global Gabor wavelets with a NN-Distance Ratio strategy [20] have been selected in the second and third scenario with run-time restriction (right columns). Different distance measures are thereby used (Jeffrey

Table 1. Recognition components

Type	Name
Visual Features	SIFT [20], Gabor Wavelets [24], MPEG-7 ColorLayout [25]
Interest Regions [5]	Dense Sampling [9], DoG, MSER
Pre-Processing	Color Normalization, Image Scaling
Filtering	High-Contrast, Minimum Region Size
Matching Strategy	Nearest Neighbor, K-NN, NN-Distance Ratio [20], Tresholding
Dissimilarity Measures	L1, L2, Canberra Metric, Jeffrey Divergence, Psi Square

Table 2. Results of the test set including the measured differences to the selection set

Selection Criterion	***Run-time < ∞***		***Run-time < 200 ms / image***	
	Recall	**Precision**	**Recall**	**Precision**
Recall > 90%	87.3% (- 2.7)	36.4% (+ 0.9)	85.9 % (- 4.2)	17.3% (+ 2.1)
Precision > 90%	79.5% (+ 1.1)	90.3% (- 2.5)	63.9% (+ 2.1)	88.6% (- 1.8)
Best F-Measure	68.6% (+ 4.2)	95.1% (- 4.4)	66.1% (- 1.3)	85.2% (- 2.9)

divergence and L1). In the high-recall scenario, a local feature DoG-SIFT approach was selected with an image scaling (to 128x128 pixels) and a high-contrast filter.

As shown in the brackets of Table 2, similar results have been achieved between the selection set and the test set in all scenarios. Differences of less than 5% are given, and the results slightly shifted to the middle of the trade-off between recall and precision. This indicates that it is possible to select recognition approaches automatically that meet the specified requirements when the sample data is highly correlated to the application data. The achieved results did not improve the state-of-the-art for face recognition from a single image per person, but the achieved results are close to the top ranked approaches in [21]. In our experiments, the auto-selection and customization process investigated between 1500 and 5000 recognition setups for each scenario and took between 5 and 12 hours to complete.

5 Conclusions

The proposed framework facilitates the selection and customization of recognition approaches for complex task specifications that can hardly be achieved otherwise. The entire recognition process is investigated to select an appropriate setup for a given task, domain, or dataset. In contrast to this holistic approach, related works optimize only specific components of the recognition process. We use all kinds of visual features, matching strategies and so on, as well as different parameter settings of all components. For this purpose, we extend an object recognition infrastructure to generate many recognition setups from a simple configuration file and to execute them in parallel. In order to cope with the complexity of thousands of setups, we further propose an iterative analysis strategy.

As a proof-of-concept it was shown that the presented framework works efficiently for face recognition from a single image per person. Different recognition approaches were selected thereby for each task requirement and the achieved results are close to the state-of-the-art. However, we intend to support a broad range of multimedia content annotation and retrieval systems in order to use object recognition as an ancillary tool. Further evaluations on different applications are therefore planned as next steps.

Acknowledgements. The author would like to thank Horst Eidenberger for his feedback and support. The research leading to this publication has received funding from the Austrian FIT-IT project ‘IV-ART – Intelligent Video Annotation and Retrieval Techniques’.

References

1. Stavens, D., Thrun, S.: Unsupervised Learning of Invariant Features using Video. In: CVPR (2010)
2. Babenko, B., Dollár, P., Belongie, S.: Task Specific Local Region Matching. In: ICCV (2007)
3. Winder, S., Hua, G., Brown, M.: Picking the best DAISY. In: CVPR (2009)
4. Everingham, M., Van Gool, L., Williams, C., Winn, J., Zisserman, A.: The Pascal Visual Object Classes (VOC) Challenge. IJCV 88(2), 303–338 (2010)
5. Tuytelaars, T., Mikolajczyk, K.: Local Invariant Feature Detectors: A Survey. Foundations and Trends in Computer Graphics and Vision 3(3), 177–280 (2008)
6. Mikolajczyk, K., Schmid, C.: A Performance Evaluation of Local Descriptors. In: PAMI (2005)
7. Hsu, C., Chang, C., Lin, D.: A Practical Guide to Support Vector Classification. Technical report, Nat. Taiwan University, Taipei (2003), `http://www.csie.ntu.edu.tw/~cjlin/papers/guide/`
8. Varma, M., Ray, D.: Learning the Discriminative Power-Invariance Trade-Off. In: ICCV (2007)
9. Jiang, Y., Ngo, C., Yang, J.: Towards Optimal Bag-of-Features for Object Categorization and Semantic Video Retrieval. In: Int. Conf. Image and Video Retrieval (2007)
10. Winder, S.A.J., Brown, M.: Learning Local Image Descriptors. In: CVPR (2007)
11. Jahrer, M., Grabner, M., Bischof, H.: Learned Local Descriptors for Recognition and Matching. In: Computer Vision Winter Workshop (2008)
12. Torralba, A., Russell, B.C., Yeun, J.: LabelMe: Online Image Annotation and Applications. Proceedings of the IEEE 98(8), 1467–1484 (2010)
13. Doermann, D., Mihalcik, D.: Tools and Techniques for Video Performance Evaluation. In: ICPR, vol. 4, pp. 167–170 (2000)
14. Leistner, C., Godec, M., Schulter, S., Saffari, A., Werlberger, M., Bischof, H.: Improving Classifiers with Unlabeled Weakly-Related Videos. In: CVPR (2011)
15. Klemmer, S.R.: Papier-Mâché: Toolkit support for tangible interaction. In: Human Factors in Computing Systems (2004)
16. Maynes-Aminzade, D., Winograd, T., Igarashi, T.: Eyepatch: Prototyping Camera-based Interaction through Examples. In: Symp. User Interface Software and Technology (2007)
17. Muja, M., Rusu, R., Bradski, G., Lowe, D.: REIN - A Fast, Robust, Scalable REcognition INfrastructure. In: International Conference on Robotics and Automation (2011)
18. Sorschag, R.: CORI: A Configurable Object Recognition Infrastructure. In: Int. Conf. on Signal and Image Processing Applications (2011)
19. Bradski, G., Kaehler, A.: Learning OpenCV, Computer Vision with the Open Source Computer Vision Library. O'Reilly Press (2008), `http://opencv.willowgarage.com`
20. Lowe, D.: Distinctive Image Features from Scale-invariant Keypoints. IJCV (2004)
21. Tan, S., Chen, S., Zhou, Z.-H., Zhang, F.: Face Recognition from a Single Image per Person: A Survey. Pattern Recognition 39, 1725–1745 (2006)
22. Viola, P., Jones, M.J.: Robust Real-time Face Detection. IJCV 57(2) (2004)
23. Phillips, P.J., Wechsler, H., Huang, J., Rauss, P.J.: The FERET Database and Evaluation Procedure for Face Recognition Algorithms. In: Image and Vision Computing (1998)
24. Frigo, M., Johnson, S.: The Design and Implementation of FFTW3. In: Proc. Program Generation, Optimization, and Platform Adaptation, vol. 93(2), pp. 216–231 (2005)
25. Manjunath, B., Ohm, J.-R., Vasudevan, V., Yamada, A.: Color and Texture Descriptors. Trans. on Circuits and Systems for Video Technology 11, 703–715 (2001)

A Fuzzy Rank-Based Late Fusion Method for Image Retrieval

Savvas A. Chatzichristofis[1,2], Konstantinos Zagoris[1],
Yiannis Boutalis[1,3], and Avi Arampatzis[1]

[1] Department of Electrical and Computer Engineering
Democritus University of Thrace, Xanthi 67100, Greece
[2] Informatics and Telematics Institute (ITI)
Centre for Research and Technology Hellas (CE.R.T.H.),Greece
[3] Department of Electrical, Electronic and Communication Engineering
Chair of Automatic Control, Friedrich-Alexander University of Erlangen-Nuremberg
Erlangen 91058, Germany
{schatzic,kzagoris,ybout,avi}@ee.duth.gr

Abstract. Rank-based fusion is indispensable in multiple search setups in lack of item retrieval scores, such as in meta-search with non-cooperative engines. We introduce a novel, simple, and efficient method for rank-based late fusion of retrieval result-lists. The approach taken is rule-based, employs a fuzzy system, and does not require training data. We evaluate on an image database by fusing results retrieved by three MPEG-7 descriptors, and find statistically significant improvements in effectiveness over other widely used rank-based fusion methods.

Keywords: Image Retrieval, Rank-Based Late Fusion, Fuzzy Systems, Heterogeneous Databases.

1 Introduction

Fusion in image retrieval is critical for the future of image retrieval research [5] and is not trivial [15]. Two main approaches to fusion have been taken: *early fusion*, where multiple image descriptors are composed to form a new one before indexing, and *late fusion*, where result rankings from individual descriptors are fused during query time. In general, late fusion approaches concern every technique for combining outputs of distinct systems [12] and can be accomplished either as a function of retrieval scores, or as a function of the position in which the results appear in each rank-list. In most cases, score-based late fusion is a better performer [2], but since in some practical situations scores are unknown, the use of rank-based fusion is necessary. A typical need for rank-based fusion arises in meta-search setups with non-cooperative search engines. Additionally, the score-based strategies, require a normalization among all systems in order to balance the importance of each of them, which is not the case of the rank-based strategies [12].

A commonly used method for rank-based fusion is Borda Count (BC), which originates from social theory in voting back in 1770. The image with the highest rank on each rank-list gets n votes, where n is the collection size. Each subsequent rank gets

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 463–472, 2012.

one vote less than the previous. Votes across rank-lists are summed. Borda count is strictly equivalent to combSUM on ranks [12]. The literature about the Borda rule is very extensive (see [6] for references).

Alternatively, in methods such as Borda Count - Max (BC-MAX) and Borda Count - Min (BC-MIN) the final rank-list does not originate from the sum of the votes. In BC-MAX, the images are rated with the highest vote they get across rank-lists while in BC-MIN with the lowest. Another method often used is the Inverse Rank Position (IRP), which merges rank-lists in the decreasing order of the inverse of the sum of inverses of individual ranks. More details about IRP as well as about Borda Count derivatives are given in [7].

In [16], the traditional Borda method is extended by using the Ordered Weighted Averaging (OWA) operator to consider the risk-attitudinal characteristics. This new approach, entitled Borda-OWA, solves the group decision making problem in a more intelligent procedure. Classic BC does not consider the optimistic/pessimistic view of the system, which has a great effect on group decisions. Fusing several rank-lists, a system faces various types of uncertainty so the decision making process will be under risk. If the system strongly avoids the risk of making bad decisions, it will consider more rank-lists in the decision process. However, this will result in conservative decisions which are different than the decisions of a neutral or optimistic decision maker. In common words, the authors are using the terms 'Most of Them' and 'Few of Them' which could be modeled by fuzzy linguistic quantifiers and are used to characterize the aggregation inputs in an OWA operator. The term 'Most of Them' corresponds to the 'Pessimism' optimistic nature, while 'Few of Them' corresponds to the 'Optimism' optimistic nature.

In this paper we introduce a novel, simple, and efficient, rank-based late fusion method. The approach utilizes a Mamdani-type rule-based fuzzy system, and it does not require training data. We evaluate the effectiveness of the proposed method by fusing image rankings of a benchmark database for three MPEG-7 descriptors [10]: the Scalable Color Descriptor (SCD), the Edge Histogram Descriptor (EHD), and the Color Layout Descriptor (CLD). As illustrated from the experimental results, the proposed method provide statistically significant improvements in retrieval quality over other widely used rank-based fusion techniques such as IRP, Borda Count and derivatives.

The rest of the paper is organized as follows: Section 2 provides some details about fuzzy inference systems while Section 3 describes the proposed fuzzy rank-based late fusion technique. The experimental results are depicted in Section 4 and finally the conclusions are drawn in Section 5.

2 Fuzzy Inference Systems

Fuzzy inference is the process of determining the response of a system to a given input by using fuzzy logic and fuzzy linguistic rules for expressing the system's i/o relation. Its main characteristic is that it inherently performs an approximate interpolation between "neighboring" input and output situations [14]. The process comprises of four parts: Initially, (1) the fuzzification of the inputs using appropriately defined membership functions, (2) the designation of the type of the linguistic connection (fuzzy operator AND or OR) in the input variables, (3) the determination of the fuzzy output variables consequences using the fuzzy inference engine and a preset set of rules, and finally, (4) the defuzzification process. Fuzzy inference systems have been successfully applied in fields such as automatic control, decision analysis, expert systems, and computer vision.For more details, see [8].

Two main types of fuzzy modeling schemes are the Takagi-Sugeno model and the fuzzy relational model. The Mamdani scheme is a type of fuzzy relational model where each rule is represented by an *IF-THEN* fuzzy relationship which is numerically built by considering the linguistic rule, the type of the participating fuzzy membership values and the appropriate implication operator. Mamdani scheme is also called a linguistic model because both the antecedent and the consequent are fuzzy propositions [1]. Mamdani fuzzy rule-based systems are among the most popular approaches used in classification problems.

3 Fuzzy Rank-Based Late Fusion

In this section we are describing a Mamdani fuzzy rule-based system for rank-based late fusion. The parameters of the proposed fuzzy inference system are given in Table 1.

Table 1. Parameters of the fuzzy inference systems

Fuzzy modeling scheme	Mamdani
Inputs	3 membership functions for each input (Fig. 1)
Fuzzy operator in the input variables	AND
Fuzzy output variables	7 membership functions (Fig. 2) with 27 Rules
Defuzzification process	Centroid defuzzification method

Initially, we assume that the results of each rank-list can be divided into 3 fuzzy clusters according to their probability degree of similarity, i.e. **High**, **Medium** and **Low**. The membership functions (MF) of each class are illustrated in Fig. 1. The horizontal axis corresponds to the total number of results in the rank-list (in percentage) while the vertical one represents the membership value for each class. Position A defines the center of the class **Medium**, as well as the lower limits of the other 2 classes. A can be moved to the left or to the right of the position shown in Fig. 1 according to design preferences.

When dividing a rank-list in this manner, we assume that each result participates in all 3 classes but with a membership value. In the example outlined in Fig. 1, the result, activates the first membership function by 0.7 and the second by 0.3. This means that this result participates in the first class by 0.7, the second by 0.3 and the third by 0.0.

In each of the rank-lists of the 3 descriptors we employed, there is a corresponding fuzzy system which classifies the results into the 3 classes, with a participation value in each. The shape of all 3 systems is the same. The principle of the system operation is as follows:

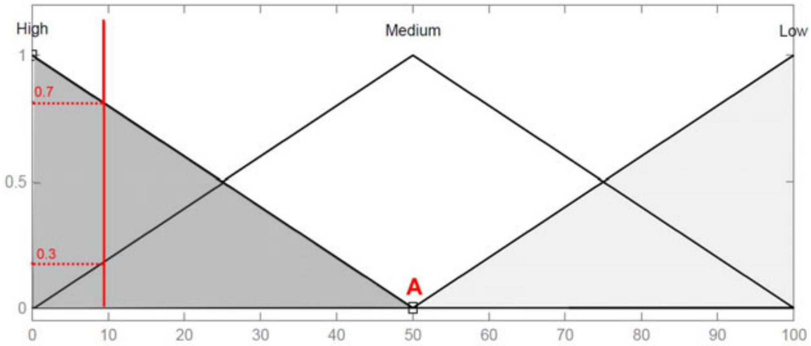


Fig. 1. The fuzzy input of the proposed system

The position of each result in each rank-list is defined as R_i^j, where i is the id of the result and j is the rank-list from which it originates. The value of R_i^1 interacts with the fuzzy membership functions of system 1 to get a membership degree in each class of the system. Similarly, R_i^2 and R_i^3 interact with the membership functions of systems 2 and 3 respectively.

The system output consists of $(2 \times i)+1 = 7$ triangular membership functions, which are illustrated in Fig. 2. The fuzzy system employs 27 rules. These rules are given in Table 2.

Next, we explain how we build the rules using a “voting” concept. Activation (to any degree) of the **High** membership function (MF) of each input contributes with +1 vote, activation of the **Medium** MF contributes 0 votes and activation of the **Low** MF contributes with -1 votes. The output that corresponds to a particular input combination depends on the summation of the votes carried by the three inputs and is determined in respect to the central output MF which is the **MM** in Fig. 2.

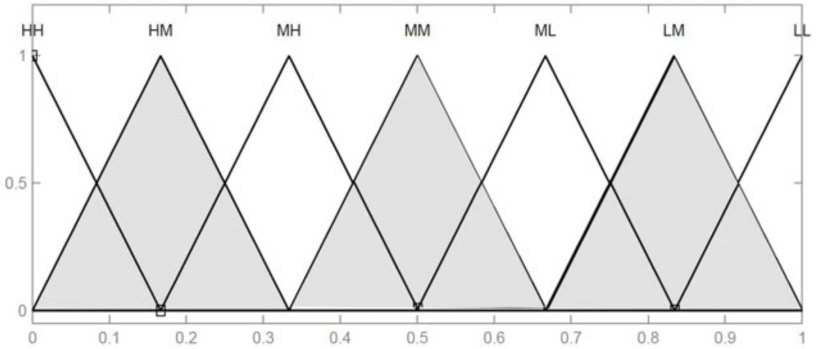


Fig. 2. The fuzzy output of the proposed system

Table 2. Fuzzy Inference Rules

RULE	IF **Input 1** is	AND **Input 2** is	AND **Input 3** is	THEN **Output** is
1	HIGH	HIGH	HIGH	LOW-LOW (LL)
2	HIGH	HIGH	MEDIUM	LOW-MEDIUM (LM)
3	HIGH	HIGH	LOW	MEDIUM-LOW (ML)
4	HIGH	MEDIUM	HIGH	LOW-MEDIUM (LM)
5	HIGH	MEDIUM	MEDIUM	MEDIUM-LOW (ML)
6	HIGH	MEDIUM	LOW	MEDIUM-MEDIUM (MM)
7	HIGH	LOW	HIGH	MEDIUM-LOW (ML)
8	HIGH	LOW	MEDIUM	MEDIUM-MEDIUM (MM)
9	HIGH	LOW	LOW	MEDIUM-HIGH (MH)
10	MEDIUM	HIGH	HIGH	LOW-MEDIUM (LM)
11	MEDIUM	HIGH	MEDIUM	MEDIUM-LOW (ML)
12	MEDIUM	HIGH	LOW	MEDIUM-MEDIUM (MM)
13	MEDIUM	MEDIUM	HIGH	MEDIUM-LOW (ML)
14	MEDIUM	MEDIUM	MEDIUM	MEDIUM-MEDIUM (MM)
15	MEDIUM	MEDIUM	LOW	MEDIUM-HIGH (MH)
16	MEDIUM	LOW	HIGH	MEDIUM-MEDIUM (MM)
17	MEDIUM	LOW	MEDIUM	MEDIUM-HIGH (MH)
18	MEDIUM	LOW	LOW	HIGH-MEDIUM (HM)
19	LOW	HIGH	HIGH	MEDIUM-LOW (ML)
20	LOW	HIGH	MEDIUM	MEDIUM-MEDIUM (MM)
21	LOW	HIGH	LOW	MEDIUM-HIGH (MH)
22	LOW	MEDIUM	HIGH	MEDIUM-MEDIUM (MM)
23	LOW	MEDIUM	MEDIUM	MEDIUM-HIGH (MH)
24	LOW	MEDIUM	LOW	HIGH-MEDIUM (HM)
25	LOW	LOW	HIGH	MEDIUM-HIGH (MH)
26	LOW	LOW	MEDIUM	HIGH-MEDIUM (HM)
27	LOW	LOW	LOW	HIGH-HIGH (HH)

Assuming the **MM** is the starting MF, each positive vote denotes a transition by one MF to the left, while a negative vote denotes a transition to the right. This way, if the inputs contribute with a sum of 1 vote, the output MF of the rule will be the **MH**. A +3 votes contribution means that the output MF of the rule will be the **HH**. On the contrary, -3 votes determines that the output MF of the rule is the **LL**.

Let that R_i^1 activates the input MF **High** by an activation degree $AV_{i,1} = 0.1$, R_i^2 activates the input MF Medium by $AV_{i,2} = 0.2$, and R_i^3 activates the input MF Low by $AV_{i,3} = 0.7$. Then the total votes will be:

$$(+1) + (0) + (-1) = 0$$

This means that the output MF will be the **MM** and the rule will be:

"*If R_1^i is **High** and R_2^i is **Medium** and R_3^i is **Low**, then the output is **MM***".

This procedure can be followed in all possible input combinations deriving 27 rules. In this particular example the rules' degree of fullfilment (DOF) is given by:

$$min(AV_{i,1}, AV_{i,2}, AV_{i,3}) = 0.1$$

An input combination may normally activate more than one rule, each one by a different DOF. The final crisp output is produced by using a conventional fuzzy inference procedure for Mamdani type systems, employing the min implication operator and the centroid defuzzification method. Centroid defuzzification method is also known as center of gravity or center of area defuzzification. This technique can be expressed as:

$$x^* = \frac{\int \mu_i(x) x \partial x}{\int \mu_i(x) \partial x}$$

where x^* is the defuzzified output, $\mu_i(x)$ is the aggregated membership function and x is the output variable.

The 3 rank-lists are fused into a new one, with their results being sorted based on the values (in the range [0,1]) provided by the fuzzy system.

4 Experimental Results

In this study, we evaluate the retrieval effectiveness of the proposed late fusion technique which enable the combined use of the Scalable Color Descriptor (SCD), Edge Histogram Descriptor (EHD), and Color Layout Descriptor (CLD)[1], on a heterogeneous database suggested in [2]. This database consist of 20230 images; 9000 grayscale images are from the IRMA 2005 database[2]; 10200 are natural color images from the NISTER[13] database and 1030 artificially generated images are from the Flags database [4]. The database includes 40 fully-judged queries. The first 20 are natural color image queries from the NISTER database and the second 20 are grayscale queries of the IRMA 2005 database.

A detailed description of the experiment is demonstrated in the following steps and illustrated in Figure 3.

Initially, a query image interacts with the image retrieval system. The three MPEG-7 descriptors are calculated and the application executes the searching procedure using each one of the descriptors. For every descriptor the similarity matching technique recommended for this descriptor is employed.

For each descriptor, when the procedure is complete, the application arranges the images contained in the database according to their proximity to the query image, generating a ranking list. Overall, the system generates three individual ranking lists. Then, using either the proposed method, or a method from the literature, these three result lists are fused in order to generate the final ranking list.

For the evaluation of the performance of the proposed image retrieval method one of the metrics we employed is the Averaged Normalized Modified Retrieval Rank (ANMRR) [11]. The average rank AVR(q) for query q is:

[1] The source code for the MPEG-7 Descriptors is a modification of the implementation that can be found in the LIRe[9] retrieval library.

[2] IRMA is courtesy of TM Deserno, Dept. of Medical Informatics, RWTH Aachen.

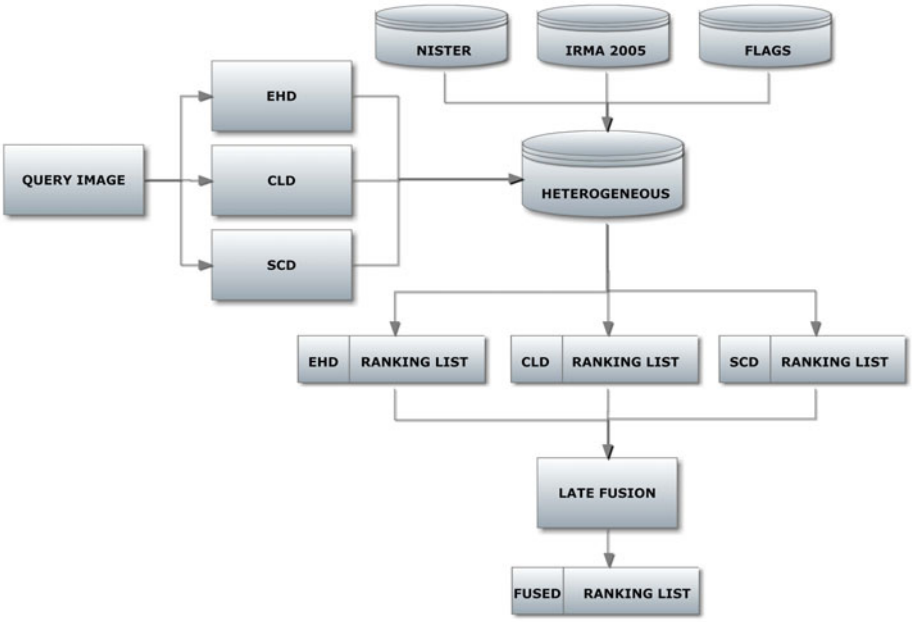


Fig. 3. Late fusion implementation process

$$\mathrm{AVR}(q) = \sum_{k=1}^{NG(q)} \frac{\mathrm{Rank}(k)}{NG(q)} \tag{1}$$

- $NG(q)$ is the number of ground truth images for query q
- $K = \min(X_{NG} \times NG(q), 2 \times \mathrm{GTM})$
- $\mathrm{GTM} = \max(NG)$.
- If $NG(q) > 50$ then, $X_{NG} = 2$ else $X_{NG} = 4$.
- $\mathrm{Rank}(k)$ is the retrieval rank of the ground truth image. Consider a query and assume that the kth ground truth image for this query q is found at position R. If this image is in the first K retrievals then $\mathrm{Rank}(k) = R$ else $\mathrm{Rank}(k) = (K + 1)$.

The modified retrieval rank is:

$$\mathrm{MRR}(q) = \mathrm{AVR}(q) - 0.5 \times [1 + NG(q)] \tag{2}$$

The normalized modified retrieval rank is defined as:

$$\mathrm{NMRR}(q) = \frac{\mathrm{MRR}(q)}{1.25 \times K - 0.5 \times [1 + NG(q)]} \tag{3}$$

and finally the average of NMRR over all queries is computed as:

$$\mathrm{ANMRR}(q) = \frac{1}{Q} \sum_{q=1}^{Q} \mathrm{NMRR}(q) \tag{4}$$

where Q is the total number of queries. The ANMRR has a range of 0 to 1 with the best matching quality defined by the value 0 and the worst by 1.

Apart from the ANMRR metric, we also evaluated the performance of the method using the Mean Average Precision (MAP) metric:

$$\text{Percision} = P = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images retrieved}} \tag{5}$$

$$\text{Recall} = R = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images retrieved}} \tag{6}$$

The average precision AP is:

$$\text{AP}(q) = \frac{1}{N_R} \sum_{n=1}^{N_R} P_Q(R_n) \tag{7}$$

where R_n is the recall after the nth relevant image retrieved and N_R the total number of relevant documents for the query. MAP is computed by:

$$\text{MAP} = \frac{1}{Q} \sum_{q \in Q} \text{AP}(q) \tag{8}$$

where Q is the set of queries q.

The last evaluation metric that we employ is the Precision at 10 (P@10) and Precision at 20 (P@20) metrics that describe the system's capability to retrieve as many relevant results as possible in the first 10 and 20 ranked positions, respectively. This evaluation of the system's performance is critical for web based retrieval systems where the users are particularly interested in the credibility of the first results.

Additionally, we calculate how significant is the performance deviation between the methods. Significance test tell us whether an observed effect, such as a difference between two means, or a correlation between two variables, could reasonably occur just by chance in selecting a random sample. This application uses a bootstrap test, one-tailed, at significance levels 0.05, 0.01, and 0.001, against a baseline run.

The results are outlined in Table 3. As a baseline we assumed the Borda Count, which is one of the most commonly used methods in the literature for rank-based fusion.

All fusion methods beat the single descriptor performance. The best effectiveness overall is achieved by the proposed method; it beats BC by wide margins for all the A levels. BC-OWA (Neutral) results are the same with BC. This is inline with [16] which shows that BC is a special case of the Borda-OWA approach.

In Table 3, we also present the significance test results at significance levels of 0.05 (△▽), 0.01 (▴▾), and 0.001 (▲▼) against the BC baseline. The proposed fuzzy method significantly improves the results, for all the three A levels we experimented with, and in all 4 evaluation measures. MAP value improved by 6.25% comparing to BC, 21.7% comparing to EHD, 25% comparing to CLD and 93.9% comparing to SCD. ANMRR value improoved by 10.2% comparing to BC, 37.89% comparing to EHD, 34.33% comparing to CLD and 85.15% comparing to SCD.

Table 3. Experimental Results

	MAP	P@10	P@20	ANMRR
CLD	0.5046	0.4600	0.3837	0.4198
EHD	0.5183	0.5225	0.4525	0.4309
SCD	0.3254	0.2625	0.1875	0.5786
Borda Count (BC)	0.5973	0.5175	0.4237	0.3444
Fuzzy Fusion, $A = 5\%$	**0.6308**▲	0.5300▲	**0.4450**▲	**0.3125**▼
Fuzzy Fusion, $A = 10\%$	0.6147▲	0.5325△	0.4337△	0.3253▼
Fuzzy Fusion, $A = 50\%$	0.6123▴	**0.5350**▴	0.4350▲	0.3244▼
BC-OWA (Pessimism)	0.5540 ▼	0.4825 ▼	0.3962 ▾	0.3947▲
BC-OWA (Optimism)	0.5802-	0.4650 ▾	0.3837 ▽	0.3453 -
BC-OWA (Neutral)	0.5973-	0.5175-	0.4237-	0.3444-
BC-MAX	0.5552 ▼	0.4875 ▼	0.3962 ▼	0.3935▲
BC-MIN	0.5263 ▼	0.4375 ▼	0.3600 ▼	0.3849△
IRP	0.5574 ▾	0.4550 ▼	0.3687 ▾	0.3574-

5 Conclusions

We proposed a new, simple, and efficient, rank-based late fusion method, employing a fuzzy rule-based system with no need of training data. The method was found to provide statistically significant improvements in retrieval quality over other widely used rank-based fusion techniques such as IRP, Borda Count and derivatives. Although we evaluated on an image database, the method can be directly applied to other media as well. In order to have the proposed method to make sense, we assume that all the rank-lists in the group are considered to contribute equally to the final fused ranking. For the future, we suggest the dynamic calculation of both the number and limits of the Membership Functions of fuzzy system, based on training data.

The proposed method is implemented in the image retrieval system img (Rummager)[3] and is available online[3] along with the image database and the queries.

References

1. Babuska, R.: Fuzzy modeling for control. Kluwer Academic Publishers (1998)
2. Chatzichristofis, S.A., Arampatzis, A., Boutalis, Y.S.: Investigating the behavior of compact composite descriptors in early fusion, late fusion, and distributed image retrieval. Radioengineering 19(4), 725–733 (2010)
3. Chatzichristofis, S.A., Boutalis, Y.S., Lux, M.: Img(rummager): An interactive content based image retrieval system. In: Skopal, T., Zezula, P. (eds.) SISAP, pp. 151–153. IEEE Computer Society (2009)
4. Chatzichristofis, S.A., Boutalis, Y.S., Lux, M.: Spcd - spatial color distribution descriptor - a fuzzy rule based compact composite descriptor appropriate for hand drawn color sketches retrieval. In: Filipe, J., Fred, A.L.N., Sharp, B. (eds.) ICAART (1), pp. 58–63. INSTICC Press (2010)

[3] www.img-rummager.com

5. Datta, R., Joshi, D., Li, J., Wang, J.: Image retrieval: Ideas, influences, and trends of the new age. ACM Computing Surveys 40(2), 1–60 (2008)
6. Garcia-Lapresta, J., Martínez-Panero, M., Meneses, L.: Defining the borda count in a linguistic decision making context. Information Sciences 179(14), 2309–2316 (2009)
7. Jović, M., Hatakeyama, Y., Dong, F., Hirota, K.: Image Retrieval Based on Similarity Score Fusion from Feature Similarity Ranking Lists. In: Wang, L., Jiao, L., Shi, G., Li, X., Liu, J. (eds.) FSKD 2006. LNCS (LNAI), vol. 4223, pp. 461–470. Springer, Heidelberg (2006)
8. Lee, C.: Fuzzy logic in control systems: fuzzy logic controller. I. IEEE Transactions on systems, man and cybernetics 20(2), 404–418 (1990)
9. Lux, M., Chatzichristofis, S.A.: Lire: lucene image retrieval: an extensible java cbir library. In: El-Saddik, A., Vuong, S., Griwodz, C., Bimbo, A.D., Candan, K.S., Jaimes, A. (eds.) ACM Multimedia, pp. 1085–1088. ACM (2008)
10. Manjunath, B., Salembier, P., Sikora, T.: Introduction to MPEG-7: multimedia content description interface. John Wiley & Sons Inc. (2002)
11. MPEG-7. Subjective Evaluation of the MPEG-7 Retrieval Accuracy Measure (ANMRR). ISO/WG11, Doc. M6029 (2000)
12. Muller, H., Clough, P., Deselaers, T.: ImageCLEF: Experimental Evaluation in Visual Information Retrieval, vol. 32. Springer, Heidelberg (2010)
13. Nistér, D., Stewénius, H.: Scalable recognition with a vocabulary tree. In: CVPR (2), pp. 2161–2168. IEEE Computer Society (2006)
14. Ross, T.: Fuzzy logic with engineering applications. Wiley (2004)
15. van Leuken, R.H., Pueyo, L.G., Olivares, X., van Zwol, R.: Visual diversification of image search results. In: Quemada, J., León, G., Maarek, Y.S., Nejdl, W. (eds.) WWW, pp. 341–350. ACM (2009)
16. Zarghami, M.: Soft computing of the borda count by fuzzy linguistic quantifiers. Appl. Soft Comput. 11(1), 1067–1073 (2011)

Annotated Free-Hand Sketches for Video Retrieval Using Object Semantics and Motion

Rui Hu, Stuart James, and John Collomosse

Centre for Vision, Speech and Signal Processing (CVSSP)
University of Surrey, Guildford, Surrey, U.K.
{R.Hu,S.James,J.Collomosse}@surrey.ac.uk

Abstract. We present a novel video retrieval system that accepts annotated free-hand sketches as queries. Existing sketch based video retrieval (SBVR) systems enable the appearance and movements of objects to be searched naturally through pictorial representations. Whilst visually expressive, such systems present an imprecise vehicle for conveying the semantics (e.g. object types) within a scene. Our contribution is to fuse the semantic richness of text with the expressivity of sketch, to create a hybrid 'semantic sketch' based video retrieval system. Trajectory extraction and clustering are applied to pre-process each clip into a video object representation that we augment with object classification and colour information. The result is a system capable of searching videos based on the desired colour, motion path, and semantic labels of the objects present. We evaluate the performance of our system over the TSF dataset of broadcast sports footage.

1 Introduction

Text keywords are the dominant query mechanism for multimedia search, due to their expressivity and compactness in specifying the semantic content (e.g. car, horse) desired within a scene. However, keywords lack the descriptive power to concisely and accurately convey the visual appearance, position and motion of objects. Querying by Visual Example (QVE) offers a solution, yet most video QVE techniques require a photo-real query (e.g. image [33], or video [5]) and so are unsuitable in cases where exemplar footage is absent. Free-hand sketch is a complementary query mechanism for specifying the appearance and motion of multimedia assets, and has recently been applied to video retrieval [8,16]. However the throw-away act of sketch, combined with limited artistic skill of non-expert users, can make unambiguous depiction of objects challenging. Such ambiguity limits the size and diversity of the dataset that can be queried purely by pictorial means. The contribution of this paper is to fuse the orthogonal query methods of *sketch* and *text* — for the first time presenting a QVE system for searching video collections using *textually annotated sketch* queries.

Our system accepts a colour free-hand sketched query annotated with text labels indicating object classification (semantics), and motion cues (arrows) that indicate the approximate trajectory of the desired object. We focus upon these cues to assess relevance, following recent studies [9,8] that observe users to draw upon their *episodic memory* during sketch recall — resulting in sketches exhibiting low spatial and temporal fidelity [34]. Users typically recall the names of a few salient objects in a scene,

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 473–484, 2012.

and their approximate trajectories, rather than their detailed appearance (e.g. shape). Object appearance tends to be depicted coarsely, using a limited yet approximately correct colour palette. Therefore, although users naturally depict an object's shape within a sketch, we *do not currently use shape information* to influence the type of object to retrieve. Rather, our contribution is to combine spatio-temporal position information in the sketch with colour, and the semantic tags associated with the object to create a more scalable solution than that offered by shape alone [8,16].

We represent video as a set of video objects, identified during video ingestion by motion segmentation based on an unsupervised clustering of sparse SIFT feature tracks. A super-pixel representation of video frames is used to aggregate colour information local to each video object. An object class distribution is also computed local to each video object, based on a per-pixel labelling of frames via a random-forest classifier. Thus each spatio-temporal video object is accompanied by colour, semantic and motion trajectory data. At query-time sketched trajectories are matched to the trajectories of video objects using an adapted Levenshtein (edit) distance measure, alongside a measurement of similarity between the colour and semantic distributions of the query and candidate objects.

We describe the extraction and matching of the video object representation in Sec. 3 and 4 respectively, evaluating over a subset of the public TSF dataset in Sec.5.

1.1 Related Work

Sketch based retrieval (SBR) of visual media dates back to the nineties, and the development of image retrieval systems where queries comprised sketched blobs of colour and texture [13,18,30]. Image retrieval using sketched line-art depictions has been addressed by exploring the relationship between image edges and sketched lines. Matusiak *et al.* [27] proposed curvature scale-space [28] as a depiction invariant descriptor. Affine invariant contour representations for SBR were also explored by [17]. The relationship between edge detail and sketches was made explicit by Del Bimbo and Pala [10] where an deformable model derived from the sketch was fitted over image edges via non-linear optimization. More scalable solutions to image SBR have been proposed via the Structure Tensor[11], and the combination of Gradient-Field HoG descriptor and the Bag of Visual Words (BoVW) framework [15] initially proposed for QVE using photographic queries.

Although such sketch based image retrieval (SBIR) may be extended to video through key-frame extraction, motion also plays an important role within video content. A number of approaches [14,2,23,3,1] have explored the description of object trajectory through sketch, but neglect the appearance and semantic properties of the video content. Collomosse *et al.* combined sketched shape, colour and motion cue through free-hand storyboard sketches [8] — solving an inference problem to assign super-pixels in video to sketched objects at query-time. The expense of the inference step motivated Hu *et al.* to consider alternative approaches to matching storyboard sketches [16] using a trellis-based edit distance.

Our system directly builds upon [16], also adopting a edit-distance measure to match tokenized motion trajectories. However our system is unique in considering not only motion and colour, but also the semantic labelling of content within the video. This

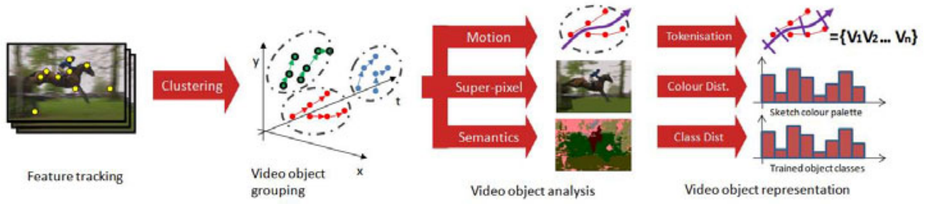


Fig. 1. Video pre-processing pipeline. Space-time video objects are first segmented from sparse tracked feature correspondences. Medial axis of each object's point cloud is extracted and augmented with colour and semantic label distributions obtained from local super-pixels.

overcomes the scalability limitations inherent in the basic appearance features (e.g. colour, shape) considered in recent work [8,16], and inherent in the medium of sketch, by relying instead on a user-annotated semantic labelling of sketched objects.

The consideration of semantics in SBR is currently sparsely researched. Semantic SBIR systems proposed by Liu *et al.* [24] and Wang *et al.* [35,6] demonstrate how annotated examplar images can be exploited to retrieve *images*. Both systems used example images from a database of images (Mindfinder – found by interactive keyword search) to construct the query either using boxes or freehand shapes to depict the shape of the object. These approaches showed how semantic based retrieval is useful in adding spatial information to the user query, but are not suitable to be extended to video. To the best of our knowledge, the extension of semantic sketch to video has not been previously explored in literature.

2 System Overview

Our system parses videos upon ingestion to extract a set of video objects, identified as tracked clouds of sparse feature points (SIFT keypoints) undergoing self-consistent spatio-temporal motion. This motion segmentation step is performed via Affinity Propagation, as outlined in Sec. 3.1. The resulting video objects are analysed further to extract motion, colour and semantic labelling information. Motion trajectory is identified by a space-time curve representing the medial axis of motion, and sampling regular intervals along this curve to encode a series of 'tokens' that are later matched to the sketched curve using a modified Levenshtein (edit) distance. Mean-shift segmentation is applied to each video frame to yield a super-pixel representation, under which we can compute a colour distribution as later described in Sec. 3.2. A per-pixel semantic label is assigned to each video frame using a random-forest based labelling algorithm [32]. Thus the image pixels local to each tracked feature point within a video object grouping contributes to a colour and semantic distribution for that object. These three components (motion, colour, semantics) comprise the video object representation that we match against sketches at query-time (Sec. 4). Fig. 1 outlines the sequence of these pre-processing steps.

3 Video Feature Extraction

Upon addition of a new video to the dataset, we segment each video into clips using shot-detection [37]. Each clip is then processed to identify objects in the video that exhibit coherent motion relative to the background. Color, motion and semantic information is then extracted for each object.

3.1 Motion Segmentation and Trajectory Extraction

Extracting and clustering motion trajectories is crucial for video processing and has been used for event analysis [29,19], pedestrian counting [2] and video retrieval [16]. In this paper, we adopt an unsupervised motion clustering method to group the trajectories, generated by SIFT feature tracking, into different categories. The dominant trajectory of each motion category is represented with a piece-wise cubic β-spline.

Trajectory Extraction. SIFT feature tracking has been used for video stabilization [4], object recognition and tracking [26,36], as well as video retrieval [16]. In this paper we use SIFT keypoints matching to compensate the camera motion and generate the individual trajectories.

SIFT keypoints are detected on each frame and matched between each two adjacent frames. We iteratively correspond descriptors using the L^1 norm, the correspondences where the distance ratio of the best two matches falls below tolerance are disregarded as in [25]. Keypoints within the TV logo areas are not considered, and due to the constant location of such logos in our dataset (TSF [8]), they are trivially masked out. The inter-frame homography is estimated via MAPSAC using the keypoint correspondences. The locations of SIFT keypoints are transformed using the inverse homography to effect compensate for camera motion during the clip. Keypoints moving below a threshold velocity are discarded as unwanted background detail.

The correspondences of keypoints after camera motion compensation generate a set of individual trajectories. In order to filter and remove erroneous correspondences, we delete and interpolate the position keypoints where the inter-frame displacement deviates from the local average. Trajectories are fragmented into separate individual trajectories from the point of sudden changes of velocity [16].

Trajectory Clustering. Tracklet representations, such as our SIFT trajectories, are frequently adopted as a basis for motion clustering in structure-from-motion applications [2,23,3,1], though often at the expense of imposing a simplifying motion model (e.g. near-linear motion [16]). In [16] we construct a $5D$ feature space from the mean space-time location (x, y, t) and velocity $(\Delta x, \Delta y)$ of each trajectory. However, despite the simplicity of individual trajectories, a grouping of trajectories can encode non-linear motion. In this paper, we perform this grouping via Affinity Propagation clustering as follows.

Given the trajectory set, we compute the *affinity* of each trajectory pair and represent each as a node in an *affinity graph*. Each edge of the graph is weighted in proportional the affinity between the two nodes. Only trajectory pairs that share at least one common frame are considered to compute the affinity; the similarity between trajectories that do not share a common frame is set to be 0.

Let A and B be two trajectories sharing at least one common frame. The dissimilarity between A and B is defined as the distance of these two trajectories at a time instance where they are the most dissimilar:

$$d^2(A,B) = max_t d_t^2(A,B). \tag{1}$$

$d_t^2(A,B)$ is the distance of A and B at the particular time instant t:

$$d_t^2(A,B) = d_{sp}(A,B)\frac{(u_t^A - u_t^B)^2 + (v_t^A - v_t^B)^2}{3\sigma_t^2}. \tag{2}$$

where $d_{sp}(A,B)$ is the average spatial distance of A and B in the common time window; $u_t := x_{t+3} - x_t$ and $v_t := y_{t+3} - y_t$ measures the motion aggregation of the two trajectories over 3 frames; $\sigma_t = min_{a\in\{A,B\}}\Sigma_{t'=1}^{3}\sigma(x_{t+t'}^a, y_{t+t'}^a, t+t')$.

The similarity of trajectory A and B is then computed as:

$$sim(A,B) = exp(-kd^2(A,B)). \tag{3}$$

where in our experiments, constant $k = 0.1$. Having computed the affinity matrix, we apply the Affinity Propagation (AP) algorithm [12] to group the trajectories into different motion categories. In contrast to k-means clustering, AP requires only the similarity between trajectories as input, and does not require prior knowledge of the number of the clusters. Rather, AP considers all data points as potential exemplars and iteratively exchanges messages between data points until the corresponding clusters gradually emerges.

Motion Representation. We extract a representative *medial axis* from each clustered component by approximating its global trajectory with a piece-wise cubic β-spline. The solution is unavailable in closed-form due to the typical presence of outliers and piece-wise modelling of complex paths. We therefore fit the spline using RANSAC to select a set of control points for the β-spline from the set of keypoints in the corresponding cluster. One keypoint is selected at random from each time instant spanned by cluster, to form the set of control points. The fitness criterion for a putative β-spline is derived from a snake [20] energy term, which we seek to minimize:

$$E = \alpha * E_{int} + \beta * E_{ext} \tag{4}$$

$$E_{int} = \int_{s=0}^{1} \left|d^2B(s)/ds^2\right|^2 \tag{5}$$

$$E_{ext} = \Sigma_{t=0}^{T}\left[\frac{1}{|\mathcal{P}_t|}\Sigma_{p\in\mathcal{P}}|p - B(t/T)|^2\right] \tag{6}$$

where $B(s)$ is the arc-length parameterised β-spline, and $\mathcal{P}_t, t = \{0..T\}$ is the subset of keypoints within the cluster at time t. We set $\alpha = 0.8$, $\beta = 0.2$ to promote smooth fitting of the motion path.

3.2 Color Feature Extraction

After motion clustering, each group of individual trajectories represents motion from one moving object which we term a *video object*. However, the sparsely detected SIFT keypoints within the video object typically exhibit insufficient pixel coverage to sample the colour appearance information of the video object.

We therefore segment each video frame into super-pixels of homogeneous color, using mean-shift [7] algorithm. The color of each keypoint along the trajectory is deemed as the mean color of the underlying region, and a weighted contribution is made to the histogram proportional based on the area of the region and the number of times that region been touched by trajectories from the according group.

The color distribution histogram is computed on all the keypoints along the trajectories of that category.

3.3 Semantic Labelling

Pixelwise Semantic Segmentation has started to gain attention in recent years, approaches such as TextonBoost[32] and ALE[21] provide a accurate way of segmenting images. These approaches suffer from the computation of complex filter banks and assignment at test time, and the addition of K-Means at train time. An alternative to these approaches Semantic Texton Forests (STF)[31] used Extremely Randomised Decision Forests to classify pixels, these ensembles of decision trees are fast to train and test their inherent random approach allows them to be flexible to a variety of applications. In evaluation the STF computational performance makes it an attractive approach for semantically segmenting videos allowing for database scalability, the alternative texton based approaches are generally too slow to handle large datasets.

The STF approach is composed of two components, training of an ensemble of random decision trees. These trees are trained based on CIELab colour value differences within a window around the training point. The comparisons of values are based on a random comparison function, these can be addition, subtraction, absolute difference for example. The second component of this approach is a global image classification, this trains a OneVsOthers SVM other each of the classes, the approach uses a unique kernel based on Pyramid Matching Kernel(PMK). The PMK is adapted from the random decision forest based on node counts of the ensemble classified image, this adds some spatial consistency of class adjacent class context within images.

We apply the STF classifier to label the pixels in each the video frame as being in one of a pre-trained set of categories. In our experiments we train STF over twelve categories — corresponding to object classes with the TSF dataset, e.g. horse, grass, person, car. We count the frequency of label occurrence over all keypoints present within the spatio-temporal extend of the video object. The resulting frequency histogram is normalized via division by the number of keypoints, yielding a probability distribution for the video object's semantic label over the potential object categories.

4 Matching the Annotated Sketch

The basic unit of retrieval in our system is the video object, parsed via the process of Sec. 3. Video retrieval proceeds by independently estimating the similarity of each video object $v_i \in \mathcal{V}$ to the annotated sketch Q supplied by the user. This provides both a video and temporal window containing relevant content, which can be presented in a ranked list to the user.

The probability of object v_i corresponding to a given sketch query is proportional to a product of three orthogonal cues:

$$p(V|Q) \propto \operatorname*{argmax}_i \left[sim_C(v_i) \times sim_M(v_i) \times sim_S(v_i) \right]. \tag{7}$$

where sim_C, sim_M and sim_S denote the color, motion and semantic similarity of the i^{th} video volume to the query sketch Q respectively — as defined below.

4.1 Motion Similarity (sim_M)

We follow the observations of [9], who observe that users depict object motion against a static background (the drawing canvas) regardless of any global camera motion present in the scene. This leads to a mapping between the sketch canvas and the camera-motion compensated frame derived from the inter-frame homographies computed within the video. Introducing a further assumption, we consider the sketched trajectory to depict the entirety of a video object's motion. We are then able to construct a space-time (x,y,t) trajectory from the sketched motion path — with time (t) spanning the temporal extent of the video object being matched, and (x,y) spanning the total camera panorama covered during that time.

The problem of matching the sketched motion path is thus reduced the problem of assessing the similarity of two space-time trajectories; that derived from the sketch, and that derived from the medial axis ($\beta-$spline) fitted to the video object's keypoints in subsec. 3.1.

Tokenization. We match the sketched motion trajectory to that of the video object by considering the path as a sequence of discrete movements, which we achieve by sampling the trajectory at regular arc-length intervals. In our experiments we sample ten intervals. A codebook of space-time moves is generated, and each trajectory segment assigned to a token in the codebook. The two strings are compared efficiently using the Levenshtein (edit) distance [22]; the minimal cost of operations required to transform one token sequence to the other. In our system we use the classical Levenshtein distance comprising insertion, deletion and substitution operators. The cost of insertion and deletion are defined as unity (i.e. high), with the substitution cost defined as the probability of two motion token being similar (derived from their space-time position and angle). The use of an edit distance measure enables two similar trajectories that exhibit temporally misaligned segments (due to inaccuracies in the sketch) being matched with low cost.

4.2 Colour Similarity (sim_C)

Colour similarity is measured by comparing the non-parametric colour distribution of the sketched object with that of the video object being compared. The colour distribution is determined by computing a normalised frequency histogram from the colours of pixels comprising the sketch. The set of colours comprising the histogram bins are derived from the discrete 16 colour palette available to the user during sketching; a similar palette is used when extracting the colour distribution from video objects with pixel colours conformed to this palette via nearest-neighbor assignment in CIELab space. The L^2 norm distance is used to compute the distance between two color histograms.

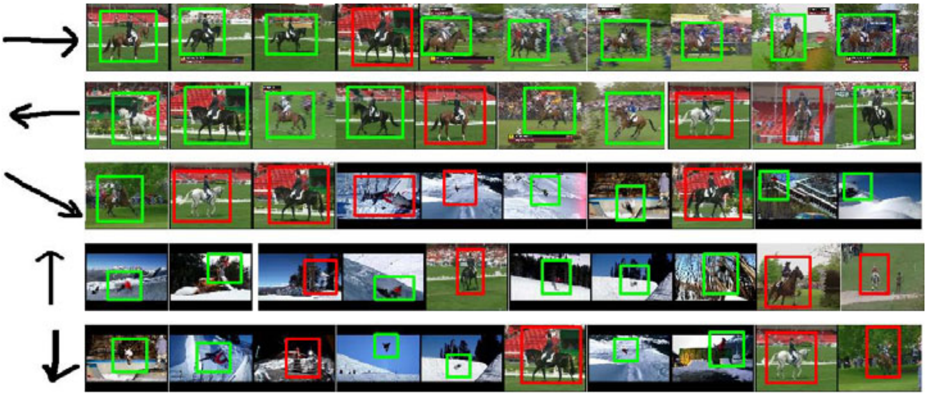


Fig. 2. Motion stroke queries and their top 10 returned results

4.3 Semantic Similarity (sim_S)

Our system enables the user to tag objects with a single object class, creating a class distribution for the sketched object with all contribution assigned to a single bin (object category). This histogram is directly compared with that of the video object, using the L^2 norm distance.

5 Experiments and Discussion

We evaluate our system over a subset of the TSF dataset, composed of 71 horse racing and 71 snow skating clips. For semantic labeling of the video frames, we define eight different semantic categories: person, horse, grass, snow, stands, plants, sky and void – although the void class is ignored for training. We manually label 143 frames from 12 video clips as training set, to classify the rest video frames.

Our system is tested in four different ways: use motion information alone as query; motion with color; motion with semantics, motion together with color and semantic information as queries. The example queries and their top 10 returned results are shown in Fig. 2 - Fig. 5 respectively. The positive results are highlighted in green and negative results are highlighted in red.

In Fig. 2 we demonstrate the effectiveness of our motion extraction and matching approach. The results over the selection of queries available for this dataset produce a Mean Average Precision(MAP) of 38.6%. Within the combination of motion and colour as queries as shown in Fig. 3, there is no shape information encoded therefore objects despite there depiction are referred to abstractly as a colour blob. These results demonstrate MAP of 42.7%.

The fusion of annotated class and motion as shown in Fig. 4, achieves a MAP of 75.85%. This improvement in contrast to motion alone demonstrates the advantages of annotated class as a facet of information.

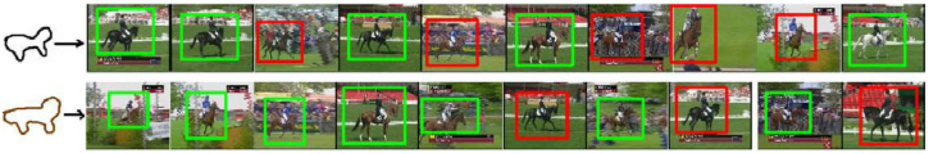


Fig. 3. Motion with color queries and their top 10 returned results

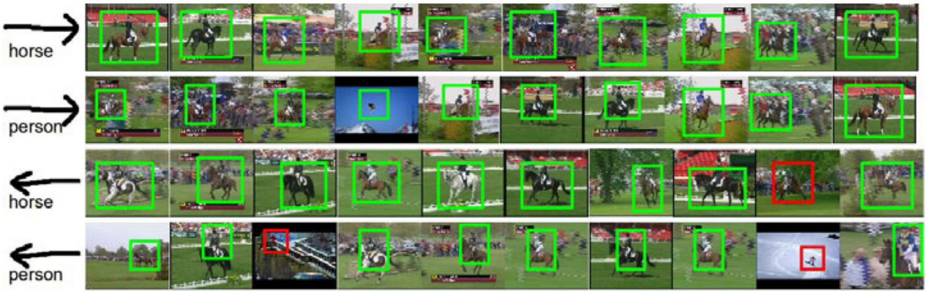


Fig. 4. Semantic with motion strokes as queries, and their top 10 returned results

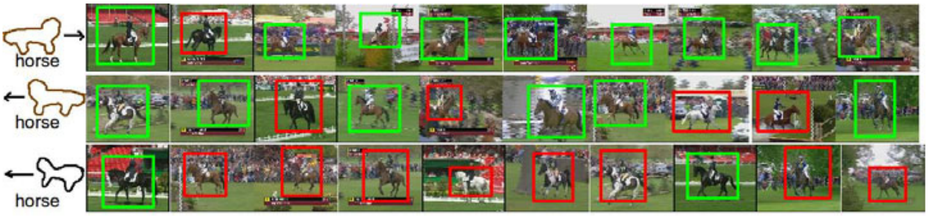


Fig. 5. Semantic query sketches and their top 10 returned results

When using the mix of all the different information sources as shown in Fig. 5, We achieve a MAP of 51.22%. The reduction in MAP is due to two main reasons, the difficulty in describing a feature points colour accurarly – generally in most scenarios the horse has a variety of colours on them even with the mean-shift filtering there are still regions such as the leg of the rider that are difficult for both the semantic segmentation and the colour description to deal with. Also with the amalgamation of all the different facets of information reduces the possible accurate results in the dataset down making it very difficult to get an accurate result.

Average Precision Recall curves of the four evaluated systems are ploted in the left of Fig. 6. From the curves we can see that by adding semantic information into the color, and motion query can significantly improve the performance of the retrieval system. The figure on the right side of Fig. 6 show the precision recall curve of each of the three queries in Fig. 5.

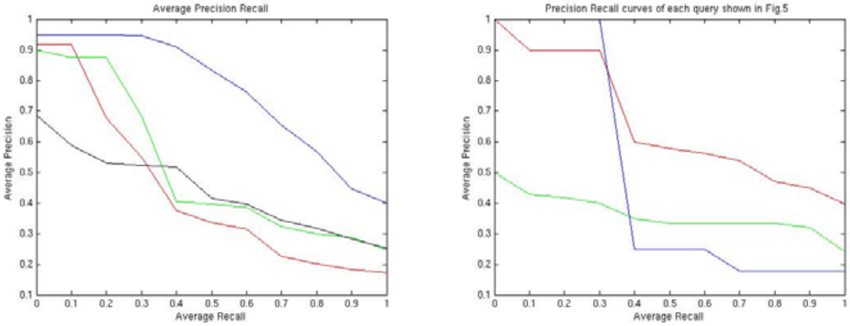


Fig. 6. (left)Average Precision Recall curves of using motion (black curve), motion with color (red curve), motion with semantics (blue curve), motion with color and semantics together (green curve) based retrieval. (right) Precision Recall curves of the three queries shown in Fig. 4. The curve for the query on the top is shown in red; the middle is shown in green; and the bottom one is shown in blue.

6 Conclusion

We have presented a video retrieval system driven by annotated sketched queries. Salient objects are identified within video through unsupervised clustering of SIFT keypoint trajectories in a camera-motion compensated frame. Each object is analysed to develop an augmented object description comprising data on space-time locus (spatial position and motion path), colour and object category. The motion is derived from a $\beta-$spline robustly fitted in space-time to keypoints comprising the object. Although semantic sketch based retrieval has been recently applied to images [35,6], our system is the first to explore the use of semantic (annotated) sketches for video retrieval. We have demonstrated improved retrieval performance through the integration of semantics, over previous sketch based video retrieval techniques using colour and motion alone [16].

Having incorporated multiple orthogonal cues into a video retrieval system, a natural direction for future work is explore the relative weightings of those cues. Such weightings seemingly cannot be prescribed in advance; a user sketching a red blob labelled “car” travelling right, may assign greater worth to red cars travelling left — or to yellow cars travelling right. Interactive relevance feedback, enabling re-weighting of the terms of eq. 7 seems a promising approach to resolving this ambiguity behind a user’s intention.

References

1. Anjum, N., Cavallaro, A.: Multifeature object trajectory clustering for video analysis. IEEE Trans. on Circuits and Systems for Video 18(11), 1555–1564 (2008)
2. Antonini, G., Thiran, J.P.: Counting pedestrians in video sequences using trajectory clustering. IEEE Tran. on Circuits and Systems for Video 16(8), 1008–1020 (2006)
3. Bashir, F.I., Khokhar, A.A., Schonfeld, D.: Real-time motion trajectory-based indexing and retrieval of video sequences. IEEE Trans. Multimedia 9(1), 58–65 (2007)

4. Battiato, S., Gallo, G., Puglisi, G., Scellato, S.: Sift features tracking for video stabilization. In: International Conference on Image Analysis and Processing, pp. 825–830 (2007)
5. Bertini, M., Del Bimbo, A., Nunziati, W.: Video Clip Matching Using MPEG-7 Descriptors and Edit Distance. In: Sundaram, H., Naphade, M., Smith, J.R., Rui, Y. (eds.) CIVR 2006. LNCS, vol. 4071, pp. 133–142. Springer, Heidelberg (2006)
6. Cao, Y., Wang, H., Wang, C., Li, Z., Zhang, L., Zhang, L.: Mindfinder: interactive sketch-based image search on millions of images. In: ACM Multimedia, pp. 1605–1608 (2010)
7. Christoudias, C.M., Georgescu, B., Meer, P.: Synergism in low level vision. In: ICPR, vol. 4, p. 40150 (2002)
8. Collomosse, J., Mcneill, G., Qian, Y.: Storyboard sketches for content based video retrieval. In: ICCV (2009)
9. Collomosse, J., Mcneill, G., Watts, L.: Free-hand sketch grouping for video retrieval. In: ICPR (2008)
10. del Bimbo, A., Pala, P.: Visual image retrieval by elastic matching of user sketches 19(2), 121–132 (1997)
11. Eitz, M., Hildebrand, K., Boubekeur, T., Alexa, M.: Sketch-based image retrieval: Benchmark and bag-of-features descriptors. In: IEEE TVCG, vol. 99 (2010)
12. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. Science 315, 972–976 (2007)
13. Hafner, J., Sawhney, H.S., Equitz, W., Flickner, M., Niblack, W.: Effcient color histogram indexing for quadratic distance. IEEE PAMI 17(7), 729–736 (1995)
14. Hsieh, J., Yu, S., Chen, Y.: Motion-based video retrieval by trajectory matching. IEEE Tran. on Circuits and Systems for Video 16(3), 396–409 (2006)
15. Hu, R., Barnard, M., Collomosse, J.: Gradient field descriptor for sketch based retrieval and localization. In: ICIP, pp. 1025–1028 (2010)
16. Hu, R., Collomosse, J.: Motion-sketch based video retrieval using a trellis levenshtein distance. In: Intl. Conf. on Pattern Recognition, ICPR (2010)
17. Ip, H.H.S., Cheng, A.K.Y., Wong, W.Y.F., Feng, J.: Affine-invariant sketch-based retrieval of images. In: International Conference on Computer Graphics, pp. 55–61 (2001)
18. Jacobs, C.E., Finkelstein, A., Salesin, D.H.: Fast multi-resolution image querying. In: Proc. ACM SIGGRAPH, pp. 277–286 (1995)
19. Jung, C.R., Hennemann, L., Musse, S.R.: Event detection using trajectory clustering and 4-d histograms. IEEE Trans. Circuits Syst. Video Techn. 18(11), 1565–1575 (2008)
20. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. Intl. Journal of Computer Vision 4(1), 321–331 (1987)
21. Kohli, P., Ladický, L., Torr, P.H.S.: Robust Higher Order Potentials for Enforcing Label Consistency. International Journal of Computer Vision 82, 302–324 (2009)
22. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Technical Report 8, Soviet Physics Doklady (1966)
23. Li, X., Hu, W., Hu, W.: A coarse-to-fine strategy for vehicle motion trajectory clustering. In: ICPR, pp. 591–594 (2006)
24. Liu, C., Wang, D., Liu, X., Wang, C., Zhang, L., Zhang, B.: Robust semantic sketch based specific image retrieval. In: Proc. Intl. Conf. and Multimedia Expo. (2010)
25. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV 60, 91–110 (2004)
26. Lpez-Garca, F.: Sift features for object recognition and tracking within the ivsee system. In: ICPR, pp. 1–4. IEEE (2008)
27. Matusiak, S., Daoudi, M., Blu, T., Avaro, O.: Sketch-Based Images Database Retrieval. In: Jajodia, S., Özsu, M.T., Dogac, A. (eds.) MIS 1998. LNCS, vol. 1508, pp. 185–191. Springer, Heidelberg (1998)
28. Mokhtarian, F., Mackworth, A.K.: A theory of multiscale, curvature-based shape representation for planar curves. IEEE Trans. Pattern Anal. Mach. Intell. 14, 789–805 (1992)

29. Piciarelli, C., Foresti, G.L.: On-line trajectory clustering for anomalous events detection. Pattern Recogn. Lett. 27, 1835–1842 (2006)
30. Di Sciascio, E., Mingolla, G., Mongiello, M.: CBIR over the web using query by sketch and relevance feedback. In: Proc. Intl. Conf. VISUAL, pp. 123–130 (1999)
31. Shotton, J., Johnson, M., Cipolla, R.: Semantic texton forests for image categorization and segmentation. In: CVPR, pp. 1–8 (2008)
32. Shotton, J., Winn, J.M., Rother, C., Criminisi, A.: *TextonBoost*: Joint Appearance, Shape and Context Modeling for Multi-class Object Recognition and Segmentation. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 1–15. Springer, Heidelberg (2006)
33. Sivic, J., Zisserman, A.: Video google: a text retrieval approach to object matching in videos. In: ICCV, pp. 2:1470–2:1477 (2003)
34. Tulving, E.: Elements of episodic memory (1983)
35. Wang, C., Li, Z., Zhang, L.: Mindfinder: image search by interactive sketching and tagging. In: WWW, pp. 1309–1312 (2010)
36. Xu, J., Ye, G., Zhang, J.: Long-term trajectory extraction for moving vehicles. In: IEEE International Workshop on Multimedia Signal Processing, pp. 223–226 (2007)
37. Zhang, H., Kankanhalli, A., Smoliar, S.W.: Automatic partitioning of full-motion video. Multimedia Systems 1(1), 10–28 (1993)

Finding Suits in Images of People

Lei Huang[1,2], Tian Xia[1], Yongdong Zhang[1], and Shouxun Lin[1]

[1] Advanced Computing Research Laboratory, Institute of Computing Technology,
Chinese Academy of Sciences, Beijing 100190, China
[2] Graduate University of Chinese Academy of Sciences, Beijing 100049, China
{huanglei,txia,zhyd,sxlin}@ict.ac.cn

Abstract. Clothing style is a salient feature for understanding images of people. To automatically identify the style of clothing that people wear is a challenging task. Suit as one of the clothing style is a key element in many important activities. In this paper, we propose a novel suits detection method. By analyzing the style of clothing, we propose the color features, shape features and statistical features for suits detection. Experiments with five popular classifiers have been conducted to demonstrate that the proposed features are effective and robust. Comparative experiments with Bag of Words (BoW) method demonstrate that the proposed features are superior to BoW which is a popular method for object detection. The proposed method has achieved promising performance over our dataset, which is a challenging web image set with various styles of clothing.

Keywords: Suits Detection, Photo Ranking, People Search, Clothing Style.

1 Introduction

People always wear different styles of clothing when participate in different activities. As shown in Fig.1, when people attend a meeting they always wear suits and when people do sports they always wear sportswear. Therefore, detecting the style of clothing is very beneficial to understand the content of images with people. Meanwhile, clothing provides significant information in people search in consumer photo albums and surveillance videos, e.g., Daniel et al. [1] used the color feature of clothing to perform attribute-based people search in surveillance environments. Besides the color feature, the style is also an important factor to reflect the characteristic of clothing. As shown in Fig. 2, in consumer photo selection systems, finding photos of people wearing suits from an increasing amount of personal photos is also very useful.

Many researchers have noted that clothing feature is important in two computer vision tasks [2-7], including human detection [2] and human recognition [3-7]. Sprague et al. [2] used the segmentation results of clothing to detect human in still images. In [3-7], researchers took the clothing feature as context information to aid human recognition. Song et al. [3] used the trained code-words to represent the clothing region, and this is an effective method concerning clothing's different types of color and texture. Gallagher et al. [4, 5] represented the clothing via three color features and two texture features. For the color features, they used the values of luminance-chrominance

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 485–494, 2012.

(a) Meeting (b) Playing Football & Baseball (c) Cooking

Fig. 1. Clothing style is a salient feature to understand images of people. (a) When people wear suits, they may be attending a meeting. (b) When people wear sportswear, they may be playing. (c) When people wear chef apparel, they may be cooking.

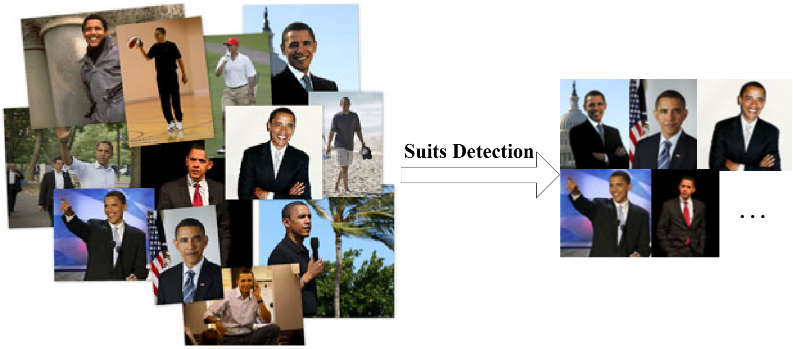


Fig. 2. Finding Obama in suits through suits detection, which is useful in consumer photo selection systems and attribute-based portrait search

space (LCC) and for the texture features, they used the responses to a horizontal and vertical edge detector. Zhang et al. [6] exploited the spatial relationship of features. They used the color signature, color pyramid and texture pyramid to represent the clothing. Khoury et al. [7] employed 3D histogram of the dominant color for clothing matching. In addition, there are many other studies on clothing analysis. Chen et al. [8] proposed an And-Or graph to represent the wide diversity of cloth configurations for clothing sketching. Tian et al. [9] proposed a clothing matching method to help the blind or color blind people.

All the above researches concentrated on extracting clothing features for clothing matching, using the segmentation of clothing for human detection and sketching. There are few researches focusing on recognizing style of clothing automatically.

There are various styles of clothing, e.g. suit, t-shirt and skirt. In this paper, we focus on finding suits in images of people for that suits have the uniform appearance and are generally a key element in many important activities. However, the features proposed in [3-7] [9] cannot be used in suits detection for the colors and textures are different in suits. Furthermore, there are some popular local features which have received a lot of research attention in recent years [3]. These features have been successfully used in many applications. However, the detectors of most of the local features are based on local extreme which cannot work if the clothing regions don't have textures (e.g., a suit always has a single color). Thus some more effective features are desired. The overview of our approach is shown in Fig. 3. To the best of our knowledge, we are the first one exploring visual features to find suits in images of people.

The contribution of the paper is that, by analyzing lots of suits and non-suits images, we discover several regular patterns for suits and propose three color features, one shape feature and two statistical features for suits detection (section 2). The proposed features will also be helpful for other clothing style detection (section 4).

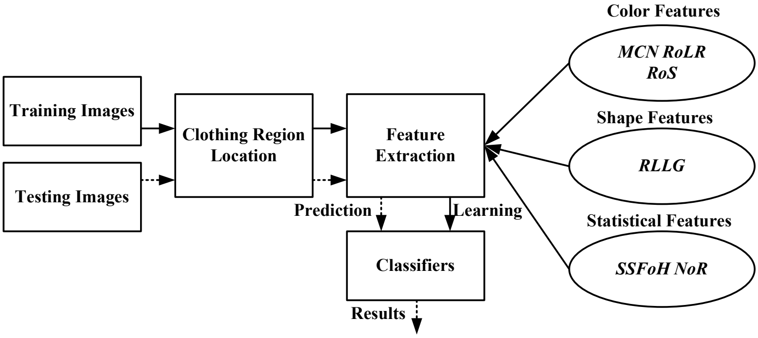


Fig. 3. Overview of the proposed method

2 Visual Features for Suits Detection

By analyzing the appearance of clothing, color features, shape features and statistical features are proposed for detecting photos of people wearing suits. The details of these features are described in this section.

2.1 Clothing Region Location

Before extracting clothing features, the clothing region should be located. A modified version of [10] which can handle rotated faces is performed first to get the face regions. And the clothing regions are acquired in rectangular regions below the face [3]. All images are normalized by the face size. In our experiments, the face region is normalized to 100*80 pixels. The clothing regions are located 10 pixels below the faces. And the size of clothing regions is 200*180. Fig.4 shows the results of the clothing region location. All the features presented below are computed in the clothing region CR . Since all features are computed in CR, our method focuses on finding photos of frontal view people wearing suits.

2.2 Color Features

Color is an important factor of clothing. Since the colors are different for different suits, conventional color features (e.g., values of luminance-chrominance space [4], color pyramid [6]) are not appropriate for suits detection, some novel color-related features should be explored. By observing lots of clothing images, we find that there are some

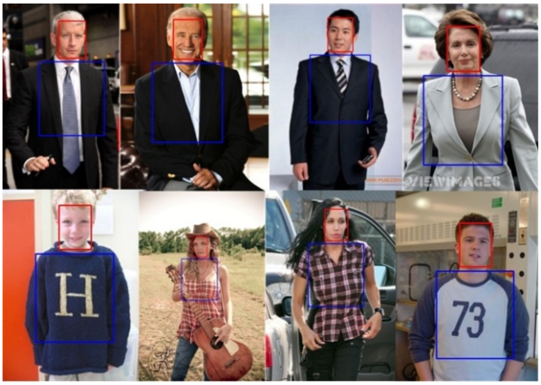


Fig. 4. Results of clothing region location. Face regions are shown in red rectangles and clothing regions are shown in blue rectangles.

rules of suits on the aspect of color: (1) Suits usually have no more than 3 main colors. (2) In the case of frontal view portrait, the distribution of the main color in the left region and right region of the suits is symmetric. (3) When people wear suits, there will be small skin region exposed in the suits location. Thus we define three features: Main Color Number (MCN), Ratio of Left-Right ($RoLR$) and Ratio of Skin(RoS).

To solve the problem caused by the lighting changes and self-shadow we use the color quantization method proposed in [9]. The clothing region CR_{rgb} is quantized to 11 colors image CR_{11} in HSI color space. First, “white”, “black” and “gray” are defined based on saturation S and luminance I. Then, “red”, “orange”, “yellow”, “green”, “cyan”, “blue”, “purple” and “pink” are defined based on the hue information. The results of color quantization are shown in Fig.5. After color quantization, MCN, $RoLR$ and RoS are defined as follows:

MCN: MCN is the number of colors which occupy larger than 10% of clothing region CR_{11}.

$RoLR$: $RoLR$ is the ratio of $ratio\_left$ to $ratio\_right$ which is computed by Eq. (1).

$$RoLR = \frac{ratio\_left}{ratio\_right} \quad , \tag{1}$$

where $ratio\_left = \frac{\#MC\_LCR}{\#LCR}$, $ratio\_right = \frac{\#MC\_RCR}{\#RCR}$, $\#MC\_LCR$ is the area of the main color regions in left clothing region, $\#LCR$ is the area of left clothing region. $\#MC\_RCR$ is the area of the main color regions in right clothing region. $\#RCR$ is the area of right clothing region. The left clothing region (LCR), middle clothing region (MCR) and right clothing region (RCR) are defined in Fig.5.

This feature also can get rid of profile-clothing images, e.g. Fig.5 (g).

RoS: RoS is the ratio of the skin area in the clothing region. Skin detection [11] is performed on CR_{rgb} to get the skin mask CR_{skin}. Morphology processing is

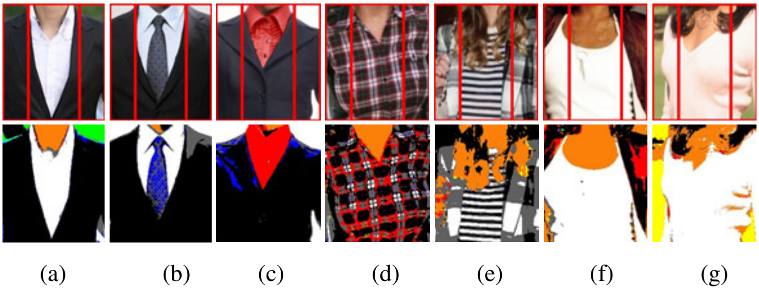


Fig. 5. Three portions of clothing region (top row) and the results of color quantization (bottom row). (a)-(c) Suits. (d)-(g) Non-suits. The left region and right region occupy 0.25 the width of the clothing region each and the middle region occupy 0.5 the width of the clothing region.

implemented on CR_{skin} to remove small regions. The results of skin detection are shown in Fig.6. RoS is computed as follows:

$$RoS = \frac{\#skin}{\#CR} \quad , \tag{2}$$

where $\#skin$ is the area of skin region in the skin mask CR_{skin} , and $\#CR$ is the area of clothing region.

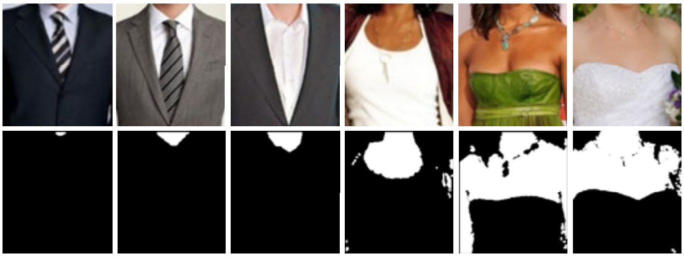


Fig. 6. Results of skin detection in clothing regions

2.3 Shape Features

There are some regular shape patterns of the suits. The most significant pattern is the line feature as shown in Fig.7. The lines at the middle of the clothing region form an inverted triangle. Meanwhile, the two lines are not intersecting in some cases as shown in the fourth column of Fig.7. Therefore, we use the ratio of the longest lines to the clothing region girth ($RLLG$) to represent this feature. $RLLG$ can be obtained by Eq.(3).

$$RLLG = \frac{L_{\max}}{Girth_{CR}} , \tag{3}$$

where $L_{\max} = \max_{l \in LineSet} |l|, Girth_{CR}$ indicates the girth of the clothing region CR , $|l|$ means the length of l . $LineSet$ denotes a line set acquired as follows: First, the segmentation method JSEG [12] is implemented to get the edges in clothing region.

JSEG is more robust to clothing images than canny edge detection which will cause more false edges induced by self-shadow and self-fold of the clothing. Then, the Hough line detection [13] is carried out on the edge set acquired by JSEG to get the candidate line set. There are still some useless lines in the candidate line set, such as too short lines, horizontal lines, lines induced by the self-fold and self-shadows of the clothing and lines induced by the boundary of the skin region. Therefore, the line filter algorithm is carried out. Following lines are filtered: with length shorter than one-fifth of the face width, with slope smaller than 0.7, with same color on both sides, with too much skin region on one side. The results of *LineSet* are shown in Fig.7.

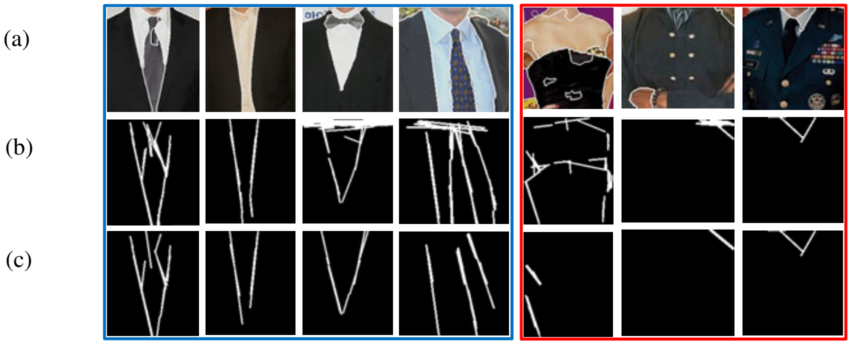


Fig. 7. Results of line detection in clothing regions. (a) Results of JSEG; (b) Results of Hough line detection; and (c) Results of our line filter algorithm.

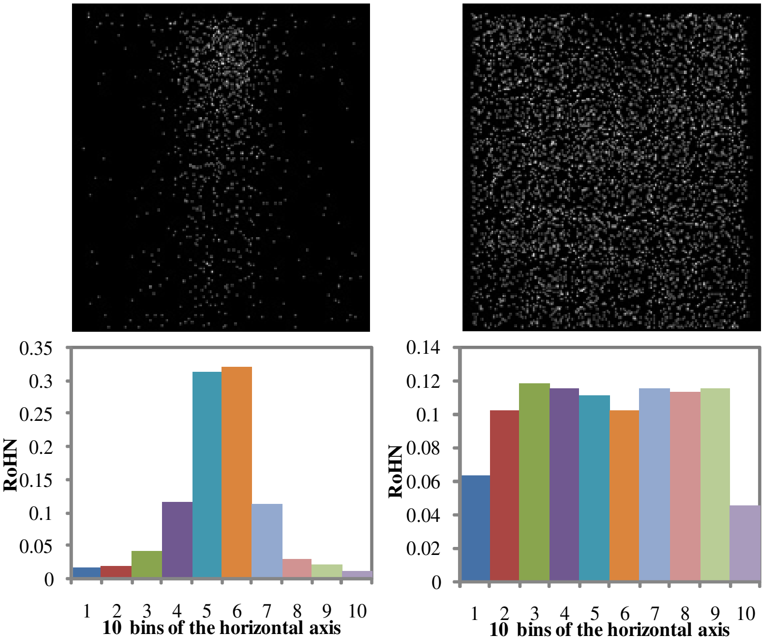
2.4 Statistical Features

There are some salient statistical features in suits. We compute a map which comes from the statistics of the location distribution of the Harris corners in the clothing regions. The map is shown in Fig.8. We use 100 suits images and 100 non-suits images (training data set in section 3.1) to get the map. For the suits images we can see, the corners mainly concentrate in the middle of the clothing regions. For non-suits images, the corner distribution is uniform. Therefore we define the spatial statistical feature of Harris Corner (*SSFoH*) to represent this regular pattern. Clothing regions are divided into three regions as shown in Fig.5. *SSFoH* can be calculated by Eq.(4).

$$SSFoH = \frac{HNoM}{HNoLR} , \tag{4}$$

where *HNoM* is the number of Harris Corners in the middle clothing region, *HNoLR* is the number of Harris Corners in the left and right clothing region.

Another significant statistical feature of suits is the blob statistical features. Suits always have simple texture and a small number of blobs (connected regions), e.g. Fig.5 (a)-(c). But for some non-suits clothing, the texture is complex and the number of connected regions is large, e.g. Fig.5 (d)-(e). Thus, we define Number of Connected Regions (*NoR*). *NoR* can be obtained on the quantized image CR_{11}. Morphological processing is processed on CR_{11} first to remove small regions.



(Suits) (Non-Suits)

Fig. 8. Statistical results of Harris corner location for suits and non-suits. The point in the images at top row means the location of Harris corners in clothing region and the brightness of the point indicates the corner number, the whiter the more. The bar charts are the statistical results projecting to horizontal axis. The horizontal axis is divided into 10 bins, each bar indicates the ratio of the corner points falling in the bin to the total corner number (RoHN).

3 Data Set and Experiments

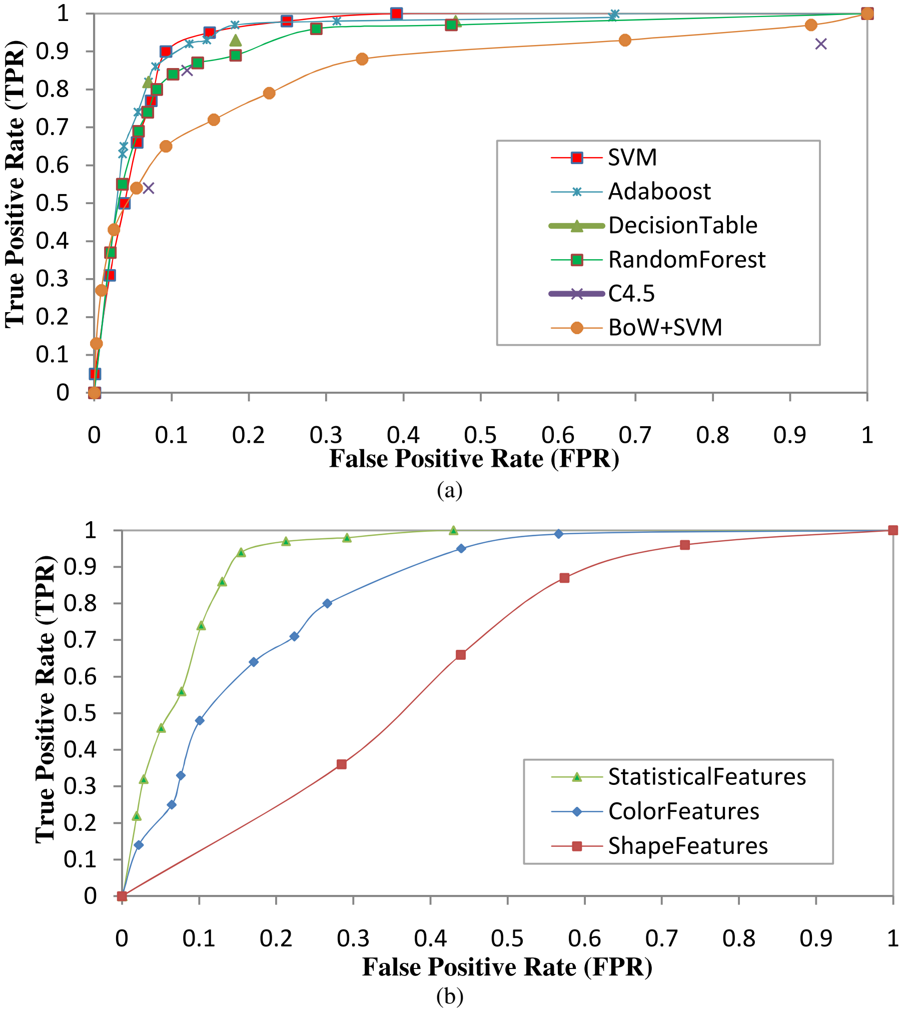
3.1 Data Set

A consumer image collection of 1500 images of people is constructed to evaluate the proposed features. All the images come from “flickr”, “baidu” and “google”. We use the keywords like *“portrait”*, *“portrait+suits”*, *“people+suits”* to search these images. The image collection contains 200 suits images and 1300 non-suits images. 100 suits and 200 non-suits are used for training, other 100 suits and 1100 non-suits are used for testing. Some samples of our dataset are shown in Fig.9.

3.2 Experimental Results

To demonstrate the effectiveness and robustness of the features, we use five popular classifiers including SVM [14], Random Forest [15], Adaboost [16], Decision Table [17] and C4.5 [18]. True Positive Rate (TPR) and False Positive Rate (FPR) are used

Fig. 9. Some samples in our dataset. The first line is suits images and the second line is non-suits images. Most of the suits images are frontal. The non-suits images contain various style of clothing, e.g. skirt, t-shirt, shirt and sweater.



(a)

(b)

Fig. 10. Performance comparison. (a) Performance of five classification and BoW method. (b) Performance of each type features based on SVM.

for evaluation. The results are shown in Fig.10(a). We can see that all the five classifiers get TPR 80% under FPR less than 10%. This shows the effectiveness and robustness of our features. The SVM method gets the best result with TPR of 90% under FPR of 9.2%. To illustrate that our features are superior to state of the art, comparison to Bag of Words (BoW) [19] is performed. In the BoW method, we use harris corner detector since we have normalized the images based on face regions. The descriptor we use is SIFT descriptor [20]. SVM is used as the classifier. The performance of BoW method is shown in Fig.10(a): BoW+SVM. We can see that our features outperform BoW method.

To further evaluate the contribution of each type of features, comparative experiments are performed. SVM which gets the best performance in the above experiment is used as the classifier. The results are shown in Fig.10(b). From the results we can see that every type of our features is effective. In low True Positive Rate, False Positive Rate of all three types of features is under 20%. The statistical features are the most discriminative of all the features. The shape features give the minimum contribution which doesn't meet our intuition. By analyzing this issue, we find that this is caused by the non-rigid deformation of clothing which reduces the performance of Hough line detection.

Some images are falsely detected by our method. As for the suits images that are not detected by our method, most of them are profile-suits images or suits images with occlusion. As for the non-suits images incorrectly detected as suits images, most of them have similar appearance with suits.

4 Conclusions and Future Work

Clothing style is an important feature to understand images of people including what he/she is doing and what is his/her profession etc. Actually, recognizing the style of clothing is a very hard task due to two reasons: (1) no-rigid deformation and (2) various styles of clothing. In this paper, we proposed a novel method to detect suits in images of people. This method is useful in photo selection systems and attribute-based people search task. We proposed six novel features including color features, shape features and statistical features for this task. The experimental results show the effectiveness of the proposed features.

Also these features will be helpful for other clothing style detection. For example, skin feature (one of our color features) is useful in t-shirt detection. The limitation of our method is that the results for suits images with occlusion and profile-suits images are not satisfactory. In the future, we will find ways to solve these problems and expand our work to other clothing style detection. We will also investigate on how to further combine the multiple features in a multiple graph framework [21][22] to learn a better detector for a particular style of clothing.

Acknowledgments. This work is supported by the National High Technology and Research Development Program of China (863 Program, 2009AA01A403); National Nature Science Foundation of China (61100087); Beijing Natural Science Foundation (4112055); Beijing New Star Project on Science & Technology (2007B071); Co-building Program of Beijing Municipal Education Commission.

References

1. Daniel, A.V., Rogerio, S.F., Duan, T., Lisa, B., Arun, H., Matthew, T.: Attribute-Based people search in surveillance Environments. In: WACV, pp. 1–8 (2009)
2. Sprague, N., Luo, J.: Clothed people detection in still images. In: ICPR, pp. 585–589 (2002)
3. Song, Y., Leung, T.: Context-Aided Human Recognition – Clustering. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part III. LNCS, vol. 3953, pp. 382–395. Springer, Heidelberg (2006)
4. Gallagher, A.C., Chen, T.: Clothing cosegmentation for recognizing people. In: CVPR, pp.1–8 (2008)
5. Gallagher, A.C., Chen, T.: Using context to recognize people in consumer images. IPSJ Transactions on Computer Vision and Applications 1(5), 115–126 (2009)
6. Zhang, W., Zhang, T., Tretter, D.: Clothing-based person clustering in family photos. In: ICIP, pp. 4593–4595 (2010)
7. Khoury, E.E., Senac, C., Joly, P.: Face-and-Clothing based people clustering in video content. In: MIR, pp. 295–304 (2010)
8. Chen, H.Z., Xu, J., Liu, Z.Q., Zhu, S.C.: Composite templates for cloth modeling and sketching. In: CVPR, pp. 943–950 (2006)
9. Tian, Y., Yuan, S.: Clothes Matching for Blind and Color Blind People. In: Miesenberger, K., Klaus, J., Zagler, W., Karshmer, A. (eds.) ICCHP 2010, Part II. LNCS, vol. 6180, pp. 324–331. Springer, Heidelberg (2010)
10. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: CVPR, pp. 511–518 (2001)
11. Jones, M.J., Rehg, J.M.: Statistical color models with application to skin detection. International Journal of Computer Vision 46(1), 81–96 (2002)
12. Deng, Y., Manjunath, B.S.: Unsupervised segmentation of color-texture regions in images and video. IEEE Transactions on Pattern Analysis and Machine Intelligence, 800–810 (2001)
13. Duan, R.O., Hart, P.E.: Use of the hough transformation to detect lines and curves in pictures. In: Communications of the Association for Computing Machinery, pp. 11–15 (1972)
14. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2, 27:1–27:27 (2011)
15. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
16. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Proceedings of the Thirteenth International Conference on Machine Learning, pp. 148–156 (1996)
17. Kohavi, R.: The Power of Decision Tables. In: Lavrač, N., Wrobel, S. (eds.) ECML 1995. LNCS (LNAI), vol. 912, pp. 174–189. Springer, Heidelberg (1995)
18. Quinlan, J.R.: Induction of decision trees. Machine Learning 1(1), 81–106 (1986)
19. Sivic, J., Zisserman, A.: Video Google: a text retrieval approach to object matching in videos. In: ICCV, pp. 1470–1478 (2003)
20. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
21. Wang, M., Hua, X.S., Hong, R.C., Tang, J.H., Qi, G.J., Song, Y.: Unified Video Annotation via Multi-Graph Learning. IEEE Trans. on Circuits and Systems for Video Technology 19(5) (2009)
22. Xia, T., Tao, D.C., Mei, T., Zhang, Y.D.: MultiView Spectral Embedding. IEEE Trans. on Systems, Man, and Cybernetics: Part B 40(6), 1438–1446 (2010)

The Video Face Book

Nipun Pande, Mayank Jain, Dhawal Kapil, and Prithwijit Guha

TCS Innovation Labs, New Delhi, India
{nipun.pande,mayank10.j,prithwijit.guha}@tcs.com,dhawalkapil@gmail.com

Abstract. Videos are often characterized by the human participants, who in turn, are identified by their faces. We present a completely unsupervised system to index videos through faces. A multiple face detector-tracker combination bound by a reasoning scheme and operational in both forward and backward directions is used to extract face tracks from individual shots of a shot segmented video. These face tracks collectively form a face log which is filtered further to remove outliers or non-face regions. The face instances from the face log are clustered using a GMM variant to capture the facial appearance modes of different people. A face Track-Cluster-Correspondence-Matrix (TCCM) is formed further to identify the equivalent face tracks. The face track equivalences are analyzed to identify the shot presences of a particular person, thereby indexing the video in terms of faces, which we call the "*Video Face Book*".

1 Introduction

Videos are generally identified by actors, sceneries or specific activities. Home videos (e.g. brother's wedding, papa's birthday etc.), movies (e.g. Mel Gibson's "Braveheart") and TV series (e.g. Jennifer Aniston's "Friends") are generally referred to by the human participants. Human face is one of the most important objects in news programs. Identifying such actors from videos become a tough computer vision task if performed in a supervised framework. In such a scenario, one has to undergo a tedious supervised learning procedure to perform the task of face recognition for individual actors or for each friend/relative in a home video. In contrast, an unsupervised approach would detect and track the face regions and cluster them to generate video intervals where a certain face appears. This is also similar to the way humans perform, by associating scene intervals with the occurrence of (previously) unseen faces. The explosive growth of image and video data available both off-line and on-line further stresses the need for unsupervised methods to index, search and manipulate such data in a semantically meaningful manner.

A system for building extremely large face datasets from archival video has been introduced by [7]. The system does shot detection, tracking using color histograms for hair,face and torso followed by grouping the tracks using agglomerative clustering. For handling large number of objects in large number of dimensions a technique using Relevant Set Correlation (RSC) has been proposed by [5]. News videos are decomposed into shots followed by face detection and

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 495–506, 2012.

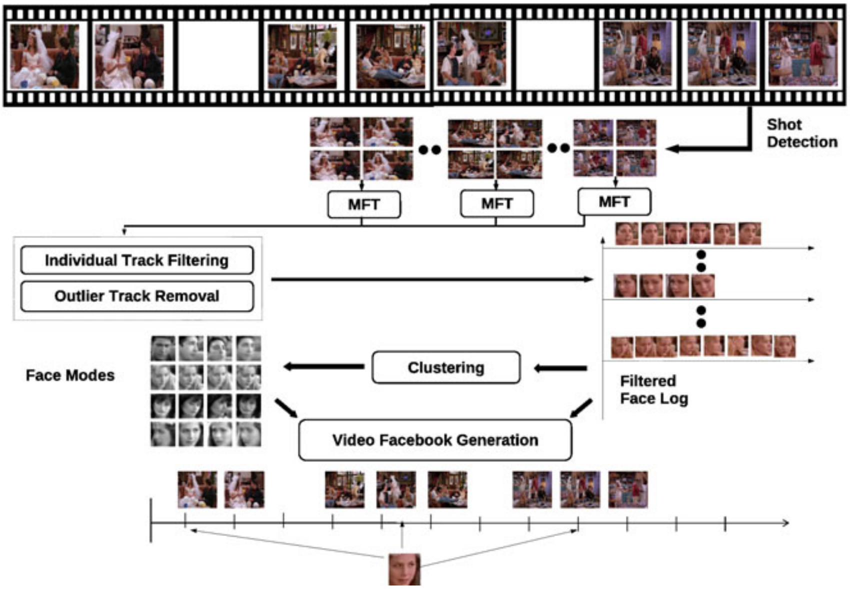


Fig. 1. The shot segmented input video is subjected to multiple face tracking (Section 2) to extract face tracks from individual shots, which are filtered in two stages to remove outliers (Section 3). The resulting face log is clustered using a GMM variant to discover the modes of facial appearances of different people in varying facial poses. The face based video index is generated by analyzing the track and cluster correspondences, which we call the *Video Face Book*.

simple tracking based on estimating the sizes and locations of faces in consecutive frames. Principal Component Analysis (PCA) is used for reducing the number of dimensions of the feature vector for face representation. For finding repeated faces, a clustering method based on RSC is used. For automatic labeling of faces of characters in TV or movie material with their names, using only weak supervision from automatically aligned subtitle and script-text [8] follow an approach where frontal/profile detections of the same face are merged using agglomerative clustering based on the overlap of the detections. Kanade-Lucas-Tomasi (KLT) feature tracker is used for feature point tracking. An approach which uses face features extracted using Discrete Cosine Transform (DCT) is proposed by [2]. Nearest neighbor classification is used to merge the tracks with distances less than a threshold. On similar lines a technique for efficient face retrieval from large video datasets using Local Binary Patterns (LBP) is proposed in [6]. A novel face indexing system that takes advantage of the internet connection of a Set Top Box (STB) to construct a Face Recognition (FR) engine has been proposed by [3]. Faces are clustered and the clustered images are combined using a weighted feature fusion scheme.

The proposed approach (Figure 1) – The video is first segmented into shots using hue-saturation histograms computed from images. The component frames of each shot interval are subjected to frontal/profile face detection and shots without any detection success are rejected as they are irrelevant to our purpose of face extraction which is dependent on such detection results. The individual shots are subjected to multiple face tracking using a detector-tracker reasoning scheme operational in both backward and forward directions (Section 2). The extracted face tracks are filtered through a two stage process where non-face regions are first removed track-wise followed by outlier track removal (Section 3). All the faces from the filtered face log are clustered to capture the facial appearance modes of different persons (Section 4). We further compute a face Track-Cluster-Correspondence-Matrix ($TCCM$) to identify the equivalent tracks and hence acquire the different shot presences of the same person. This results in the generation of the face based video index, which we call the "*Video Face Book*".

2 Multiple Face Tracking

We have used the Haar feature based face detectors [9] to segment the regions of left/right profile or frontal faces in the image sequence. However, these detectors are extremely sensitive to the facial pose. Thus, although they are very accurate in detecting faces in left/right profile or frontal faces, they fail when the facial pose changes. It is also not practical to use a lot of detectors, each tuned to different face orientations as that would lead to both high memory and processor usage. Thus, a detection reduced to a local neighborhood search guided by face features is advantageous to satisfy real-time constraints. Such a necessity is achieved by the procedure of tracking. We initialize the tracker with a face detection success, continue tracking where detection fails (due to facial pose variations) and update the target face features at times when the detectors succeed during the frame presence of the face.

Existing works in multiple face tracking have generally focused on methodologies for face detection and tracking using (skin) color distributions and/or motion cues [7,8]. These satisfy the tracking algorithm necessities of "*target representation*" and "*inter-frame target region correspondence*". However, in cases involving multiple targets, a "*reasoning*" method is required for handling various situations like tracking failure, new target acquisition, entry/exit etc. We next describe the proposed face region representation/localization schemes (Sub-section 2.1) and the adopted methodology of reasoning for tracking multiple faces (Sub-section 2.3).

2.1 Face Representation and Localization

The location of the face F in the image is identified by the face bounding rectangle $\mathbf{BR}(F)$ with sides parallel to image axes. We use a second order motion model (constant jerk), continuously updated from the 3 consecutive centroid positions of $\mathbf{BR}(F)$. Using this model, The centroidal position $\hat{\mathbf{C}}_t(F)$ at the t^{th} instant is predicted as $\hat{\mathbf{C}}_t(F) = 2.5\mathbf{C}_{t-1}(F) - 2\mathbf{C}_{t-2}(F) + 0.5\mathbf{C}_{t-3}(F)$. The color distribution $\mathbf{H}(F)$ of the face F is computed as a normalized color histogram, position weighted by the Epanechnikov kernel supported over the maximal elliptical region $\mathbf{BE}(F)$ (centered at $\mathbf{C}(F)$) inscribed in $\mathbf{BR}(F)$ [4]. Mean-shift iterations initialized from the motion model predicted position converge to localize the target face region in the current image. The mean-shift tracking algorithm maximizes the Bhattacharya co-efficient between the target color distribution $\mathbf{H}(F)$ and the color distribution computed from the localized region at each step of the iterations. The maximum Bhattacharya co-efficient obtained after the mean-shift tracker convergence is used as the tracking confidence $tc(F)$ of the face F [4]. We combine this color based representation with an appearance model to encode the structural information of the face. The RGB image region within $\mathbf{BR}(F)$ is first resized and then converted to a $q \times q$ monochrome image which is further normalized by its brightest pixel intensity to form the normalized face image nF of the face F. The normalization is performed to make the face image independent of illumination variations.

2.2 Normalized Face Cluster Set

During the course of tracking, a person appears with various facial poses. We propose to cluster the normalized faces obtained from the different facial poses to learn the modes of his/her appearances thereby forming a *Normalized Face Cluster Set* ($\mathbf{NFCS}(F)$, henceforth). The normalized face image nF is re-arranged in a row-major format to generate the $d = q \times q$ dimensional feature vector $\mathbf{X}(nF)$. To achieve computational gain, we assume that the individual dimensions of the feature vector are un-correlated and hence, a diagonal co-variance matrix is sufficient to approximate the spread of the component Gaussians. A distribution over these feature vectors is approximated by learning a variant of the Gaussian mixture models where we construct a set of normalized face clusters.

The **NFCS** with K clusters is given by the set $\mathbf{NFCS} = \{(\mu_r, \sigma_r, \pi_r); r = 1, \ldots K\}$, where μ_r, σ_r are the respective mean and standard deviation vectors of the r^{th} cluster and the weighing parameter π_r is the fraction of the total number of normalized face vectors belonging to the r^{th} cluster. The **NFCS** initializes with $\mu_1 = \mathbf{X}(nF_1)$ and an initial standard deviation vector $\sigma_1 = \sigma_{init}$ and $\pi_1 = 1.0$.

Let there be K_{l-1} clusters in the **NFCS** until the processing of the vector $\mathbf{X}(nF_{l-1})$. We define the belongingness function $B_r(u)$ for the u^{th} dimension of the r^{th} cluster which is set to 1.0 if $|\mathbf{X}(nF_l)[u] - \mu_r[u]| \leq \lambda\sigma_r[u]$ and to 0.0, otherwise. Here λ is the *cluster membership threshold* and is generally chosen between $1.0 - 5.0$ (Chebyshev's inequality). The vector $\mathbf{X}(nF_l)$ is considered to belong to the r^{th} cluster if $\sum_{u=1}^{d} B_r(u) \geq (1 - \eta_{mv})d$, where $\eta_{mv} \in (0, 1)$ is the *cluster membership violation tolerance threshold* such that $\eta_{mv} \times d$ denotes the upper limit of tolerance on the number of membership violations in the normalized face vector. If $\mathbf{X}(nF_l)$ belongs to the r^{th} cluster, then its parameters are updated as,

$$\pi_r \leftarrow (1-\alpha_l)\pi_r + \alpha_l \tag{1}$$
$$\sigma_r^2[u] \leftarrow (1-\beta_r(l,u))[\sigma_r^2[u] + \beta_r(l,u)D_{lr}^2[u]] \tag{2}$$
$$\mu_r[u] \leftarrow \mu_r[u] + \beta_r(l,u)D_{lr}[u] \tag{3}$$

where $\alpha_l = \frac{1}{l}$, $\beta_r(l,u) = \frac{\alpha_l B_r(u)}{\pi_r}$ and $D_{lr}[u] = \mathbf{X}(nF_l)[u] - \mu_r[u]$. For all other clusters $r' \neq r$, the mean and standard deviation vectors remain unchanged while the cluster weight $\pi_{r'}$ is penalized as $\pi_{r'} \leftarrow (1-\alpha_l)\pi_{r'}$. However, if $\mathbf{X}(nF_l)$ is not found to belong to any existing cluster, a new cluster is formed ($K_l = K_{l-1}+1$) with its mean vector as $\mathbf{X}(nF_l)$, standard deviation vector as σ_{init} and weight $\frac{1}{l}$; the weights of the existing clusters are penalized as mentioned before.

The parameter updates in equation 3 match the traditional Gaussian Mixture Model (GMM) learning. In GMMs, all the dimensions of the mean vector are updated with the incoming data vector. However, here we update the mean and standard deviation vector dimensions selectively with membership checking to resist the fading out of the mean images. Hence, we call the **NFCS** as a variant of the mixture of Gaussians. Figure 2(a) shows a few mean images of the normalized face clusters learned from the tracked face sequences of the subject.

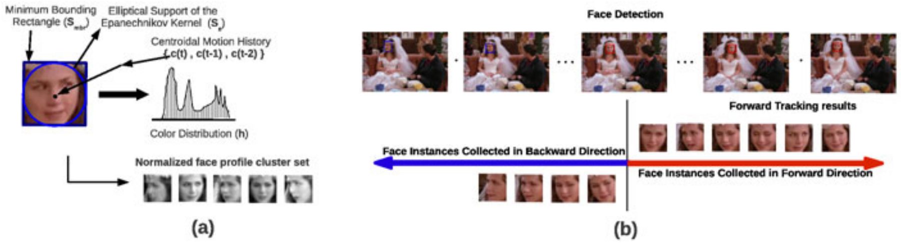

Minimum Bounding Rectangle (S_{mbr})
Elliptical Support of the Epanechnikov Kernel (S_e)
Centroidal Motion History {c(t), c(t-1), c(t-2)}
Color Distribution (h)
Normalized face profile cluster set
(a)
Face Detection
Forward Tracking results
Face Instances Collected in Backward Direction
Face Instances Collected in Forward Direction
(b)

Fig. 2. (a) Color distribution $\mathbf{H}(F)$, second order motion model and the *normalized face cluster set* ($\mathbf{NFCS}(F)$) are used for *face representation and tracking*. (b) *Backward-Forward tracking* – Jennifer Aniston's face gets detected somewhere at the middle of the shot interval; multiple face tracker detects a new face region and starts tracking (marked with red bounding box) in forward direction. Mean-shift tracker initialized from the first detection is used to localize the face in backward direction (marked with blue bounding box).

2.3 Handling Multiple Faces

Tracking multiple faces is not merely the implementation of multiple trackers but a reasoning scheme that binds the individual face trackers to act according to problem case based decisions. For example, consider the case of tracking a face which gets occluded by another object. A straight through tracking approach will try to establish correspondences even when the target face disappears in the image due to complete occlusion by some scene object leading to tracking failure. A reasoning scheme, on the other hand, will identify the problem situation of the disappearance due to the occlusion of the face and will accordingly wait for the face to reappear by freezing the concerned tracker. Our approach to multiple face tracking proposes a reasoning scheme to identify the cases of face grouping/isolation along with the scene entry/exit of new/existing faces.

The process of reasoning is performed over three sets, viz. the sets of *active*, *passive* and *detected* faces. The active set $\mathcal{F}_a(t)$ consists of the faces that are well tracked until the t^{th} instant. On the other hand, the passive set $\mathcal{F}_p(t)$ contains the objects for which either the system has lost track or are not visible in the scene. The set of detected faces $\mathcal{F}_d(t)$ contains the faces detected in the t^{th} frame. The system initializes itself with empty active/passive/detected face sets and the objects are added or removed accordingly as they enter or leave the field of view. During the process of reasoning, the objects are often switched between the active and passive sets as the track is lost or restored. We start the process of reasoning at the t^{th} frame based on the active/passive face sets available from the $(t-1)^{th}$ instant. The faces in the active set are first localized with motion prediction initialized mean-shift trackers (Sub-section 2.1. We compute the extent of overlap between the tracked face regions from the active set and the detected face regions to identify the isolation/grouping state of the faces. The reasoning scheme based on the tracked-detected region overlaps is described next.

Consider the case where m faces are detected ($\mathcal{F}_d = \{dF_j; j = 1 \ldots m\}$) while n faces were actively tracked till the last frame ($\mathcal{F}_a = \{aF_i; i = 1 \ldots n\}$). We define the fractional overlap between the faces F_1 and F_2 as $\gamma(F_1, F_2) = \frac{|\mathbf{BR}(F_1) \cap \mathbf{BR}(F_2|)}{\mathbf{BR}(F_1)}$ to analyze the correspondence between F_1 and F_2. We consider aF_i and dF_j to have significant overlap with respect to a certain threshold η_{ad}, if the predicate $\textsc{Overlaps}(aF_i, dF_j) \Rightarrow [\gamma(aF_i, dF_j) \geq \eta_{ad}] \vee [\gamma(dF_j, aF_i) \geq \eta_{ad}]$ is satisfied.

Let $\mathbf{S_{df}}(i) = \{dF_k : [dF_k \in \mathcal{F}_d] \wedge \textsc{Overlaps}(aF_i, dF_k)$ denote the set of detected faces which has significant overlap with the face aF_i in the active set and $\mathbf{S_{af}}(j) = \{aF_r : [aF_r \in \mathcal{F}_a] \wedge \textsc{Overlaps}(aF_r, dF_j)$ represent the set of faces in the active set which has significant overlap with the detected face dF_j. Based on the cardinalities of these sets associated with either of aF_i/dF_j and the tracking confidence $tc(aF_i)$, we identify the following situations during the process of tracking.

Isolation and Feature Update – The face aF_i is considered to be isolated if it does not overlap with any other face in the active set – $\forall r \neq i \neg\textsc{Overlaps}(aF_i, aF_r)$; $aF_i, aF_r \in \mathcal{F}_a$. Under this condition of isolation of the tracked face, we update its color distribution and motion features from the associated detected face if there exists a pair (aF_i, dF_k) which significantly overlap only with each other and none else – $\exists k \textsc{Overlaps}(aF_i, dF_k) \wedge |\mathcal{S}_{df}(i) = 1| \wedge |\mathcal{S}_{af}(k) = 1|$.

Face Grouping – The face is considered to be in a group (e.g. multiple persons with overlapping face regions) if the bounding rectangles of the tracked faces overlap. In this case, even if a single detected face dF_k is associated to aF_i, we only update the motion model of aF_i as we are not confident about the correspondence on account of multiple overlaps.

Detection and/or Tracking Failure – This is the case where face detection fails due to facial pose variations. However, if the face aF_i is tracked well ($tc(aF_i) \geq \eta_{tc}$), we update only the motion model of aF_i and do not update the color distribution. However, in case of both detection and tracking failure, aF_i is not associated with any detected face and the tracking confidence also drops below the threshold (η_{tc}). In this case, we consider aF_i to disappear from the scene and transfer it from $\mathcal{F}_a$ to $\mathcal{F}_p$ i.e. $\textsc{Disappears}(aF_i) \Rightarrow |\mathcal{S}_{df}(i) = 0| \wedge [tc(aF_i) < \eta_{tc}]$.
New Face Identification – A new face in the scene does not overlap with any of the the bounding rectangles of the existing (tracked) faces. Thus, dF_j is considered a new face if $\mathbf{S_{af}}(j)$ is a null set i.e. $\textsc{NewFace}(dF_j) \Rightarrow |\mathbf{S_{af}}(j)| = 0$. Note that, the system might lose track of an existing face whose re-appearance is also detected as the occurrence of a new one. Hence, the newly detected face region is normalized first and checked against the $NFCS$ of the faces in $\mathcal{F}_p$. If a match is found, the track of the corresponding face is restored by moving it from $\mathcal{F}_p$ to $\mathcal{F}_a$ and its color and motion features are re-initialized from the newly detected face region. However, if no matches are found, a new face is added to $\mathcal{F}_a$ whose color and motion features are learned from the newly detected face region.

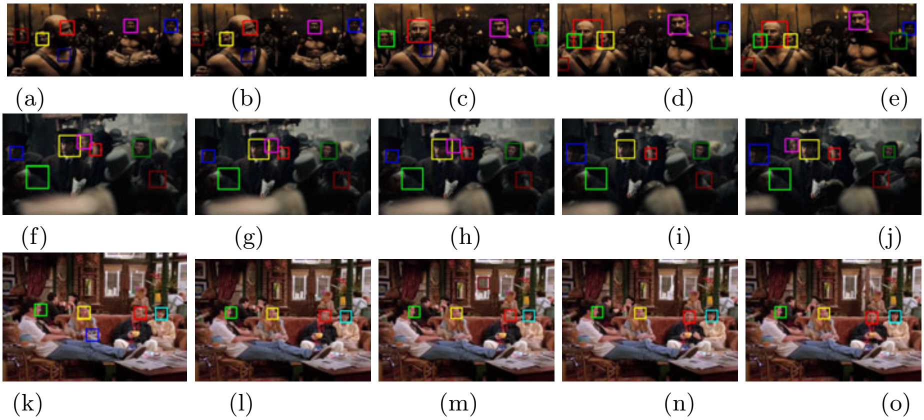
During the course of multiple object tracking, the faces in the active set are identified in one of the above situations and the feature update or active to passive set transfer decisions are taken accordingly. By reasoning with these conditions, we initialize new trackers as new faces enter the scene and destroy them as the faces disappear.

2.4 Backward-Forward Tracking

Our work assumes that a certain person will be detected in either front/profile face at some time in a shot (of duration $[t_s, t_e]$, say). However, it may well happen that the person gets detected only at the t^{th} instant ($t_s < t < t_e$), although he/she was present from the very beginning (t_s) with a facial pose different from either frontal or left/right profile. In such cases, tracking in only forward direction will not provide us with all the face instances of the person. To avoid this, we also run a backward tracker initialized with the first detection to provide us with all the facial pose variations of the tracked person. The tracker is terminated when the tracking confidence dips below the threshold η_{tc}. Figure 2(b) illustrates the combined scheme for tracking in both backward and forward direction for acquiring the face instances in varying poses; including the ones prior to first detection.

2.5 Results: Multiple Face Tracking

We present results from 3 shots from the movies "*300*" (624 images) and "*Sherlock Holmes*" (840 images); and the TV Series "*Friends*, an episode from Season 1 (143 images). The results of multiple face tracking in these videos are shown in figure 3. The proposed approach for multiple face tracking is implemented on a single core 1.6 GHz Intel Pentium-4 PC with semi-optimized coding and operates at 13.33 FPS (face detection stage included).



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

(k) (l) (m) (n) (o)

Fig. 3. Results of multiple face tracking under occlusions. (a)-(e) Movie *300* - faces are relatively unoccluded; (f)-(j) Movie *Sherlock Holmes* – The face marked with the pink bounding rectangle undergoes partial and full occlusion and the track is successfully restored as it reappears. (k)-(o) TV series *Friends*, Season 1. Note that apart from faces, trackers are also initialized on non-face regions in (f)-(o) due to false detections which are filtered later. (Section 3)

Performance Analysis – We present an object centric performance analysis by manually inspecting the surveillance log for computing the average rates of tracking precision and track switches. Consider the case of a tracker with a life span of T frames, of which for the first T_{trk} frames, the tracker successfully tracks the same face over which it is initialized and then successively switches track to N_{switch} number of (different) faces(s) during the remaining $T - T_{trk}$ frames. The *tracking precision* of an individual object is then defined as $\frac{T_{trk}}{T}$ and the average tracking precision computed over the entire set of extracted faces is called the **Tracking Success Rate** for the entire video. In the same line, the **Tracker Switch Rate** is evaluated as the average number of track switches over the entire set of extracted objects. After a track switch from the $T_{trk} + 1$ frame onwards, a different tracker may pick up the trail of this object through a track switch from some other face or through the initialization of a new tracker – let there be N_{reinit} number of tracker re-initializations on some face region. The **Tracker Re-initialization Rate** is defined as the average number of tracker re-initializations per face computed over the entire set of extracted faces. Refer to Figure 4.

3 Face Log Processing

The cropped face regions acquired by tracking are stored in a face log. However, the face log also contain non-face regions (outliers) on account of detection/tracking failure. We note that such outliers are of two types – first, trackers initialized on proper face regions which occasionally drift to non-face regions

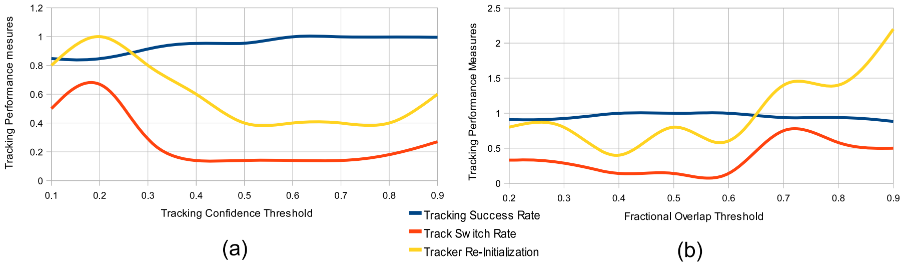


Fig. 4. Multiple face tracking performance analysis. The rates of tracking success, track switches and tracker re-initialization are plotted with respect to (a) tracking confidence threshold (η_{tc}) and (b) fractional overlap threshold (η_{fo}) varied in the interval of $[0.1, 0.9]$ in steps of 0.1. We choose $\eta_{fo} = 0.4$ and $\eta_{tc} = 0.6$ for optimal performance by referring to these graphs.

due to motion-model failure or pre-mature mean-shift convergence; and second, trackers initialized from non-face regions (false detections) continuously tracking these outlier regions during the entire shot. We propose a two-stage filtering scheme to remove such outliers based on three assumptions – first, hue-saturations histograms computed from face regions will have similar distributions for the skin pixels while non-face regions will have completely different distribution profiles; second, in each track the face regions are in the majority and hence the average color distribution will be considerably different from the color distributions of non-face regions; and third, in face-tracks initialized on false detections, there will be hardly any face region and thus the average hue-saturation distribution of that track will be significantly different from an average distribution computed from only face regions.

Consider the case where N face tracks ($T_i; i = 1, \dots N$) are extracted where the i^{th} track contains n_i faces ($T_i = \{F_{ij}; j = 1, \dots n_i\}$). Let $H_{hs}(i, j)$ denote the hue-saturation distribution computed from F_{ij} and we compute the average $\bar{H}_{hs}(i) = \frac{1}{n_i} \sum_{j=1}^{n_i} H_{hs}(i, j)$ from all the faces in T_i. Based on our assumptions, we declare the q^{th} face as an outlier if $\mathbf{B}_c(H_{hs}(i, q), \bar{H}_{hs}(i)) < \eta_{cm}$ where η_{cm} is a color distribution match threshold. The outliers, if present are removed from each track and leaves us with $T_i = \{F_{ij}; j = 1, \dots n_i'\}; i = 1, \dots N$. Note that this process only removes outliers from each track but can not filter the ones where the trackers were initialized on non-face regions due to erroneous face detections (Figure 5(a)).

The process of individual track filtering leaves us with two kinds of tracks – first, the "pure" ones with only face regions; and second, the ones containing mostly outliers where the tracker was initialized on non-face regions. We compute the average hue-saturation distributions $\bar{H}_{hs}(i)$ from each track and obtain their average as $\tilde{H}_{hs} = \frac{1}{N} \sum_{i=1}^{N} \bar{H}_{hs}(i)$. Proceeding on the same assumptions outlined earlier, we describe the i^{th} track as an outlier, if $\mathbf{B}_c(\tilde{H}_{hs}, \bar{H}_{hs}(i)) < \eta_{cm}$ (Figure 5(b)). The faces belonging to the filtered tracks are clustered further to group the similar faces and are described next (Section 4).

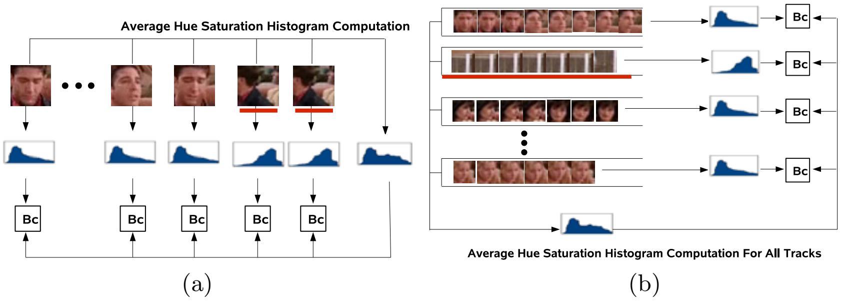
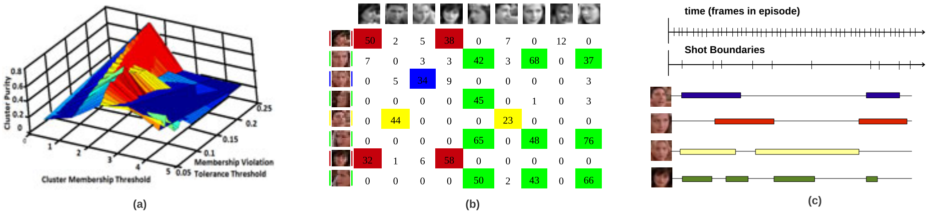

Average Hue Saturation Histogram Computation
Bc
Bc
Bc
Bc
Bc
(a)
Bc
Bc
Bc
Bc
Average Hue Saturation Histogram Computation For All Tracks
(b)

Fig. 5. Two stage face log filtering with Bhattacharya coefficient $\eta_{cm} = 0.6$. (a) Non-face instances are removed from individual tracks in first stage. (b) Outlier tracks initialized from non-face regions are filtered next.



50	2	5	38	0	7	0	12	0
7	0	3	3	42	3	68	0	37
0	5	34	9	0	0	0	0	3
0	0	0	0	45	0	1	0	3
0	44	0	0	0	23	0	0	0
0	0	0	0	65	0	48	0	76
32	1	6	58	0	0	0	0	0
0	0	0	0	50	2	43	0	66

Fig. 6. (a) The cluster purity is evaluated by varying the cluster membership threshold (λ) and membership violation tolerance threshold (η_{mv}) for clustering performance analysis. (b) The marked cells of $TCCM$ indicate the face track-cluster linkages which satisfy a thresholded association criterion. A linkage transitivity analysis is performed further to identify the tracks linked through the common cluster(s). (c) A small segment of the *Video Face Book* generated from the TV series "Friends" (episode 1, season 1). Horizontal colored bars indicate the shot presences of different human participants.

4 Face Clustering

The face regions obtained from all tracks of the filtered face log are clustered using the approach outlined in Sub-section 2.2. Ideally, each cluster should contain faces of the same person. However, such a *cluster purity* varies with different values of the *cluster membership threshold* (λ) and *cluster membership violation tolerance threshold* (η_{mv}). Consider the case where K clusters are formed, where the k^{th} cluster contains nC_k faces, of which mC_k number of faces belong to the same person and satisfies the plurality criterion. Then, we define the average cluster purity $cP(\lambda, \eta_{mv})$ for a certain set of chosen thresholds as $cP(\lambda, \eta_{mv}) = \frac{\sum_{k=1}^{K} mC_k}{\sum_{k=1}^{K} nC_k}$. The clustering performance is analyzed by varying λ in $[0.5, 4.5]$ in steps of 0.1 and η_{mv} in $[0.05, 0.25]$ in steps of 0.005. The performance analysis is performed on 3 test data sets (Figure 3) and we have chosen $\lambda = 1.8$ and $\eta_{mv} = 0.215$ by referring to Figure 6(a) for which we achieve the maximum cluster purity of 0.804.

5 Video Index Generation

Consider the case where the filtered face log contains N' face tracks and K clusters are obtained by face clustering. We form the $N' \times M$ Track-Cluster-Correspondence-Matrix ($TCMM$) to analyze the equivalences of the different tracks present in the face log. Let $cL(i,j)$ denote the cluster index of the j^{th} face in the i^{th} track, i.e. $cL(i,j) \in [1,M]$. The $TCMM$ is thus formed as $TCMM[i][k] = \sum_{j=1}^{n'_i} \delta(cL(i,j) - k)$ where, the i^{th} track contains n'_i faces and $\delta(\bullet)$ is the Kronecker Delta function.

Tracking provides us with various facial poses of the same person while clustering helps us discover the modes of facial appearance. The similar facial appearances are grouped through clustering while the different facial appearances of the same person are linked through tracking. Each row of the $TCCM$ signify the number of occurrences of different facial appearance modes in a certain track and each column of $TCCM$ denote the frequencies of assuming the same facial appearance mode by different tracks. We link a track i to the cluster k if more than 25% faces of the i^{th} track assume the k facial appearance mode i.e. if $TCMM[i][k] \geq 0.25n'_i$. Consider the case where the i^{th} track is linked to the clusters k and p while the r^{th} track is linked with clusters p and q. We perform a linkage transitivity analysis to identify that the tracks i and p have a common link to the p^{th} cluster and use the same to declare the tracks i and j as equivalent. A similar analysis is performed on the entire $TCMM$ to identify the equivalent tracks (Figure 6(b)). Since the face tracks are obtained from indexed shots, analyzing the equivalent tracks reveal the shot presences of the same person. This is illustrated in Figure 6(c) where a part of the *Video Face Book* formed by analyzing the TV series “Friends” (episode 1, season 1) is shown.

6 Conclusion

We present an unsupervised scheme for indexing videos with human participants by using facial information and hence the name *Video Face Book*. The video is initially decomposed into a sequence of shots using the criterion of intra-shot frame hue-saturation distribution consistency. A combination of backward-forward tracking is used to extract the tracks of multiple faces from individual shots. Such tracks obtained from each shot collectively form the crude face log containing outliers along with face instances. Outliers are removed in two stages – first, the non-face regions are filtered from each track and second, the outlier tracks formed due to false detections are removed. All the face instances from all tracks are clustered next to form the face clusters. A person may appear with varying facial poses in the same track and hence traverse the different modes (mean faces of clusters) of facial appearance. Thus people appearing in different shots can be linked through strong correspondences of different tracks with the same cluster. We form a Track-Cluster-Correspondence-Matrix ($TCMM$) to identify such track linkages and hence generate the video index in terms of shot presences of a certain person.

We have demonstrated an unsupervised approach to indexing videos through faces. However, recent research has also proposed unsupervised means of discovering objects from images/videos [1]. These approaches may be used to discover objects from videos first, and the proposed scheme can be used next to detect/track and cluster objects of different categories for indexing videos. However, this will only be the indexing of videos with the *actors*, whose interactions might be discovered and grouped further to index videos in terms of *actions* thereby proceeding a few steps further to achieve the final goal of a cognitive vision system.

References

1. Alexe, B., Deselaers, T., Ferrari, V.: What is an object? In: IEEE Computer Vision and Pattern Recognition (CVPR), San Francisco, pp. 1–8 (June 2010)
2. Bauml, M., Fischer, M., Bernardin, K., Ekenel, H.K., Stiefelhagen, R.: Interactive person-retrieval in tv series and distributed surveillance video. In: MM 2010 Proceedings of the International Conference on Multimedia (2010)
3. Choi, J.Y., Neve, W.D., Ro, Y.M.: Towards an automatic face indexing system for actor-based video services in an iptv environment. IEEE Transactions on Consumer Electronics 56, 147–155 (2010)
4. Comaniciu, D., Ramesh, V., Meer, P.: Real-time tracking of non-rigid objects using mean shift. In: Computer Vision and Pattern Recognition, vol. 2, pp. 142–149 (2000)
5. Le, D.D., Satoh, S., Houle, M.E., Nguyen, D.P.T.: An efficient method for face retrieval from large video datasets. In: Proceedings of the ACM International Conference on Image and Video Retrieval (2010)
6. Nguyen, T.N., Ngo, T.D., Le, D.D., Satoh, S., Le, B.H., Duong, D.A.: An efficient method for face retrieval from large video datasets. In: Proceedings of CIVR 2010, pp. 382–389 (2010)
7. Ramanan, D., Baker, S., Kakade, S.: Leveraging archival video for building face datasets. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, pp. 1–8 (2007)
8. Sivic, J., Everingham, M., Zisserman, A.: Who are you?- learning person specific classifiers from video. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1145–1152 (2009)
9. Viola, P., Jones, M.: Robust real-time face detection. International Journal on Computer Vision 57(2), 137–154 (2004)

Content Based Image Retrieval Using Bag-Of-Regions

Rémi Vieux, Jenny Benois-Pineau, and Jean-Philippe Domenger

LaBRI - CNRS UMR 5800 - Université de Bordeaux
firstname.name@labri.fr

Abstract. In this work we introduce the Bag-Of-Regions model, inspired from the Bag-Of-Visual-Words. Instead of clustering local image patches represented by SIFT or related descriptors, low level descriptors are extracted and clustered from image regions, as given by a segmentation algorithm. The Bag-Of-Region model allows to define visual dictionaries that capture extra information with respect to Bag-Of-Visual-Words. Combined description schemes and ad-hoc incremental clustering for visual dictionnaries are proposed. The results on public datasets are promising.

Keywords: Content Based Image Retrieval, Bag-Of-Regions, Incremental Clustering, Meta-Search.

1 Introduction

Image retrieval is a challenging topic that has been a research challenge for several decades. The task is difficult for several reasons. Smeulders *et. al.* [24] introduced the concept of *semantic gap*, that is the discrepancy between the low level descriptors that can be computed from the images and the interpretation of the image done by humans. A query to an image retrieval system is ill-defined by nature. Such a query could take several forms. One of the earliest successful system, QBIC[8], accepted queries as a user defined color palette, that images should matches. A query can be formulated using an example image (Query-By-Example – QBE– paradigm). The system must retrieve the most similar images to the query. In this case, the notion of similarity is implicit for the user, and the system must approximate this notion into a computable quantity. In the best case, it can be related to several measurements in terms of low level descriptors.

In the last decade, a breakthrough in image retrieval and object recognition have been achieved using the Bag-Of-Visual-Words (BOVW) model based on interest-point descriptors such as SIFT[16]. In the mean time, methods based on region-based properties of the image have known a decrease of popularity for CBIR and classification tasks, since the fundamental work of Duygulu *et. al.* [5]. Few examples include Souvannavong *et. al.* [25] for video content indexing and retrieval and Gokalp and Askoy [10] for scene classification. However, current state-of-the-art for accurate object class image segmentation rehabilitates image segmentation and region-based visual description of the image content [12,26,27].

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 507–517, 2012.

In this paper, we study an image retrieval system that extends the traditional notion of BOVW vocabulary not only to keypoint-based descriptors, but to region based descriptors, as obtained by a segmentation algorithm. Region-based descriptors open the way to exploit a vast amount of different visual cues, such as colour, texture and shape, that are not captured by keypoint descriptors. With this extension arise several challenges: first, in section 2, we clarify the definition of a region-based visual dictionary. The computation of visual dictionary has been considered traditionally as an offline, time independent process. This point becomes more problematic when several such dictionaries must be built. In section 3, we propose to rely on an incremental clustering method that has lower memory and computational complexity than k-means, the reference algorithm. Able to express the image content through several visual dictionaries, we combine them to improve the single-modality retrieval results. We introduce in section 4 the topic of *meta-search* and its most famous strategies. In section 5, we perform deep experiments of the method on three public datasets. We conclude the paper in section 6 and give research perspectives for the future of this work.

2 Bag-Of-Regions Model

The BOVW model has been inspired by the Bag-Of-Words (BOW) model for text document representation. In the BOW model, a text document is represented by the number of occurrences of the words in the document. Despite the simplicity of the model, which neither takes into account the order of the words, nor the relationships between them, this model is very efficient for document classification tasks [11]. Sivic and Zisserman proposed to compute a visual dictionary by clustering similar visual entities inspired by BOW model. Hence, to build a visual dictionary, we must define two key concepts: what entity defines the *spatial support* for a visual word and which *descriptor* underpins the notion of similarity in the clustering process. In the BOVW model, local interest points are used as salient image patches and SIFT or related [18] descriptors used to describe the patches. We propose to define as the basic entities supporting the visual words the *image regions* obtained by a segmentation algorithm. An extremely rich collection of descriptors can be extracted from image regions providing new kind of visual dictionaries. We call the Bag-Of-Regions dictionary BOR. Figure 1 shows the different steps for BOR extraction. If these are similar to BOVW, BOR is trickier to compute due to the number of parameters in the process. The parameters are represented by the colored box in figure 1: there can be several segmentations, many visual features and different quantization of the visual space to compute a single BOR dictionary. Hence, the number of single BOR models that can be computed explode with the number of parameters. The time spent in the vocabulary computation is usually considered as irrelevant, since clustering is performed offline. Time does matter with such an amount of dictionaries to compute. There is a need for an efficient clustering algorithm able to produce these results in a reasonable time.

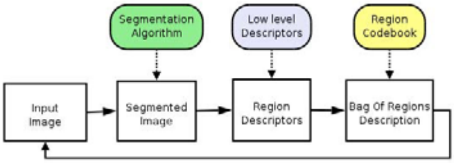


Fig. 1. Bag-Of-Regions extraction pipeline

3 Incremental Clustering for Visual Dictionnaries

Clustering for codebook construction is a difficult problem because of the large number of data samples as well as clusters. K-means clustering has been the reference and most popular method so far [23]. Alternative for to k-means have been proposed by Nister and Stewenius, for handling large amount of data [19]. Yeh *et. al.* proposed the dynamic computation of visual vocabulary using adaptive vocabulary forests [28]. We propose to replace the traditional k-means algorithm with the incremental vector quantization of Lughofer [17]. The principles of vector quantization are the following:

1. Choose initial values for the k cluster centers, $\mathbf{c}_k, k = 1, \dots, K$.
2. Fetch out the next data sample $\mathbf{x}$ of the data set $\mathcal{D}$
3. Calculate the distance of the selected data point to all cluster centers.
4. Elicit the cluster center which is closest to the data point as

$$\mathbf{c}_{win} = \arg\min_{k} d(\mathbf{x}, \mathbf{c}_k) \tag{1}$$

5. If $d(\mathbf{x}, \mathbf{c}_{win}) \geq \rho$ then the current sample $\mathbf{x}$ becomes the center of a new cluster. Otherwise move the cluster center towards the new point:

$$\mathbf{c}_{win}^{new} = \mathbf{c}_{win}^{old} + \eta(\mathbf{x} - \mathbf{c}_{win}^{old}), \eta \in [0, 1] \tag{2}$$

The main advantage of incremental clustering with respect to k-means is the lower computational complexity. All the data are processed in a single pass. The computational complexity of the incremental clustering is $O(KNd)$ with K the number of clusters, N the number of vectors and d their dimension. K-means is $O(IKNd)$ with I the number of iterations. Extensions to vector quantization to fit a real incremental clustering task with unknown number of clusters are given in [17] that we do not detail here.

However, we have seen that those extensions are not directly suitable for our task. Indeed, clustering model can differ significantly while processing the same data in different order. We propose to choose the initial cluster centers according to the k-means++ initialisation [1]. In this way, we ensure that a minimal number of clusters (visual words) is reached. Moreover, as the position of the centroid is crucial during the incremental clustering process, it is natural to chose centers that reflect well the organisation of the data as k-means++

does, leading to a more robust model. In section 5, in order to compare the incremental clustering with k-means, we ensured that no clusters were created incrementally (*i.e.* setting up a high threshold ρ).

4 Fusion of Multiple Retrieval Systems

The BOR model allows a profusion of different representations of image content based on the nature of the low level descriptor, the granularity of the segmentation or the quantization method used for vocabulary construction. Retrieval systems based on these vocabularies are likely to return different sets of images. The optimal combination of this set is the problem of *meta-search* [2]. We assume that the retrieval systems return results in a decreasing order of similarity to the query. Two major types of error can occur for any such system[9]: 1) giving a high rank to non relevant documents and 2) giving a low rank to relevant ones. In table 1, we present the most widely known strategies for combination [9,13]. $S(q, d)$ is the similarity value of document d to query q. CombMIN minimizes probability of 1), while CombMax minimizes probability of 2). CombMED tries to handle 1) and 2). The other three methods consider the relative similarity values given by each method, instead of selecting a value from the set of runs. CombSUM gives the numerical mean of similarity values, CombANZ ignore effects of single runs failing to retrieve relevant documents and CombMNZ provides higher weights to documents retrieved by multiple retrieval methods. Experiments have shown that CombSUM and CombMNZ usually offer the best increase in performances [2,9,13]. We considered these methods as they are very simple and are almost a standard in information retrieval, despite the existence of more advanced literature on this subject.

Table 1. Classical combination strategies for multiple retrieval system results

Name	Formula
CombMIN	$S(q,d) = \min_i(S_i(q,d))$
CombMAX	$S(q,d) = \max_i(S_i(q,d))$
CombMED	$S(q,d) = median(S_i(q,d))$
CombSUM	$S(q,d) = \sum_i S_i(q,d)$
CombANZ	$S(q,d) = CombSUM/\sum_{i\|S_i(q,d)\neq 0} 1$
CombMNZ	$S(d) = CombSUM \times \sum_{i\|S_i(d)\neq 0} 1$

5 Experiments

The goal of the experiments our threefold:

1. Test that the proposed incremental clustering approach for visual dictionary computation does not affect the performances of the retrieval systems.
2. Show that BOR is a suitable approach that can be as efficient as traditional BOVW.

3. Effectively combine the results of multiple systems to build a meta-search engine with increased performances.

The three points are addressed in the following subsections. We used three publicly available datasets, namely WANG, SIVAL and CALTECH101. WANG [15] is a subset of Corel dataset containing 1000 images classified in 10 different categories. We chose WANG to compare our results with the in-depth evaluation of features for image retrieval of Deselaers *et. al.* [4]. SIVAL is a more challenging dataset which has been specifically built for *localized* CBIR, *i.e.* where the user is interested in retrieving images of a specific object[21]. 25 objects have been pictured at different locations on the same set of complex backgrounds. There are 1500 images in this dataset. Using this dataset, we will show that the BOR representation is suitable for performing local queries as is the BOVW. Finally, we used CALTECH101 dataset [6] to provide larger scale experiments. CALTECH101 contains approximately 9000 images grouped into 101 categories. The number of images per category differs from one another.

We fixed the parameters to compute BOR visual dictionaries for all the datasets to the following:

- 7 different segmentations per image. 5 segmentations were computed with the algorithm of Felzenszwalb and Huttenlocher [7], tuning the parameters to produce different levels of region granularity. 2 segmentations were computed using Turbopixels [14]. We the number of regions to k and $2k$ with $k = 50$ for WANG, $k = 1000$ for SIVAL and $k = 100$ for CALTECH101. We used different k because images in SIVAL have much larger resolution than images in WANG. Examples are given in figure 2.
- 2 low level descriptors were computed from the regions: HSV color histogram and the histogram of Local Binary Patterns (LBP) [20] as a texture descriptor.
- 5 different size for visual vocabularies were used: $\{500, 1000, 2000, 5000, 10000\}$ words.

Hence, we computed $5 \times 2 \times 7 = 70$ BOR vocabularies for each dataset. We also computed the BOVW by the clustering of SURF points [3] using the same dictionary sizes. When computing the actual image signature for the BOR representation, we can weight the contribution of each word either by the area of the regions or by the number of regions in the image. We tried both approaches. Thus we have $70 \times 2 + 5 = 145$ visual vocabularies for each dataset. In the case of SIVAL dataset, we will consider 2 cases: global queries, where the BOR and BOVW signature are computed using the full image, and local queries, where they are computed considering only the regions or keypoints that are inside the object bounding box, that have been manually annotated.

For all datasets, we computed the visual vocabularies using the whole dataset, as no supervised learning is employed which would require the definition of a training and test set. We evaluate the Mean Average Precision (MAP) to asses the performances of the systems. The MAP is evaluated using every image of the datasets as query.

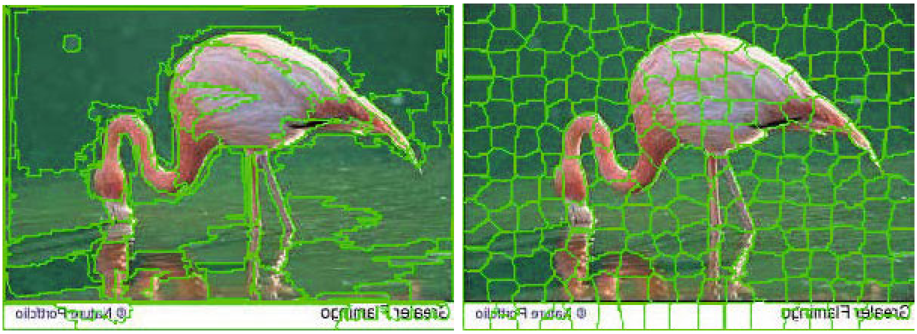


Fig. 2. Example of segmentation with Felzenszwalb [7] (left) and TurboPixels [14] (right). Image from CALTECH101 flamingo category.

5.1 Incremental Clustering

We compared k-means clustering and the proposed incremental clustering on WANG and SIVAL, as it was not possible to compute k-means on CALTECH101 due to memory requirements. For incremental clustering and k-means, we used k-means++ initialisation of centroids. We set up the incremental clustering to ensure that no clusters were added incrementally, in order to have a fair comparison using vocabularies of the same size. Figure 3 shows the results obtained with the two methods on the 145 systems for each. The curve of performances of systems built with k-means and incremental clustering always have the same shape. This shows that the incremental clustering does not affect the retrieval performances, despite the lower computational and memory requirements. The incremental clustering curve seems even slightly higher than the k-means curves in all cases (*i.e.* WANG, SIVAL global and SIVAL local). Note that the increase is not really significant and we are far from claiming that incremental clustering should be favored to build retrieval systems with improved retrieval efficiency, but use it for computational efficiency.

5.2 Retrieval Efficiency Using Bag-Of-Regions Vocabulary

In figure 4 we compare the results obtained by the HSV, LBP and original BOVW with SURF descriptors. The results are presented in the same way as in figure 3, *b*y increasing MAP scores. There are only 5 systems based on SURF and 70 systems based on HSV and LBP, which is why the SURF curve is shorter.

For WANG dataset, the results obtained with SURF descriptors is outperformed by HSV and LBP. More than half of the HSV and LBP systems are better than SURF. The best results are achieved by HSV descriptors, which is in accordance with the experiments of [4], where global color histograms were the best features for this dataset. Note that the best MAP reported in [4] was 0.505, while our best system obtains 0.548 MAP, and many systems outperform 0.505. The best run based on SURF is 0.443. For the SIVAL and

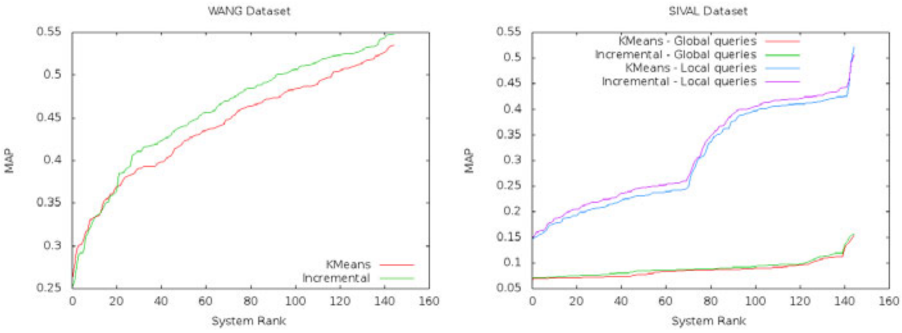


Fig. 3. Comparative performances of retrieval systems built with k-means clustering and the proposed incremental clustering scheme. x-axis denotes the system rank (from worst to best), y-axis the system MAP score.

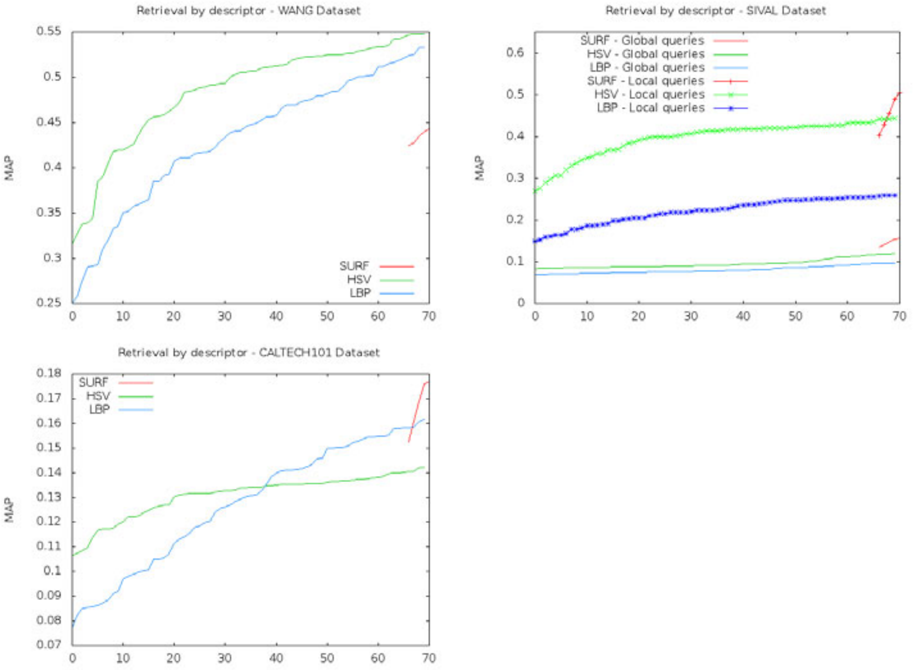


Fig. 4. Performances of retrieval systems using HSV, LBP and SURF descriptors

CALTECH101 datasets, the best overall systems are obtained with SURF (0.157 on SIVAL global, 0.505 on SIVAL local, 0.177 CALTECH). However, SURF does not clearly outperforms color and texture systems. In figure 5, we detail the MAP per categories of the best performing SURF, HSV and LBP systems. The retrieval performances of the descriptors is linked with the queries. The most explicit example can be seen in CALTECH101 dataset, where there are 2 blue peaks corresponding to a high MAP obtained with HSV. While HSV is overall the worst descriptor(figure 4), it is particularly well suited for **car sides** and **leopards** categories. This is surprising for the first category, but is actually due to an artifact of the dataset. All **car sides** images are black and white, while other are color images.

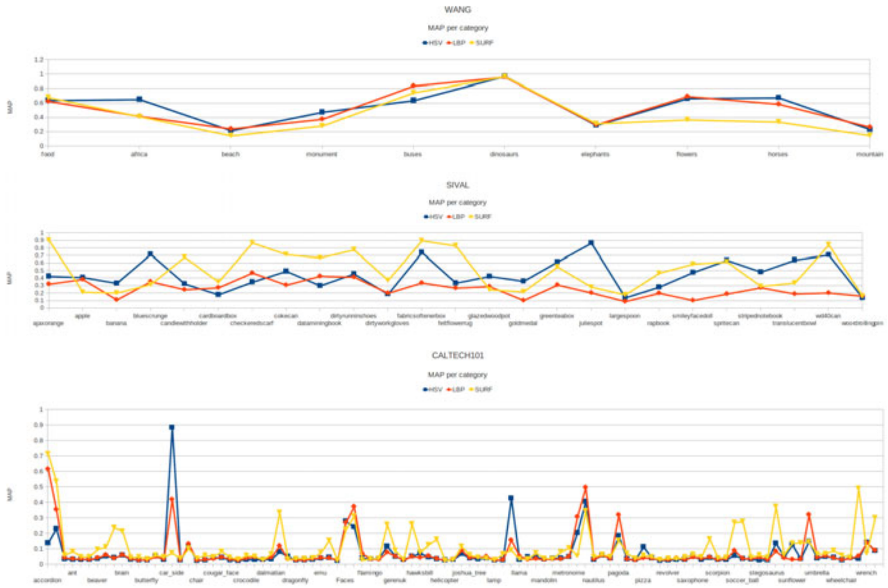


Fig. 5. MAP per category for the best performing SURF, HSV and LBP systems

5.3 Combining Multiple Retrieval Systems

The results obtained in the previous section naturally let us think that the combining the runs could greatly improve the efficiency of the method. In this section, we present the results of combined systems using the strategies shown in table 1. Since we returned for each query the ranked list of all images in the database, CombANZ and CombMNZ are not applicable as they are equivalent to CombSUM. We defined two sets of systems to combine. In the first set, we combine the best HSV, LBP and SURF runs. In the second set, we combine the top 5 systems overall, independently of the descriptors. Results are shown in table 2. It is clear from table 2 that combining the different systems is beneficial to the overall retrieval efficiency. The best performing system, shown in bold, is always the result of a meta-search. As reported in [9,13], CombSUM seems to be the best choice. In these experiments, we have fixed a set of systems to combine. The choice of this set is still an interesting research challenge. The greatest increase in result is achieved when selecting high performing single systems with complementary results. This is the case when combining SURF, HSV and LBP for WANG, while the 5 best systems on this dataset are HSV-based

Table 2. MAP results for different combination strategies

Single runs			Combined runs							
SURF	HSV	LBP	(SURF,HSV,LBP)				(Top5)			
			MIN	MAX	MED	SUM	MIN	MAX	MED	SUM
WANG										
.443	.548	.533	.539	.551	. 522	**.639**	.553	.551	.556	.563
SIVAL Global										
.157	.120	.097	.131	.120	.122	.132	.135	**.157**	.148	.149
SIVAL Local										
.505	.443	.260	.461	.437	.508	.555	.541	.475	.579	**.603**
CALTECH										
.177	.142	.162	.173	.178	.178	.197	.172	.178	.186	**.223**

and the improvement is small. Results are greatly improved for SIVAL-Local and CALTECH101 when combining the top 5, which are composed of 2 HSV and 3 SURF, 4 SURF and 1 LBP respectively. To our knowledge, there is no comparable CBIR results on SIVAL dataset. Ramanathan *et. al.* reported 0.0978 MAP on CALTECH101 using a quadtree extended vector space model [22], but they queried the system using only 10 images per category. This corresponds to a 19.2% of increase compared to the regular BOVW in their experiments (8.3% increase using spatial pyramid matching). We increase the BOVW results by 26%. An example for a SIVAL-local query is shown in figure 6.

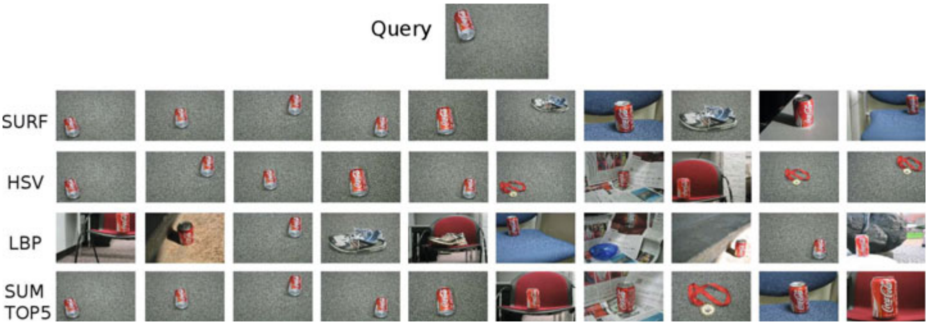


Fig. 6. Example query from SIVAL Dataset (local). First row: best SURF run. Second row: best HSV run. Third row: best LBP run. Fourth row: CombSUM of the top 5 ranking system results, showing a higher average precision for the query.

6 Conclusion

In this work, we proposed a new kind of image signatures based on the clustering of image regions, BOR. We have demonstrated that the BOR signatures are well adapted to build efficient CBIR systems, that can outperform traditional

BOVW signatures on some datasets. BOR signatures are a good counterpart of BOVW, and can be more appropriate depending on the specific queries. Upgrading the traditional BOVW framework to BOR leads to a multiplication of possible visual dictionaries. With this increase, the computational time to build the dictionaries becomes problematic. We proposed to rely on a single pass, incremental clustering algorithm with appropriate k-means++ bootstrapping method. Experiments have shown that visual dictionaries built upon the incremental clustering lead to results as efficient as k-means. Finally, we proposed to combine the results of the different systems using well known meta-search techniques from text information retrieval. In all the cases, the combined results have favorably impacted the performances of the single retrieval systems.

This work has demonstrated that a system based on BOVW and BOR signatures is already very effective. Yet, it opens the way for further research on at least two different aspects. The impact of the segmentation algorithms on the retrieval performances would be interesting to study. Is an accurate segmentation really necessary? Should we over-segment, under-segment, or both? Such a study could allow to fix a few set of segmentation parameters that enable to capture complementary information from the BOR signatures. The second research topic that we will investigate is concerning the meta-search. In our experiments, we just fixed the set to combine under what seemed to be a reasonable choice. However, we know for sure that those choices are far from optimal. Moreover, the choice of an optimal combination is likely to differ between queries. In the future we will investigate meta-models that allow the weighted combination of each single system (*e.g.* weighted Borda-Fuse of Aslam and Montague [2]), where the weights are interactively computed using relevance feedback.

References

1. Arthur, D., Vassilvitskii, S.: k-means++: The advantages of careful seeding. In: SODA 2007, pp. 1027–1035 (2007)
2. Aslam, J.A., Montague, M.: Models for metasearch. In: ACM SIGIR (2001)
3. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. Computer Vision and Image Understanding 110, 346–359 (2008)
4. Deselaers, T., Keysers, D., Ney, H.: Features for image retrieval: An experimental comparison. Information Retrieval 11(2), 77–107 (2008)
5. Duygulu, P., Barnard, K., de Freitas, J.F.G., Forsyth, D. A.: Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part IV. LNCS, vol. 2353, pp. 97–112. Springer, Heidelberg (2002)
6. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In: CVPR 2004 (2004)
7. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. International Journal of Computer Vision 59, 167–181 (2004)
8. Flickner, M., Sawhney, H., Niblack, W., et al.: Query by image and video content: the qbic system. IEEE Computer 28(9), 23–32 (1995)

9. Fox, E.A., Shaw, J.A.: Combination of multiple searches. In: Third Text Retrieval Conference, TREC 1994 (1994)
10. Gokalp, D., Aksoy, S.: Scene classification using bag-of-regions representations. In: CVPR 2007, pp. 1–8 (2007)
11. Hofmann, T.: Learning the similarity of documents: an information-geometric approach to document retrieval and categorization. In: Advances in Neural Information Processing Systems (2000)
12. Ladicky, L., Russel, C., Kohliwu, P.: Associative hierarchical crfs for object class image segmentation. In: ICCV 2009 (2009)
13. Lee, J.H.: Analyses of multiple evidence combination. In: ACM SIGIR 1997 (1997)
14. Levinshtein, A., Stere, A., Kutulakos, K.N., Fleet, D.J., Dickinson, S.J., Siddiqi, K.: Turbopixels: Fast superpixels using geometric flows. IEEE PAMI 31, 1–9 (2009)
15. Li, J., Wang, J.Z.: Automatic linguistic indexing of pictures by a statistical modeling approach. IEEE PAMI 25, 1075–1088 (2003)
16. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 2(60), 91–110 (2004)
17. Lughofer, E.: Extensions of vector quantization for incremental clustering. Pattern Recognition 41, 995–1011 (2008)
18. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE PAMI 27, 1615–1630 (2005)
19. Nister, D., Stewenius, H.: Scalable recognition witha vocabulary tree. In: CVPR 2006 (2006)
20. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE PAMI 24, 971–987 (2002)
21. Rahmani, R., Goldman, S.A., Zhang, H., Cholleti, S.R., Fritts, J.E.: Localized content based image retrieval. IEEE PAMI 30, 1902–1912 (2008)
22. Ramanathan, V., Mishra, S.S., Mitra, P.: Quadtree decomposition based extended vector space model for image retrieval. In: IEEE WACV (2011)
23. Sivic, J., Zisserman, A.: Video google: a text retrieval approach to object matching in videos. In: ICCV 2003, vol. 2, pp. 1470–1477 (2003)
24. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE PAMI 22(12), 1349–1380 (2000)
25. Souvannavong, F., Merialdo, B., Huer, B.: Region-based video content indexing and retrieval. In: CBMI 2005 (2005)
26. Tighe, J., Lazebnik, S.: SuperParsing: Scalable Nonparametric Image Parsing with Superpixels. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 352–365. Springer, Heidelberg (2010)
27. Vieux, R., Benois-Pineau, J., Domenger, J.-P., Braquelaire, A.: Segmentation-based multi-class semantic object detection. Multimedia Tools and Applications, 1–22 (October 2010)
28. Yeh, T., Lee, J., Darrell, T.: Adaptive vocabulary forests br dynamic indexing and category learning. In: ICCV 2007, pp. 1–8 (October 2007)

Active Cleaning for Video Corpus Annotation

Bahjat Safadi[1], Stéphane Ayache[2], and Georges Quénot[1]

[1] UJF-Grenoble 1 / UPMF-Grenoble 2 / Grenoble INP / CNRS, LIG UMR 5217,
Grenoble, F-38041, France
{Bahjat.Safadi,Georges.Quenot}@imag.fr

[2] LIF UMR 6166, CNRS / Université de la Méditerranée / Université de Provence,
F-13288 Marseille Cedex 9, France
Stephane.Ayache@univmed.fr

Abstract. In this paper, we have described the *Active Cleaning* approach that was used to complete the active learning approach in the TRECVID collaborative annotation. It consists of using a classification system to select the samples to be re-annotated in order to improve the quality of the annotations. We have evaluated the actual impact of our active cleaning approach on the TRECVID 2007 collection, using full annotations collected from the TRECVID collaborative annotations and the MCG-ICT-CAS annotations.

From our experiments, a significant improvement of our annotation systems performance was observed when selecting a small fraction of samples to be re-annotated by our cleaning strategy, denoted as *Cross-Val*, than using the same fraction to annotate more new samples. Furthermore, it shows that higher performance can be reached with double annotations of 10% of negative samples, or 5% of all the annotated samples that were selected by the proposed cleaning strategy.

Keywords: Corpus annotation, active learning, annotation cleaning.

1 Introduction

Concept indexing in image and video documents is very important for content-based retrieval. It is a fundamental image/video retrieval problem: given a data set of images and a query (visual concept), which images do present the given visual concept? Generally, classical keyword based search is not possible due to the frequent absence of appropriate text annotation. Signal-level descriptions (e.g. color and texture) are also known to be inappropriate for the task since they do not represent the semantic content well, and are not easy to handle for users. Automatic concept indexing has been one of the main focus of the TRECVID campaigns (evaluation of video retrieval systems, [12]) since 2002.

Most concept indexing systems use a supervised learning approaches [7,13], in which concepts are learned from sets of positive and negative samples. The models and training algorithms are important for systems' performance, but the training data also play an important role. While it is quite easy and cheap to get large amounts of raw data, it is usually very costly to have them annotated, due to the involvement of human intervention for judging the "ground truth".

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 518–528, 2012.

While the volume of data that can be manually annotated is limited due to the cost of manual intervention, it still possible to select the data samples that will be annotated, so their annotation is "as useful as possible" [1]. Deciding which samples will be the most useful is not trivial. *Active learning* is an approach in which an existing system is used to predict the usefulness of new samples. This approach is a particular case of *incremental learning* in which the system is trained several times with a growing set of labeled samples. The objective is to select as few samples as possible to be manually annotated so that these annotations lead to better classification performance.

The quantity of the annotated samples is important for system's performance, Their quality is also very important since most advanced classification methods are sensitive to mislabeled training examples. Using crowd-sourcing [3,14] methods leads to quickly changing the landscape for the quantity and the quality of labeled data available to supervised learning. While such data can now be obtained more quickly and cheaply than ever before, the generated labels also tend to be far noisier due to limitations of quality control mechanisms. The quality of the labels obtained from annotators varies. Some annotators provide random or bad quality labels in the hope that they will go unnoticed and still be paid, and yet others may have good intentions but completely misunderstand the task at hand or they become distracted or tired over time. The standard solution to the problem of "noisy" labels is to assign the same labeling task to annotators, in the hope that at least a few of them will provide high quality labels or that a consensus emerges from a great number of labels. In either case, a large number of labels is necessary, and although a single label is cheap, the costs can accumulate quickly. It can be observed, that if one is aiming to produce a quality labels within minimum time and money, it makes more sense to dynamically decide on the number of labelers needed. For instance, if an expert annotator provides a label, we can probably rely on it being of high quality, and we may not need more labels for that particular task. On the other hand, if an unreliable annotator provides a label, we should probably ask for more labels until we find an expert or until we have enough labels on which we can apply the majority vote to decide the final label.

Given the substantial human effort required to gather good training sets -as well as the expectation that more data is almost always advantageous-, researchers have begun to explore new ways to collect labeled data. Both active learning and crowd-sourced labeling are promising ways to efficiently build up training sets for concept indexing and retrieval. The active learning techniques aim to minimize human effort by focusing label requests on those that appear to be the most informative samples to the classifier [8,4,15,10,2], whereas crowd-sourcing work explores how to package annotation tasks in such a way that they can be dispersed effectively [15,5,11]. The interesting questions raised in these areas - such as dealing with noisy labels, measuring reliability, mixing strong and weak annotations - make it clear that data collection is no longer an ordinary necessity, but a thriving research area in itself.

Recent years have seen significant growth in label aggregation researches. For example, Vijayanarasimhan et al. presented an approach for live learning of object detectors [15], in which the system autonomously refines its models by actively requesting crowd-sourced annotations on images crawled from the worldwide web. Kumar et al. showed that generating additional labels for labeled examples reduces the potential label noise

[5], and faster learning can be achieved by incorporating knowledge of worker accuracies into consensus labeling. Sheng et al. in [11] presented repeated-labeling strategies of increasing complexity, and their results show clearly that when labeling is not perfect, selective acquisition of multiple labels is a strategy that data miners should have in their repertoire; and for certain label-quality/cost regimes, the benefit is substantial.

Using multiple annotations to reduce labeling noise have also been used in the context of crowd-sourcing; although a full double or triple annotation is even more costly than a simple full one; and it is not in the spirit of data annotation based active learning approaches, in which we do not need to annotate all the samples in the data set. In this paper, we propose to use an active learning approach for selecting samples for second or third annotations. We call this approach *Active Cleaning*. Using the simulated active learning approach and all the available annotations on TRECVID 2007 development set, we have designed different experiments in order to evaluate the benefits of the active cleaning approach, as well as the relative efficiency of the associated strategies.

The outline of the paper continues as follows: the annotation type is presented in section 2; the active cleaning approach is discussed in section 3; section 4 describes the experimental results, while Section 5 presents concluding remarks.

2 Annotation Type

We consider the binary annotations, which are often used for image/video classification, such as "Does the video-shot contain an instance of the given visual concept C or not?". Let t_x the target value for the sample x and y_{xk} the k^{th} label for the sample x given by an annotator. The set of target values T and the set of labels Y are binary scalars, hence $y_{xk}, t_x \in \{-1, 1\}$. T values are decided by applying the majority vote on Y values. In the collaborative annotation we have a third case that we call *skipped*: the user already saw the shot but he/she was confused of its label. Three possible annotations were considered: *Positive, Skipped* and *Negative* we name them *pos, skip* and *neg* respectively.

3 Active Cleaning

Active cleaning is the method of using an existing classification system for selecting samples for re-annotation, in order to improve the quality of an annotated corpus. It may be implemented in an incremental way, in conjunction with an active learning based annotation algorithm. In this case, the annotations may be cleaner and more correct, which makes the active learning more effective and efficient. Active cleaning may also be used for cleaning an already existing annotation, which can be either complete or partial. In this case, the benefits are only at the level of the resulting annotation.

Cleaning during active learning is the approach that was used in TRECVID collaborative annotation system. The active cleaning algorithm based concept annotation is detailed in Algorithm 1, which applies the classical active learning algorithm in which we added the cleaning process. Let *D* be the data set which needs to be labeled as containing a target concept (e.g. Airplane, Person..); L, U the labeled and unlabeled subsets respectively, thus $L \cup U = D$ and $L \cap U = \phi$. N a set of the possible choices of the user to label sample x as containing a given concept or not. Three possible choices are allowed by the annotation system: *Positive, Skipped* and *Negative*, (see section 2). We denote Q_{al} and Q_{cl} to be the selection strategies of the active learning and cleaning respectively (see section 3.1). Before explaining the algorithm let us introduce some definitions in order to facilitate the understanding of our algorithm:

1. The set of available annotations: $Y = \{y_{xk} \in N : x \in L; k \in \{1, 2, \ldots, t\}\}$, where y_{xk} defines the k^{th} label of sample x given from an annotator. Hence we ask, orderly, for up to three annotations for each sample, we set $t = 3$.
2. The subset of conflicting samples: $ConfANN = \{x \in L : y_{x1}, y_{x2} \in Y \wedge y_{x1} \neq y_{x2}\}$, a subset of L that have two different annotations for each sample.
3. The subset of second-annotations: $SANN_{Q_{cl}} = \{x \in L : y_{x1} \in Y \wedge y_{x2} \notin Y\}$, a subset of L that have only one annotation for each sample, selected according to the cleaning strategy Q_{cl}.
4. The subset of primary-annotations: $PANN_{Q_{al}} = \{x \in U\}$ samples have no available annotations, selected according to the active learning strategy Q_{al}.

Algorithm 1. Active Cleaning Algorithm Based Concept Annotations

```
$D$: all data samples.
$L_i, U_i$: labeled and unlabeled subsets of $S$, ($L_i \cup U_i = D$).
$A$=(train, predict): the elementary learning algorithm.
$Q_{al}, Q_{cl}$: the selection strategies, respectively, for the active learning and cleaning.
$Y_i$: available annotations for $L_i$.
Initialize $L_0$ and $Y_0$.
while $D \setminus L_i \neq \emptyset$ do
    $m_i \leftarrow$ Train($A$, $L_i, Y_i$)
    $P_u \leftarrow$ Predict($U_i$,$m_i$)
    $P_l \leftarrow$ Predict($L_i$,$m_i$)
    (*) Select the subset $ConfANN \subset L_i$
    (**) Apply $Q_{cl}$ on $P_l$ in order to select the subset $SANN \subset L_i$.
    (***) Apply $Q_{al}$ on $P_u$ in order to select subset $PANN \subset U_i$.
    $\tilde{Y}$ = (Label ($ConfANN$)) $\cup$ (Label ($SANN$)) $\cup$ (Label ($PANN$))
    $Y_{i+1} \leftarrow Y_i \cup \tilde{Y}$
    $L_{i+1} \leftarrow L_i \cup PANN$
    $U_{i+1} \leftarrow U_i \setminus PANN$
end while
```

The algorithm is iterative, for implementation purposes, the elementary learning algorithm A is split into two parts: train and predict. The algorithm starts by initializing the L_0 set, which can be done by collecting initial labels Y_0 for some samples of D, through the annotators. Iteratively, the development set D is split into two parts: labeled samples L_i, and unlabeled samples U_i. Then classifier A is trained using L_i with its associated labels Y_i and obtains the model m_i, which is then used to predict the scores - likeliness to contain the target concept - P_l and P_u of the samples in L_i and U_i sets respectively. These predicted scores are used to select the samples to be labeled in the next iteration. However, the selection is done in three steps: first the algorithm chooses the samples with conflicting labels *ConfANN* (*); then it apply the cleaning strategy Q_{cl}

on the predicted scores P_l of the samples in L_i, and selects the samples of the *SANN* set to be re-annotated by different users (**). Finally, the predicted scores P_u of unlabeled samples in U_i are passed to the Q_{al} strategy, which selects the *PANN* set (***). The annotators are asked to annotate all the samples in these three sets, taking into account that a data sample x can be examined maximum once by the same annotator, and annotators cannot access the judgments of other annotators. When the new annotations set $\tilde{Y}$ is completed, it will be added to the global annotations set Y. The set *PANN* is added to the L_i set to produce the set L_{i+1}, and it is also removed from the U_i set to produce the U_{i+1} set. Thus a new iteration is started.

3.1 Active Learning and Cleaning Strategies, Q_{al} and Q_{cl}

In this paper, the selection strategy of the active learning, Q_{al} has been chosen to implement the relevance sampling, which selects the most probable positive samples regarding to their classification scores (samples with high prediction scores). It was observed that this is a good strategy for sparse concepts [2,10] where the objective is to find as many positive samples as possible from the unlabeled set U to be annotated.

For the active cleaning, several strategies Q_{cl} can be used for the selection of samples to be re-annotated. They may depend upon the type of annotation (number of possible judgments for instance) and the problem of highly imbalanced dataset, which is a very frequent case in video indexing. Furthermore, these strategies can depend on whether the first annotations were done incrementally or at once. We propose here a cleaning strategy, denoted *Cross-Val*. It is based on re-annotating the wrongly labeled samples due to an error of the annotator (for instance if the annotator missed the change of the concept to annotate). Detecting the wrongly labeled samples is done by training classifiers on these labeled samples and using the trained models to predict the correctness of these labeled samples. Thus, through the predicted score of each sample we can expect if the sample has a correct label or not. The wrongly labeled samples are then those having positive labels with low scores, or negative labels with high scores. Basically, this strategy selects fractions of the labeled samples. These fractions denoted as $P\%, N\%$ and $S\%$ and refer to annotated samples as positive, negative and skipped respectively, (see section 2). Furthermore, the selected samples are then proposed to annotators for a second annotation round.

In *Cross-Val* strategy, the *N%, P% and S%* correspond to the percentage of the labeled samples as *Negative, Positive* and *Skipped*. This includes the baseline (no second annotations), when *N=P=S=0*, re-annotating all skipped and positive samples (*SkipPos*) by *P=S=100 and N=0*, and the extreme fully cleaning *N=P=S=100*. In this paper, we evaluated the Cross-Val strategy with different fractions and several ways of re-annotations as in table 1. Our goal is to study the system performance with the Cross-Val strategy for cleaning annotations, furthermore to find the best fraction values for this process.

3.2 A Posteriori Cleaning

In the case of a posteriori cleaning, we assume that first annotations have been done, thus we have one annotation for each sample, and they will be cleaned globally with a single iteration. A system is trained using the available annotations and the samples are ranked according to their probability of being positive by the system. The given fractions P%, S% and N% of samples annotated as positive, skipped and negative will be used respectively to select the samples for second annotation round. For the positive samples, the system chooses the P% of positive samples with false prediction (have lowest predicted scores). For the negative samples, it chooses the first N% of negatives samples with the highest predicted scores annotated. For the skipped samples we chose the S% of the skipped samples that have uncertainty scores (predicted score is close to the classifier boundaries). In all cases, a third annotation is required from the annotators when conflicting is detected, between the first and second annotations.

4 Experiments

We have evaluated the active cleaning approach based on the *Cross-Val (Q_{cl})* strategy in a variety of contexts. It has been applied with a classification system using four types of image descriptors, which are taken from IRIM GDR-ISIS partners [9], including *the combination of Histogram and Gabor, Global-Tlep, Global-Qwm and Bow-Sift*. The multiple-SVM classifiers with RBF kernel was applied as the classification algorithm, which was implemented as in [10]. The evaluations were conducted using the TRECVID 2007 collection metrics and protocol. The TRECVID 2007 collection contains two main sets: the development set consists 21532 sub-shots with 36 concepts (or “high level features”) selected from the LSCOM-lite [6] set for annotation, and the test set which consists of 22084 sub-shots. In TRECVID 2007, the evaluation was done on the test set using only 20 concepts which were chosen by the National Institute of Standards and Technology (NIST). In order to carry out the experiments on the simulated active cleaning, three annotations are needed for each concept (c)× sub-shot (x) in this dataset. We have collected and completed all the annotations, which were produced by the collaborative annotation on the considered database, that we get at least two labels for each $c \times x$. In addition, we used a complete set of annotations: one label for each video shot, produced independently by a group from the Multimedia Content Group, Institute of Computing Technology, Chinese Academy of Sciences (MCG-ICT-CAS).

Since our goal, in this work, is to study the system performance with the *Cross-Val (Q_{cl})* strategy for cleaning annotations, we present the different fractions that were used in our experiments in table 1. In which $E1$ is the baseline, $E8$ refers to the cleaning of all skipped and positive samples, and $(E2, E3, \ldots, E7)$ indicates the cross-validation strategy with different *(N%, P%, S%)* fractions.

Table 1. The *(P%,N%,S%)* fraction values that were used in our experiments with our active cleaning strategy

Q_{cl}	E1	E2	E3	E4	E5	E6	E7	E8
pos %	0	10	0	0	5	10	20	100
neg %	0	0	0	10	5	10	20	0
skip %	0	0	10	0	5	10	20	100

4.1 The Active Learning Steps and the Cold-Start

In calculating the number of the required annotations at each active learning iteration (including the third, second and first annotations), a variable step size function can be used. In practice we used 30 steps in total, considering the geometric scale function with the following formula: $s_k = s_0 \times (n/s_0)^{k/K}$, where n is the total size of the development set, s_0 is the size of the training set at the cold-start, K is the total number of steps and k is the current step. At each step (or iteration) the algorithm calculates the s_k to be the size of the new training set and it chooses the number of samples that needs to be cleaned cl_k, and the new samples to be labeled with size equal to $new_k = s_k - s_{k-1} - cl_k$.

In this evaluation, the harmonic mean has been applied as a fusion function for the multiple-SVM results (scores). The cold start problem was solved by using another TRECVID collection, the 2005 one. We trained SVM classifiers on the TRECVID 2005 collection and predicted the usefulness on the development set of TRECVID 2007; we have started with annotating the first 100 samples at the top of the ranked list (samples having high scores), then the Active learning and cleaning system was run to label all the shots within the development set.

4.2 Available Annotations

In the following we present the two resources of the considered annotations:

1- Collaborative Annotations (*CA*): Annotations were done in collaboration with 32 groups of participants at TRECVID, each group contributed with several annotators. The annotation system used is based on the active learning approach. For each concept (c) × sub-shot (x) in the data set, the annotators have left the choice to label x as containing an instance of concept t or not, *pos* and *neg* respectively, they also can skip annotating it in the case of confusing on its label. This can be considered as crowd-sourcing, since each shot could be proposed to several annotators to judge whether it contains c or not. Since we were limited in time of the annotating phase of TRECVID, this data set was not fully annotated. Furthermore, there are multiple annotations for the annotated samples L for each concept c, and they are still available and can be used as multiple judgments for the experiments on simulated active cleaning approach. For our experiments, these judgments have been completed to have at least two judgments for each sample.

2- MCG-ICT-CAS Annotations (*MCG*): The MCG-ICT-CAS team has produced, on its own, complete and independent annotations of all the concepts (c)× sub-shots (x). The annotations were made by a pool of students. Each student could annotate shots to contain only a specific concept, and the annotations were done on all the data set (active learning was not considered). Each $c \times x$ has only one label, since only one annotator (student) could examine and label it, which means that it does not contain multiple annotations. This annotations set has the advantage of being complete, and since it was made using a smaller number of annotators, one can say it is more consistent.

These annotations were taken by different annotators and two different systems, and they have some noise in annotations. These noises came from the annotation systems used and the annotators themselves. For instance, given concept *Sports*: we got 482

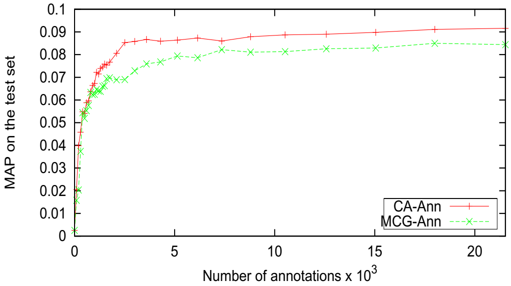


Fig. 1. The MAP calculated on 20 concepts of the TRECVID 2007 test set, with two different annotation sources

positive samples from the CA annotations, while from MCS annotations we got only 226 positives; furthermore, the two sources were agreed on only 168 positive samples.

The performance of our baseline system, by using only single annotations from the two annotation resources (CA and MCG), is shown in figure 1. This figure shows the effectiveness in performance, of the classification system, with the number of the annotated sub-shots from the development set. Thus, it presents the MAP, of the 20 concepts, calculated on the test set. For both curves, we consider a better curve to be: the fastest in growing, and the highest MAP value, it reaches, especially in the beginning. As we can see, the system performance using the annotations produced by the CA is much higher than using the MCG annotations. This can be due to the annotation strategy, which is different in the two cases as described above, and it may also be related to the annotators themselves.

From this result, we assume that for each concept $(c) \times$ sub-shot (x), the annotations taken from CA are cleaner than the MCG, and we planned two main experiments to study the effectiveness of the active cleaning strategies:

1. (MCG-CA): the first annotation, for each $c \times x$, is taken from low-quality annotators, (MCG), and the second annotation was taken from better-quality annotators (CA).
2. (CA-MCG): the first annotation, for each $c \times x$, is taken from good-quality annotators, (CA), and the second annotation was taken from lower-quality annotators (MCG).

In both experiments, we have used the second annotation produced by CA as the third annotation, and it was used when the two annotations (CA and MCG) are conflicting.

4.3 Active Cleaning Effectiveness

We have studied the performance of the annotation system using the cleaning strategy, *Cros-Val* with different P%, N% and S% fractions as set in table 1. Thus, we report the obtained results from our two main experiments MCG-CA and CA-MCG. For simplicity, we report the results of the last iteration of the active cleaning, in table 2. Furthermore, in figure 2 we present the full iterative results of the cleaning performance, for some experiments.

Table 2. The result of the cleaning strategies with the eight experiments described in table 1

	MCG-CA	#Annotations	CA-MCG	#Annotations
E1=N0P0S0	0.084	21532	0.091	21532
E2=N0P10S0	0.084 +0%	+65	0.091 +0%	+46
E3=N0P0S10	0.086 +2%	+50	0.092 +1%	+11
E4=N10P0S0	0.095 +14%	+2100	0.096 +5%	+2150
E5=N5P5S5	0.096 +14%	+1100	0.095 +4%	+1100
E6=N10P10S10	0.097 +15%	+2200	0.090 -1%	+2215
E7=N20P20S20	0.097 +15%	+4400	0.095 +4%	+4420
E8=N0P100S100	0.086 +2%	+1150	0.093 +2%	+580

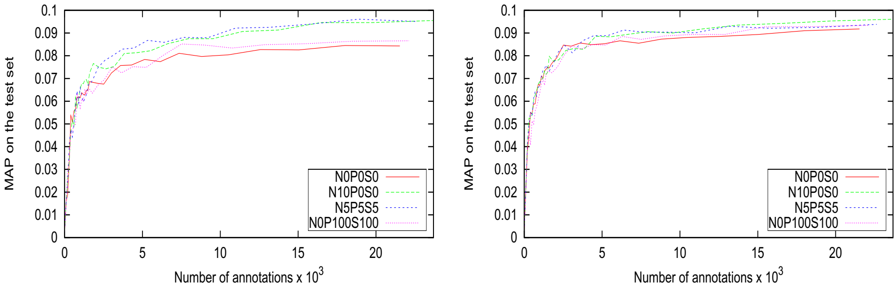


Fig. 2. Active cleaning strategies: Cleaning *MCG* annotations by *CA* in left, and in right Cleaning *CA* by *MCG* annotations

Table 2 presents the evaluation results of the two main combinations MCG-CA and CA-MCG, using the cleaning strategy, *Cros-Val* with different P%, N% and S% fractions as set in table 1. Moreover, it presents the number of cleaning annotations required for each experiment in the two considered combinations. As we can see from this table, some experiments do not have a real effect on the system performance, especially when the cleaning system does not include the negative samples, as in E2, E3 and E8. This is due to the fact, that the number of re-annotated samples is very small, since there are few positive and skipped samples in the data set. However, the performance is higher when the negative samples were included in the cleaning system; moreover it goes up to 15% in the case of MCG-CA and 5% in CA-MCG. This is expected since, as shown in figure 1, we consider that annotations from MCG have lower-quality than CA.

Figure 2 shows the effectiveness of the active cleaning strategies E4 and E5 compared to the baseline (E1) and the *Skip-Pos* (E8) strategy, with the two considered experiments, the MCG-CA (left) and CA-MCG (right). As we can see in this figure, in both experiments, the system performance (using the MAP) was increased when the cleaning system considered the re-annotations of negative samples, as in E4 and E5. Hence, the Cross-Val strategy E4 works in re-annotating only 10% of the negative samples, and E6 re-annotating 5% of each type of the annotations (positive, negative, skipped). Moreover, the active cleaning maintains the purpose of using the active learning approaches to annotate large scale image/video databases. Thus, the best performance could be obtained when annotating only 15-30% of the development set. The enhancement in the performance is more important when cleaning the lower-quality annotations than better-quality annotations. Furthermore the active cleaning can better enhance the performance under the condition that the number of annotations is the same.

4.4 A Posteriori Cleaning Effectiveness

Table 3 shows the same results in the case of a posteriori cleaning. The results are similar to the results obtained by active cleaning, as shown in the previous section, but Active cleaning is is more effective and efficient. In this table, as we can see, using the full three annotations (N100P100S100) leads to a better performance than using different fractions as in table 1. Even though, it requires three times as many annotations as the baseline, while each of the other combinations requires only few more annotations than the baseline. This is due to either the fraction is small (e.g. N5* or N10*) or because the target concepts are sparse.

Table 3. The result of the posteriori cleaning with the eight experiments described in table 1

	MCG-CA	CA-MCG
E1=N0P0S0	0.0840	0.0910
E2=N0P10S0	0.0833	0.0917
E3=N0P0S10	0.0847	0.0927
E4=N10P0S0	0.0858	0.0917
E5=N5P5S5	0.0841	0.0921
E6=N10P10S10	0.0852	0.0910
E7=N20P20S20	0.0877	0.0921
E8=N0P100S100	0.0866	0.0931
Full3=N100P100S100	0.0962	0.0962

5 Conclusions

We have described the active cleaning approach that was used to complement the active learning approach in the TRECVID collaborative annotation. The actual impact of the active cleaning approach was evaluated on TRECVID 2007 collection. The evaluations were conducted using complete annotations that were collected from different resources, including the TRECVID collaborative annotations and the MCG-ICT-CAS annotations.

From our experiments, a significant improvement of the annotation quality was observed when applying the cleaning by cross-validation strategy, which selects the samples to be re-annotated. Experiments show that higher performance can be reached with minimum double annotations of 10% of negative samples or 5% of all the annotated samples selecteded by the proposed cleaning strategy using cross-validation. It has been shown that, with an appropriate strategy, using a small fraction of the annotations for cleaning improves much more the system's performance than using the same fraction for adding more annotations.

Acknowledgments. This work was partly realized as part of the Quaero Program funded by OSEO, French State agency for innovation. Experiments presented in this paper were carried out using the Grid'5000 experimental testbed, being developed under the INRIA ALADDIN development action with support from CNRS, RENATER and several Universities as well as other funding bodies (see https://www.grid5000.fr).

References

1. Angluin, D.: Queries and concept learning. Machine Learning 2, 319–342 (1988)
2. Ayache, S., Quénot, G.: Evaluation of active learning strategies for video indexing. In: Signal Processing: Image Communication (2007)
3. Howe, J.: The rise of crowdsourcing. Wired Magazine 14(6) (June 2006)
4. Joshi, A.J., Porikli, F., Papanikolopoulos, N.: Multi-class active learning for image classification. In: CVPR, pp. 2372–2379 (2009)
5. Kumar, A., Lease, M.: Modeling annotator accuracies for supervised learning. In: Proceedings of the Workshop on Crowdsourcing for Search and Data Mining (CSDM) at the Fourth ACM International Conference on Web Search and Data Mining (WSDM), Hong Kong, China, pp. 19–22 (February 2011)
6. Naphade, M., Smith, J.R., Tesic, J., Chang, S.-F., Hsu, W., Kennedy, L., Hauptmann, A., Curtis, J.: Large-scale concept ontology for multimedia. IEEE MultiMedia 13, 86–91 (2006)
7. Naphade, M.R., Smith, J.R.: On the detection of semantic concepts at trecvid. In: MULTIMEDIA 2004: Proceedings of the 12th Annual ACM International Conference on Multimedia, pp. 660–667. ACM Press, New York (2004)
8. Qi, G.-J., Hua, X.-S., Rui, Y., Tang, J., Zhang, H.-J.: Two-dimensional active learning for image classification. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
9. Quénot, G., Delezoide, B., le Borgne, H., Moëllic, P.-A., Gorisse, D., Precioso, F., Wang, F., Merialdo, B., Gosselin, P., Granjon, L., Pellerin, D., Rombaut, M., Bredin, H., Koenig, L., Lachambre, H., Khoury, E.E., Mansencal, B., Benois-Pineau, J., Jégou, H., Ayache, S., Safadi, B., Fabrizio, J., Cord, M., Glotin, H., Zhao, Z., Dumont, E., Augereau, B.: Irim at trecvid 2009: High level feature extraction. In: TREC 2009 notebook, November 16-17 (2009)
10. Safadi, B., Quénot, G.: Active learning with multiple classifiers for multimedia indexing. In: CBMI, Grenoble, France (June 2010)
11. Sheng, V.S., Provost, F., Ipeirotis, P.G.: Get another label? improving data quality and data mining using multiple, noisy labelers. In: Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2008, pp. 614–622. ACM, New York (2008)
12. Smeaton, A.F., Over, P., Kraaij, W.: Evaluation campaigns and trecvid. In: MIR 2006: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, pp. 321–330. ACM Press, New York (2006)
13. Snoek, C.G.M., Worring, M.: Multimodal video indexing: A review of the state-of-the-art. Multimedia Tools and Applications 25(1), 5–35 (2005)
14. Snow, R., O'Connor, B., Jurafsky, D., Ng, A.Y.: Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, pp. 254–263. Association for Computational Linguistics, Stroudsburg (2008)
15. Vijayanarasimhan, S., Grauman, K.: Multi-level active prediction of useful image annotations for recognition. In: NIPS, pp. 1705–1712 (2008)

Improving Cluster Selection and Event Modeling in Unsupervised Mining for Automatic Audiovisual Video Structuring

Anh-Phuong Ta[1], Mathieu Ben[2], and Guillaume Gravier[3]

[1] INRIA-Rennes, Campus Beaulieu, F-35042 Rennes, Cedex, France
anh-phuong.ta@inria.fr
[2] Powedia, 12A avenue des Peupliers, F-35510, Rennes, Cedex, France
mathieu.ben@powedia.com
[3] CNRS-IRISA, Campus Beaulieu, F-35042 Rennes, Cedex, France
guillaume.gravier@irisa.fr

Abstract. Can we discover audio-visually consistent events from videos in a totally unsupervised manner? And, how to mine videos with different genres? In this paper we present our new results in automatically discovering audio-visual events. A new measure is proposed to select audio-visually consistent elements from the two dendrograms respectively representing hierarchical clustering results for the audio and visual modalities. Each selected element corresponds to a candidate event. In order to construct a model for each event, each candidate event is represented as a group of clusters, and a voting mechanism is applied to select training examples for discriminative classifiers. Finally, the trained model is tested on the entire video to select video segments that belong to the event discovered. Experimental results on different and challenging genres of videos, show the effectiveness of our approach.

Keywords: Video mining, Video structuring, Multimodality, Mutual Information, Event discovery, Structural event, Audiovisual consistency.

1 Introduction

The purpose of video structuring is to automatically find structural events in video sequences. Obviously, a structural element is represented as a key content of videos. Examples of such structural events are jingles in news videos, anchorpersons or participants in TV program videos, choruses from karaoke music videos, etc. Due to its potential applications in various fields, such as video summarization, video indexing and browsing, and content based video retrieval, video structuring can be considered as a crucial step in content-based video analysis. Existing methods can be broadly classified into two groups: (i) dense segmentation of the entire video, in which the video is mapped to a predefined structure [5] [6], and (ii) detection of a specific event like goals in sport videos [7] [8], advertisements, or anchorpersons [3] [4]. Despite their good results, most of these methods have two shortcomings: (a) they require manually annotated data for training models, thus these methods lack the generality to cope with diverse video sources; (b) they use clustering techniques to group similar video

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 529–540, 2012.

segments, which form structural events. However, the problems that many clustering algorithms encounter are the choice of the optimal number of clusters, and how to deal with outliers.

It should be noted that there are very few research results available on discovering audio-visual events in an unsupervised way. Previous methods that are closely related to our work are those concerning video mining (video structuring). Video mining approaches [9] [10] [11] [12], which aim to detect regularly repeating patterns, have focused on the discovery of near-duplicate repetitions. However, these methods cannot deal with the structural events exhibiting content and temporal variations, i.e., the repetitions are not exact. There have been several works in unsupervised mining from videos [2] [5] [6]. The main idea of these methods is to exploit structural elements through mapping the entire video to predefined models. Because they are based on the assumption of dense segmentation, these approaches cannot handle the discovery of sporadic structural events.

In order to overcome the limitations mentioned above, in our previous work [1] we have proposed an unsupervised method to detect structural events from audio-visual documents without prior knowledge of the genre of the video. As presented in [1], two hierarchical clustering trees (called dendrogram) are first constructed for both audio segments and video shots (keyframes). We then measure the consistency between all pairs of audio-visual clusters by using mutual information (MI). Finally, several heuristics are used to select the best (i.e., the most relevant) audio-visual pair that represents a structural event. In this paper, we propose an extension to this method that makes it more robust to deal with variability and easier to extract multiple structural events. Compared to our previous work [1], the main contributions of this paper are summarized as follows:

- We present a new measure for selecting audio-visually consistent elements. The proposed measure is slightly different from the one used in [1], in that instead of using the original mutual information measure, which has 4 possible states (correlations) for two input binary variables, we use only two positive correlations (see section 3.1 for more details).
- We take into account the structure of the two given dendrograms to represent events, ie., each event is represented by a group of audio-visual cluster pairs, as opposed to only one pair used in [1], which may result in partial events (i.e. events whose occurrences are incompletely detected).
- In order to construct a model for each event, we propose a voting mechanism to select positive and negative examples as input vectors for Support Vector Machines (SVM). Note that, in [1], positive and negative examples are selected entirely based on the audiovisual segments of the best pair of clusters. Due to errors in clustering, however, not all these segments belong to the same structural event. In this work, we propose a more accurate method, in which each audiovisual segment will first cast its votes for negative or positive. Then, thresholds are used to select positive and negative examples (see section 3.3 for more details).

The rest of this paper is organized as follows. After briefly summing up the early work [1] for discovering structural events in section 2, we introduce our extensions of this method in section 3. Section 4 describes experimental results. Finally, we conclude and give some perspectives of this work.

2 Discovering Audio-Visually Consistent Events

In this section, we briefly summarize our early work on unsupervised video structuring [1], on which our approach is based. Figure 1 illustrates the main components of our approach, as well as the original approach[1] in [1]. Recall that our main objective was to design a generic approach to find events of interest that share common characteristics, including audio and visual presentations. First, the audio and video streams are respectively segmented into audio and video (shot) segments. We then extract commonly used audio and visual features (Gaussian components for audio, and RGB histograms for video) for segmentation and clustering. Classical bottom-up clustering algorithms are then applied for each modality to provide two different dendrograms representing video and audio clustering results. Given these two dendrograms, the work in [1] consists of 3 steps:

1. The first step measures mutual information (MI) between all pairs of audio and video clusters. These pairs of clusters are then ranked according to their MI values, and a list of n-best top pairs of clusters (i.e., the ones having highest MI values), called N-best list, is extracted.
2. The second one selects the best consistency pair among different pairs in the N-best list using several heuristics. This selected pair is considered to be representative of the most relevant event.
3. In the last step, based on the found event (the pair of clusters selected from the above step), positive and negative samples are extracted (see section 3.3 for more details). Then, a binary SVM classifier is applied to construct the event model.

In this paper, we aim at improving the cluster selection (step 1), and the event modeling techniques (step 3). We do not focus on using heuristics (step 2), which depend on the mining objectives, but we propose a new way to represent candidate events. The details of our approach is described in the next section.

3 Improving Cluster Selection and Event Modeling Techniques

Despite the promising initial results, the method introduced in [1] has several limitations:

a) It only allows to discover a single audio-visually consistent event.
b) It relies entirely on the consistent power of a single pair of clusters, i.e., a structural event is represented by a pair of audio-visual clusters only. Thus this method may result in partial structural events.

[1] This scheme, which illustrates the different components of our work, is slightly different from the original one presented in [1], but both operate on the same principle.

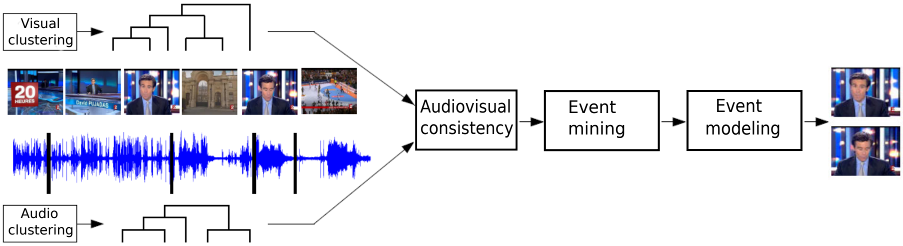


Fig. 1. Our general scheme for mining structural events from videos (this figure is adapted from [1]). In this figure, we show a simple example illustrating the detection of an anchor-person.

c) The original mutual information measure used in [1] to compute the consistency between two binary variables, which represent a pair of audio-visual clusters, is symmetric. That is, $MI(A,V) = MI(\bar{A},V) = MI(A,\bar{V}) = MI(\bar{A},\bar{V})$, where A, V are two binary variables denote the existence of audio, and video clusters, respectively; and $MI(\cdot,\cdot)$ is a function that measures mutual information of two given variables. This measure can give good results for detecting events that appear quite regularly, or for events, for which the corresponding audio and video segments from the selected clusters are quite strongly consistent. However, it cannot distinguish between relevant and irrelevant events (an irrelevant event appears when one of the two binary variables, or both is absent, i.e., for the cases of $MI(\bar{A},V)$, $MI(A,\bar{V})$, and $MI(\bar{A},\bar{V})$).

In this work we overcome these limitations by exploiting the N-best list based on the structure of the trees (i.e., dendrograms).

3.1 Estimating the Consistency of Audio-Visual Clusters

After the hierarchical clustering of audio and video segments extracted from an input video, we obtain two corresponding dendrograms, where each node has a set of segments from the corresponding modality. Now we measure audio-visual consistency between an audio cluster and a visual cluster by using mutual information (MI) for two random variables. A large MI value indicates that the two corresponding clusters are closely consistent with each other and share more mutual information. In our case, for finding a repeating event, we aim at finding the consistency of audio-visual segments from a pair of clusters, which can be represented by the two positive correlations of the original mutual information. Let (C_i^A, C_j^V) be the i-th and the j-th nodes of the audio and video dendrograms, respectively. The mutual information between C_i^A and C_j^V is given as follows:

$$MI(C_i^A, C_j^V) = \sum_{(a,v)\in\{(0,0),(1,1)\}} p(a,v)\ln(\frac{p(a,v)}{p(a)p(v)}) \quad (1)$$

where a and v are binary random variables which respectively denote membership in C_i^A and C_j^V. The probabilities $p(a,v)$, $p(a)$, and $p(v)$ are estimated from the temporal distribution of segments. For instance, the join probability $p(a = 1, v = 1)$ is measured as the sum of the amount of time each segment of C_i^A co-occurs with a segments of C_j^V, normalized by the total duration of the video.

Equation 1 is applied for measuring MI between all audio and visual cluster pairs. Then, a list of pairs of clusters is established, and sorted in descending order according to its MI value. The objective of sorting is to help discover candidate events from more consistent to less consistent, and to filter out the irrelevant pairs (inconsistent pairs), i.e., the ones having low MI values. After removing inconsistent pairs, the final list remains n best pairs, called N-best list. In the next sub-section, we will analyze this list to discover structural events.

3.2 Event Mining

From the N-best list obtained in the above step, we extract candidate events. Due to the nature properties of hierarchical clustering algorithms, there are redundancies (in terms of structural relationships and similar characteristics) in the N-best list. Obviously, the pairs of clusters in the N-best list, which share redundant contents, should be part of an event (i.e., they should belong to the same structural event). More precisely, using a pair of audio-visual clusters (a, v) as example, if (a, v) represents a structural event, all other pairs of clusters from the two sub-trees constructed from (a, v) will represent (sub)instances (or other instances) of this event. Therefore, this event should be represented by a group of cluster pairs, including (a, v) and its *neighboring* pairs. See figure 2 for an explanation of this principle. Based on the analysis above, we extract multiple structural events in the following way: starting from the first pair in the N-best list (i.e., the one having highest MI value), we search from the N-best list to find all successor pairs from the two sub-trees constructed from this pair. The found successor pairs are grouped together with the current pair to establish a representative group for

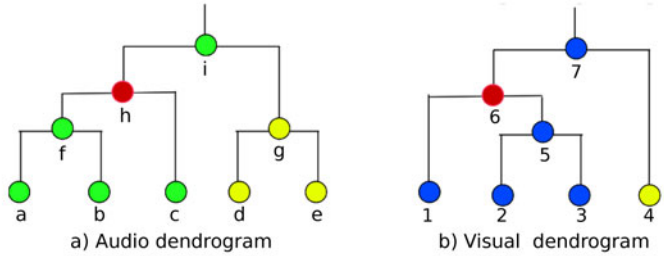


Fig. 2. Partial views of the audio (a) and visual (b) dendrograms, which illustrate the principle of the construction of a representative group for a given candidate event represented by a pair of clusters (see section 3): assuming that the pair of audio and video clusters in the N-best list (h,6) represents a candidate event. Then, the list of audiovisual candidate pairs contains: {(a,1) (a,2), (a,3),(a,5), (a,6), (a,7), (b,1), (b,2), (b,3), (b,5), (b,6), (b,7), (c,1), (c,2), (c,3), (c,5), (c,6), (c,7), (f,1), (f,2), (f,3), (f,5), (f,6), (f,7), (h,1), (h,2), (h,3), (h,5), (h,6), (h,7), (i,1), (i,2), (i,3), (i,5), (i,6), (i,7)}. From this list, a representative group for the candidate event is constructed by grouping together pairs that are present in the N-best list. Note that the yellow clusters from both dendrograms are not considered to be part of the representative group, because they may belong to another event.

Fig. 3. Algorithm for extracting groups of consistent audio-visual clusters

```
Data: A ranked list of n pairs of clusters: nbestList; Two dendrograms corresponding to
      the two hierarchical clustering results: A, V;
Result: A list of groups containing consistent audio-visual clusters: E
k = 0;
while $not\ empty(nbestList)$ do
    $E(k) = \{\}$;
    $a, v \leftarrow getFirstAV(nbestList)$;
    $sub_a = getSubtree(A, a)$;
    $sub_v = getSubtree(V, v)$;
    for $\forall(a, b)\ |(a \in sub_a,\ b \in sub_b)$ do
        if $(a, b) \in nbestList$ then
            $E(k) = E(k) \cup (a, b)$;
            remove(nbestList,(a,b));
        end
    end
    $k = k + 1$;
end
```

this event. This process is applied for each of the remaining pairs in the N-best list. The result is a list of group containing consistent audio-visual clusters, in which each group represents a structural event. The overall algorithm is presented in Figure 3, where $getSubtree(\cdot, \cdot)$ is a function that returns a subtree from a given node (cluster); $remove(\cdot, \cdot)$ function that removes a pair of clusters from a list; and $getFirstAV(\cdot)$ function returns the first element (ie., the pair of audio-visual clusters having highest MI value) from a list. Please note that we define a subtree for a given node as a list of nodes created from all paths that pass through it from a leaf to the root of the tree. An illustration example is shown in figure 2a, where the subtree for the red node is a list of nodes including all green nodes and the red node.

In our context, a consistent audio-visual cluster belongs to only one structural event. In other words, there are no structural events that share the same cluster pairs. Thus, after having constructed the list of groups, we remove for each group from this list, all common elements of any two groups. Consequently, the obtained list contains only non-intersecting groups, each of which represents a structural event. Note that although the algorithm presented in figure 3 returns all possible structural events from an input video, quantitative evaluations for multiple structural events require the knowledge of a human expert, and therefore out of the scope of this paper. In the next sub-section, we present a new method to model an event from a given group of audio-visual clusters.

3.3 Event Modeling and Recognition

After the event mining step, a candidate event is characterized by a group of audio and video clusters $E = \{e_1, e_2, ..., e_m\}$, where e_i represents a pair of audio and video clusters (C^A, C^V) with the corresponding temporal segments (S^A, S^V). Note that the audio/video segmentation and clustering steps are performed independently for each

Fig. 4. Feature selection for SVM

```
Data: A group of clusters E; thresholds $T_p$,$T_n$; set of audiovisual segments SEG
Result: Set of input vectors (including positive samples and the negative ones) for SVM:
        $v_p$, $v_n$
forall $s \in SEG$ do
    SN(s) ← 0 ;
    SP(s) ← 0 ;
end
foreach $e \in E$ do
    negVote(e, SN);
    posVote(e, SP);
end
$v_p \leftarrow (\boldsymbol{AV_{s_i}} \mid s_i \in SEG \; and \; SP(s_i) > T_p)$;
$doubtSet \leftarrow (s_i \in SEG \mid (0 < SP(s_i) \leq T_p \; and \; SN(s_i) > 0))$;
$v_n \leftarrow (\boldsymbol{AV_{s_i}} \mid s_i \in SEG \; and \; (SN(s_i) > T_n \; and \; s_i \notin doubtSet))$;
```

modality. Thus S^A and S^V may be different from each other, and we need to combine them for building representative segments (i.e., audiovisual segments) of the pair in consideration e_i. For this end, an audiovisual segment of the video is constructed by merging the boundaries of an audio and a visual segments, and the corresponding feature vector (audiovisual feature vector) is the concatenation of the two component feature vectors. Recall that our goal here is to build a model based on audiovisual feature vectors, which can be used to predict a structural event from the entire video. Our early experiments [1] show that taking audiovisual segments belonging to the intersections of audio and video segments as positive examples[2], and those corresponding to neither their intersections nor their unions as negative examples, gives good results. Particularly, for the above pair e_i characterized by the two clusters (C^A, C^V), two sets of positive and negative training samples are determined as follows:

$$\boldsymbol{AV_{s_k}} \in +\mathbb{1} \text{ if } s_k \subset S^A \cap S^V$$
$$\boldsymbol{AV_{s_k}} \in -\mathbb{1} \text{ if } s_k \not\subset S^A \cup S^V$$

where s_k is an audiovisual segment, and $\boldsymbol{AV_{s_k}}$ is its corresponding audiovisual feature vector.

Now we extend this technique for the selection of training samples for the group of clusters E. The idea is as follows: for each element e_i (a pair of clusters) in E, we cast its votes for negative and positive samples (i.e., each audiovisual segment from e_i casts vote for positive or negative), and the voting results are accumulated for all elements in the group. This voting algorithm is described in figure 4, where $negVote(\cdot,\cdot)$ is a function that casts votes for negative examples for the audiovisual segments from a given pair of clusters. Similarly, $posVote(\cdot,\cdot)$ casts votes for positive examples. Note that, SN and SP are used to keep voting results accumulated over all elements in the input group. T_p, T_n are thresholds, which will be set experimentally, used to filter out the

[2] Note that, in our case we use a binary classifier to distinguish between an candidate event, i.e., positive class (+1), and non-candidate event, i.e., negative class (-1).

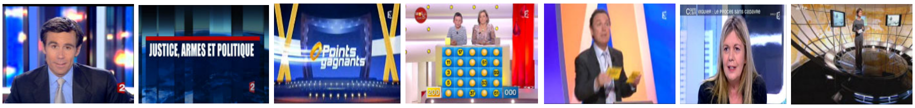


Fig. 5. Illustration of typical structural events in our dataset. From left to right: anchor person in news; a separator screen in flash news; a separator, two participants, and a presenter in games; a guest in a talk show; and magazine anchor person.

outliers. And $doubtSet$ is a set of audiovisual segments, where we do not have enough information to choose between negative and positive samples. Thus these segments should not belong to both negative and positive sets. Once the positive and negative samples are selected, discriminative classifiers can be used to train a binary model. In our work, we use a binary SVM with 5-fold cross-validation procedure on the training set to optimize Hyper-parameters. We then apply the trained model to all audiovisual segments in the input video. These segments are classified as corresponding to either the event under consideration or not.

4 Experimental Results

Discovering structural events based on audiovisual consistency should be evaluated on datasets, in which such events are present. Unfortunately, up to our knowledge no standard database is available for such difficult tasks. To evaluate our method, we captured TV programs with different genres from various French television channels (see fig. 5 for several examples of structural events), including: flash news, news, magazines, investigation reports, talk shows, and games. These videos are then annotated by a human expert. There are 18 video sequences (over 16 hours), in which the longest annotated event has about 120 occurrences and the shortest one has 9 occurrences. We used the classical recall, precision, and F-measure to evaluate the performance of our method. Note that, given the occurrences of an discovered event and its corresponding annotated event, recall is defined as the amount of time of correctly detected occurrences with respect to the total amount of time of all the occurrences from the corresponding annotated event in the ground truth, whereas precision is the amount of time of correctly detected occurrences with respect to the total amount of time of the detected occurrences. We performed two different experiments: the first was designed to evaluate the performance of the proposed measure for cluster selection. The second experiment aimed to evaluate the effectiveness of the proposed event modeling method. These two experiments are described below.

Evaluation of the Proposed Measure. In this evaluation, we compare the performance of the proposed measure for cluster selection with the baseline measure presented in [1]. For more convenience, we denote our measure by MI2 (i.e., it has two possible correlations, cf. eq. 1), and the method in [1] by MI4. The testing protocol is as follows: for each video, both MI2 and MI4 are first applied to establish two corresponding N-best lists (ranked lists). Then SVM models are trained on samples extracted from the first element for each list. Note that ranking results in the N-best list provided by these two

Table 1. Comparison of the performance between MI4 and MI2

Genre	MI4			MI2			#videos
	R	P	F1	R	P	F1	
Flash News	0.78	0.88	**0.83**	0.78	0.88	**0.83**	4/4
News	0.81	0.85	0.82	0.83	0.85	**0.84**	2/2
Magazine	0.91	0.97	**0.93**	0.91	0.97	**0.93**	2/5
Investigation	0.24	0.75	0.35	0.33	0.69	**0.44**	3/4
Games	0.71	0.71	0.71	0.79	0.72	**0.75**	2/3

Table 2. Comparison of the performance for MI4 with and without applying the proposed event modeling technique

Genre	Single pair			Group		
	R	P	F1	R	P	F1
Flash News	0.78	0.88	0.83	0.88	0.82	**0.85**
News	0.81	0.85	0.82	0.82	0.84	**0.83**
Magazines	0.78	0.87	0.82	0.79	0.87	**0.83**
Investigation	0.35	0.75	0.44	0.52	0.76	**0.57**
Talk shows	0.50	0.93	0.65	0.68	0.90	**0.72**
Games	0.63	0.81	0.68	0.75	0.76	**0.75**

measures are not always in the same order. Therefore, for a fair comparison, for each genre of programs, we selected to report only results from the videos, for which both methods give exactly the same structural event corresponding to the first element in the N-best list. In the case that these two methods return the same event with different orders, we will discuss qualitative results later. Table 1 presents a comparison of these methods, where the last column indicates the number of videos for which the two methods detect the same event (in terms of the first element in the two N-best lists) with respect to the total number of videos tested. For flash news, these two methods give exactly the same pair of audio and visual clusters, yielding the same results from SVM classification. This is due to the fact that such kinds of videos contain only one structural event and have little variations. From this table, we can observe that for the more challenging videos (eg., investigation videos), MI2 gives much better results. It should be noted that, except for flash news and magazines, MI2 always returns a pair comprising either the same audio cluster as MI4 and associated with a higher (i.e., higher level in the dendrograms) video cluster, or vice-versa. This indicates that MI2 is more robust to variability. Please note that, we cannot directly compare our results with those from the previous method [1], because the experimental set-ups are different, and we used more challenging videos for the tests in this work.

Table 3. Comparison of the performance for MI2 with and without applying the proposed event modeling technique

Genre	Single pair			Group		
	R	P	F1	R	P	F1
Flash News	0.78	0.88	0.83	0.84	0.88	**0.86**
News	0.83	0.85	**0.84**	0.85	0.83	0.83
Magazine	0.65	0.77	**0.71**	0.67	0.76	**0.71**
Investigation	0.45	0.65	0.49	0.61	0.66	**0.61**
Talk shows	0.65	0.85	**0.74**	0.67	0.80	0.73
Games	0.82	0.81	0.81	0.90	0.79	**0.83**

Evaluation of Event Mining and Modeling Techniques. In this experiment, we test the performance of the proposed event modeling method, i.e., evaluating the voting mechanism based on a representative group for a candidate event. To this end, we compare the performance for both MI2 and MI4 with and without using group (i.e., with or without using the proposed event mining and modeling techniques). For the case of without using group, the experimental set-up is the same as in experiment 1 above. For the case of using group, the testing protocol is given as follows: given the N-best list (obtained by MI2 or MI4), we apply the algorithm presented in Fig. 3 to obtain a list of representative groups of candidate events. The first group is then extracted from this list (i.e.., the group corresponding to the first element in the N-best list). For each group of audiovisual cluster pairs, we keep only 5 elements that have highest MI values. In all our experiments, thresholds[3] T_p is set to 1, and T_n is 4 for MI4, and T_p is set to 2, and T_n is 4 for MI2 (i.e., T_p and T_n are respectively set to be of 40% and 80% with respect to the group size). Finally, the algorithm presented in Fig. 4 is used to select training samples and SVM is applied to train the event model.

Table 2 and table 3 give the average results (in terms of recall, precision, and F1-measure) by using the measure introduced in [1] (MI4) and our measure (MI2), respectively. Where, the column, namely "Single pair", shows the obtained results from the best pair of clusters; and the column "group" shows the obtained results applying the proposed event modeling technique (i.e., the voting mechanism). From these tables, it can be easily seen that applying the proposed event modeling method (using a group of pairs) to represent events significantly outperforms the case of using only one pair of clusters (the most consistent pair in the N-best list). Taking the investigation videos for example, the performance of using the event modeling technique is increased by more than 10% with respect to that of using only one single pair. The results in terms of the average of F1-measure for MI2 using the voting method moderately decrease by roughly 1% for talkshows and news, however the corresponding recalls are higher. To evaluate the stability of our voting method, we also performed the tests for MI2 by varying the size of the group and setting T_p to 40%, and T_n to 80% of the group size, respectively, the performance changes by less than 1.5%.

[3] The choice of the optimal thresholds T_p and T_n is beyond the scope of this paper.

Qualitative Analysis. Although quantitative analysis for multiple structural events is beyond the scope of this paper, we still performed the tests for this task, and observed qualitative trends of both MI4 and MI2 with and without applying the proposed event modeling technique. This allows us to point out some particular points: (a) using the proposed voting method gives much better results in general; (b) MI4 seems to be suited for sparse events, and events having little variations, and provides quite limited potential for discovery of multiple events; (c) if both measures detect the same events but with different orders (the detected order from the N-best list), MI2 often returns a more complete event.

5 Conclusion and Discussion

In this paper, we have presented an improvement on cluster selection and event modeling in unsupervised mining for automatic audiovisual video structuring. The experimental results, using different genres of videos, demonstrate that the proposed method gives significant improvement compared to our previous work [1]. Our current work can be extended in several ways: first, we plan to evaluate multiple structural events, and to automatically determine the thresholds used for selecting training features. Second, we will explore how to automatically select n best pairs in the N-best list, and the relevant events among the different candidate groups. Finally, other features would be useful for the discovery of events in specific domains, eg., optical flows could be interesting for event discovery in sport videos.

Acknowledgments. This work was partly funded by OSEO, French State agency for innovation, in the framework of the Quaero research program. Many thanks to Monica Corlay for doing the ground truth annotation work, and to Sébastien Campion for providing Python code for key frame clustering, as part of the PimPy library[4].

References

1. Ben, M., Gravier, G.: Unsupervised mining of audiovisually consistent segments in videos with application to structure analysis. In: IEEE International Conference on Multimedia and Exhibition ICME 2011, Barcelona, Spain (July 2011)
2. Naphade, M., Li, C., Huang, T.: Discovering Recurrent Events in Multichannel Data Streams Using Unsupervised Methods. In: Data Mining: Next Generation Challenges and Future Directions. AAAI Press (2004)
3. Hauptmann, A., Baron, R.V., Chen, M.Y., Christel, M., Duygulu, P., Huang, C., Jin, R., Lin, W.H., Ng, T., Moraveji, N., Snoek, C.G.M., Tzanetakis, G., Yang, J., Yan, R., Wactlar, H.D.: Analyzing and searching broadcast news video. In: Proc. of TRECVID (2003)
4. Tat-Seng, C., Shih-Fu, C., Lekha, C., Winston, H.: Story boundary detection in large broadcast news video archives: techniques, experience and trends. In: Proceedings of the 12th ACM International Conference on Multimedia (2004)
5. Clarkson, B., Pentland, A.: Unsupervised clustering of ambulatory audio and video. In: IEEE International Conference on Proceedings of the Acoustics, Speech, and Signal Processing, vol. 6, pp. 3037–3040 (1999)

[4] http://pim.gforge.inria.fr/pimpy/

6. Xie, L., Chang, S., Divakaran, A., Sun, H.: Unsupervised Mining of Statistical Temporal Structures. In: Rosenfeld, A., et al. (eds.) Video Mining, ch.10. Kluwer Academic Publishers (2003)
7. Petkovic, M., Mihajlovic, V., Jonker, W., Djordjevic-Kajan, S.: Multi-Modal Extraction of Highlights from TV Formula 1 Programs. In: Proceedings of the IEEE International Conference on Multimedia and Expo, ICME (2002)
8. Wang, F., Ma, Y.-F., Zhang, H.-J., Li, J.-T.: A Generic Framework for Semantic Sports Video Analysis Using Dynamic Bayesian Networks. In: International MultiMedia Modeling Conference, pp. 115–122 (2005)
9. Covell, M., Baluja, S., Fink, M.: Detecting Ads in Video Streams Using Acoustic and Visual Cues. IEEE Computer Magazine 19(12) (2006)
10. Herley, C.: ARGOS: automatically extracting repeating objects from multimedia streams. IEEE Transactions on Multimedia 8(1) (2006)
11. Jacobs, A.: Using Self-similarity Matrices for Structure Mining on News Video. In: Antoniou, G., Potamias, G., Spyropoulos, C., Plexousakis, D. (eds.) SETN 2006. LNCS (LNAI), vol. 3955, pp. 87–94. Springer, Heidelberg (2006)
12. Yang, X.-F., Tian, Q., Xue, P.: Efficient Short Video Repeat Identification With Application to News Video Structure Analysis. IEEE Transactions on Multimedia 9(3), 600–609 (2007)

Pedestrian Attribute Analysis Using a Top-View Camera in a Public Space

Toshihiko Yamasaki[1,2,3] and Tomoaki Matsunami[1]

[1] Department of Information and Communication Engineering, The University of Tokyo,
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
[2] School of Electrical and Computer Engineering, Cornell University,
116 Ward Hall, Ithaca, NY 14853, USA
[3] JSPS Postdoctoral Fellow for Research Abroad
{yamasaki,matsunami}@hal.t.u-tokyo.ac.jp

Abstract. In this paper, we propose a method to analyze gender of the pedestrian and whether he or she has a baggage or not in a public space. The challenging part of this work is we only use top-view camera images to protect the pedestrians' privacy. We focused on temporal changes in their position, shape, and contours over the frames because their appearances do not provide much information. We extracted the pedestrians' features using their position, area, aspect ratio, histogram of oriented gradients (HoG), and Fourier descriptors. The temporal information was taken into consideration by employing Gaussian mixture models (GMM), GMM universal background model (GMM-UBM), and bag of features (BoF) model. The attributes were classified by using support vector machines (SVM). We conducted experiments using 60-minute video captured by a top-view camera attached at an airport. Experimental results show that the classification accuracy is 69% for the gender classification and 79% for baggage possession classification.

Keywords: Human attributes, surveillance, gender classification, bag possession classification.

1 Introduction

Visual surveillance has been one of the most active research areas in computer vision [1][2]. Surveillance cameras have been installed in a lot of places in such as stations, airports, or on the streets for security purposes. Visual surveillance data are easy to analyze for humans. On the other hand, analyzing the data by computers requires a wide range of algorithms such as moving object detection, object classification, counting, tracking, behavior labeling, human identification, abnormal object/event detection, flux analysis, data fusion collected from multiple cameras, and so on.

Understanding human attribute and behavior, in particular, is getting more attention not only for security reasons but for better services, marketing, and so on. If surveillance systems can recognize gender and age range of the passengers, digital

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 541–550, 2012.

singnage dedicatedly designed for a particular target can be displayed. If systems detect children who are alone, they might be lost and looking for their parents. In addition, systems can alert person who is carrying a large suitcases widely spread behind him/her, which is dangerous and is becoming a significant safety issue in crowded airports and stations. For activity recognition, Chen and Hauptmann proposed MoSIFT [3]. MoSIFT was an extension of the Scale Invariant Feature Transform (SIFT) [4] features to the temporal domain and showed its superiority to Histogram of Oriented Gradients (HoG) [5] and Histogram of oriented Optical Flow (HoF) [6] based approaches. Vezzani et al. proposed projection histogram features and used Hidden Markov Models (HMM) to classify the human activities [7]. These activity recognition algorithms focused on general activity classification such as walking, running, jumping, and so forth. Ozturk et al. [8] proposed body and head orientation detection to analyze what pedestrians are looking at in a public space. They used omega-like-shape detection for head pose estimation and SIFT-based feature tracking for temporal analysis.

In this paper, we analyze gender of pedestrians in an airport and to judge whether they have bags by using only top-view images. A lot of work on bag detection [9], gender classification [10], and face attribute analysis [11][12] can be found in literatures. Tao et al. proposed general tensor discriminant analysis using Gabor filter based gait analysis to analyze human carrying status [9]. Zhang et al. analyzed the optimal camera angle for the gender classification using SMV classifiers [10], in which only yaw angles were considered. In these approaches, however, the quality of the images was well-controlled: target objects were large enough, taken from the frontal-view, and so on. On the other hand, only top-view images taken by a surveillance camera is used in this work, which could protect the pedestrians' privacy. Another challenging point is the data used in our experiment were "real-life" data taken at an airport, not simulated data.

In our proposed approach, we detect pedestrians and track them using conventional background subtraction and blob tracking. Then, position, area, contour, shape, and their changes are extracted. In order to analyze the shape feature of the contour images, three kinds of Fourier descriptors [13] and HoG are employed. Features over the multiple frames are considered by using GMM [14], GMM-UBM [15], and BoF [16]. The validity of the proposed algorithm was evaluated using 60 minutes' real-life video taken at Haneda airport, in which 788 pedestrians walked through the view area. The experimental results have demonstrated that the performance for the gender classification was 69% when the change in the position, area, and aspect ratio is directly considered. On the other hand, the P-type Fourier descriptor with GMM-UBM yielded the best score of 79% for baggage possession classification followed by the P-type descriptor with BoF a little behind.

The organization of this paper is as follows. Section 2 describes pedestrian detection and tracking. Feature extraction from each frame and feature vector generation over the frames are explained in Section 3 and Section 4, respectively. Experimental results are demonstrated in Section 5. Concluding remarks are given in Section 6.

2 Detection and Tracking

Since multiple pedestrians could be observed in a frame, detection and tracking of them is mandatory. A conventional approach was employed because detection and tracking themselves are out of scope of this paper. A background image is generated for every frame by averaging the previous 60 frames. After simple background subtraction, graph-cuts [17] based segmentation is applied to extract the pedestrians' silhouettes. The graph-cuts based segmentation is important because some pedestrians wear a white or cream color shirt, whose color is very close to that of the floor and simple background subtraction cannot detect the silhouette properly. The flowchart and some results are shown in Fig. 1.

Since the pedestrians' paths in the view area are rather simple, a simple blob tracking algorithm is employed to save computational time. Once blobs representing pedestrians are detected, blob matching is done between neighboring frames searching for the nearest blob in terms of the position and the size. If there is no correspondence in the previous frame, the blob is detected as a new pedestrian and the same procedure is applied to the pedestrians who are getting out of the view area. In this paper, erroneously detected or tracked blobs were eliminated in advance. And only the pedestrians who existed in the area for more than 20 frames are analyzed because the temporal change is also considered. Some examples of our blob tracking results are demonstrated in Fig. 2.

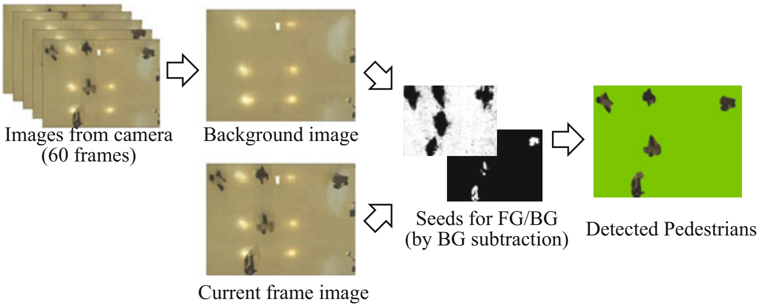


Fig. 1. Pedestrian detection using background subtraction and graph-cuts

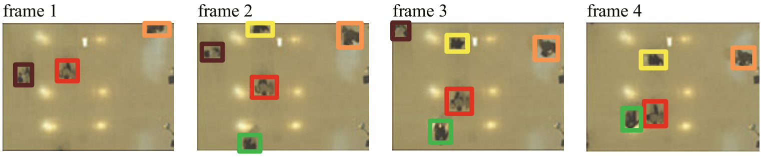


Fig. 2. Blob tracking by comparing position and size

3 Feature Extraction from Each Frame

Analyzing human attributes using only top-view images is a challenging task because no face and no details are recorded while it protects the pedestrians' privacy. The underlying assumption is that their silhouette and how they walk would differ depending on their gender and their belongings.

3.1 Position and Area Based Features

The change in pedestrians' position is calculated as follows. We assume that they walk straight in a short distance so the direction of the pedestrian is estimated by the least square linear fitting using five previous positions of the pedestrian. Then, the shift from the previous position in the perpendicular and parallel directions to the estimated moving direction is defined as dx(t) and dy(t), where t is the frame ID. In addition, area, aspect ratio, and their changes from the previous frame are also used.

3.2 Shape Based Features

The shape feature of the detected pedestrians is analyzed by three kinds of Fourier descriptors: G-type, P-type, and Z-type. Approximating the contour by a closed loop of lines is common to all of them but to what components the Fourier transform is applied is different. In our case, the contour is represented by 100 lines. In the G-type Fourier descriptor, the Fourier transform is applied to the vertex position in a form of $z(i) = x(i) + jy(i)$, $(i = 0, \ldots ,99)$ directly:

$$e(k) = \frac{1}{n}\sum_{i=0}^{n-1} z(i)\exp\left(-2\pi j\frac{ik}{n}\right), (\mathrm{k} = 0,\ldots,99)\,. \tag{1}$$

Here, $e(k)$ is the G-type descriptor and i is the ID for the lines. The P-type descriptor is obtained applying the Fourier transform to the length-normalized vector from $z(i)$ to $z(i+1)$ as shown below:

$$c(k) = \frac{1}{n}\sum_{i=0}^{n-1} w(i)\exp\left(-2\pi j\frac{ik}{n}\right), (\mathrm{k} = 0,\ldots,99) \tag{2}$$

where $w(i) = (z(i+1)-z(i))/|z(i+1)-z(i)|$.

The Z-type is the Fourier coefficients of the angles of the lines instead of using the vectors as in the P-type descriptor.

$$d(k) = \frac{1}{n}\sum_{i=0}^{n-1} \varphi(i)\exp\left(-2\pi j\frac{ik}{n}\right), (\mathrm{k} = 0,\ldots,99) \tag{3}$$

where $\varphi(i) = \theta(i)-\theta(0)-2\pi l/L$.

Table 1. Summary of pedestrians' attributes

	With bag	W/o bag	Total
Male	272	187	459
Female	179	150	329
Total	451	337	788

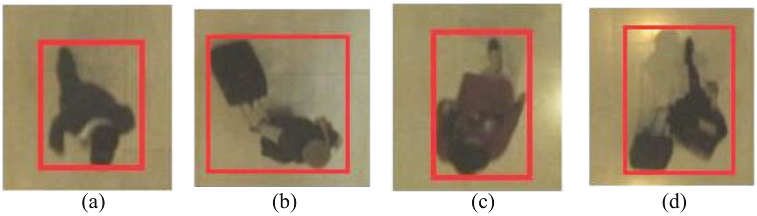


Fig. 3. Examples of pedestrians: (a) male w/o bag, (b) male w/ bag, (c) female w/o bag, (d) female w/ bag

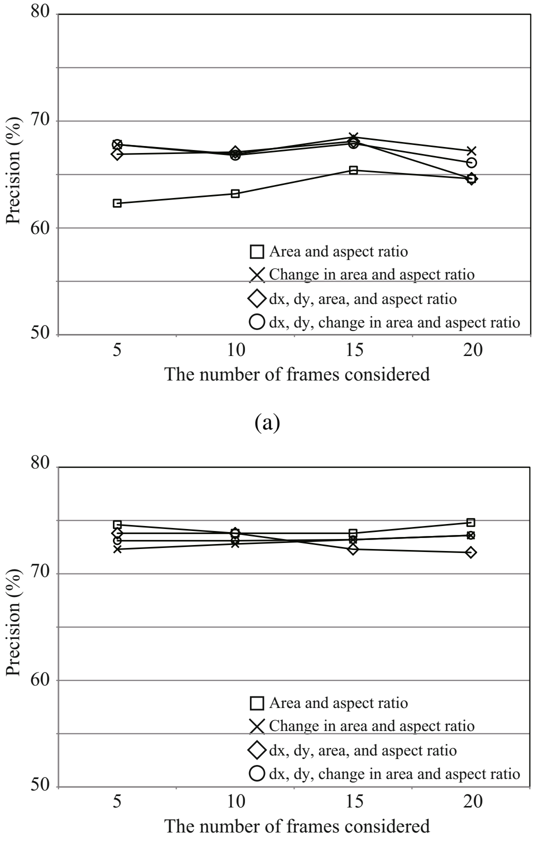
Here, $\theta(i)$ is the angle of the ith line to the x-axis, l is the length of the lines accumulated from the 0th to the ith, and L is the total length of all the lines. The lengths of the lines need be stored for the inverse transform in the P-type and the Z-type descriptors.

Only a limited number of Fourier coefficients from lower frequency components, which describes rough shape of the object, are used for the classification. The performance dependency on the number of coefficients will be discussed in Section 5. HoG-based feature vectors are generated as in the original paper [5].

4 Feature Extraction over the Frames

For the position and area related feature, the dimension of the feature description is small. Therefore, they are simply concatenated chronologically. The other features such as Fourier descriptors and HoG descriptors tend to have higher dimension. Therefore, the feature vector distribution over the frames is transformed into a single vector by applying GMM, GMM-UBM, and BoF.

In the GMM approach, a mixture of Gaussians in the feature space is estimated using the expectation-maximization (EM) algorithm and the mean vectors of the estimated Gaussians are concatenated to form a feature vector. In a simple GMM approach, GMM is generated independent of other pedestrians' set of vectors. Therefore, the order of concatenating multiple mean vectors is not consistent among the pedestrians. In the GMM-UBM approach, however, the seed vectors for the EM process are generated by putting all the feature vectors of all the pedestrians first and then the seeds are used in generating a GMM for each pedestrian. Therefore, generated feature vectors are expected to be more robust than those generated by a simple GMM approach. The BoF vectors are generated by clustering all the feature vectors of all the pedestrians and generating a frequency histogram of the cluster IDs for each pedestrian. The BoF vectors are normalized by the number of frames.



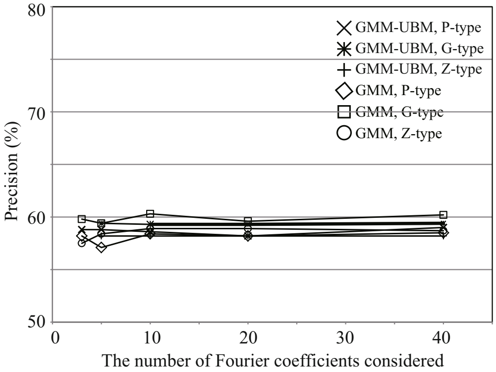
(a)

(b)

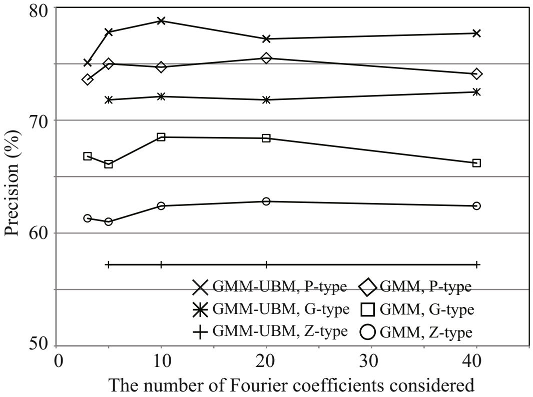
Fig. 4. Classification performance using position, area, and aspect ratio: (a) gender, (b) with/without bag

5 Experimental Results

The data were captured in Haneda airport, Japan, which is one of the top-5 busiest airports in the world. A top-view camera was attached at 12m height. The view area was about 6m x 4.5m. The images were captured at 720 x 540 resolutions and at 6.25 frames per second because the system was originally designed to record images for a long period. Typical pedestrians' sizes are about 100 x 100. Note here is that all the pedestrians were actual travelers; there were no "simulated (or pretended)" pedestrians. Only the pedestrians who were detected for more than 20 frames were used. Erroneously detected/tracked blobs such as those including two or more pedestrians in them were eliminated by hand. Such miss detection and tracking and occlusion/overlap problems are still difficult problems [18] and therefore they are out of focus of this paper. This paper concentrates only on human attribute analysis



(a)



(b)

Fig. 5. Classification performance using Fourier descriptors with GMM and GMM-UBM: (a) gender, (b) with/without bag

assuming that such pre-processing is done perfectly. Ground truth was annotated by the authors. The number of detected pedestrians and their attributes after the pre-processing are summarized in Table 1 and some sample images are shown in Fig. 3. The extracted feature vectors were classified using SVM with the Gaussian kernels optimized for each case. The accuracy was calculated by the 10-cross validation.

Figure 4 shows the classification performance using position, area, aspect ratio and their temporal changes. The changes between frames are more significant for gender classification. The best performance is obtained when the changes in the area and aspect ratio over 15 frames are used and its accuracy is 69%. On the other hand, raw data of area and aspect ratio are better than the others for the with/without bag classification. It can be observed that the number of frames is not so important except for only a few exceptions. Also, it is interesting to see that the pedestrian's gender and bag possession status affects how they walk to some extent and it can be observed in such simple features.

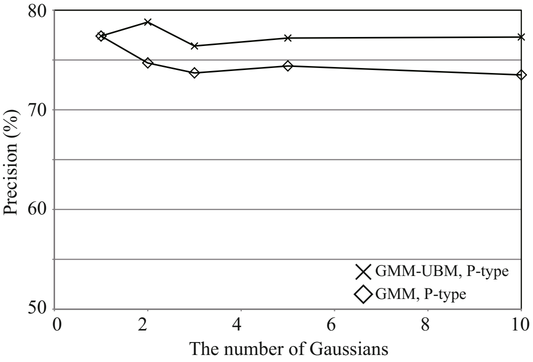


Fig. 6. Performance of with/without bag classification using GMM and GMM-UBM based features as a function of the number of Gaussians

The classification performance using Fourier descriptors with GMM or GMM-UBM is demonstrated in Fig. 5. The number of Fourier coefficients is altered in the x-axis. The number of the Gaussians is set as two. We can see that the gender classification accuracy is around 58-60% for all the cases. Besides, the gender classification accuracy was always within the range of 58-60% for all the experiments hereafter. Therefore, graphs are not shown to save the limited space. For with/without bag classification, the P-type Fourier descriptor performs the best and GMM-UBM yields better results than simple GMM. This tendency coincides with [19], which compared the P-type and Z-type Fourier descriptors and Zernike moments in the context of motion retrieval. The best performance of 79% is obtained when the number of coefficients is 10 for the P-type descriptor with GMM-UBM. 5-20 coefficients out of 100 are enough for the classification, showing that the other coefficients do not contribute to shape description and can be regarded as noise.

The classification performance as a function of the number of Gaussians is shown in Fig. 6. The GMM model performs the best when the number of Gaussians is only one. On the other hand, for the GMM-UBM model, the performance gets the maximum when the number of Gaussian is two. The computational cost and the memory usage for the feature vector storage are almost the same for GMM and GMM-UBM. GMM-UBM is better from the view point of the classification performance.

The BoF representation works well with the P-type Fourier descriptor as shown in Fig. 7. The accuracy is the best (78%) when the number of clusters is set at 200. Since the number of frames for each pedestrian is only 20-40 frames, the generated BoF vector is very sparse. On the other hand, BoF using HoG features performs with less than 75% of accuracy. In addition, gender classification accuracy using HoG was 58% for GMM and 59% for GMM-UBM and that for baggage possession classification was 71% for GMM and 69% for GMM-UBM.

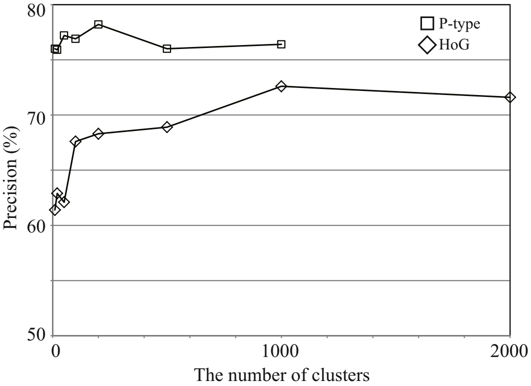


Fig. 7. Performance of BoF-based with/without bag classification using p-type Fourier descriptor and HoG

6 Conclusions

In this paper, we have presented the algorithms to analyze the pedestrians' attributes such as gender and whether they have bags or not using top-view images in the airport. After the pedestrian detection and blob tracking, the features for each frame were extracted such as temporal change in position and area, shape feature using HoG and Fourier descriptors. Then GMM, GMM-UBM, and BoF were applied to the feature vectors over the frames to form final feature vectors for the classification. The experiments using 60 minutes' video demonstrated that the gender could be classified with 69% of accuracy. And the accuracy for the with/without bag classification was 79%. It has been shown that simple features such as temporal change in position and area performs well for the gender classification and the P-type Fourier descriptor with either GMM-UMB or BoF is suitable to judge the pedestrian possesses a bag or not. Performance improvement up to 96% is expected by employing a multi-stage classifier framework along with a HoG-based BoF model [20].

The future direction of this work is analyzing more attributes such as age range and group/family detection, which would be moving synchronously, among multiple blobs.

Acknowledgments. The authors would like to thank Prof. K. Aizawa of the University of Tokyo for valuable discussions and comments.

References

1. Hu, W., Tan, T., Wang, L., Mayban, S.: A survey on visual surveillance of object motion and behaviors. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 34(3), 334–352 (2004)
2. Candamo, J., Shreve, M., Goldgof, D.B., Sapper, D.B., Kasturi, R.: Understanding transit scenes: a survey on human behavior-recognition algorithms. IEEE Transactions on Intelligent Transportation Systems 11(1), 206–224 (2010)

3. Chen, M.Y., Hauptmann, A.: MoSIFT: recognizing human actions in surveillance videos date of original version. CMU Technical Report (September 2009)
4. Lowe, D.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (IJCV) 60(2), 91–110 (2004)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. IEEE CVPR, pp. 886–893 (2005)
6. Chaudhry, R., Ravichandran, A., Hager, G., Vidal, R.: Histograms of oriented optical flow and Binet-Cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In: Proc. IEEE CVPR, pp. 1932–1939 (2009)
7. Vezzani, R., Baltieri, D., Cucchiara, R.: HMM Based Action Recognition with Projection Histogram Features. In: Ünay, D., Çataltepe, Z., Aksoy, S. (eds.) ICPR 2010. LNCS, vol. 6388, pp. 286–293. Springer, Heidelberg (2010)
8. Ozturk, O., Yamasaki, T., Aizawa, K.: Estimating Human Body and Head Orientation Change to Detect Visual Attention Direction. In: Koch, R., Huang, F. (eds.) ACCV Workshops 2010, Part I. LNCS, vol. 6468, pp. 410–419. Springer, Heidelberg (2011)
9. Tao, D., Li, X., Maybank, S.J., Wu, X.: Human Carrying Status in Visual Surveillance. In: IEEE CVPR, vol. 2, pp. 1670–1677 (2006)
10. Zhang, D., Wang, Y.: Investigating the separability of features from different views for gait based gender classification. In: Proc. ICPR, pp. 1–4 (2008)
11. Kumar, N., Berg, A.C., Belhumeur, P.N., Nayar, S.K.: Attribute and simile classifiers for face verification. In: IEEE ICCV, pp. 365–372 (2009)
12. Guo, G., Mu, G.: A study of large-scale ethnicity estimation with gender and age variations. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 79–86 (2010)
13. Uesaka, Y.: Spectral analysis of form based on Fourier descriptors. In: Proc. the First International Symposium for Science on Form, pp. 405–412 (1986)
14. Goldberg, M., Shlien, S.: A clustering scheme for multispectral images. IEEE Transactions on Systems, Man and Cybernetics 8(2), 86–92 (1978)
15. Campbell, W.M., Sturim, D.E., Reynold, D.A.: Support vector machines using GMM supervectors for speaker verification. IEEE Signal Processing Letters 13(5), 308–311 (2006)
16. Sivic, J., Zisserman, A.: Video Google: a text retrieval approach to object matching in videos. In: Proc. IEEE ICCV, pp. 1470–1477 (2003)
17. Boykov, Y., Jolly, M.-P.: Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. In: Proc. IEEE ICCV, pp. I-105–I-112 (2001)
18. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. ACM Comput. Surv. 38(4) Article 13 (December 2006)
19. Kasai, D., Yamasaki, T., Aizawa, K.: Retrieval of Time-Varying Mesh and Motion Capture Data Using 2D Video Queries Based on Silhouette Shape Descriptors. In: Proc. IEEE ICME, pp. 854–857 (2009)
20. Yamasaki, T., Matsunami, T.: Human Attribute Analysis using a Top-View Camera Based on Multi-Stage Classification. In: Proc. 5th ACM/IEEE ICDSC, #61 (2011)

A Fast GPU-Based Motion Estimation Algorithm for H.264/AVC

Rafael Rodríguez-Sánchez[1], José Luis Martínez[2], Gerardo Fernández-Escribano[1], José Luis Sánchez[1], and José Manuel Claver[3]

[1] Instituto de Investigación en Informática de Albacete,
Universidad de Castilla-La Mancha, Avenida de España s/n, 02071, Albacete, Spain
{rrsanchez,gerardo,jsanchez}@dsi.uclm.es
[2] Architecture and Technology of Computing System Group,
Complutense University, Madrid, Spain
joseluis.martinez@fdi.ucm.es
[3]Departamento de Informática. Universidad de Valencia,
Avenida de Vicente Andrés Estelles, s/n, 46100 Burjassot, Valencia, Spain
jose.claver@uv.es

Abstract. H.264/AVC is the most recent predictive video compression standard to outperform other existing video coding standards by means of higher computational complexity. In recent years, heterogeneous computing has emerged as a cost-efficient solution for high-performance computing. In the literature, several algorithms have been proposed to accelerate video compression, but so far there have not been many solutions that deal with video codecs using heterogeneous systems. This paper proposes an algorithm to perform H.264/AVC inter prediction. The proposed algorithm performs the motion estimation, both with full-pixel and sub-pixel accuracy, using CUDA to assist the CPU, obtaining remarkable time reductions while maintaining rate-distortion performance.

Keywords: H.264/AVC, Heterogeneous computing, Motion Estimation, GPU.

1 Introduction

Most of the applications which have recently appeared in the multimedia community, such as digital TV, streaming, video conferencing or DVDs, require a video coding algorithm to meet their requirements and operate. The most recent video coding standard is H.264/AVC [1], which is able to outperform the video codecs of previous coding standards [2]. The compression gains are mainly related to the variable and smaller block size motion compensation, improved entropy coding, motion estimation with multiple reference frames, and smaller block transforms, among others. However, these new/improved video coding tools increase both the encoder and decoder complexity substantially.

Fortunately, heterogeneous computing has emerged as a real solution for high-performance computing [3]. There are many examples of this kind of systems, but

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 551–562, 2012.

possibly the systems composed of one or more Graphics Processing Units (GPUs) and one or more multi-core Central Processing Units (CPUs) are the most widely used. GPUs are small devices with hundred of similar processing cores organized to achieve higher performance. GPUs are highly parallel and are normally used as a coprocessor to assist the CPU for computing massive data. GPUs have a parallel architecture that focuses on executing many concurrent threads slowly, rather than executing a single thread very quickly. In order to assist programmers, the main GPU manufacturers provide them with different tools. For example, Nvidia® proposes a powerful GPU architecture called Compute Unified Device Architecture (CUDA) [4]. CUDA is basically a Single Instruction Multiple Data (SIMD) computing device.

Therefore, it is mandatory to explore new and efficient implementations of H.264/AVC video coding systems on different computing platforms in order to support these applications. At this point, this paper proposes an algorithm to perform the inter prediction carried out in H.264/AVC using CUDA to assist the CPU. The present approach performs the Motion Estimation (ME) with both full-pixel and sub-pixel accuracy over the GPU. It should be pointed out that the ME algorithm is one of the most computationally expensive tasks in an H.264/AVC encoder; it performs the same operations (Sum of Absolute Differences) over a large amount of data (over the search area). Therefore, ME fits well in the SIMD programming model. On the other hand, the ME algorithms implemented in the H.264/AVC reference software are sequential, where each MacroBlock (MB) is encoded based on its neighboring MBs. One of the major issues of our proposed algorithm is how to remove or mitigate these dependencies between MBs in order to minimize the Rate-Distortion (RD) penalties. Performance evaluation is carried out for High Definition (HD) video sequences. The results show a remarkably time reduction of up to 99% with a negligible RD penalty. Moreover, the proposed algorithm outperforms the fastest ME algorithms included in the H.264/AVC reference software as well as some of the approaches available in the literature in terms of coding efficiency and time savings.

The rest of the paper is organized as follows: Section 2 contains a brief overview of H.264/AVC and GPU programming; in Section 3 some related proposals are shown; Section 4 shows details about the approach presented in this paper; Section 5 describes the performance evaluation and, finally, conclusions are given in Section 6.

2 Technical Background

In the H.264/AVC standard [1] inter prediction is carried out by means of the process of variable block size ME, which is able to eliminate the temporal redundancy between two or more adjacent frames. This approach supports motion compensation block sizes ranging between 16x16, 16x8, 8x16 and 8x8, where each of the sub-divided regions is an MB partition. If the 8x8 mode is chosen, each of the four 8x8 block partitions within the MB may be further split in 4 ways: 8x8, 8x4, 4x8 or 4x4, which are known as sub-MB partitions. The ME is carried out for each of these partitions. Furthermore, Motion Vectors (MVs) from spatially adjacent blocks and from other MB partitions are used to initialize the search area for the current partition. These are known as Motion Vector predictors (MVp). In addition, to compute the rate term R in the Lagrangian cost the MVs of the neighboring MBs are required [5].

In terms of Multi-Core graphics processors, GPUs are small accelerator devices with hundreds of cores which are organized in several SIMD blocks. GPUs are characterized by a high level of parallelism and are usually used as a coprocessor to assist the CPU in computing massive data. For instance, the architecture of the Nvidia® GPUs consists of a set of SIMD multiprocessors called Stream Multiprocessors (SM). Each SM has up to 48 processing elements called cores and a set of resources shared by all cores, such as 32-bit registers, local shared / texture memory or caches. More detail about the Nvidia GPU architecture can be found in [4].

3 Previous Work

Most of the proposals available in the literature for accelerating the H.264/AVC encoding algorithm are sequential-based approaches, but so far there have not been many solutions which make use of Many-Core graphics hardware to accelerate this highly complex algorithm. At this point, the main objective of this paper is to combine powerful Multi-Core architectures to accelerate traditional video coding algorithms, such as H.264/AVC. In 2007, Lee et al. [6] presented a multi-pass and frame parallel algorithm to accelerate H.264/AVC ME using a GPU. They unroll and rearrange the multiple nested loops by using the multi-pass method. The multiple reference frames method is implemented at frame parallel level by the use of SIMD vector operations of the GPU. In 2008, Ryoo et al. in [7] and Chen and Hang [8] presented some optimization principles of a multithreaded GPU using CUDA. In [8] the algorithm is based on an efficient block-level parallel algorithm for the variable block size motion estimation in H.264/AVC. They decompose the H.264/AVC ME algorithm into 5 steps so that they can achieve highly parallel computation. The major failing of all these approaches is that they do not analyze the RD performance, they only show timing results; although the speedup and time reduction are acceptable, they are only valid if they keep the RD as close as possible to the sequential approach.

More recently, in 2010, Cheung et al. proposed a GPU implementation of the simplified Unsymmetrical Multi-Hexagon search (smpUMHexagonS) [9] ME algorithm, which is a fast ME technique implemented in the H.264/AVC reference software. The authors divide the current frame into multiple tiles. Each tile is processed by a single GPU thread, and different tiles are processed by different independent threads concurrently on the GPU. They report significant bitrate increases (12%) with a penalty in quality (0.4dB) depending on the sequence and the tile length.

Many-Core architectures have been also used for accelerating other modules of the H.264/AVC encoding algorithm, such as the Intra Prediction [10]. Based on a dependency analysis of intra-mode decision. they propose to encode the video blocks following the greedy order, leading to highly parallel RD cost computations. They obtain an speed of up to 80 with negligible RD penalty.

4 Proposed GPU-Based Motion Estimation Algorithm

This section describes the algorithm for implementing the inter prediction performed by the H.264/AVC encoding algorithm via a system composed of a CPU with the support of a GPU. The ME as part of the inter prediction process has been implemented in the JM v17.2 reference software [11], but this algorithm can be easily adapted to other H.264 implementation such as x264 [12]. The present approach is based on the *Full Search* (FS) ME algorithm implemented in the JM reference software. At this point, this paper proposes a modified GPU-based FS ME algorithm with quarter-pixel accuracy.

ME is tackled in two steps, full-pixel and sub-pixel accuracy ME; each one is performed following a highly-parallel procedure over the GPU. Firstly, the image to be processed (with full-pixel accuracy) is moved from CPU to GPU and it performs the full-pixel ME. Then, the GPU is able to generate the sub-pixel image from the reference image. Finally, the GPU performs the sub-pixel ME. In other words, once the frame is moved to the GPU, all the computations relating to the ME are carried out over the GPU. In this way, we avoid memory transfers between the CPU and GPU, which is the bottleneck in this kind of systems. The proposed GPU-based ME algorithm is performed concurrently for the complete image at the beginning of coding each P frame, where the inter prediction is applied. Figure 1 shows a simplified activity diagram of our parallel ME proposal.

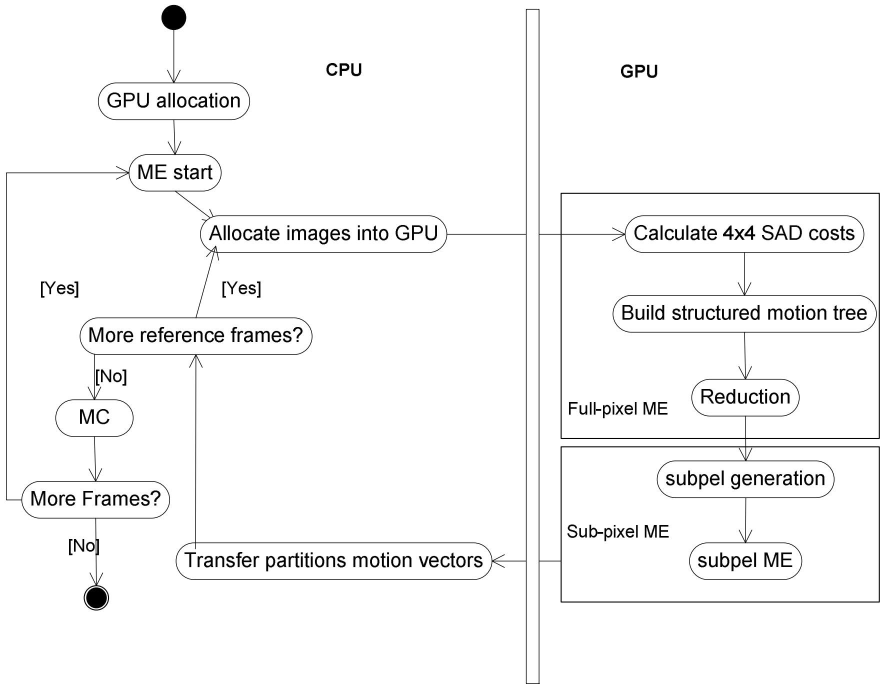


Fig. 1. Activity Diagram of Proposal

4.1 Full-Pixel Motion Estimation

The proposed Full-pixel ME process is divided into three steps. The goal of the first step is to obtain the Sum Absolute Differences (SAD) calculation between the current

MB (split into sixteen 4x4 partitions) and all MB positions in the reference frame inside the search area. Then, the second step, using the previous 4x4 block SAD calculations, is able to obtain the SAD costs for all other MB partitions. Finally, the last step reduces the SAD costs to one SAD cost for each one of the 41 MB partitions of each MB. This three-step algorithm is implemented using two GPU kernels.

In the first kernel, all threads from a thread block cooperate to copy its assigned MB and corresponding search area from texture memory to multiprocessor local shared memory. Shared memory is defined as an integer and it allows contiguous multiprocessor threads to read from contiguous memory banks without access conflicts in the memory banks. The SAD calculations are carried out in 4x4 blocks, therefore each MB is divided into sixteen 4x4 blocks for each search area position. These SAD costs are stored in registers to build the structured motion tree (*4x4 SAD sub-matrix* in Figure 2). The complete search area is computed by rows, one or more rows corresponding to a thread block, so contiguous search area positions for a certain MB are computed by the same thread block, with normally 256 positions for each thread block.

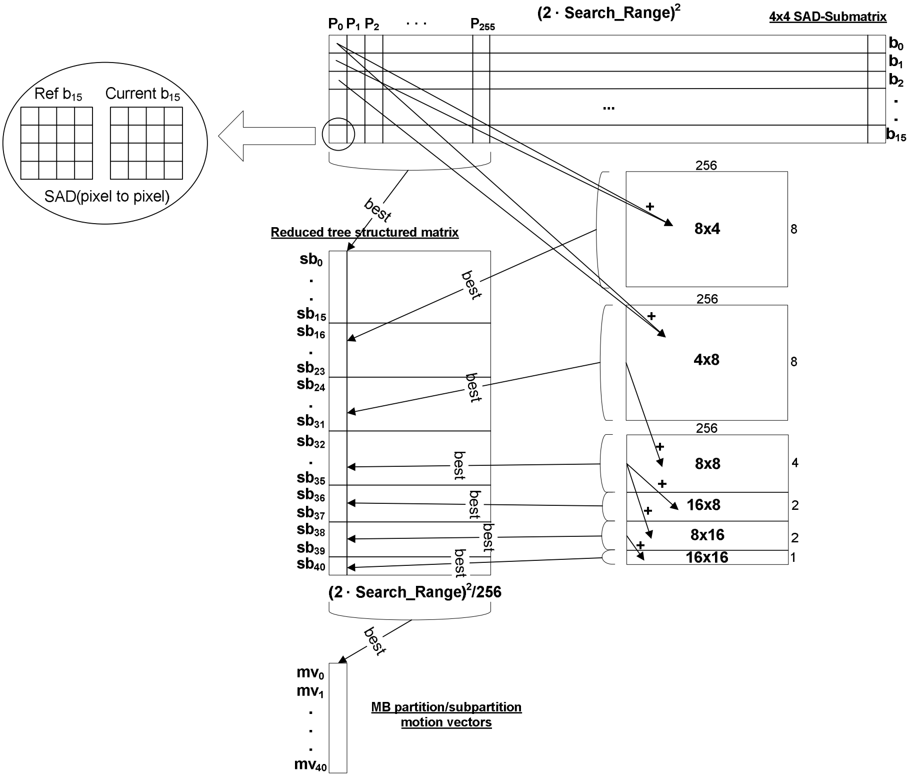


Fig. 2. Proposed Full-pixel ME Algorithm

In the same GPU kernel used to obtain the 4x4 SAD costs, the structured motion tree is obtained. Using the information previously stored in registers (4x4 SAD costs) our algorithm is able to obtain the SAD costs for higher partitions. As depicted in Figure 2, by adding two 4x4 SAD costs it is able to obtain the 4x8 and 8x4 SAD costs, by adding two 4x8s it is able to obtain the 8x8 SAD costs and so on. Intermediate results are stored in multiprocessor shared memory for faster memory accesses.

Finally, this kernel also carries out a first reduction due to the large amount of data generated. The reduction starts with 256 SAD costs per MB partition and finishes with 1 SAD cost per MB partition (*Reduced tree-structured matrix* in Figure 2). For this reduction procedure, a binary reduction has been implemented. Note that this kernel performs the reduction procedure over the 256 positions configured in a thread block; the final reduction is performed by an independent kernel. This last kernel obtains the best SAD cost for each one of the MB partitions in each MB, using the same binary reduction procedure as the previous kernel.

Our GPU-based algorithm is executed concurrently for a complete frame, but each MB coding depends on their neighbors in two ways: 1) to compute the Lagrangian cost and, 2) to locate the search area (MVp). These dependencies mean that the optimal MV may not be found, resulting in a bitrate increase and in a PSNR drop.

Therefore, the proposed algorithm also tries to mitigate the effect of MVp, which is one the biggest challenges of performing the ME process in parallel. The idea for solving these impairments consists of reusing the MV of the previous frame to adjust the MB search area. The MVp does not have a big impact on low resolution and/or low motion sequences, but the lack of MV predictors for higher resolutions (such as HD) and/or high motion sequences may result in a big impact on RD performance. After a large set of experiments, we conclude that the best way to estimate the motion is to use the MV from the higher partition (16x16) for the MB located in the same position in the reference frame, but with the constraint that the MVp cannot be higher than the search range to ensure that the MV (0,0) is inside the search area.

Furthermore, to compute the Lagrangian cost, the MVp is required. The Lagrangian cost is defined as $SAD_{cost} + \lambda * vector_{bits}$, where $vector_{bits}$ is the number of bits required to encode the $MV - MV_P$. Nevertheless, the MVp is required to obtain the final cost for all positions inside the search area, which is affected by the dependencies between neighboring MBs.

4.2 Sub-pixel Accuracy Motion Estimation

In order to further improve compression, the H.264 /AVC standard assumes that the best match can be found at a region offset from the current MB (search area) by an integer number of pixels. However, for many MBs a better match can be obtained by searching a region interpolated to sub-pixel accuracy; for this case, a new prediction pixel is created by means of an interpolation of its neighbor. H.264/AVC reference software supports quarter-pixel accuracy, which means that the image sizes are multiplied by four on each dimension or, in other words, one pixel is converted into sixteen sub-pixels. One of these sub-pixels is the pixel with full-pixel accuracy; three of them are the sub-pixels with half-pixel accuracy and the other twelve pixels are the sub-pixels with quarter-pixel accuracy.

As mentioned at the beginning of this section, the images are located in GPU DRAM with full-pixel accuracy. So, we need firstly to extend the reference images to sub-pixel accuracy. The sub-pixels with half-pixel accuracy are obtained by means of a 6-tap filter and the sub-pixels with quarter-pixel accuracy are obtained by a bilinear filter. A GPU thread per pixel is generated and it applies both filters to obtain the fifteen sub-pixels.

The sub-pixel accuracy ME is performed in two steps: the first one is the half pixel refinement and the second one is the quarter pixel refinement, both of which are performed for all partitions. The best matching obtained for full-pixel accuracy becomes the center point for half-pixel refinement, and the best matching for half-pixel refinement becomes the center for quarter-pixel refinement. The algorithm for half- and quarter-pixel refinement is the same, but applied over different data.

The algorithm for sub-pixel ME is similar to the algorithm used for full-pixel ME: we divide the MB into sixteen 4x4 blocks and each one takes as its starting point the appropriate MV, i.e., all 4x4 blocks will take the same MV to perform the 16x16 partition and the final cost will be obtained using atomic GPU operations. On the other hand, all 4x4 blocks will take different MVs to perform the 4x4 partition and no extra operations will be needed. The same reduction procedure used for full-pixel accuracy ME is used to obtain the best MV. However, there are two main aspects to take into account. First, we cannot reuse the motion information from the smallest partition to obtain the Lagrangian cost of the higher partition because each partition has a different starting point (Full-pixel MV or Half-pixel MV). We have to recalculate the 4x4 cost for each partition. Second, the metric to compute the Lagrangian cost is the Hadamard SAD instead of SAD, as configured for the baseline profile in the H.264/AVC JM 17.2 [11] reference software used.

5 Performance Evaluation

In order to show the performance of the proposed algorithm, it was implemented in the H.264/AVC JM v17.2 reference software encoder. The parameters used in the H.264/AVC encoder configuration file were those included in the baseline profile of the mentioned reference software. However, some parameters were changed in the configuration file: the number of reference frames was set to 1 in order to keep the complexity as low as possible because higher values imply excessive time consumption, but higher number of reference frames can be used; RD-Optimization was disabled for the same reason as the NumberReferenceFrames parameter; the GOP pattern was set to one I frame followed by eleven P frames (I11P); the tests were carried out with popular sequences in 720p format (1280 x 720) and 1080p format (1920 x 1080); the frame rate parameter was set to 50 because sequences were sampled at 50Hz; the parameter FramesToBeEncoded was adjusted according to the sequence, in order to encode the full sequence; the Quantization Parameter (QP) called QPISlice and QPPSlice was varied among 28, 32, 36 and 40 according to [13].

The performance evaluation of our proposal for H.264/AVC based on the JM v17.2 encoder was carried out on a system composed of an Intel® Core™ i7 @930 running at 2.80 GHz, with 6GB DDR3 memory and the GPU Nvidia GTX480. The operating system was Ubuntu 10.4 with the Nvidia GPU driver 260.19 and CUDA SDK 3.2 was used.

5.1 Metrics

In order to evaluate the time saved by the proposed algorithm with respect to the reference H.264/AVC encoder, two metrics were used: Time Reduction (TR), which is based on Equation 1, and Speedup, which is based on Equation 2.

$$TR\ (\%) = \frac{T_{JM} - T_{FI}}{T_{FI}}\ x\ 100 \qquad (1)$$

$$Speedup = \frac{T_{JM}}{T_{FI}} \qquad (2)$$

where T_{JM} denotes the coding time used by the reference software, and T_{FI} is the time taken by the algorithm proposed in this paper. The times measured by T_{JM} and T_{FI} refer to the time employed to carry out the ME. T_{FI} also includes all the computational costs for the operations needed in order to prepare the information required by our proposal.

5.2 Results

Table 1 shows the RD performance and time reduction of our proposed GPU-based ME algorithm for 720p sequences against three of the most well-known ME algorithms implemented by the reference H.264/AVC encoder (Full search (FS), Unsymmetrical Multi-Hexagon search (UMHexagonS) and simplified Unsymmetrical Multi-Hexagon search (smpUMHexagonS) [14])). The results show that the RD performance obtained by our proposed ME algorithm is very similar to the sequential-based implementations. Our GPU-based ME algorithm compared with FS ME and smpUMHexagonS ME have a PSNR drop of up to 0.1 dB for a given bitrate, and a bitrate increase of up to 3.52% for a given PSNR. On the other hand, our algorithm has a PSNR increase of up to 0.199 dB for a given bitrate and a bitrate drop of up to 7.07% for a given bitrate, compared with UMHexagonS ME.

Table 1. RD Performance and TR of the proposed GPU-Based Algorithm. 720p sequences.

Sequence	Full Search			UMHexagonS			smpUMHexagonS		
	ME TR	ΔPSNR (dB)	ΔBitrate (%)	ME TR	ΔPSNR (dB)	ΔBitrate (%)	ME TR	ΔPSNR (dB)	ΔBitrate (%)
Dolphins	98.82	-0.072	2.59	83.01	0.199	-7.07	68.60	-0.039	1.35
Mobcal	99.06	-0.037	1.15	81.64	0.031	-1.12	74.46	-0.073	2.35
Parkrun	99.28	-0.043	1.40	86.31	-0.018	0.57	80.72	-0.049	1.59
Shields	98.94	-0.043	1.38	83.50	-0.038	1.00	73.95	-0.100	3.21
Stockholm	98.96	-0.040	1.35	82.67	-0.021	0.45	74.74	-0.097	3.52
Average	***99.01***	***-0.047***	***1.57***	***83.43***	***0.031***	***-1.23***	***74.50***	***-0.072***	***2.40***

Table 2 shows the RD performance and time reduction of our proposed algorithm for 1080p sequences. The RD performance conclusions are similar to those obtained for 720p format. Our GPU-based algorithm obtains slightly worse results than the FS ME algorithm and the smpUMHexagonS ME algorithm (the bitrate increases and the

PSNR drops) and it obtains slightly better results than UMHexagonS ME (the bitrate drops and the PSNR increases). However, for both resolutions our proposal obtains considerable time reductions. The average ME time reduction comparing with FS ME is better than 99% (Speedup over 100), the average ME time reduction comparing with UMHexagonS ME is better than 81% (Speedup over 5) and finally, the average ME time reduction comparing with smpUMHexagonS ME is better than 74.5% (Speedup close to 4). At this point, the negligible RD penalty is an acceptable solution because of the very high time reductions achieved.

Table 2. RD Performance and TR of the proposed GPU-Based Algorithm. 1080p sequences.

Sequence	Full Search			UMHexagonS			smpUMHexagonS		
	ME TR	ΔPSNR (dB)	ΔBitrate (%)	ME TR	ΔPSNR (dB)	ΔBitrate (%)	ME TR	ΔPSNR (dB)	ΔBitrate (%)
Crowd	99.15	-0.118	3.86	83.14	0.035	-1.25	78.57	-0.103	3.32
Ducks	99.31	-0.037	1.20	84.19	-0.002	-0.12	81.18	-0.036	1.05
IntoTree	99.13	-0.074	3.66	84.19	0.026	-1.18	77.56	-0.113	5.00
OldTown	98.92	-0.048	2.31	80.15	-0.017	0.36	71.17	-0.102	3.78
ParkJoy	99.25	-0.077	2.33	85.10	0.017	-0.61	79.38	-0.087	2.61
Average	***99.15***	***-0.071***	***2.67***	***81.89***	***0.012***	***-0.56***	***77.57***	***-0.088***	***3.15***

Figure 3 shows the RD graphic (PSNR against bitrate) results for the reference algorithms and the proposed approach, using different 1080p sequences. In general, the PSNR against bitrate curves are quite similar to those achieved by the reference algorithm while our proposal is always much faster than the reference implementations. Due to space limitations only 1080p sequences are shown. Similar RD results are obtained for 720p sequences.

5.3 Comparison with Other Known Results

In this section, a comparative performance evaluation in terms of the RD performance and execution time is presented. We compare the results of the proposed algorithm with those shown in one of the papers available in the literature with the most promising results. In [9], the authors proposed a GPU-based implementation of the well-know smpUMHexagonS ME algorithm. They partition each frame into multiple tiles, where each tile contains one or more MBs and each tile is processed by a single GPU thread.

Table 3 shows the RD results for their algorithm as well as our RD results using the same encoding conditions. We have employed the same 720p sequences sampled at 60Hz, selecting 64 as the search range and all pictures are encoded as P-frames except the initial I-frame. The comparison is achieved when comparing our results against the reference smpUMHexagonS (implemented in JM) with their results against smpUMHexagonS too. In their implementation, they obtain more degradation as many tiles are used due to the dependencies between neighboring MBs. However, we mitigate the degradation in our approach. Our algorithm outperforms the RD performance obtained by their fastest configurations (90 or more tiles); our algorithm has lower bitrate increments and lower PSNR losses than their algorithm for all video sequences.

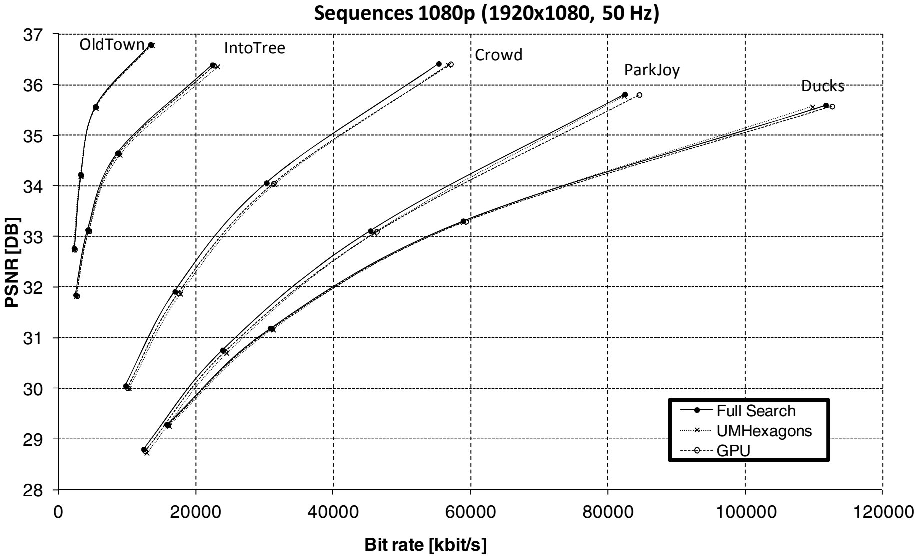


Fig. 3. RD results comparing the performance of the proposal and the H.264 reference. 1080p Sequences.

Table 3. RD comparison with *Cheung et al.* results[9]

	Sequence							
	Crew		City		Harbor		Night	
Number of Tiles	ΔBitrate (%)	ΔPSNR (dB)	ΔBitrate (%)	ΔPSNR (dB)	ΔBitrate (%)	ΔPSNR (dB)	ΔBitrate (%)	ΔPSNR (dB)
3,600	3.14	-0.082	12.93	-0.407	5.58	-0.221	4.64	-0.170
900	3.08	-0.079	11.12	-0.352	2.39	-0.094	3.55	-0.130
225	3.12	-0.080	11.17	-0.350	2.25	-0.089	3.42	-0.125
90	3.22	-0.083	10.82	-0.339	2.21	-0.087	3.40	-0.124
12	0.63	-0.016	1.41	-0.044	0.57	-0.022	1.19	-0.043
3	0.09	-0.003	0.26	-0.008	0.07	-0.003	0.16	-0.006
Our algorithm	3.08	-0.071	6.68	-0.309	0.88	-0.028	1.55	-0.047

Table 4 shows the execution time for the experiments carried out to fill the previous table. Table 4 also shows the average execution time for each configuration. Note that the peak performance for our GPU is 1350 GFlops and the peak performance for the GPU used in [9] is 345.6 GFlops, which means that our GPU is 3.9 times more powerful. For this reason and for a fair comparison, we have included the column labeled as Index in Table 4, which shows the ratio between the average execution time obtained by their implementation for a certain encoder configuration and the average execution time by our implementation using the same encoder configuration. Higher values than 3.9 for this index mean that our algorithm is faster than their algorithm. In conclusion, our algorithm is as fast as their best configuration (index of 3.85) and it outperforms the execution time for the other configurations (higher index than 3.9). The execution time using 3 and 12 tiles is not specified in [9]; however we expect a higher execution time than the other tile configuration since they use less GPU threads.

Table 4. Execution time comparison with *Cheung et al.* results[9]

	Sequence				Average GPU time (ms)	Index
	Crew	City	Harbor	Night		
Number of Tiles	GPU Time (ms)	GPU Time (ms)	GPU Time (ms)	GPU Time (ms)		
3,600	835.05	927.32	1,248.95	1,688.50	1,174.95	3.85
900	959.16	1,005.55	1,341.45	1,975.95	1,320.53	4.33
225	2,169.25	2,108.71	2,763.79	4,175.44	2,804.30	9.19
90	4,373.63	4,165.28	5,318.38	6,920.73	5,194.51	17.02
12	Unknown					
3						
Our algorithm	305.09	306.16	304.96	304.71	305.23	

6 Conclusions

This paper proposes an algorithm to perform the inter prediction carried out in H.264/AVC using a system composed of a CPU with the support of a GPU. The proposed algorithm performs the ME over a GPU by means of an efficient distribution of complexity and management of GPU resources. The algorithm includes the ME with full-pixel and sub-pixel accuracy, as well as sub-pixel interpolation. Furthermore, the proposed approach is further adapted to avoid coding dependencies between MBs, which is one of the major issues when the ME is carried out in parallel. The results show that the proposed algorithm achieves almost the same coding efficiency but outperforms all the ME algorithms available in the JM reference software in terms of time. The performance is also compared with the state-of-the-art approach available in the literate, achieving a faster execution time together with better coding efficiency.

Acknowledgements. This work was supported by the Spanish MEC and MICINN, as well as European Commission FEDER funds, under Grants CSD2006-00046 and TIN2009-14475-C04. It was also partly supported by The Council of Science and Technology of Castilla-La Mancha under Grants PEII09-0037-2328 and PII2I09-0045-9916.

We would like to thank *N.M. Cheung*, *O. C. Au* and the rest of the authors of the paper [9] for their invaluable support in improving this manuscript.

References

1. ISO/IEC International Standard 14496-10:2005, Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding
2. Wiegand, T., Sullivan, G.J., Bjontegaard, G., Luthra, A.: Overview of the H.264/AVC Video Coding Standard. IEEE Transaction on Circuits and System for Video Technology 13(7), 560–576 (2003)
3. Feng, W.-C., Manocha, D.: High-performance computing using accelerators. Parallel Computing 33(10-11), 645–647 (2007)
4. Nvidia, Nvidia CUDA Compute Unified Device Architecture-Programming Guide, Version 3.2 (August 2010)
5. Wiegand, T., Girod, B.: Lagrange multiplier selection in hybrid video coder control. In: Proc. IEEE International Conference on Image Processing, ICIP (2001)
6. Lee, C.-Y., Lin, Y.-C., Wu, C.-L., Chang, C.-H., Tsao, Y.-M., Chien, S.-Y.: Multi-Pass and Frame Parallel Algorithms of Motion Estimation in H.264/AVC for Generic GPU. In: Proceedings of IEEE International Conference on Multimedia and Expo 2007 (ICME 2007), Beijing (China), pp. 1603–1606 (July 2007)
7. Ryoo, S., Rodrigues, C., Baghsorkhi, S., Stone, S., Kirk, D., Hwu, W.-M.: Optimization Principles and Application Performance Evaluation of a Multithreaded GPU Using CUDA. In: Proceedings of the 13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, Salt Lake City (USA), pp. 73–82 (February 2008)
8. Chen, W.-N., Hang, H.-M.: H.264/AVC motion estimation implementation on Compute Unified Device Architecture (CUDA). In: Proceedings of IEEE International Conference on Multimedia and Expo 2008 (ICME 2008), Hannover (Germany), pp. 679–700 (June 2008)
9. Cheung, N.-M., Fan, X., Au, O.C., Kung, M.-C.: Video Coding on Multi-Core Graphics Processors. IEEE Signal Processing Magazine 27(2), 79–89 (2010)
10. Cheung, N.-M., Au, O.C., Kung, M.-C., Wong, P.H.W., Liu, C.H.: Highly Parallel Rate-Distortion Optimized Intra Mode Decision On Multi-Core Graphics Processors. IEEE Transactions on Circuit and System for Video Technology 19(11), 1692–1703 (2009)
11. Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, Reference Software to Committee Draft. JVT-F100 JM17.2 (2010)
12. X.264 reference software, `http://www.videolan.org/developers/x264.html`
13. Sullivan, G., Bjøntegaard, G.: Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low-Resolution Progressive-Scan Source Material, ITU-T VCEG, Doc. VCEG-N81 (2001)
14. Richardson, I.E.G.: Video codec design, 2nd edn. John Willey & Sons LTD. (2003)

A Low Complexity Macroblock Layer Rate Control Scheme Base on Weighted-Window for H.264 Encoder

Huibo Zhong, Sha Shen, Yibo Fan, and Xiaoyang Zeng

State Key Lab of ASIC and System, Fudan University
825 Zhangheng Road, Shanghai, 201203, China
fanyibo@fudan.edu.cn

Abstract. Rate control plays a very important role in video coding. A low complexity macroblock (MB) layer rate control scheme for H.264 encoder is presented in this paper. Based on the analysis of the relationship among the quantization parameter (QP), mean absolute distortion (MAD) and the coded bits, a weighted-window model is proposed. A weighted-window based QP decision and MAD prediction model is proposed to reduce the computational complexity of MB-layer rate control. A new rate control scheme based on these models is presented in detail. The experimental results show that the proposed scheme gives a quality improvement of about 0.80dB on the average for all sequences, and about 58% reduction in bit rate mismatch.

Keywords: H.264, rate control, weighted-window, Video coding.

1 Introduction

Rate control plays a very important role in real-time video communication applications. The goal of rate control is to regulate the coded bitstream to meet certain given constraints, such as bit rate, buffer overflow/underflow prevention. As one of the key technologies in regarded to coding performance, rate control has drawn significant research attentions.

1.1 Brief Review of Rate Control Scheme

In the field of video transmission, as the available channel bandwidth for the coding process can be constant or time varying, the rate control schemes can be classified into two major categories: constant-bit-rate (CBR) control for constant channel bandwidth and variable-bit-rate (VBR) control for variable channel bandwidth. The existing rate control schemes focus on the CBR case. There are two main problems in rate control schemes: the first is how to allocate proper bits to each coding unit, and the second is how to select the quantization parameter (QP) to encode each unit with the allocated bits. The problem of optimum bits allocation can be described by the formula as follows:

$$\min\{D\},\ \ \textit{subject to } R_c \le R_t \tag{1}$$

Where D denotes the distortion of current encoding unit, R_c and R_t denote the number of bits used to encode the unit and the bits budget respectively. The key point of

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 563–573, 2012.

quantization parameter (QP) selection is to find the relation between rate and QP. The relation between rate and QP is usually derived based on a rate-quantization (R-Q) model, and the most common R-Q models are either linear or quadratic. Linear R-Q model has been studied in MPEG-2 TM5 rate control [1]. The quadratic R-Q model, originally proposed for MPEG-4 Q2 rate control [2] [3], was considered better and more accurate than the linear one. This quadratic R-Q model has been adopted in the JM reference software for H.264 rate control [4~6].

1.2 Rate Control in H.264

According to the terms of the unit of rate control operation, rate control schemes for H.264 can be classified into MB-layer, frame-layer, group-of-picture- (GOP) layer rate control [6]. The rate control for H.264 is more difficult than those for other standards such as MPEG-2, MPEG-4, H.263, and so on. This is because the quantization parameters are used in both rate control algorithm and rate distortion optimization (RDO), which results in the following chicken-and-egg-dilemma: to perform RDO for MBs in the current frame, a quantization parameter should be first determined for each MB by using the mean absolute distortion (MAD) of current frame or MB, however, the *MAD* of current frame or MB is only available after the RDO. To solve this dilemma, a linear model using the actual *MAD* of the basic unit in the same position of previous frame is proposed in [6]:

$$MAD_a[k,i] = a_1 \times MAD_a[k,i-1] + a_2 \tag{2}$$

Where a_1 and a_2 are two coefficients of prediction model, MAD_a[k,i] and MAD_a[k,i-1] are the *MAD* of k^{th} MB in current (i) and previous (i-1) frame, respectively. By the value of MAD_a[k,i], the corresponding QP can be computed by following quadratic R-Q model:

$$R = c_1 \frac{MAD_a[k,i]}{Qs} + c_2 \frac{MAD_a[k,i]}{Qs^2} \tag{3}$$

Where the two parameters c_1 and c_2 are model coefficients, and *R* denotes the target bits used for encoding the current basic unit, *Qs* is quantizer step size. QP can be obtained by converting *Qs* as defined in [4].

It is noted that by employing a bigger coding unit, a higher PSNR can be achieved while the bit fluctuation is also bigger. On the other hand, by using a smaller coding unit, the bit fluctuation is less severe. This is very useful in some communication application. By employing a MB-layer rate control, the bit fluctuation can be the smallest; however, this will introduce the highest computational complexity and make it hard for real-time applications. Many previous works have paid attention to the accuracy of (1) and (2). To improve the performance of (2), frame bits allocation and MAD estimation accuracy have been enhanced using a PSNR-based frame complexity measure in [7]. Multi-pass rate control can achieve higher performance than one-pass rate control, [8] has proposed an optimized two-pass rate control with a linear R-Q model, however, it will introduce higher complexity. A weighted model for MAD prediction is proposed using for rate control in embedded systems [9], a simple QP decision method is proposed by judging the MAD of current MAD and previous frame average MAD, only a little better RD performance can be achieved compared with the JM's. Since any model is actually an approximate model and cannot always match ideally with real application, the estimation of model parameters usually has some deviation or errors.

In this paper, base on the analysis among QP, MAD, and coded bit, a Weighted-Window prediction model is proposed. This model removes the complex update of the coefficients in (1) and (2) [10]. A weighted-window model based QP decision and MAD prediction model is proposed to reduce the computational complexity of MB-layer rate control, a new rate control scheme based on these models is presented in detail. The experiment results show that our scheme achieves higher PSNR and smaller overall bit rate mismatch compared with JVT-G012 [6].

The rest of this paper is organized as follow. Section 2 presents the proposed rate control algorithm in detail. The experiment results and the comparison are given in Section 3. Finally, section 4 shows the conclusion.

2 Proposed Weighted-Window Based Rate Control Scheme

Temporal and spatial information are always been used to predict MAD and QP. In this section, a weighted-window model is proposed in this section. The relationship between QP, MAD and coded bit is analyzed to decide the size of the window. Based on this weighted-window model, temporal and spatial information are used. A new QP decision and MAD prediction model are proposed. And finally, the framework of the proposed rate control is presented in detail.

2.1 Weighted-Window Model

Experiments have shown that the average MAD of the current frame becomes bigger if the previous frame is quantized with a larger QP [9]. This is because the motion estimation has to refer to the reconstructed previous frame with more distortion. On the other hand, if the QP in the previous frame is smaller, the average MAD of the current frame should be decreased. Hence, we can jointly considering the average MAD and the average QP of current frame and the previous reference frame. Fig.1 illustrates two windows in current frame and previous frame. The window in current frame can be seen as spatial information and the window in previous frame as the temporal information. We use *Ws* to measure the size of the window, *Ws* equal to 3 denotes that the window is 3x3 macroblock square. Fig.1 is a case with a window size equals to 3. MB0$_c$ means current MB in current frame, and MB1$_c$~ MB8$_c$ are the corresponding MB of Left, Right, Top, Down, Top left, Top right, Down left, Down right, respectively. To find the relationship between the QP and MAD in the two windows, we introduce two variables N_c and N_p whose values are given by:

$$N_c = \sum_{k \in Wc} (\alpha_k \times QP[k,i]) / \sum_{k \in Wc} (\alpha_k \times MAD_a[k,i]) \tag{4}$$

$$N_p = \sum_{k \in Wp} (\beta_k \times QP[k,i-1]) / \sum_{k \in Wp} (\beta_k \times MAD_a[k,i-1]) \tag{5}$$

Where *Wc* and *Wp* denote the windows in current and previous frame respectively, *QP*[k,i], *QP* [k,i-1] are the QP of the MBs in the window of current and previous frame, respectively. *MAD$_a$*[k,i], *MAD$_a$*[k,i-1] are the MAD of the MBs in the two windows, α_k and β_k are weight factors.

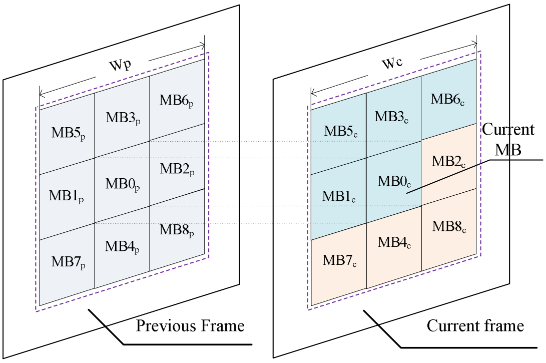
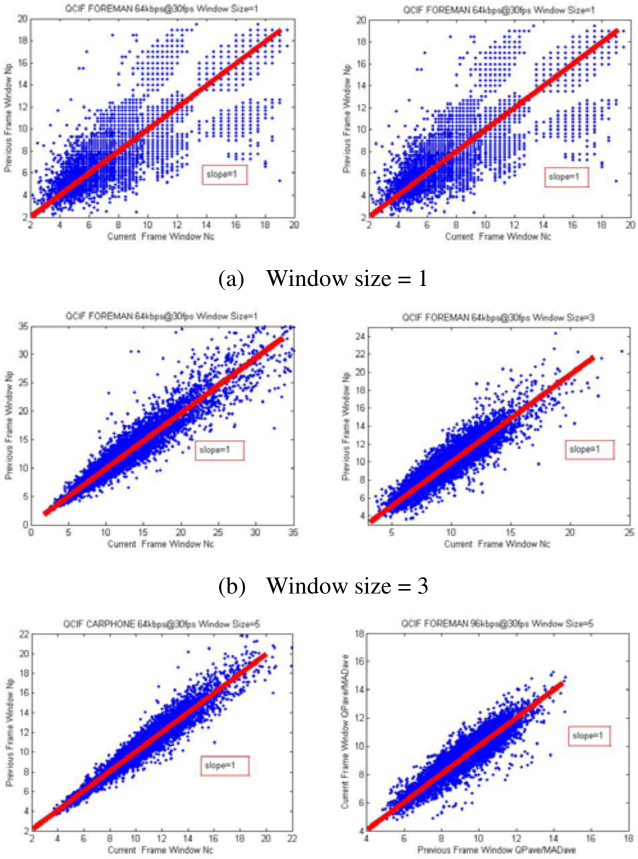


Fig. 1. Proposed weighted-window model with *Ws* is equal to 3



(a) Window size = 1

(b) Window size = 3

(c) Window size = 5

Fig. 2. The relationship N_c and N_p at different window size for test sequences: Carphone (left) and Foreman (right) 96kbps@30fps

To analyze the relationship between N_c and N_p, let $\alpha_0=\ldots=\alpha_k$, $\beta_0=\ldots=\beta_k$, and numerous experiments have been done according (4) and (5) at different window size. The results are shown in Fig.2, the bigger the window size, the closer the correlation between N_c and N_p; when *Ws* is 3 or larger, N_p is almost linear to N_c as follows:

$$N_c = N_p \quad if \;\; Ws \geq 3 \tag{6}$$

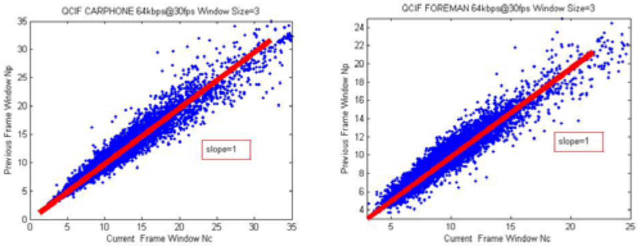


Fig. 3. Relationship between $N_{c,2}$ and N_p:Carphone (left) and Foreman (right) 64kbps @30fps

2.2 QP Computation Model Based on Weighted-Window

As mentioned above, when the window size is 3 or larger, we can have $N_c = N_p$. The more candidates taken into account, the more parameters should be determined, which will introduce more computational complexity. The value of *Ws* is set as 3 in this paper. As shown in Fig.1, at the current frame, the data of MB2$_c$, MB7$_c$, MB4$_c$, and MB8$_c$ is not available when the encoder is coding MB0$_c$, so (4) has to be adjusted. The data of MB2$_c$, MB4$_c$, MB7$_c$ and MB8$_c$ are substituted by MB2$_p$, MB4$_p$, MB7$_p$ and MB8$_p$ as follows:

$$\begin{aligned} N_{c,2} = \{ & \sum_{k=0,1,3,5,6} \alpha_k \times QP[k,i] + \sum_{k=2,4,7,8} \alpha_k \times QP[k,i-1]\} \\ & / \{ \sum_{k=0,1,3,5,6} \alpha_k \times MAD_a[l,i] + \sum_{k=2,4,7,8} \alpha_k \times MAD_a[k,i-1]\} \end{aligned} \tag{7}$$

Similar relationship between $N_{c,2}$ and N_p can be obtained as Fig.3 shown when *Ws* = 3.

$$N_{c,2} = N_p \qquad when \;\; Ws = 3 \tag{8}$$

For MB layer, the coded bit of each MB should also be considered. The bit allocated for each MB is by judging the value of the current MAD, the remaining bit, and the complexity of current picture. Combine (5), (7) and the coded bit, we have:

$$QP[0,i] = \{S_c \times N_p \times \phi - \sum_{k=1,3,5,6} \alpha_k \times QP[k,i] - \sum_{k=2,4,7,8} \alpha_k \times QP[k,i-1]\} / \alpha_0 \tag{9}$$

Where S_c is computed as:

$$S_c = \sum_{k=0,1,3,5,6} \alpha_k \times MAD_a[k,i] + \sum_{k=2,4,7,8} \alpha_k \times MAD_a[k,i-1] \tag{10}$$

According to experiment results, the weight factors are set as follows:

$$\alpha_k = \begin{cases} 3, & k=0 \\ 2, & k=1,2,3,4 \\ 1, & k=5,6,7,8 \end{cases} , \quad \beta_k = \begin{cases} 4, & k=0 \\ 2, & k=1,2,3,4 \\ 1, & k=5,6,7,8 \end{cases} \tag{11}$$

Where 0~8 denote the position of MB0~MB8 as shown in Fig.1. ϕ is a regulated factor which value is computed by :

$$\phi = \frac{Bit[0,i]}{\sum_{k\in Wp} \beta_k \times Bit[k,i-1] / \sum_{k\in Wp} \beta_k} \tag{12}$$

Where *Bit*[0,i] is the coded bit for current MB, *Bit*[k,i-1] is the coded bit of the MBs in the window of previous frame. ϕ is used for regulating the obtained QP according to the consumed bits. Before encoding current MB, *Bit*[0,i] is always obtained by bit allocation according to the predicted MAD of current MB. If *Bit*[0,i]> $\sum_{k\in Wp} \beta_k \times Bit[k,i-1] / \sum_{k\in Wp} \beta_k$, it means the bit used for current MB is large than the number of the bits previous window used, ϕ>1 is obtained to achieve a relative large QP, else if *Bit*[0,i]< $\sum_{k\in Wp} \beta_k \times Bit[k,i-1] / \sum_{k\in Wp} \beta_k$, the bit used for current MB is smaller than the previous window used, ϕ<1 is obtained to achieve a relative small QP . The QP obtained by (9) is rounded as follows:

$$QP = \lfloor QP[0,i] + 0.5 \rfloor \tag{13}$$

Where $\lfloor \ \rfloor$ is the floor operation, the further bound of the QP is presented in the following parts.

2.3 MAD Prediction Model Based on Weighted-Window

As we mentioned above, for the current MB, we can obtain the value of QP according to (9), however, *MAD*$_a$[0,i] can only be obtained until RDO is done. As the one which is used for QP decision model, we use the same weighted-window to predict the MAD of current MB. According to our experiment, similar relationship exists between the MAD of previous frame and current frame. Same as the QP decision model, a similar MAD prediction model based on weighted-window can also be established. As Fig.1 shows, we use the previous window to predict the MAD of current window in current frame. Two variables M$_p$ and M$_c$ are introduced as follows:

$$M_p = \sum_{k=0}^{8} \beta_k \times MAD_a[k,i-1] / \sum_{k\in Wp} \beta_k \tag{14}$$

$$M_c = \{ \sum_{l=0,1,3,5,6} \gamma_l \times MAD_a[l,i] + \sum_{l=2,4,7,8} \gamma_l \times MAD_a[l,i-1] \} / \sum_{l\in Wp} \gamma_l \tag{15}$$

Where γ

Where γ_l and β_k are the weight factors, the value of β_k is the same as (11) shown. We analyze the relationship between M_p and M_c in the same way as N_p and N_c. The results are presented in Fig.4. We can have:

$$M_c = M_p \quad when\ Ws = 3 \tag{16}$$

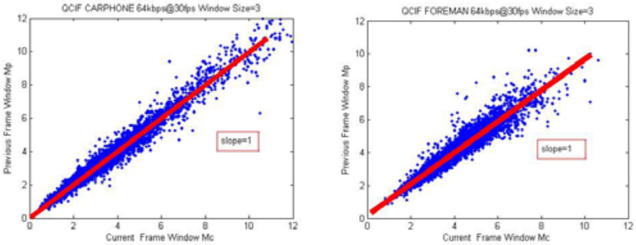


Fig. 4. Relationship between M_c and M_p @Ws=3 for test sequences: Carphone (left) and Foreman (right) 64kbps@30fps

As shown in Fig.4, we can have $M_p = M_c$, $MAD_a[0,i]$ can be computed by:

$$MAD_a[0,i] = \{M_p - (\sum_{l=1,3,5,6} \gamma_l \times MAD_a[l,i] - \sum_{l=2,4,7,8} \gamma_l \times MAD_a[l,i-1]) / \sum_{l \in Wp} \gamma_l\} / \gamma_0 \tag{17}$$

Where the weight factor γ_l is given as following according to the experimental results:

$$\gamma_l = \begin{cases} 3, & l = 0 \\ 1, & l = 1 \sim 8 \end{cases}$$

2.4 Proposed Rate Control Framework

With the proposed QP and MAD model based on the weighted-window, we now present our rate control scheme for H.264. The proposed rate control scheme includes three different coding granularities: the GOP-layer, frame-layer, and MB- layer. At the GOP-level and frame-layer, it is the same way as [6] to allocate target bits and perform the post-encoding regulation. Now we focus on a step-by-step description of the proposed scheme at MB-layer for P-frames as Fig.5 shows.

In Fig.5, for the first I/P frame, the initial QP is computed by the same way as Li's [6], QP_{ave} is the average QP of previous frame, QP[k-1,i] is the QP of the previous MB in current frame. The MB layer bits allocation is according to the predicted MAD obtained from (17) and the average MAD of all coded MBs in current frame as following shows:

$$T_c = T_r \times MAD_a[0,i] / (MAD_{ave,c} \times N) \tag{18}$$

T_c and T_r are the target bits for the current MB and the remaining bits for all MBs which are not encoded yet in the current frame. $MAD_{ave,c}$ denotes the average MAD of all coded MB in current frame and N is the number of the remaining MBs to be encoded. T_r is updated by substracting the total encoded bits of encoded MB from it.

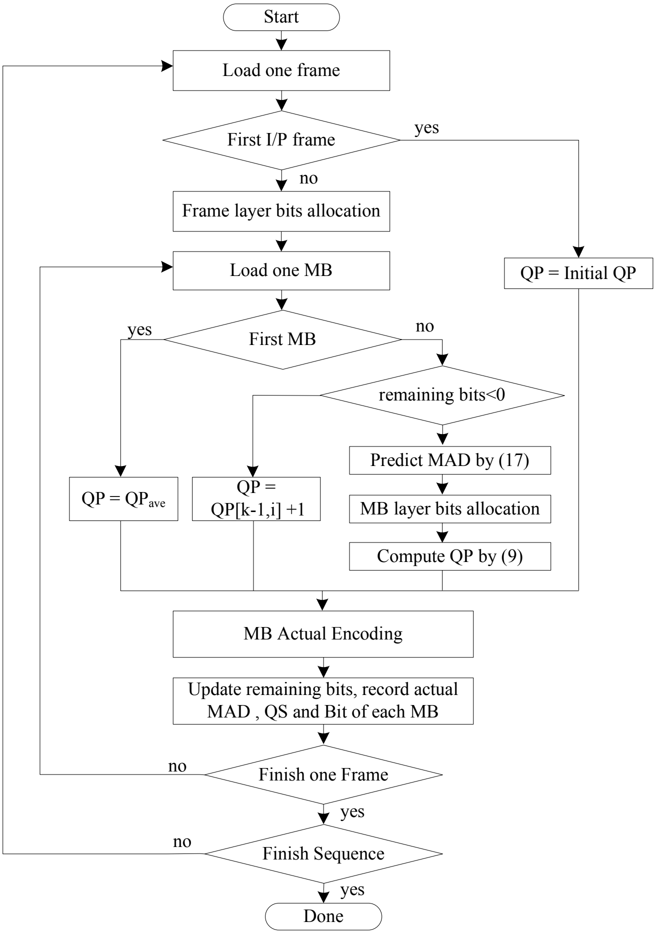


Fig. 5. Proposed rate control scheme framework base on the weighted-window

If current MB is the first MB in current frame, current QP is set to be the average QP of the previous P-frame. For other MBs, QP can be calculated as follows. If the remaining bits are negative, the current QP is set to *QP*[k-1,i]+1 to achieve frame level actual bits that are closer to target bits, otherwise, allocate bits for the current MB by (18), and calculate QP for the MB by (9). The derived QP value should also be restricted by the QP value of the previously encoded MB to reduce blocking artifacts by (19):

$$QP[k,i] = \min\{QP[k-1,i]+1, \max\{QP[k-1,i]-1, QP\}\} \tag{19}$$

Qp[k,i] is the QP value for the kth MB in current frame. To maintain the smoothness of the visual quality within one sequence, the QP value is further adjusted by (18):

$$QP[k,i] = \min\left\{QP_{ave} + 2, \max\{QP_{ave} - 2, QP\}\right\} \tag{20}$$

And finally, the QP value should be restricted between 1 and 51, which is provided in H.264/AVC [4]:

$$QP[k,i] = \min\left\{51, \max\{1, QP\}\right\} \tag{21}$$

After encoding each MB, the encoder should update the remaining bits and record the data such as the actual QP, coded bit and MAD.

3 Experimental Results

Our proposed rate control scheme is implemented in JM15.1 [5], to evaluate the performance of the proposed rate control scheme, the test parameters for encoding are: 1) CABAC is used; 2) Hadamard transform is used; 3) Max reference frame number is 5, and search range is 16; 4) Fast full search is used; 5) RDO is on, and rate control mode is 0. All other parameters are carefully selected for both algorithms to be equivalent. For each sequence, 300 frames are encoded at 30fps, and the GOP structure is IPPP, the GOP length is 300 if not specified, for each GOP, the first frame is IDR frame, and the following 299 frames are P-frame.

Table 1. Rate control performance comparison between JM15.1 and the proposed

Sequence	Target rate (kbps)	JM 15.1		Proposed	
		PSNR	Rate	PSNR(dB)	Rate
akiyo	48	40.64	48.08(+0.08)	41.70(+1.06)	48.05(+0.05)
	64	42.12	64.10(+0.10)	43.03(+0.91)	64.06(+0.06)
	96	44.51	96.19(+0.19)	45.07(+0.56)	96.07(+0.07)
Carphone	48	32.02	48.09(+0.09)	32.67(+0.65)	48.03(+0.03)
	64	33.26	64.09(+0.09)	33.88(+0.66)	64.03(+0.03)
	96	35.30	96.12(+0.12)	35.83(+0.53)	96.01(+0.01)
Container	48	36.43	48.04(+0.04)	37.17(+0.74)	48.02(+0.02)
	64	37.66	64.04(+0.04)	38.25(+0.59)	64.02(+0.02)
	96	39.34	96.06(+0.06)	39.84(+0.50)	96.04(+0.04)
foreman	48	31.10	48.10(+0.10)	31.50(+0.40)	48.03(+0.03)
	64	32.49	64.08(+0.08)	32.89(+0.40)	64.05(+0.05)
	96	34.50	96.17(+0.17)	34.82(+0.32)	96.06(+0.06)
grandma	48	37.25	48.08(+0.08)	38.18(+0.93)	48.05(+0.05)
	64	38.55	64.06(+0.06)	39.46(+0.91)	64.03(+0.03)
	96	40.92	96.10(+0.10)	41.72(+0.80)	96.09(+0.09)
salesman	48	35.21	48.07(+0.07)	36.25(+1.04)	48.02(+0.02)
	64	37.07	64.04(+0.04)	37.94(+0.87)	63.98(-0.02)
	96	39.90	96.06(+0.04)	40.48(+0.58)	96.04(+0.04)

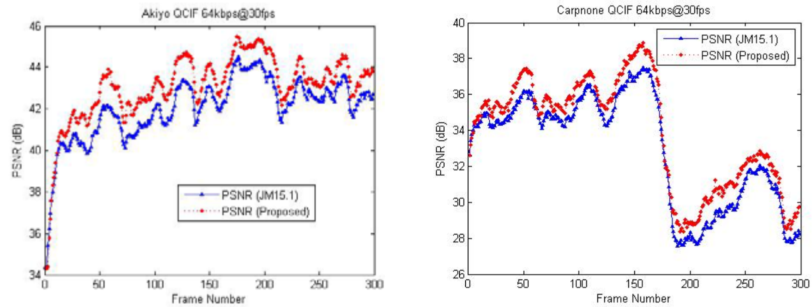


Fig. 6. PSNR comparison frame by frame: Akiyo (left), Carphone (right) 64kbps @30fps

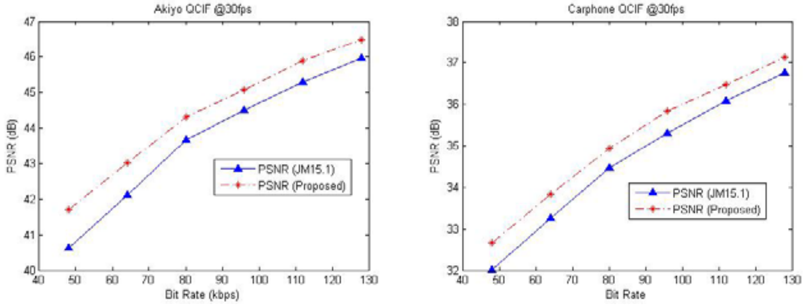


Fig. 7. PSNR comparison at different bit rate: Akiyo (left), Carphone (right) 64kbps @30fps

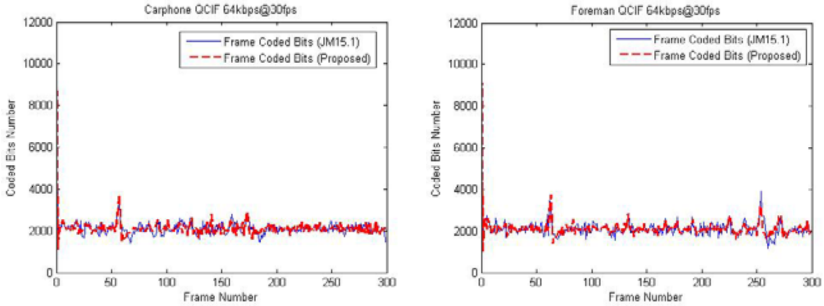


Fig. 8. Coded bits comparison frame by frame: Carphone (left), Foreman (right) 64kbps@30fps

The rate distortion performance comparison is summarized in Table 1. For all test sequences, the target bit rate is set to 48, 64 and 96kbps. It shows that our proposed rate control scheme achieves better results with a largest increase in PSNR about 1.06dB (Akiyo @48kbps) and an average increase for all test sequences of about 0.80dB @48kbps, 0.72dB @64kbps, and 0.55dB @96kbps. Fig.6 illustrates the comparison of the PSNR curse for QCIF sequences Akiyo and Carphone @64kbps frame by frame. Fig.7 gives the PSNR comparison at different bit rate. The results prove that the proposed rate control scheme outperforms the original rate-control scheme proposed in JM15.1 [5].

Table 1 also shows that the bit rate mismatches in our proposed rate control scheme is smaller than that of JM15.1. The average bit rate mismatch in our proposed rate control scheme for all sequences is about 0.069%, 0.055% and 0.054% at different bit rate. The corresponding values are 0.188%, 0.107% and 0.122% respectively in JM 15.1. The average bit rate mismatch for all video sequences is reduced by 58% with our proposed rate control scheme. Fig.8 shows the number of coding bits frame-by-frame of QCIF sequence Carphone and Foreman 64kbps @30fps.

4 Conclusion

Based on the analysis of the relationship among the quantization parameter (QP), mean absolute distortion (MAD) and the coded bits, a weighted-window model is proposed in this paper. A weighted-window model based QP decision and MAD prediction model is proposed to reduce the computational complexity of MB-layer rate control. A new rate control scheme based on these models is presented in detail. The experimental results show that the proposed scheme gives a quality improvement of about 0.80dB on the average for all sequences, and about 58% reduction in bit rate mismatch.

Acknowledgments. This work is supported by the State Key Lab of ASIC & System, Fudan University. Project Number: 11MS004.

References

1. Test Model Editing Committee, MPEG-2, Test Model 5, Doc. ISO/IEC JTC1/SC29 WG11/93-225b (April 1993)
2. Lee, H.J., Chiang, T.H., Zhang, Y.Q.: Scalable Rate Control for MPEG-4 Video. IEEE Trans. CSVT 10, 878–894 (2000)
3. Chiang, T., Zhang, Y.Q.: A New Rate Control Scheme using a New Rate-Distortion Model. IEEE Trans. CSVT, 246–250 (February 1997)
4. ITU-T, H.264, Advanced video coding for generic audiovisual services (March 2005)
5. Joint Video Team (JVT) of ITU-T VCEG and ISO/IEC MPEG, Joint Model (JM) Reference Software Version 15.1, http://iphome.hhi.de/suehring
6. Li, Z.G., Pan, F., Lim, K.P., Feng, G.N.: Adaptive basic unit layer rate control for JVT, JVT-G012. In: 7Th meeting, Pattaya, Thailand (2003)
7. Jiang, M., Ling, N.: Low-delay rate control for real-time H.264/AVC video coding. IEEE Trans. Multimedia, 467–477 (June 2006)
8. Kwon, D.-K., Shen, M.Y., Kuo, C.C.: Rate control for H.264 Video With Enhanced Rate and Distortion Models. IEEE Trans. CSVT, 517–529 (2007)
9. Kuo, C.H., Chang, L.C., Fan, K.W., Liu, B.D.: Hardware/software co-design of low cost rate control scheme for H.264/AVC. IEEE Trans. CSVT 20(2), 250–261 (2010)
10. Chang, L.C., Kuo, C.H., Liu, B.D.: A Two-Stage Rate Control Mechanism for RDO-Based H.264/AVC Encoders. IEEE Trans. CSVT, 660–673 (May 2011)

Forward Wyner-Ziv Fast Video Decoding Using Multicore Processors

Alberto Corrales-Garcia[1], José Luis Martínez[2],
Gerardo Fernández-Escribano[1], and Francisco Jose Quiles[1]

[1] Instituto de Investigación en Informática de Albacete (I3A),
University of Castilla-La Mancha, Campus Universitario, 02071 Albacete, Spain
{albertocorrales,gerardo,paco}@dsi.uclm.es
[2] Architecture and Technology of Computing Systems Group. Complutense University
Ciudad Universitaria s/n, 28040 Madrid, Spain
joseluis.martinez@fdi.ucm.es

Abstract. With the aim of providing low complexity encoders, Wyner-Ziv video coding provides a new paradigm where the complexity of the encoder is moved to the decoder. However, this high decoding complexity could involve a problem in some applications which have delay restrictions. Nowadays parallel computing is a growing field into the computation market. In particular, most of personal computers and hardware for video coding includes multicore processors, which allows a parallel execution by means of several independent cores in a same chip. As a consequence, several DVC parallel decoding approaches are beginning to appear. This work proposes a parallel DVC decoding scheme for multicore processors, which decodes each GOP in an independent and parallel way. This scheme achieves above 70% time reduction without any rate-distortion penalty.

Keywords: Distributed Video Coding, Parallel Computing, Multicore Processors, OpenMP.

1 Introduction

Traditionally, the digital video codecs adopted by all MPEG and ITU-T video coding Standards have based their design in architectures where encoders are more complex than decoders [1]. Nowadays, new devices (such as surveillance systems, sensor networks, micro cameras, etc) with low-cost hardware can integrate cameras, but they should carry out a low-cost encoding. For this kind of applications, Wyner-Ziv (WZ) video coding [2] (which is a particular case of Distributed Video Coding (DVC) [3]) is an attractive paradigm since it provides a framework where the complexity of the encoder is displaced to the decoder allowing low cost encoding. This low-complexity is achieved because the encoding process does not exploit the temporal correlation to compress more the video information. In addition, DVC can achieve theoretically a similar Rate-Distortion (RD) performance than the joint video coding. At the same time, it provides robustness over noisy channel (as it often happens in wireless networks). Despite all this advantages, the complexity of the decoder is highly

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 574–584, 2012.

increased. Most of this complexity is caused by the iterative turbo decoding algorithm. In particular, the feedback channel contributes to a large degree in the cost of the decoder [4]. Although the amount of time taken by the decoder could not seem important, for many applications which the decoding delay plays an important role (such as WZ to H.264 video transcoding [3]), a fast decoding is desired and sometimes mandatory.

On the other hand, the technological advancement in microprocessors has introduced new architectures, which allow high-performance computing [5]. In particular, regarding processors, the new architectures tend to include several processors (called cores) in the same chip. This kind of processors is called Multicore Processors and nowadays they are widely extended in the market. However, although multicore processors can help to reduce the time spent by high-complex tasks, most of applications are designed to be executed in a sequential way. As a consequence, high complex tasks follow spreading much time and they do not take advance of the available computational capacity. To exploit this research field, the researching community should invest effort to propose new architectures and methods to take advance of the parallel computing to use efficiently the computational capacity that the new hardware offers to us.

At that point, this paper proposes to reduce the complexity of the WZ decoder (the WZ decoding complexity is even higher than traditional video coding algorithms [4]) by means of a multicore processor system. As a first attempt to achieve this, each Group of Pictures (GOP) is decoded in each processing unit. This provides good trade-off between the time reduction and the RD loss. The simulations results offer an acceptable time reduction up to 71% without any RD penalty. Moreover, the present proposal is scalable for a higher number of cores.

Accordingly, this paper is organized as follows: Section 2 presents an overview of the WZ codec and multicores; section 3 shows the previous works related to reducing WZ decoding complexity and some parallel DVC decoding approaches; section 4 proposes the parallel WZ decoder based on multicore; section 5 presents experimental results for the proposed architecture; and, finally, some final remarks are presented in Section 6.

2 Technical Background

2.1 Distributed Video Coding

Theoretical fundaments of DVC depart from the information theory [6]. However, is in [3] where one of the first practical DVC architecture was proposed by Stanford University. It is based on turbo codes as Slepian-Wolf encoder/decoder and a feedback channel used by the decoder to manage the rate control. Over the Stanford architecture many research and improvement have been carried out in the literature. Later, in [7] was proposed the DISCOVER codec architecture as a result of a European project and it could be considered as a reference codec in the DVC paradigm. In addition, DISCOVER codec was later improved by VINET-II team [8]. These codecs are focus on achieving the best RD results without considering the time spent. In this work, our architecture is an improved version of VISNET-II codec which is depicted in Figure 1.

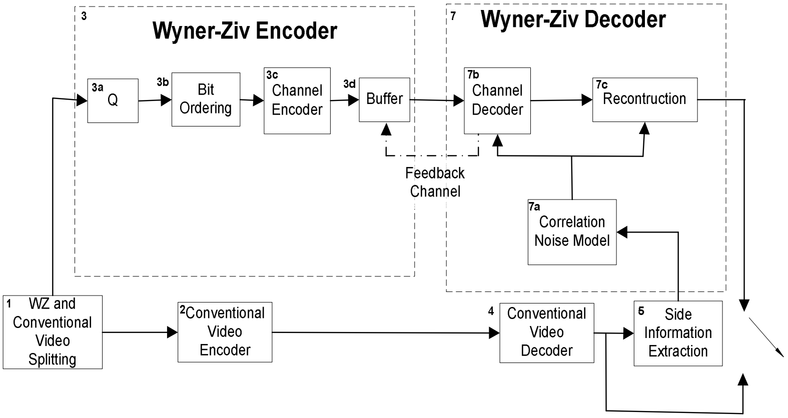


Fig. 1. Block diagram of the reference DVC architecture

The Figure 1 presents a scheme of the architecture employed in this paper. To sum up the basic WZ video coding architecture operation, we should know WZ video coding deals with two kinds of frames: Key Frames (K) and Wyner-Ziv Frames (WZ). Each frame of the sequence is sent to a different channel by means of the splitting module (1). At the encoder side, the K frames are encoded using a H.264/AVC Intra encoder [1] (2). On the other hand, the WZ frames are sent to a Wyner-Ziv Encoder (3). On the first stage, the frame information is quantized (3a). Afterwards, over the resulting quantized symbol stream, a bitplane extraction is performed per bitplanes (3b). Each bitplane is then independently channel encoded, starting with the most significant bitplane (3c). The parity bits produced by the channel encoder are stored in the buffer and transmitted in small amounts upon decoder request via the feedback channel; the systematic bits are discarded (3d).

On the other hand, in the decoder side, firstly the K frames are decoded using a H.264/AVC Intra decoder [1] (4). Then, in (5) the decoder uses each both frames like previous and next temporary references to create a Side Information (SI) frame. SI represents an estimation for each non-present original WZ frame. From each SI in (6) a Laplacian distribution models the residual statistics between corresponding WZ frame and SI. Then, the SI and the statistic model associated are used in an iterative decoding algorithm (7b) to obtain the decoder quantized symbol. In this module, each bitplane is decoded in a sequential order. Every decoding iteration new parity bits are requested to the encoder by means of the feedback channel. To decide if more parity bits are requested, a stopping criterion is defined based on error probabilities. When the decoding is considered successfully another bitplane is being decoded. Finally, the reconstructed pixels are obtained using the decoded quantized pixels, the correlation noise model estimated in (7a) and the quantized SI pixels.

2.2 Multicore Processor System

As a consequence of the computation limit of single processors, some time ago was introduced successfully the idea of having multiple cores in a same chip. Nowadays the usage of multicore processors is growing more and more. In fact, most of commercial computers include a multicore processor to increase the performance of the computers. In a muticore processor each core can execute a different application or they can work in a collaborative way to accelerate the execution of one application. However, due to heritage of simple processors, most of complex applications are designed to be executed in a sequential way and then the computational capacity of multicore systems are not fully exploited. In the particular case of multimedia applications, multicore processors can help to accelerate complex tasks. In fact, many multimedia hardware solutions are based on this kind of architectures. However, new methods and algorithms should be proposed to support the parallel execution as efficiently as possible.

In the architecture of a multicore processor, several cores share the same chip and they have some shared memory and some private memory. Regarding commercial multicore processors, the highest performance is reached by the multicore processors based on Intel Nehalem Micro-architecture [9]. In particular, this paper is based on this multicore processor Intel i7-940. The most important features of this processor are the following: four cores, clock speed of 2.93 GHz, 45nm manufacturing process, new point-to-point processor interconnect, Intel QuickPath Interconnect (QIP), Simultaneous Multi-Threading (SMT) by multiple cores which enables two threads per core (hyper-threading) and three levels of cache (32 KB L1 instruction and 32 KB L1 data cache per core, 256 KB L2 cache per core and 8 MB L3 cache shared by all cores).

On the other hand, Open Multi-Processing (OpemMP) has been proposed to develop parallel programs over this kind of multicore processors [10]. OpemMP is an Application Programming Interface (API) that supports multi-platform shared memory multiprocessing programming. It provides a portable and scalable model which consists of a set of compiler directives, library routines, and environment variables that influence run-time behavior.

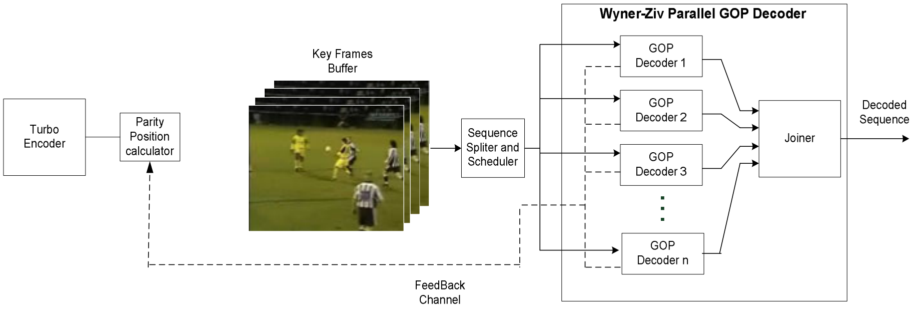


Fig. 2. Proposed Parallel DVC GOP decocing architecture

3 Related Work

DVC framework is based on displacing the complexity from the encoders to the decoders; however, a reduction of the complexity into the decoders is desirable. In traditional feedback-based DVC architectures [3], the rate control is done at the decoder and it is controlled by means of feedback channel; this is the main reason of the decoder complexity just because once a parity chunk arrives to the decoder, the turbo decoding algorithm (one of the most computational task [4]) is called. Taking this fact into account, there are several approaches which try to reduce de complexity of the decoder, which usually induces RD penalty. However, due to the technology advance, new parallel hardware is being introduced in practical video coding solutions. These new features of computers offer a new challenge for the research community to integrate its algorithms into the parallel framework; this opens a new door in the multimedia research. On the one hand, regarding the traditional standards several approaches have been proposed since multicores appeared in the market but, this paper focuses on parallel computing applied to the DVC framework.

On the other hand, in 2010 have been proposed different parallel solutions for DVC. In particular, in [11] *Oh et al.* proposed a DVC parallel execution carried out by Graphic Processing Units (GPUs). In this proposal, authors focus on design a parallel distribution for a Slepian-Wolf decoder based on rate Adaptative Low Density Check Code (LDPC) with Accumulator (LDPCA). LDPC codes are composed by many bit-nodes which do not have many dependencies between each node, so they propose a parallel execution in three kernels (steps): “kernels for check nodes calculations”, “kernels for bit nodes calculations”, and “kernels for termination condition calculations”. In a NVIDIA GeForce GTX260 216SP GPU they achieve a decoding 4~5 times faster for QCIF and 15~20 for CIF. On the other hand, in [12] *Momcilovic et al.* proposed a DVC LDPC parallel decoding based on multicore processors. In this work, the authors parallelize several LDPC approaches (Sum-Product, Min-Sum, and Algorithm E). In a Quad-Core machine, they reach and speedup up to 3.5. Both previous approaches propose a low level parallelism for a particular LDPC/LDPCA implementation.

However, the current work presents a higher level parallel WZ video decoding algorithm implemented over a multicore system. The reference WZ decoding algorithm is adapted by means of a GOP parallel decoding. In addition, the proposed algorithm is scalable because it does not depend on the hardware architecture neither the number of cores or on the implementation of the internal Wyner-Ziv decoder. Therefore, the time reduction can be increased simply by increasing the number of cores, as technology advance. Furthermore, the algorithm depicted in this paper could be extended by using another level of parallelism (frame or bitplane) as well as GPUs.

4 Proposed Wyner-Ziv GOP Parallel Decoding

Although most of commercial computers include multiple core processors, several cores are inactive regularly, while other are overloaded. As a result, the computation capacity is wasted and the complex tasks spend more time to finish. In the DVC framework, the fist aim is providing simple encoders, which are suitable to encode sequences in low cost devices. However, the decoder complexity is highly increased and sometimes, this high decoding delay does not allow including DVC in real environments. The first goal of this work is providing a simple and practical solution, which will be allow to execute WZ decoding in a parallel way, saving much decoding time and tanking advance of the computation capacity of whatever multicore processor.

4.1 Proposed Architecture

The traditional WZ decoding presents several sequential dependences such as the updating of CNM between bitplanes. Basically, once a bitplane is successfully decoded, the correlation model is updating by the following bitplanes.

Depending on the parallelism level, it is necessary to break these dependences, which could affect increasing the bitrate needed or decreasing the quality of the decoded frame. This is because a poor SI or a bad correlation model deals with a loss of performance. However, the WZ video coding offers an independent GOP decoding so we could execute each WZ GOP decoding in an independent core. In this way, we achieve a parallel decoder, which carries out a fast parallel decoding without any rate RD penalty. Figure 2 shows the proposed scheme, where the number of decoders is equal than the number of available cores, and then each decoder will decode one independent GOP. This involves that the architecture is not fixed for a specific hardware. In other words, it is architecture scalable.

In addition, the architecture has a module splitter and joiner. The first one carries out the task of splitting the key frame sequence by sending each key frame to the corresponding core. In addition, the joiner module includes the execution schedule. Each decoded GOP could have different delay due to the complexity of the scene and thus the number of the iterations needed. In addition, during a sequential execution there is one core in use and the rest are idle.

Taking this fact into account, the best schedule is a dynamic schedule (Figure 3), which assigns new tasks when any core finishes the current task. In this way, if there are GOPs to be decoded, all cores will be working and the capacity of the multicore processor will be utilized fully. In the end of the sequence, some cores could be idle, but this period is insignificant comparing with the whole sequence time decoding.

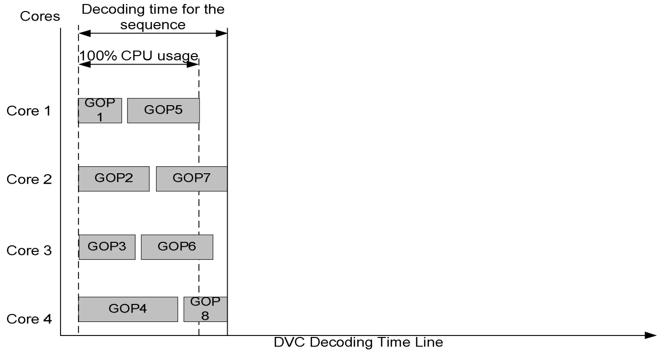


Fig. 3. DVC parallel GOP decoding time line execution for a 4 cores processor

On the other hand, the joiner module carries out the task of join each GOP in a suitable sequential way, because the parallel decoding could not maintain the source order.

In addition, as the decoding could be carried out without following a sequential order, the parity data could be also requested without a sequential order. To consider this case, the decoder sends a few bits in a header of the request to the encoder, which includes a module to estimate the relative position of the parity data related to each GOP. The *Parity Position* (*PP*) is calculated by the Equation 1, where *I* is the Intra period, *P* is position of the current GOP and *Q* is the quantification parameter. On the other hand *W* is the width of the image and *H* the height.

$$PP = (I - 1) * P * Q * \left(\left(\frac{W * H * 2}{8}\right) + 1\right) \tag{1}$$

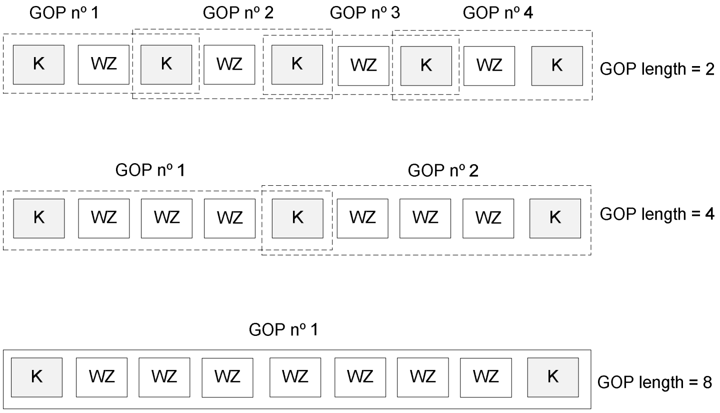


Fig. 4. Distribution of GOPs en each core and data shared for GOPs 2, 4 and 8

The WZ video coding only needs two reference frames to build any GOP length. Therefore, for consecutive frames, the same K frame is shared, as it is shown in Figure 4.Normally, in the WZ video coding the GOP sizes used are 2, 4 and 8.

In a nutshell, initially the parallel decoder needs to store several frames. This number of frames will depend on the number of available cores, but it does not depend on the GOP length. In general terms, the *Number of Frames* (*NF)* needed in the key frame buffer at the beginning is defined by the Equation 2, where *c* is the number of cores.

$$NF = \mathrm{c} + 1 \tag{2}$$

Finally, the structures created at the beginning for each core are reused for every GOP decoding, saving time and reusing the allocated memory.

5 Experimental Results

In order to evaluate the proposed parallel DVC decoding, four QCIF sequences were considered. These sequences have different motion and complexity features. For each sequence 150 frames were encoded by using the DVC VISNET-II codec [8]. The parallel implementation departs from the VISNET-II codec and was implemented by using Intel C++ compiler (version 11.1) [9] which combines a high-performance compiler as well as Intel Performance Libraries to support multi-threading applications. In addition, it provides support for OpenMP 3.0 [10]. In order to study the performance of the DVC parallel decoder, the sequences were encoded by using quantification from 1 to 4 bitplanes in pixel domain. In addition, to analyze the impact in different GOPs, several lengths of GOP were selected (2, 4 and 8).

The Time Reduction achieved (*TR*) is calculated by the Equation 3, where $Time_{seq}$ is the time spent by the sequential decoding and $Time_{par}$ the time spent by the proposed parallel version. Additionally, the speedup is calculated as the reference time divided by the parallel time.

$$TR = 100 * \frac{\left(Time_{seq} - Time_{par}\right)}{Time_{seq}} \tag{3}$$

The Table 1 shows the results for a GOP length = 2. It displays the results for several bitplanes (BPs) for each sequence. The Reference Time per frame column represents the decoding time spent by one WZ frame (on average). The reference version is composed by the VISNET-II sequential version. In the proposed parallel version a four-core processor was used (more details in section 2.2). As this multicore processor allows hyperthreading, each core runs 2 threads sharing the same core. In general, time reduction is higher in more complex sequences (such as foreman and soccer) reaching a mean of 71.05%. In most of the cases the speedup is between 3.5 and 4 (the maximum theoretically by using four core is 4). The RD results are not included due to for both versions (reference and proposed) are exactly the same.

On the other hand, tables 2 and 3 show the results for GOP length 4 and 8 respectively. As it is expected, the decoding time is a little higher for the middle frames because the distance of their references is higher and then, the SI generated is worse. To correct a worse initial SI, the decoding needs more interactions and then the decoding time per frame is increased. However, for longer GOP lengths similar conclusions are observed when sequential and parallel versions are compared.

In addition, a study about the influence of the number of cores and threads was done. Figure 5 shows the decoding time and the speedup factor (for Foreman sequence with GOP length = 2 and 3 BPs) when different threads are used in a 4 core processor with hyperthreading. As it is observed, the first 4 obtain a more significant time reduction whereas following 4 threads reach less time reduction. This is caused by the hyperthreading effect: when more than 4 threads are running, they are sharing the same physical cores and then the time reduction is lower.

Table 1. Parallel Decoder performance for GOP 2

Sequence	BP	Reference Time per frame (s)	Parallel Time per frame (s)	TR(%)	SpeedUp
Foreman	**1**	4.33	1.21	72.01	3.57
	2	7.49	1.94	74.12	3.86
	3	13.41	3.49	74.01	3.85
	4	20.05	5.39	73.13	3.72
Hall	**1**	3.18	1.21	62.07	2.64
	2	4.63	1.43	69.19	3.25
	3	8.35	2.18	73.88	3.83
	4	11.14	2.89	74.03	3.85
CoastGuard	**1**	3.05	1.22	59.95	2.50
	2	6.26	1.83	70.77	3.42
	3	11.97	3.37	71.82	3.55
	4	18.09	4.98	72.45	3.63
Soccer	**1**	7.18	2.07	71.21	3.47
	2	12.02	3.41	71.60	3.52
	3	19.47	5.24	73.06	3.71
	4	26.04	6.91	73.46	3.77
Mean		**11.04**	**3.05**	**71.05**	**3.51**

Table 2. Parallel Decoder performance for GOP 4

Sequence	BP	Reference Time per frame (s)	Parallel Time per frame (s)	TR(%)	SpeedUp
Foreman	**1**	4.33	1.21	72.01	3.57
	2	7.49	1.94	74.12	3.86
	3	13.41	3.49	74.01	3.85
	4	20.05	5.39	73.13	3.72
Hall	**1**	3.18	1.21	62.07	2.64
	2	4.63	1.43	69.19	3.25
	3	8.35	2.18	73.88	3.83
	4	11.14	2.89	74.03	3.85
CoastGuard	**1**	3.05	1.22	59.95	2.50
	2	6.26	1.83	70.77	3.42
	3	11.97	3.37	71.82	3.55
	4	18.09	4.98	72.45	3.63
Soccer	**1**	7.18	2.07	71.21	3.47
	2	12.02	3.41	71.60	3.52
	3	19.47	5.24	73.06	3.71
	4	26.04	6.91	73.46	3.77
Mean		**11.04**	**3.05**	**71.05**	**3.51**

Table 3. Parallel Decoder performance for GOP 8

Sequence	BP	Reference Time per frame (s)	Parallel Time per frame (s)	TR(%)	SpeedUp
Foreman	1	5.89	1.64	72.24	3.60
	2	10.71	2.65	75.26	4.04
	3	17.94	4.32	75.94	4.16
	4	25.72	6.40	75.10	4.02
Hall	1	2.81	1.19	57.6	2.36
	2	4.10	1.53	62.75	2.68
	3	6.75	2.12	68.66	3.19
	4	10.54	2.99	71.67	3.53
CoastGuard	1	3.61	1.41	60.86	2.56
	2	7.42	2.43	67.25	3.05
	3	13.36	4.11	69.27	3.25
	4	20.15	6.42	68.14	3.14
Soccer	1	8.89	2.26	74.51	3.92
	2	15.20	3.75	75.36	4.06
	3	23.12	5.72	75.26	4.04
	4	31.16	7.70	75.28	4.05
Mean		**12.96**	**3.54**	**70.32**	**3.48**

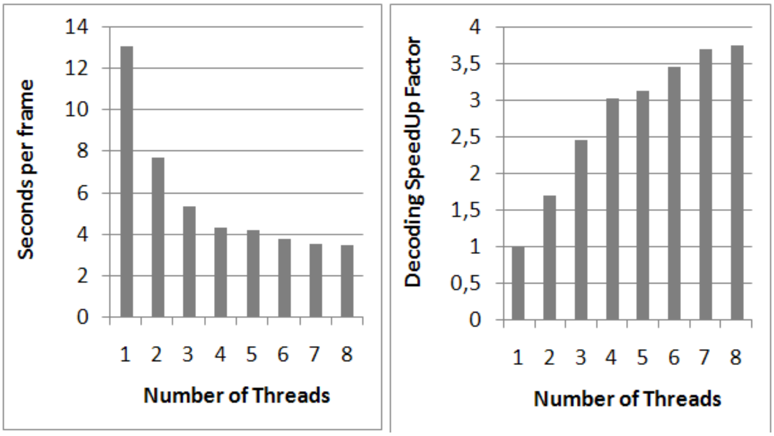


Fig. 5. DVC sequential decoding time line execution

6 Conclusions

The WZ video decoding is highly complex and this could be a problem for applications which have delay requirements. This work presents a WZ parallel decoding scheme by means of muticore processors. In this approach each GOP is decoding in an independent and parallel way, so that this scheme could be used in different WZ implementations without taking into account the implementations details of a particular approach. In addition, our proposed decoder maintains the same RD results than the sequential version. The time reduction reached is above 70% on average and it is extensible for longer GOP lengths with similar results. In spite of the feedback channel is still a bottleneck in DVC decoding, this proposed scheme could be used without modifications with more core architectures (following the current market tendency) and even in architectures without feedback channel.

Acknowledgments. This work was supported by the Spanish MEC and MICINN, as well as European Commission FEDER funds, under Grants CSD2006-00046, TIN2009-14475-C04. It was also supported by JCCM funds under grant PEII09-0037-2328 and PII2I09-0045-9916, and the University of Castilla-La Mancha under Project AT20101802. The work presented was performed by using the VISNET2-WZ-IST software developed in the framework of the VISNET II project.

References

1. ISO/IEC International Standard 14496-10:2003: Information Technology – Coding of Audio – Visual Objects – Part 10: Advanced Video Coding
2. Aaron, A., Rui, Z., Girod, B.: Wyner-Ziv coding of motion video. In: Asilomar Conference on Signals, Systems and Computers, pp. 240–244 (2002)
3. Girod, B., Aaron, A.M., Rane, S., Rebollo-Monedero, D.: Distributed Video Coding. Proceedings of the IEEE 93, 71–83 (2005)
4. Brites, C., Ascenso, J., Quintas Pedro, J., Pereira, F.: Evaluating a feedback channel based transform domain Wyner-Ziv video codec. Signal Processing: Image Communication 23, 269–297 (2008)
5. Feng, W.-C., Manocha, D.: High-performance computing using accelerators. Parallel Computing 33, 645–647 (2007)
6. Wyner, A.: Recent Results in the Shannon Theory. IEEE Trans. on Information Theory 20 (1974)
7. Artigas, X., Ascenso, J., Dalai, M., Klomp, S., Kubasov, D., Ouaret, M.: The DISCOVER codec: architecture, techniques and evaluation. In: Picture Coding Symposium (PCS), pp. 1-4. Citeseer (2007)
8. Ascenso, J., Brites, C., Dufaux, F., Fernando, A., Ebrahimi, T., Pereira, F., Tubaro, S.: The VISNET II DVC Codec: Architecture, Tools and Performance. In: European Signal Processing Conference, EUSIPCO (2010)
9. Intel Processor Core family, `http://www.intel.com/`
10. The OpenMP API specification for parallel programming, `http://openmp.org`
11. Ryanggeun, O., Jongbin, P., Byeungwoo, J.: Fast implementation of Wyner-Ziv Video codec using GPGPU. In: IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), pp. 1–5 (2010)
12. Momcilovic, S., Yige, W., Rane, S., Vetro, A.: Toward realtime side information decoding on multi-core processors. In: IEEE International Workshop on Multimedia Signal Processing (MMSP), pp. 321–326 (2010)

Fast Mode Decision Algorithm for H.264/AVC-to-SVC Transcoding with Temporal Scalability

Rosario Garrido-Cantos[1], Jan De Cock[2], Sebastiaan Van Leuven[2], Pedro Cuenca[1], Antonio Garrido[1], and Rik Van de Walle[2]

[1] Albacete Research Institute of Informatics, University of Castilla-La Mancha, Albacete, Spain
{charo,pcuenca,antonio}@dsi.uclm.es
[2] Department of Electronics and Information Systems - Multimedia Lab,
Ghent University - IBBT, Ghent, Belgium
{jan.decock,sebastiaan.vanleuven,rik.vandewalle}@ugent.be

Abstract. Scalable Video Coding (SVC) uses a notion of layers within the encoded bitstream for providing temporal, spatial and quality scalability, separately or combined. By truncating layers the bitstream can be adapted to devices with different characteristics and to varying network constraints. Since the majority of the existing video content is encoded using H.264/AVC without scalability, they cannot benefit from these scalability tools, so a transcoding process should be applied to provide scalability to this existing encoded content. In this paper, an algorithm based on Machine Learning techniques for temporal scalability transcoding from H.264/AVC to SVC focusing on mode decision task is discussed. The results show that when our technique is applied, the complexity is reduced by 82% while maintaining coding efficiency.

Keywords: Scalable Video Coding (SVC), H.264/AVC, Transcoding, Temporal Scalability, Machine Learning.

1 Introduction

The users' demand for multimedia content such as video streaming in digital video services has grown spectacularly in the last years. These video streams, generally, are encoded to reduce the necessary storage and the network bandwidth for transmission. In 2007, the scalable extension of the H.264/AVC [1][2] video coding standard was finalized. *Scalable Video Coding* (SVC) [3] provides temporal, spatial, quality scalability or a combination of these. An SVC bitstream is organized in layers (one base layer and one or more enhancement layers). The base layer represents the lowest frame rate, spatial resolution and quality resolution while the enhancement layers provide improvements allowing a higher frame rate, resolution and/or quality. The bitstream is adaptable to the channel bandwidth or the terminal capabilities by truncating the undesirable enhancement layers.

Today, most of the digital video streams are still created in a single-layer format (such as H.264/AVC video streams) so they cannot benefit from this scalability. This

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 585–596, 2012.

fact leads to the need for developing alternative techniques to enable video adaptation between non-scalable and scalable bitstreams. In this framework, transcoding approaches are one of the solutions used to adapt a video stream by reducing the temporal resolution, lowering the spatial resolution, decreasing the visual quality, or changing the coding format.

In this paper, an efficient video transcoding [4] technique is proposed for transforming H.264/AVC bitstreams to SVC bitstreams providing temporal scalability. The ultimate goal is to perform the required adaptation process faster than the straightforward concatenation of decoder and encoder while maintaining its coding efficiency. In the H.264/AVC standard and its SVC extension, inter prediction is carried out by means of variable block size motion estimation, which is able to eliminate the temporal redundancy between two or more adjacent frames. This approach supports motion compensation block sizes ranging between 16x16, 16x8, 8x16 and 8x8, where each of the sub-divided regions is a *Macroblock* (MB) partition. If the 8x8 mode is chosen, each of the four 8x8 block partitions within the MB may be further split in 4 ways: 8x8, 8x4, 4x8 or 4x4, which are known as sub-MB partitions. In the proposed approach, the reduction in complexity is obtained by reusing as much information as possible from the original bitstream, such as H.264/AVC mode decision, residual, etc. to reduce the encoding SVC time focusing on the mode decision process. This time saving is achieved by narrowing the possible modes where the standard can choose by using a decision tree built using *Machine Learning* (ML) tools. This technique is applied to different sequences using varying GOP sizes in Baseline Profile.

The remainder of this paper is organized as follows. In Sect. 2, the state-of-the-art for H.264/AVC to SVC transcoding is discussed. Sect. 3 introduces briefly the temporal scalability technique in SVC. In Sect. 4, our proposal is described. In Sect. 5 the results of applying our approach are presented and, finally, in Sect. 6 conclusions are shown.

2 Related Work

In the literature different proposals exist for using *Machine Learning* for transcoding. Some examples are [5][6].

In the framework of H.264/AVC-to-SVC video transcoding, in the last few years, different techniques have been proposed. They can be classified into three different types of scalability.

For quality-SNR scalability, in 2006 *Shen at al. proposed* a technique for transcoding from hierarchically-encoded H.264/AVC to Fine-Grain Scalability (FGS) streams [7]. In 2009, *De Cock et al.* presented different open-loop architectures for transcoding from a single-layer H.264/AVC bitstream to SNR-scalable SVC streams with *Coarse-Grain Scalability* (CGS) layers [8]. In 2010 and 2011, they proposed simple closed-loop architectures that reduce the time of the mode decision [9][10].

Regarding spatial scalability, in 2009 a proposal was presented by *Sachdeva et al.* in [11]. The idea consists of an information single layer to SVC multiple-layer for adding spatial scalability to all existing non-scalable H.264/AVC video streams. The algorithm reuses available data by an efficient downscaling of video information for different layers.

Finally, for temporal scalability, in 2008 a transcoding method from an H.264/AVC P-picture-based bitstream to an SVC bitstream was presented in [12] by *Dziri et al.* In this approach, the H.264/AVC bitstream was transcoded to two layers of P-pictures (one with reference pictures and the other with non-reference ones). Then, this bitstream was transformed to an SVC bitstream by syntax adaptation. In 2010 *Al-Muscati et al.* proposed another technique for transcoding that provided temporal scalability in [13]. The method presented was applied in the Baseline Profile and reused information from the mode decision and motion estimation processes from the H.264/AVC stream. The same year we presented an H.264/AVC to SVC video transcoder that efficiently reuses some motion information of the H.264/AVC decoding process in order to reduce the time consumption of the SVC encoding algorithm by reducing the motion estimation process time. The approach was developed for Main Profile and dynamically adapted for several temporal layers [14]. Later, in 2011, the previous algorithm was adjusted for Baseline Profile and P frames [15]. At this point, we emphasize that the present work is another step in the framework of H.264/AVC to SVC video transcoders. Our previous approaches [14][15] focused only on the motion estimation process, where the idea consists in reducing the search area dynamically based on the incoming H264/AVC motion vectors. On the contrary, in this work, while the motion estimation is kept untouched, the approach is extended to a MB mode decision algorithm which is explained in next section. As future work, we can try to combine both approaches together.

3 Temporal Scalability in SVC

Since our proposal focuses on temporal scalability, a brief explanation about this type of scalability is given in this section. For a comprehensive overview of the whole scalable extension of H.264/AVC, the reader is referred to [3].

As it was said in the introduction, in a sequence with temporal scalability, the bitstream is encoded in layers. The base layer (with an identifier equal to 0) represents the lowest frame rate while the temporal enhancement layers (with identifiers that increase by 1 in every layer) increase the available frame rate. By removing temporal layers, the frame rate can be adapted dynamically.

Fig. 1 shows a sequence with a *Group of Pictures* (GOP) of 8 encoded as four temporal layers. The base layer (layer 0) consists of frames 0 and 8 and provides 1/8 of the original frame rate. Frame 4 lies within the first enhancement temporal layer and, decoded together with layer 0, produces 1/4 of the frame rate of the full sequence. Layer 2 consists of frames 2 and 6; together with layers 0 and 1 it provides a frame rate that is 1/2 of the frame rate of the whole sequence.

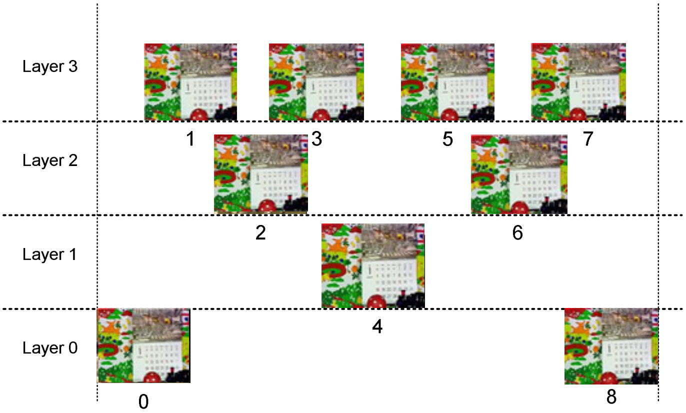


Fig. 1. Distribution per temporal layer of the frames within a GOP = 8

Temporal scalability can be achieved using P and B coding tools that are available in H.264/AVC and by extension in SVC. Flexible prediction tools were provided that make it possible to mark any picture as a reference picture, so that it can be used for motion-compensated prediction of the following pictures. In this way, to achieve temporal scalability, SVC links its reference and predicted frames using hierarchical prediction structures [16] which define the temporal layering of the final structure. As was mentioned previously, the temporal base layer represents the lowest frame rate that can be obtained. The frame rate can be increased by adding pictures of the enhancement layers. There are different structures for enabling temporal scalability, but the one used by default in the *Joint Scalable Video Model* (JSVM) reference encoder software [17] is based on hierarchical pictures with a dyadic structure where the number of temporal layers is thus equal to $1+ log_2[GOP\ size]$.

Temporal scalability based on P pictures was introduced in [18]. This technique provides lower latency and is particularly useful for multimedia communications like mobile video broadcasting or mobile digital television where the transmission of a scalable bitstream would be a good solution to address mobile terminals with several requirements.

4 Proposed H.264/AVC-to-SVC Video Transcoder

4.1 Motivation

One of the computationally most intensive tasks involved in the SVC encoding process is the MB mode decision. Therefore, this is one of the more suitable parts of the proposed H.264/AVC-to-SVC transcoder to be accelerated. H.264/AVC and its extension SVC support both intra prediction and inter prediction in P or B frames. Intra prediction only requires data from the current picture, while inter prediction uses data from pictures that have been previously coded and transmitted (reference pictures) and is used for eliminating temporal redundancy in P and B frames. As it has been depicted before, SVC supports different MB and sub-MB partitioning modes for inter prediction as shown in Fig. 2. Moreover, SVC also allows intra predicted modes, and a skipped mode in inter frames for referring to the 16x16 mode where no motion and residual information is encoded.

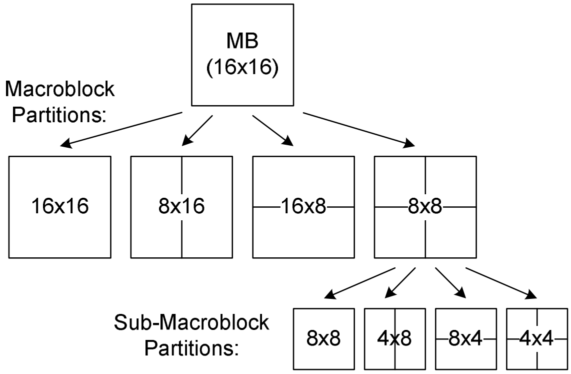
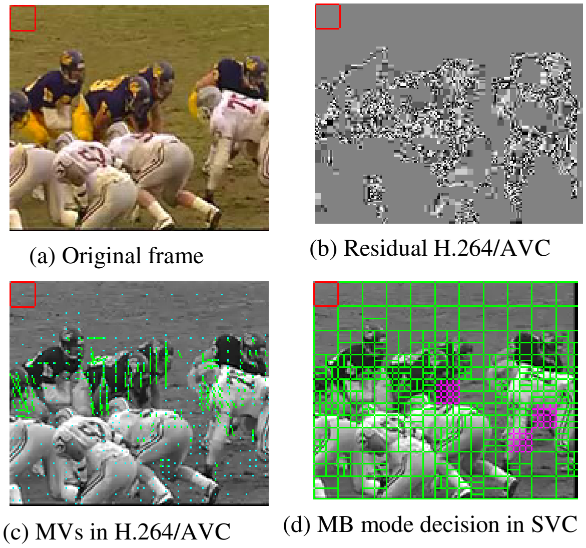


Fig. 2. Macroblock and sub-macroblock partitions for inter prediction

Since for choosing the best partitioning, every block size is checked by the SVC encoder (as part of the proposed transcoder), a way to reduce the time spent by this mode decision process is trying to narrow the set of possible MB partitions. This paper proposes an algorithm which makes use of some information gathered in the H.264/AVC decoding algorithm to reduce the MB partitions to be checked in the SVC encoder. By using a data mining procedure, an algorithm which implements a decision tree is proposed. This decision tree is generated by means of machine learning tools.

Although the prediction structure (and as result, the frames used as a reference) of H.264/AVC without temporal scalability (in this case using the well-known IPPP pattern) and SVC are not the same, some data extracted from the H.264/AVC decoding algorithm can still be reused in the transcoder. This information can help us to find out the best partitioning structure without the needed of checking all the MB partitions. This correlated information extracted from the H.264/AVC decoding algorithm is depicted in Fig. 3. In this figure, the correlation between the residual and MV length calculated in H.264/AVC with respect to the MB coded partition done in SVC are shown.

In this case, after an extensive analysis, we observed that stationary areas or objects with slow motion are often coded in MBs without sub-blocks (such as 16x16, 16x8 or 8x16) or even as Skipped where the MB contains no residual data. On the other hand, the regions with sudden changes (scene, light, an object that appears) are coded using inter modes with smaller MB mode partitions (such as 4x8, 8x4, 4x4) or even using the Intra mode. Moreover, we also found a high correlation between the length of the MVs calculated by H.264/AVC and the final MB mode decision where long MVs suggest more complicated MB partitions such as 4x4, while shorter MVs lead to simpler MB partitions. These relationships can be observed in Fig. 3 as well. Taking


(a) Original frame

(b) Residual H.264/AVC

(c) MVs in H.264/AVC

(d) MB mode decision in SVC

Fig. 3. Exploiting the correlation using Machine Learning

into account these observations, the information that needs to be extracted from the H.264/AVC decoding process is:

- Residual: The residual data of every block of 4x4 pixels is used by the decoder to reconstruct the MB, so this information will be available in the decoding process. For our purpose, only the residual data of the luma component is extracted.
- Motion vectors: This information is available as well in the decoding process. The motion vectors of each MB are extracted. Note that each MB in H.264/AVC can have more than one pair of motion vectors since each MB can be further divided in smaller partitions.
- Mode decision of H.264/AVC: The MB partitioning of each MB in H.264/AVC is related to the residual and the motion vectors and can give us valuable information.

4.2 Generating the Decision Tree

Data mining is the process of finding correlations or patterns among dozens of fields in large relational databases. This information can be converted into knowledge. Machine learning is a scientific discipline concerned with the design and development of algorithms that allow computers to evolve behaviors based on empirical data. It has the decision making ability with low computation complexity, basically, if-then-else operations. For this proposal, we used ML tools for converting the relationships between data extracted from the H.264/AVC decoding process and the MB mode partitioning of SVC into a decision tree. By using this decision tree, the possible modes that can be chosen by the encoder are narrowed down.

This decision tree was built using the WEKA software [19]. WEKA is a collection of machine learning algorithms for data mining tasks and also contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. For every MB, the extracted information is used to generate the decision tree (used to decide the MB partitioning later). Some operations and statistics are calculated for this data such as the length of the motion vectors, the variance of means of the residual of 4x4 blocks or the mean of variances of the residual of these blocks.

The information enumerated in Section 4.1 together with the SVC encoder mode decision was introduced and then, an ML classifier was run. In this case, the well-known RIPPER algorithm [20] was used. The process for building the decision tree for H.264/AVC-to-SVC transcoding is shown in Fig. 4. The obtained binary decision tree has three decision levels:

- First level: Discriminates between LOW {SKIP, 16x16, 16x8, 8x16} and HIGH COMPLEXITY {INTRA, 8x8, 8x4, 4x8, 4x4} modes.
- Second level: Inside the LOW COMPLEXITY bin, a decision between {SKIP, 16x16} or {16x8, 8x16} is made.
- Third level: Inside the HIGH COMPLEXITY bin, a decision between {8x8, 8x4, 4x8} or {4x4, INTRA} is made.

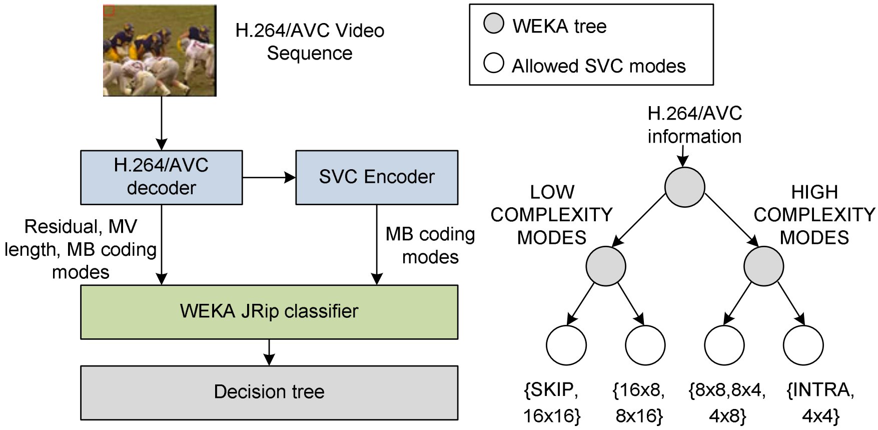


Fig. 4. Building the decision tree

This tree was generated with the information available after the decoding process and does not focus on the final MB partition, but reduces the set of MB modes that can be chosen by SVC encoder. This is represented in Fig. 4 where the white circles represent the set of MB partitions the SVC encoder can choose from. The ML process gives us a decision tree that classifies correctly in about 87% of the cases in the 1st level, 80% in the 2nd level and 93% in the 3rd level. For all this cases, training with *Football* and testing with the rest of sequences of the performance evaluation (see Section 5).

This decision tree is composed of a set of thresholds for the H.264/AVC residual and for the statistics related to it. Since the MB mode decision, and hence the thresholds, depend on the Quantization Parameter (QP) used in the H.264/AVC stage, the residual, the mean and the variance threshold will be different for each QP. The solution is to develop a single decision tree for a specific QP and adjust the mean and the variance threshold used by the trees based on the QP.

5 Implementation Results

In this section, results from the implementation of the proposal described in the previous section are shown. Test sequences with varying characteristics were used, namely *Foreman, Harbour, Mobile, City, Soccer* and *Hall* in CIF resolution (30 Hz) and QCIF resolution (15 Hz). These sequences were encoded using the H.264/AVC *Joint Model* (JM) reference software [21], version 16.2, with an IPPP pattern and a fixed QP = 28 in a trade-off between quality and bitrate. Then, for the reference results, the encoded bitstreams are decoded and re-encoded using the JSVM software [17], version 9.19.3 [17] with temporal scalability, Baseline Profile and different values of QP (28, 32, 36, 40).

For the results of our proposal, encoded bitstreams in H.264/AVC are transcoded using the technique described in the previous section. This technique was applied to the enhancement temporal layers because, as it was shown in [15], the two enhancement temporal layers with the highest identifier are where most encoding time is spent. In these results, the *Football* sequence has been excluded from the evaluation set since it was used as a training sequence for generating the decision tree.

In Table 1, 2 and 3 the results for ΔPSNR, ΔBitrate and Time Saving are shown when our technique is applied compared to the reference transcoder. ΔPSNR and ΔBitrate are calculated according to the *Bjøntegaard-Delta* metric [22].

Time Savings are calculated for the full sequence (‘Full Seq.’) and for the temporal layers where the technique is applied to (‘Partial’). To evaluate it, (1) is calculated where T_{ref} denotes the coding time used by the SVC reference software encoder and T_{pro} is the time spent by the proposed algorithm.

$$T_{saving}\,(\%) = 100 \cdot (T_{ref} - T_{prop})/T_{ref} \qquad (1)$$

Table 1. RD performance and time savings of the approach for GOP = 2 and different resolutions

RD performance and time savings of H.264/AVC-to-SVC transcoder
GOP = 2

	QCIF (15 Hz)				*CIF (30 Hz)*			
Sequence	**ΔPSNR (dB)**	**ΔBitrate (%)**	**Time Saving (%)**		**ΔPSNR (dB)**	**ΔBitrate (%)**	**Time Saving (%)**	
			Full Seq.	***Partial***			***Full Seq.***	***Partial***
Hall	0.042	-0.05	57.96	85.38	0.055	-0.08	58.94	86.64
City	0.026	0.92	57.16	84.16	0.055	0.25	58.24	85.61
Foreman	0.077	1.21	56.20	82.70	-0.059	1.51	58.12	85.46
Soccer	0.036	1.45	54.34	79.86	0.021	1.28	56.28	82.85
Harbour	0.022	-0.13	52.91	77.95	0.047	-0.35	56.12	80.58
Mobile	0.033	-0.15	52.28	76.93	0.080	-1.10	54.51	80.09
Average	***0.039***	***0.54***	***55.14***	***81.16***	***0.033***	***0.25***	***57.03***	***83.54***

ΔPSNR: Difference in quality (negative means quality loss); ΔBitrate: Bitrate increase;
Time Saving: complexity reduction.

The values of PSNR and bitrate obtained with the proposed transcoder are very close to the results obtained when applying the reference transcoder (re-encoder) while around 65% of reduction of computational complexity in the full sequence and 82% in the specific layers is achieved. The resulting Rate-Distortion (RD) curves for the CIF SVC bitstream with GOP = 4 is shown in Fig. 5 where it can be seen that our proposal for transcoding is able to approach the RD of the reference transcoded (re-encoded) without any significant coding efficiency loss. Due to the space restriction, only one RD plot can be shown; the rest of the figures are similar.

Table 2. RD performance and time savings of the approach for GOP = 4 and different resolutions

RD performance and time savings of H.264/AVC-to-SVC transcoder GOP = 4								
QCIF (15 Hz)					*CIF (30 Hz)*			
Sequence	**ΔPSNR (dB)**	**ΔBitrate (%)**	**Time Saving (%)**		**ΔPSNR (dB)**	**ΔBitrate (%)**	**Time Saving (%)**	
			Full Seq.	***Partial***			***Full Seq.***	***Partial***
Hall	0.219	0.04	74.58	85.80	0.328	-0.45	74.69	86.45
City	0.064	1.93	75.69	86.04	0.200	0.66	76.30	86.96
Foreman	0.251	2.34	72.68	83.55	-0.112	3.01	74.63	85.65
Soccer	0.043	2.24	72.11	81.83	0.021	2.37	72.35	83.05
Harbour	0.107	-0.68	68.30	78.88	0.175	-1.22	71.75	81.57
Mobile	0.142	0.15	65.37	76.51	0.229	-1.69	69.83	80.37
Average	***0.138***	***1.00***	***71.46***	***82.10***	***0.140***	***0.45***	***73.26***	***84.01***

Table 3. RD performance and time savings of the approach for GOP = 8 and different resolutions

RD performance and time savings of H.264/AVC-to-SVC transcoder GOP = 8								
QCIF (15 Hz)					*CIF (30 Hz)*			
Sequence	**ΔPSNR (dB)**	**ΔBitrate (%)**	**Time Saving (%)**		**ΔPSNR (dB)**	**ΔBitrate (%)**	**Time Saving (%)**	
			Full Seq.	***Partial***			***Full Seq.***	***Partial***
Hall	0.158	0.37	70.59	86.28	0.025	0.47	70.69	86.83
City	-0.008	2.67	70.16	85.70	0.175	1.32	70.10	86.16
Foreman	0.210	3.22	66.89	82.89	-0.001	3.58	69.96	85.91
Soccer	0.074	2.61	65.19	80.63	-0.001	2.99	68.07	83.55
Harbour	0.048	0.15	64.60	79.54	0.072	-0.18	65.54	80.60
Mobile	0.031	0.87	64.82	79.36	0.233	-0.84	65.81	81.10
Average	***0.086***	***1.65***	***67.04***	***82.40***	***0.084***	***1.22***	***68.36***	***84.02***

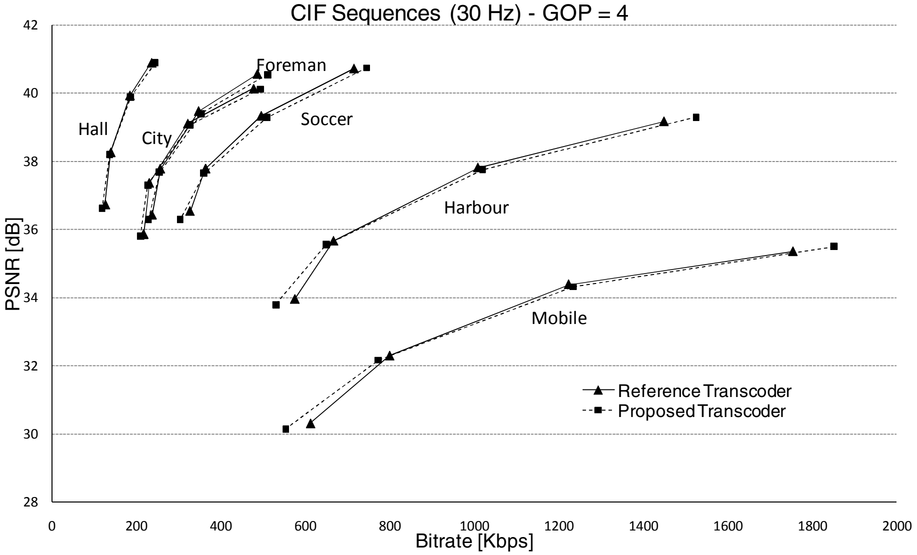


Fig. 5. Rate-distortion performance of CIF sequences with GOP = 4

Finally, our technique is capable of outperforming earlier solutions such as in [12][13][15]. In contrast to [12], we show that our proposal can be successfully applied to a wide range of test sequences with varying motion characteristics and resolutions. A comparison with these proposals is shown in Table 4. This comparison is done with the values available in the papers (PSNR and Time Saving for Foreman CIF with GOP = 2 and an average of PSNR and Time Saving for different sequences in QCIF and CIF resolutions and GOP = 8). Regarding ΔBitrate, there is not numerical information in [12] and [13] and comparing to [15] the ΔBitrate is similar.

Table 4. Comparison between different proposals

Comparison with other proposals						
	GOP = 2 ***CIF - Foreman***		***GOP = 8*** ***QCIF***		***GOP = 8*** ***CIF***	
Proposal	**ΔPSNR (dB)**	**TS (%)**	**ΔPSNR (dB)**	**TS (%)**	**ΔPSNR (dB)**	**TS (%)**
Dziri et al. [12]	-0.50	47.00	--	--	--	--
Al-Muscati et al. [13]	-0.50	37.00	-0.200	55.20	--	62.10
Garrido-Cantos et al. [15]	-0.01	41.79	-0.027	51.90	-0.040	48.83
Our technique	**-0.06**	**58.12**	**0.086**	**67.04**	**0.084**	**68.36**

ΔPSNR: Difference in quality (negative means quality loss); TS: complexity reduction.

6 Conclusions

In this paper, a proposal based on Machine Learning tools for transcoding H.264/AVC bitstreams to SVC streams with temporal scalability has been presented.

This scalability makes it possible to adapt the video contents to different mobile devices regarding frame rate. Moreover, by applying our proposal, the complexity of the macroblock mode decision process is reduced. The experimental results show that it is capable to reduce the coding complexity by around 82% where it is applied while maintaining the coding efficiency.

Acknowledgments. This work was supported by the Spanish MEC and MICINN, as well as European Commission FEDER funds, under the grants CSD2006-00046, TIN-2009-14475-C04, and it was also partly supported by JCCM funds under grants PEII09-0037-2328 and PII2I09-0045-9916.

References

1. ITU-T and ISO/IEC JTC 1: Advanced Video Coding for Generic Audiovisual Services. In: ITU-T Rec. H.264/AVC and ISO/IEC 14496-10 (including SVC extension) (March 2009)
2. Wiegand, T., Sullivan, G.J., Bjontegaard, G., Luthra, A.: Overview of the H.264/AVC Video Coding Standard. IEEE Transaction on Circuits and System for Video Technology 13(7), 560–576 (2003)
3. Schwarz, H., Marpe, D., Wiegand, T.: Overview of the Scalable Video Coding Extension of the H.264/AVC Standard. IEEE Transactions on Circuits and Systems for Video Technology 17(9), 1103–1120 (2007)
4. Vetro, A., Christopoulos, C., Sun, H.: Video Transcoding Architectures and Techniques: an Overview. IEEE Signal Processing Magazine, 18–29 (2003)
5. Martinez, J.L., Fernandez-Escribano, G., Kalva, H., Fernando, W.A.C., Fernando, Cuenca, P.: Wyner-Ziv to H.264 Video Transcoder for Low Cost Video Communications. IEEE Transaction on Consumer Electronics 55(3), 1453–1461 (2009)
6. Fernandez-Escribano, G., Bialkowski, J., Gamez, J.A., Kalva, H., Cuenca, P., Orozco-Barbosa, L., Kaup, A.: Low-Complexity Heterogeneous Video Transcoding Using Data Mining. IEEE Transactions on Multimedia 10(2), 286–299 (2008)
7. Shen, H., Sun, X.S., Wu, F., Li, H., Li, S.: Transcoding to FGS Streams from H.264/AVC Hierarchical B-Pictures. In: IEEE Int. Conf. Image Processing, Atlanta (2006)
8. De Cock, J., Notebaert, S., Lambert, P., Van de Walle, R.: Architectures of Fast Transcoding of H.264/AVC to Quality-Scalable SVC Streams. IEEE Transaction on Multimedia 11(7), 1209–1224 (2009)
9. Van Wallendael, G., Van Leuven, S., Garrido-Cantos, R., De Cock, J., Martinez, J.L., Lambert, P., Cuenca, P., Van de Walle, R.: Fast H.264/AVC-to-SVC transcoding in a mobile television environment. In: Proceedings of Mobile Multimedia Communications Conference, 6th International ICST, Lisbon (2010)
10. Van Leuven, S., De Cock, J., Van Wallendael, G., Van de Walle, R., Garrido-Cantos, R., Martinez, J.L., Cuenca, P.: Combining Open- and Closed-loop Architectures for H.264/AVC-to-SVC Transcoding. In: 18th IEEE International Conference on Image Processing (in press)
11. Sachdeva, R., Johar, S., Piccinelli, E.: Adding SVC Spatial Scalability to Existing H.264/AVC Video. In: 8th IEEE/ACIS International Conference on Computer and Information Science, Shangai (2009)

12. Dziri, A., Diallo, A., Kieffer, M., Duhamel, P.: P-Picture Based H.264 AVC to H.264 SVC Temporal Transcoding. In: International Wireless Communications and Mobile Computing Conference (2008)
13. Al-Muscati, H., Labeau, F.: Temporal Transcoding of H.264/AVC Video to the Scalable Format. In: 2nd Int. Conf. on Image Processing Theory Tools and Applications, Paris (2010)
14. Garrido-Cantos, R., De Cock, J., Martínez, J.L., Van Leuven, S., Cuenca, P., Garrido, A., Van de Walle, R.: Video Adaptation for Mobile Digital Television. In: IFIP Wireless and Mobile Networking Conference, Budapest, Hungary (2010)
15. Garrido-Cantos, R., De Cock, J., Martínez, J.L., Van Leuven, S., Cuenca, P.: Motion-Based Temporal Transcoding from H.264/AVC-to-SVC in Baseline Profile. IEEE Transactions on Consumer Electronics 57(1) (February 2011)
16. Schwarz, H., Marpe, D., Wiegand, T.: Analysis of Hierarchical B pictures and MCTF. In: IEEE Int. Conf. ICME and Expo, Toronto (2006)
17. Joint Video Team (JSVM) reference software, `http://ip.hhi.de/imagecom_G1/savce/downloads/SVC-Reference-Software.htm`
18. Wenger, S.: Temporal scalability using P-pictures for low-latency applications. In: IEEE Second Workshop on Multimedia Signal Processing, Redondo Beach, CA, USA, pp. 559–564 (December 1998)
19. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations 11(1) (2009)
20. Cohen, W.W.: Fast Effective Rule Induction. In: 20th International Conference on Machine Learning, pp. 115–123 (1995)
21. Joint Model JM reference software, `http://iphome.hhi.de/suehring/tml/download/`
22. Sullivan, G., Bjøntegaard, G.: Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low-Resolution Progressive-Scan Source Material. ITU-T VCEG, Doc. VCEG-N81 (September 2001)

Asserting the Precise Position of 3D and Multispectral Acquisition Systems for Multisensor Registration Applied to Cultural Heritage Analysis

Camille Simon[1,2], Rainer Schütze[2], Frank Boochs[2], and Franck S. Marzani[1]

[1] le2i, Université de Bourgogne, Dijon, France
{camille.simon,franck.marzani}@u-bourgogne.fr
[2] i3mainz, Mainz University of Applied Sciences, Mainz, Germany
{camille.simon,rainer.schuetze,boochs}@geoinform.fh-mainz.de

Abstract. We present a novel method to register multispectral acquisitions on a 3D model. The method is based on the external tracking of the acquisition systems using close-range photogrammetric techniques: multiple calibrated cameras simultaneously observe the successive acquisition systems in use. The views from these cameras are used to precisely determine the position of each acquisition system. All datasets can then be projected in the same coordinate system. The registration is thus independent from the quality and content of the data. This method is well suited to the study of cultural heritage or any other application where we do not wish to place targets on the object. We describe the method and the simulation pipeline used to find an adequate setup for two case studies.

Keywords: 2D-3D registration, close range photogrammetry, optical calibration, 3D digitization, multispectral acquisitions, cultural heritage.

1 Introduction

The analysis of cultural heritage objects relies on multiple techniques of which contactless analysis techniques are favored. Two such optical techniques are multispectral imaging and 3D digitization. Both are increasingly used to document and analyze cultural heritage objects. Multispectral imaging systems are used to produce more faithful color reproductions [2], for pigment identification [10] or to decipher overwritten text in palimpsests [5] for example. On the other hand, 3D models can be used to observe the surface structure of an object without manipulating it. This is useful both for conservators and for communication purposes. Art scholars can examine fine brushstrokes on paintings and chisel marks of statues. 3D models can also be used to create virtual museums and as virtual archives of an object. 3D digitization and multispectral imaging provide complementary data and it is interesting for conservators to be able to visualize both datatypes in an integrated frame. 3D models with multispectral texture can also be used for web-museums or enhanced reality applications.

Systems which simultaneously perform the 3D digitization on an object and acquire multispectral texture [9,14] generally do not reach the resolution each system can achieve independently. These systems are also very bulky and not transportable. Using

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 597–608, 2012.

separate systems for the multispectral acquisitions and the 3D digitization of a given object we benefit from the high resolution each acquisition system offers. We can also choose and adapt each acquisition system independently to the present application. This approach, however, requires to register the acquisitions in a single coordinate system. We develop a registration strategy based on close-range photogrammetry techniques to precisely asses the position and orientation of each acquisition system during its use. This paper presents a description of the registration strategy and simulation results for two real scenarios. The strengths of our multimodal registration method are:

- A method suitable for registering data with no salient features
- A registration precision independent from the content of the acquired data
- A registration method which works for many different optical sensors
- A flexible solution suitable for many different applications

2 Related Work

2D-3D registration. If the various bands that form a multispectral acquisition have been properly calibrated and registered, registering a multispectral acquisition on a 3D model is equivalent to image to 3D registration. Most 2D-3D registration techniques have the same global setup: first estimate the external camera parameters (position and orientation) of the 2D acquisition system as well as its focal length and possibly other internal parameters (lens distortion, principle point, etc); then use these parameters to project the image on the 3D model.

If we have a set of corresponding points in the two datasets, a calibration method such as Tsai's [15] can be used to estimate the camera parameters. These pairs of 2D-3D points can be natural features, or targets that are added to the scene to guide the registration. There are several drawbacks to this approach: Manually detecting corresponding points is time-consuming, yet the accuracy of the registration depends on this task. Also, image vision algorithms can rarely be used to automatically detect corresponding points due to the very different data structure. When studying cultural heritage objects we have the additional problem that often few natural salient features are present in both the 2D and 3D datasets, yet we can not use targets as they may damage the fragile and unique surfaces.

Another strategy to estimate the camera parameters is based on the maximization of mutual information developed in the late 1990s [8,16]. Here we compare successive views of the 3D model to the image and iteratively compute the camera parameters. The precision of the ensuing registration is of the order of a few pixels, though the success of such methods greatly depends on the rendering strategy (depth map, silhouette, illumination related, etc.) as illustrated in [4].

Our approach differs in that we evaluate the camera parameters using photogrammetric techniques, instead of interpolating them from the acquired data.

Photogrammetric Tracking. Photogrammetric tracking is used in industrial settings for the real-time calibration of robot arms. Two setups exist: either a calibrated camera is fixed to the arm and observes the background which has been covered with targets [6], or a target object is attached to the robot arm which is observed by several photogrammetric cameras surrounding the scene [7,13]. Our work is based on the second

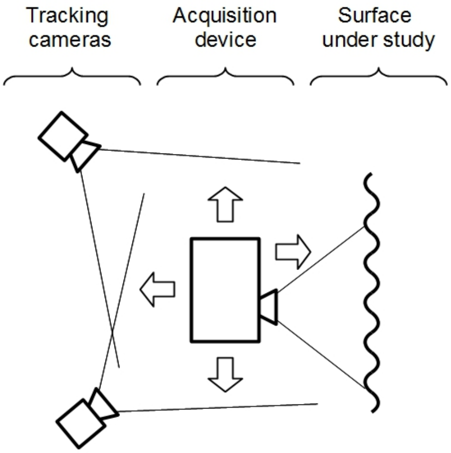


Fig. 1. Setup of the on-site acquisitions: a group of photogrammetric cameras observe each acquisition device while it images the surface from several positions

approach: multiple photogrammetric cameras observe an acquisition system defined by several targets. A similar setup has been used in [3] to guide the 3D registration procedure of a high precision scanner using a second scanner. We extend the scope of this setup to multimodal registration and demonstrate its flexibility and adaptability to several acquisition settings.

3 Material and Methods

3.1 Acquisition Pipeline

Fig. 1 represents the in-situ acquisition setup: a group of photogrammetric/tracking cameras observe the acquisition devices as they successively digitize the surface under study from various positions. Several calibration and acquisition steps are necessary before, during and after the acquisition process to obtain a precise registration.

Pre-Processing. The following two steps must be performed independently for each acquisition system, either before or after the on-site acquisitions.

- Acquisition system characterization: the relative position of the targets on the acquisition system is measured by taking multiple photos of the acquisition system with several surrounding targets as well as at least one scale bar.
- Acquisition system calibration: a calibration procedure provides us with the interior camera parameters of each acquisition system.

On-Site Acquisitions. The following steps must be performed once the photogrammetric cameras are set up to observe the acquisition devices in all their planned positions.

- Calibration of the photogrammetric cameras: the standard procedure of taking several simultaneous images of a target plate in many positions provides us with the interior and exterior camera parameters.
- Data acquisitions: simultaneously from the acquisition system in use and all tracking cameras.

Data-Processing. We now have the data necessary to perform the registration. This is done in two steps:

- The photogrammetric cameras interior and exterior orientation, the acquisition system characterization and the view of the acquisition system by the tracking cameras during the acquisition process are used to compute the position and orientation of each acquisition system for each acquisition.
- The acquisition system calibration and the position and orientation of each acquisition system in the same coordinate system enable us to project all the object data in a common coordinate system.

The precision of the final registration depends on many parameters such as the number, focal length, position and sensor of the photogrammetric cameras; the dimension of acquisition area, etc. Simulations enable us to test many configurations and evaluate how precisely we can detect the position and orientation of each acquisition system.

3.2 Simulation Pipeline

We use a three stage simulation pipeline to predict how precisely we can track the position of each acquisition system. We start by creating a scene under 3ds Max which contains the object under study, an acquisition device in different positions and the photogrammetric cameras. Each acquisition device is simplified by a box modelized by a variable number of points. These points represent the targets that we will attach to our acquisition device. The scene is exported as a *.WRL file and read by a lab-developed software. This software calculates the images seen from each camera as given by the focal length defined in 3ds Max. Lens distortion parameters can be entered manually. This software is also used to add Gaussian noise to the following four scene parameters:

Object Coordinates: The coordinates of the targets that define the object in the coordinate system defined by the photogrammetric cameras. These are usually known with a precision of 0.1 mm or 0.05 mm.

Picture Coordinates: The coordinates of the targets in the images taken by the photogrammetric cameras. These are usually known with a precision of 1/10 of a pixel. If the conditions are good (sufficient contrast and focus, well resolved targets) the picture coordinates can be resolved with an accuracy of 1/30 of a pixel.

Camera Translation: The position of the photogrammetric cameras (X, Y, Z coordinates). The accuracy of these values depends on the camera setup and the results of the calibration procedure. We can usually resolve them with an accuracy better than 0.05 mm.

Camera Rotation: The orientation of the photogrammetric cameras (Ω, Φ , K coordinates). We usually know these coordinates with an accuracy better than 0.05 mrad though this also depends on the success of the camera calibration.

The resulting data is then exported as a *.axo file to be treated by a lab-developed software based on the AXOri library [1]. This library can perform the inverse calculation of the camera positions, their interior orientation or the object position, depending on the input parameters.

3.3 Acquisition Systems

Multispectral cameras. The multispectral acquisitions are performed either by a lab-designed multispectral camera or with a commercial camera from FluxData (FD-1665-MS). A few characteristics of both cameras are given in the top portion of Table 1. The lab-designed mulitpsectral camera is based on a filter-wheel and has been used in the past to document these objects. Careful calibration and a neural network algorithm provide us with a reflectance spectra for each pixel [12]. On the other hand, the FluxData camera is based on a 3 CCD system which provides simultaneous data for each spectral band. This camera acquires less spectral bands than the lab-designed multispectral camera, but the bigger sensor and pixel size will allow us to register the data more precisely.

3D digitizing system. The digitizing of the surface is carried out by a commercial fringe projection system (Atos III, manufactured by GOM). The system can digitize a field of view of 500×500 mm^2 with a resolution of 0.25 mm. A smaller field of view of 150×150 mm^2 can be acquired with a resolution of 0.07 mm.

Photogrammetric cameras. The grayscale cameras we use have a 5 megapixel sensor (AVT Stingray F-504B). This 2/3" sensor is large enough to ensure good results, while still being compatible with good quality optics (we use an 8 mm Pentax lens).

3.4 Objects under Study

This work stems from the need to study both the surface structure and the spectral properties of two cultural heritage objects: a sandstone sarcophagus and a wall painting.

We are interested in monitoring the deterioration of the surface of a polychrome sandstone sarcophagus from the 3rd century A.D. This sarcophagus is in a crypt under the Friedhofs chapel of the St. Matthias abbey in Trier (Germany). The sarcophagus was discovered by archaeologists approximately fifty years ago. Unfortunately, the airflow and humidity in the crypt has been damaging the surface and fragmentary remains of polychromy. The area of the sarcophagus facing the entrance of the chamber is very damaged. Traces of polychromy on the surface of the sarcophagus are flaking while the stone itself erodes. We study an area of approximately 40×70 cm^2 on this face. Our goal is to precisely localize and quantify the structural and spectral degradation of the sandstone and polychromy in this zone. The need to precisely register the two datasets stems naturally from the will to find correlations in the structural and spectral aging of the surface.

We also monitor a 16^{th} century wall painting located in the Brömser Hof in Rüdesheim (Germany). In 2008 this wall painting was partially restored. Regular acquisition campaigns on both the restored and non-restored surfaces are being carried out to compare the aging of these two areas. Once again, the changes that can arise over time are both spectral and structural. This accounts for the complementary acquisition techniques and the need to register the data an integrated dataset.

4 Results and Discussion

The simulation proceeds in three steps: First we optimize the camera arrangement for a given number of cameras observing the acquisition system. This is done while adding noise only to the picture coordinates. Once an optimized setup has been found, we simulate the camera calibration. This tells us how precisely we can expect to evaluate the interior and exterior camera parameters on site. We use the output values to define reasonable noise to add to the camera parameters in the next step. We can then simulate the full tracking procedure with realistic noise. If we do not reach our target goal at this stage, there are other parameters that can be tweaked, such as the number of targets that define the acquisition system. The full strategy is illustrated in detail in the case of the sarcophagus. For the wall painting configuration, we only present a few intermediate results.

4.1 Sarcophagus

Accuracy goal. We assume the acquisition system is 50 cm from the acquisition surface of the sarcophagus. The field of view and pixel size at this distance for each multispectral camera are given in the second section of Table 1. We need 3×6 acquisitions to cover the full area of interest with sufficient overlap using the lab-designed multispectral camera (and 4×8 acquisitions using the FluxData multispectral camera). We do not perform the simulations for every 18 (respectively 32) positions. Instead, we calculate the achieved accuracy for the four corners of the rectangle thus defined, as well as for the central position. All results correspond to the worst spatial accuracy in X, Y or Z and the worst angular accuracy achieved in Ω, Φ or K over all the test positions.

Our goal is to register the multispectral data on the 3D model with an accuracy of at least half a pixel. We must thus track the imaging systems with an accuracy better than half a pixel of the multispectral camera in use. This constraint is harder to achieve with the lab-designed multispectral camera which has smaller sensor cells (see the first section of Table 1). It is also generally much harder to reach the desired angular accuracy than the desired spatial accuracy. Our goal is thus to detect the lab-designed multispectral camera with an angular accuracy of 0.128 mrad. Given the size of the area of interest, we would like to reach this target value using no more than four photogrammetric cameras.

Optimizing the camera arrangement. In these simulation runs the acquisition device is modelized by a box defined by 26 points. The dimensions of the box are that of the lab-designed multispectral camera. During this simulation phase we only add noise to the picture coordinates, with a standard deviation of 1/10 of a pixel (0.345 μm). The successive setups are described bellow. As can be seen in the corresponding simulation results (Table 2), varying the camera positions does not greatly alter the results.

Table 1. Characteristics of the multispectral images and simulation goal to detect each acquisition systems with half a pixel accuracy. The target value is typeset in boldface.

	Object – sensor distance	Lab-designed multispectral camera	FluxData multispectral camera	
Sensor size		1392 × 1040	659 × 494	(pixels^2)
Cell size		6.45	9.9	(µm)
Focal length		25	25	(mm)
Angular accuracy goal		**0.128**	0.198	(mrad)
Field of view		178 × 134	130 × 98	(mm^2)
Pixel size on object	0.5 m	0.129	0.198	(mm)
Spatial accuracy goal		0.064	0.099	(mm)
Field of view		641 × 482	469 × 352	(mm^2)
Pixel size on object	1.8 m	0.464	0.713	(mm)
Spatial accuracy goal		0.232	0.356	(mm)

Table 2. Optimizing the camera arrangement to track the lab-designed multispectral camera in front of the sarcophagus. The best results are typeset in boldface.

Camera Arrangement	Results		Mean number of visible points per camera			
	Spatial (mm)	Angular (mrad)	Camera 1	Camera 2	Camera 3	Camera 4
(a)	0.021_8	0.326	19	19	18.2	16.6
(b)	0.021_2	0.334	19	19	18.2	18.2
(c)	$\mathbf{0.021_0}$	**0.300**	19	18.8	17.4	17.4
(d)	0.024_0	0.342	19	19	17.4	17.4
(e)	$\mathbf{0.020_0}$	**0.312**	19	19	17.4	17.4

(a) Initial setup created by placing the cameras roughly at 90° angles (top row of Fig. 2). This configuration is based on the authors experience as well as general guidelines in close range photogrammetry such as those given by [11]. We notice that camera 4 is not very well placed, as it only sees a mean of 16.6 points over the five positions. Since the lowest position of the acquisition device is on the ground, it is not possible to place the bottom cameras as low as we would like to. The bottom tracking cameras are constrained to 10 cm to 20 cm above the ground and thus detect less points than the top two. On the other hand, raising the top two cameras would reduce the number of points that are well detected by all cameras, an essential factor for a stable configuration.

(b) Based on setup (a), cameras 2 and 3 are placed symmetrically to cameras 1 and 4 with respect to the y-z plane. As could be expected the results are worse, though more points are detected than in the previous setup.

(c) Based on setup (a), cameras 1 and 4 are positioned symmetrically to cameras 2 and 3 with respect to the y-z plane. The results are greatly improved, though camera 2 does not see as many points as it could.

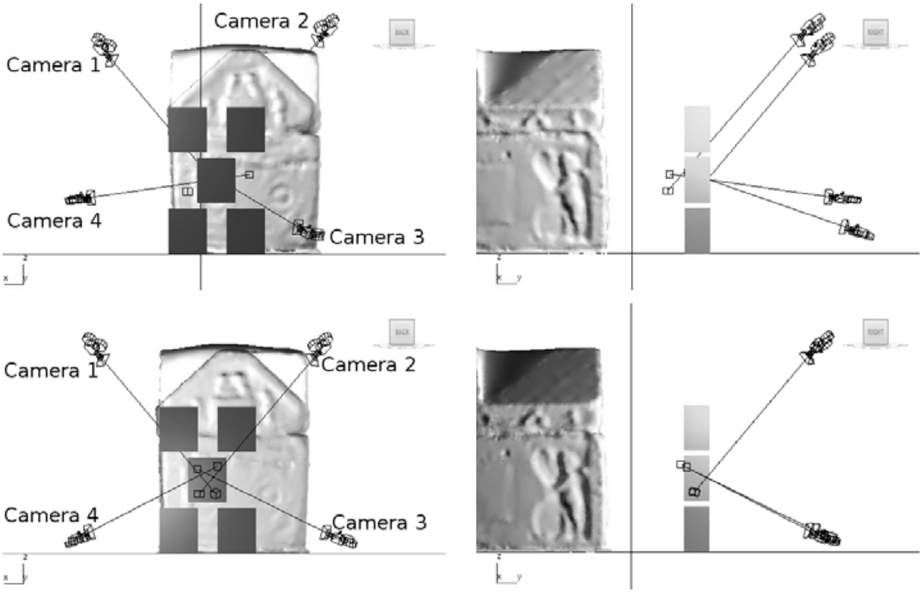


Fig. 2. Top row: view of camera arrangement (a), first configuration. Bottom row: view of camera arrangement (e), optimized configuration.

Table 3. Orientation results for the sarcophagus configuration

Noise		Results		Characteristics of next input noise		
Picture Coord.		Spatial	Angular	Camera Transl.	Camera Rot.	Description
(pixel = µm)		(mm)	(mrad)	(mm)	(mrad)	
1/10	0.345	0.021_0	0.030	0.03	0.04	Strong constraints
1/30	0.115	0.007_0	0.010	0.01	0.02	Low constraints

(d) Cameras 1 and 2 from setup (b), cameras 3 and 4 from setup (c). Taking what seems like the best from the two previous setups surprisingly gives worse results.

(e) Based on setup (c), we increase the perpendicularity of the intersections (bottom row of Fig. 2). The angular results are not increased but camera 2 detects more points. We thus base the following simulations on this setup.

Orientation. To define how precisely we can detect the position and orientation of the cameras during a real calibration we simulate positioning several target fields in the area between the cameras and the sarcophagus. These simulations are also performed by adding noise only to the picture coordinates. We evaluate how accurate the calibration is if the picture coordinates are resolved with a precision of 1/10 of a pixel and 1/30 of a pixel. Adding a reasonable margin to these results defines a realistic amount of noise to use on the subsequent simulations (see Table 3) .

Table 4. Simulation results for the sarcophagus configuration. The results that are better than our target angular accuracy are typeset in boldface.

	Noise parameters				Results	
Object	Picture Coord. (pixel = µm)	Object Coord. (mm)	Camera Transl. (mm)	Camera Rot. (mrad)	Spatial (mm)	Angular (mrad)
Lab MSC					0.020_0	0.312
FluxData MSC	$1/30$ 0.115	0.05	0.01	0.02	0.020_0	0.580
Gom Atos III					0.021_6	0.252
Frame 1	$1/30$ 0.115	0.05	0.01	0.02	0.019_2	0.132
Frame 2					0.014_0	**0.100**
Frame 2	$1/30$ 0.115	0.1	0.01	0.02	0.026_2	0.184
	$1/10$ 0.345	0.1	0.03	0.04	0.032_8	0.230

Table 5. External dimensions of the acquisition systems

Acquisition system	Width (mm)	Height (mm)	Depth (mm)
Lab designed multispectral camera	270	320	180
FluxData multispectral camera	92	112	187
GOM Atos III 3D digitization system	490	170	300

Simulations with realistic noise. We apply the lowest realistic noise (strong constraints) to three boxes of the dimension as the acquisition systems (these dimensions are given Table 5) defined by 26 points. The results are given in the top section of Table 4. The target spatial accuracy is reached for all three acquisition systems and its value is hardly influenced by the the dimension of the acquisition system. The target angular accuracy on the other hand is not reached and depends on the size of the acquisition device. Big objects are tracked with a better angular accuracy than small objects. If our acquisition systems were large enough, we could thus detect them with the angular accuracy we seek. We can enlarge the acquisition systems by fixing them to a three-dimensional frame which will also support the targets. We thus evaluate how precisely we can detect a $500 \times 500 \times 500\,\text{mm}^{3}$ cube covered with 26 targets (Frame 1) or 56 targets (Frame 2). The second section of Table 4 shows that this higher amount of targets is necessary to reach the desired angular accuracy. The third section of Table 4 shows that we need to ensure the best acquisition conditions possible to reach our goal: if we increase the noise added to the parameters, we no longer reach the target angular accuracy.

4.2 Wall Painting

Our goal is once again to register the data with an accuracy better than half a pixel. We are interested in an area that is $2 \times 1.5\ \text{m}^2$. We assume that the multispectral cameras

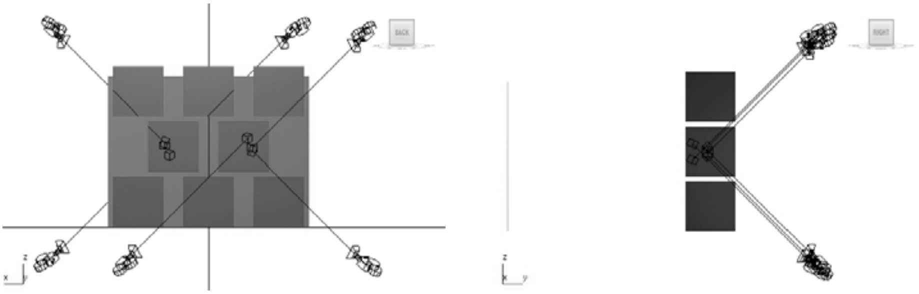


Fig. 3. View of the optimized camera arrangement in front of the wall painting

are 1.8 m from the wall surface. At this distance, we need 16 acquisitions to cover the full area of interest with the lab-designed multispectral camera and 25 acquisitions to cover the full area of interest with the FluxData multispectral camera.

We start by optimizing the camera arrangement. These simulations are performed using Frame 2. The optimized arrangement with four cameras (see the top section of Table 6) has an angular accuracy which is very far from our target goal, even though we only apply noise to the picture coordinates. We thus use 6 cameras to track the frame in front of the area of interest. Since we know that using more than four cameras to observe the same area does not greatly improve the tracking accuracy, we divide the observed area in two overlapping zones, each of which is observed by 3 cameras. The optimized arrangement is illustrated Fig. 3. The angular accuracy thus achieved is greatly improved.

We now evaluate how precisely the camera calibration can be performed for this setup. The spatial results are a factor two worse than those achieved for the setup in front of the sarcophagus while the angular results are only slightly worse (see the second section of Table 6).

As previously, we use these results to define realistic noise to add to the full setup. Once again, we reach our angular accuracy goal only if we can detect the picture coordinates with an accuracy of 1/30 of a pixel and the object coordinates with an accuracy of 0.05 mm. These are strong but achievable constraints.

The final results are valid if we move the full setup (photogrammetric cameras and acquisition systems) closer to the wall painting. However, as we move closer to the wall painting, the size of a pixel on the object decreases. If the acquisition system is too close to the object the constraining value to reach our accuracy goal of half a pixel will be the spatial accuracy instead of the angular accuracy. In the final setup, the spatial accuracy reached is 0.0152 mm. This value is larger than half a pixel as long as the distance between the wall painting and the lab-designed multispectral camera is more than 12 cm. This distance is even smaller for the FluxData multispectral camera. Using 6 photogrammetric cameras we can thus track our acquisition systems with a precision better than half a pixel in front of any area of 2×1.5 m^2 that is at least 12 cm away from the object.

Table 6. Simulation results for the wall painting configuration

	Number of cameras	Noise parameters				Results	
		Picture Coord. (pixel = µm)	Object Coord. (mm)	Camera Trans. (mm)	Camera Rot. (mrad)	Spatial (mm)	Angular (mrad)
Camera positioning	4	1/10 0.345	0	0	0	0.013_8	0.208
	6					0.016_4	0.116
Orientation	6	1/10 0.345	0	0	0	0.040_8	0.036
	6	1/30 0.115	0	0	0	0.013_8	0.012
Full results	6	1/10 0.345	0.1	0.05	0.04	0.029_2	0.200
	6	1/30 0.115	0.1	0.02	0.02	0.023_0	0.156
	6	1/30 0.115	0.05	0.02	0.02	0.015_2	**0.106**

5 Conclusion and Future Work

Simulations show that we can to evaluate the position and orientation of an acquisition system in front of an area of 40×70 cm^2 with an angular accuracy of 0.100 mrad and a spatial accuracy of 0.05 mm using four cameras photogrammetric cameras. Using six cameras we reach comparable results (0.106 mrad angular accuracy and 0.015 mm spatial accuracy) for an area of 2×1.5 m^2. These configurations will enable us to project multispectral data (or any 2D information) on a 3D model with an accuracy better than half an image pixel.

These simulation results will be validated through a series of lab tests. Then, we will test our technique on the two cultural heritage objects which motivated this study. Though only very specific results are given, this technique is widely adaptable to other setups and different constraints. Furthermore, the technique works independently from the acquisition systems used, as long as they are based on optical sensors that can be characterized and calibrated. This setup could thus be extended to other applications.

Acknowledgments. The authors would like to thank the Conseil Régional de Bourgogne (France) and i3mainz laboratory (Germany) for their financial support. We also thank the Institut für Stein Konservierung (Germany) for providing the test objects and defining the conservation problem.

References

1. Axori photogrammetric bundle block adjustment, http://www.axios3d.de/produkte/funktionlibs/ori/index_en.html
2. Berns, R.S.: Color-Accurate Image Archives Using Spectral Imaging. In: Proceedings of the National Academy of Science - Scientific Examination of Art: Modern Techniques in Conservation and Analysis, pp. 105–119 (2005)
3. Blais, F., Taylor, J., Beraldin, J.A., Godin, G., Cournoyer, L., Picard, M., Borgeat, L., Dicaire, L., Rioux, M., Lahanier, C., Aitken, G.: Ultra-High Resolution Imaging at 50 μm using a Portable XYZ-RGB Color Laser Scanner. In: International Workshop on Recording, Modeling and Visualization of Cultural Heritage (2005)

4. Corsini, M., Dellepiane, M., Ponchio, F., Scopigno, R.: Image-to-Geometry Registration: a Mutual Information Method exploiting Illumination-related Geometric Properties. Computer Graphics Forum 28(7), 1755–1764 (2009)
5. Easton Jr., R.L., Knox, K.T., Christens-Barry, W.A.: Multispectral imaging of the Archimedes palimpsest. In: Proceedings of the 32nd Applied Imagery Pattern Recognition Workshop, pp. 111–116. IEEE (2003)
6. Hefele, J.: Real-time photogrammetric algorithms for robot calibration. International Archives of Photogrammetry and Remote Sensing XXXIV(Part 5), 33–38 (2002)
7. Maas, H.G.: Dynamic photogrammetric calibration of industrial robots. In: Proceedings of SPIE, Videometrics V, vol. 3174, pp. 106–112. SPIE, San Diego (1997)
8. Maes, F., Collignon, A., Vandermeulen, D., Marchal, G., Suetens, P.: Multimodality image registration by maximization of mutual information. IEEE Transactions on Medical Imaging 16(2), 187–198 (1997)
9. Mansouri, A., Lathuiliere, A., Marzani, F., Voisin, Y., Gouton, P.: Toward a 3D Multispectral Scanner: An Application to Multimedia. IEEE MultiMedia 14(1), 40–47 (2007)
10. Pelagotti, A., Del Mastio, A., De Rosa, A., Piva, A.: Multispectral imaging of paintings. IEEE Signal Processing Magazine 25(4), 27–36 (2008)
11. Remondino, F., El-Hakim, S.: Image-based 3D modelling: a review. The Photogrammetric Record 21(115), 269–291 (2006)
12. Sanchez, M., Mansouri, A., Marzani, F.S., Gouton, P.: Spectral reflectance estimation from multispectral images using neural networks. In: Physics in Signal and Image Processing, Toulouse, France (2005)
13. Schütze, R., Raab, C., Boochs, F., Wirth, H., Meier, J.: Optopose - a multi-camera system for fast and precise determination of position and orientation for moving effector. In: 9th Conference on Optical 3D Measurement Techniques, Vienna, Austria (2009)
14. Sitnik, R., Mczkowski, G., Krzeslowski, J.: Integrated shape, color, and reflectivity measurement method for 3D digitization of cultural heritage objects. In: Proceedings of SPIE, 3D Image Processing and Applications, vol. 7526, pp. 75260Q-1–75260Q-10. SPIE, San Jose (2010)
15. Tsai, R.: A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. IEEE Journal on Robotics and Automation 3(4), 323–344 (1987)
16. Viola, P.A., Wells, W.M.: Alignment by maximization of mutual information. International Journal of Computer Vision 24, 137–154 (1997)

U-Drumwave: An Interactive Performance System for Drumming

Yin-Tzu Lin[1], Shuen-Huei Guan[1], Yuan-Chang Yao[2],
Wen-Huang Cheng[3], and Ja-Ling Wu[1]

[1] National Taiwan University, Taipei, Taiwan, R.O.C.
{known,drake,wjl}@cmlab.csie.ntu.edu.tw
[2] Digimax Inc., Taipei, Taiwan, R.O.C.
[3] Academia Sinica, Taipei, Taiwan, R.O.C.
whcheng@citi.sinica.edu.tw

Abstract. In this paper, we share our experience of applying the modern multimedia technologies to the traditional performing art in a drumming performance project, U-Drumwave. By deploying an interactive system on the drumming stage, the audience will see augmented visual objects moving on the stage in accord with the performer's drumming rhythms. The creation and display of the visual objects are integrated with the concept of story intensity curve in order to vary the perceptual degree of tension given to the audience during the performance.

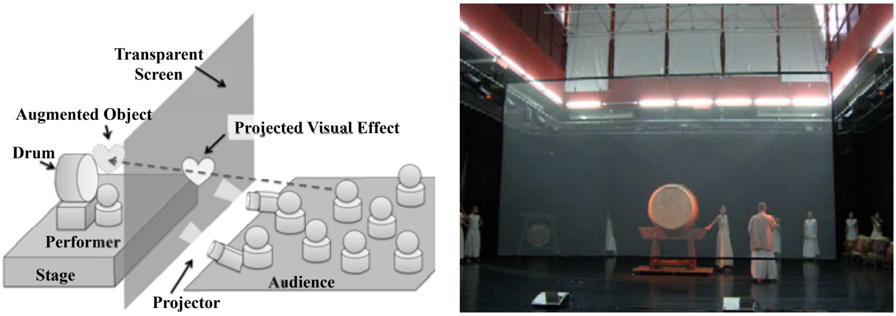
Keywords: Interactive Art, Drumming Performance, Spatial AR.

1 Introduction

U-Drumwave is a project for producing a new style drumming show with the creation of a sense of harmony between the traditional art performance and the modern multimedia technologies. We incorporated the existing interactive and spatial augmented reality technologies into the traditional East Asian drumming performance of U-Theatre[1]. In this show, the U-Theatre artists aim to convey the harmonious relation between humans and the universe. To express the idea, the stage was designed as shown in Figure 1. We set up a transparent screen between the audience and the stage. From the audience's viewpoint, visual contents will be projected on that screen, appearing as augmented objects near to the performer so as to create a perception that there is no boundary between the performer and the world (the projected contents). In addition, the visual effects will be in response to the performer's actions in a synchronized rhythmic manner to express the performer's sense of mutuality and understanding toward the world. To ensure the visual contents' style would conform to U-Theatre's conveyed messages, the contents are jointly created by both the drumming group and our visual design artists through an iterative discussion process. We also incorporate the concept of story intensity curve in the film theory [5] into the

[1] http://www.utheatre.org.tw/

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 609–620, 2012.



(a) Overview (b) Actual setup

Fig. 1. The setup of our U-Drumwave stage

design of our visual effects in order to give better comprehension of the idea behind the performer's motions to the audience. The U-Drumwave project was successfully presented to the committees of Taiwan's Council for Cultural Affairs[2] on November 17, 2010, and also granted the government's financial support. In this paper, we will share our experience of running this project and describe our design philosophy and the corresponding implementation details.

1.1 U-Theatre Introduction

U-Theatre is a renowned Taiwanese performance group. It was founded in 1988 by their artist director Ruoyu Liu in a mountain forest in Taipei's Muzha Area. Inspired by Jerzy Grotowski who trains performers in mountains, Liu emphasizes how performers sharpen their bodily sensation and inner awareness. In 1993, after studying meditation in India and Tibet, the drum master ChihChun Huang join the group. Huang requests that before practicing drumming, performers must learn meditation. This training course not only changes the temperament of U-Theatre members but also sets the tone of their performance: athletic drumming and martial arts. Because U-Theatre represents the harmonious encounter between the West and the East in its performances, it is highly evaluated by international artists. U-Theatre has been regularly invited to perform in art festivals of different countries, such as the Netherlands, Italy, Germany, Spain, France, and so forth.

2 Artist's Intentions and Design Strategies

Liu thinks that a show, in essence, is the expression of "an individual's attitude toward life." For U-Theatre members, the attitudes are "living in the moment" and the unification of "Tao[3]" and "Yi[4]". Huang believes that the spiritual states

[2] Taiwan's highest official institution for planning cultural establishments and policies.

[3] A Chinese character generally meaning "way", here means self-improvement.

[4] A Chinese character generally meaning "skill", here means live with art.

during meditating and performing are the same. When the performers drum with meditative minds, each hit on the drums strikes the heart chord of the audience. Contrarily, the loud pounding of drums makes the audience stay calm. Then, the performance becomes a comfort for the audience. In this show, the U-Theatre artists tend to further emphasize the relation between humans and the universe. That is, with open minds, there will be no boundaries among people. Then, there will be no boundary between humans and the world. Humans will develop a sense of mutuality and understanding toward natural beings. To convey the above ideas, our design strategies are listed below.

2.1 Transparent Screen

The transparent screen between the audience and the stage (cf. Figure 1) will help to provide the feelings of no boundary between the performer (humans) and the projected visual contents (the world). From the audience's viewing direction, the contents becomes virtual objects near the performer. There are many means that could create this experience (c.f. Section 6.1). We use a near-transparent projection screen made of gauze. The gauze is coated with special painting materials so that it reflects non-black lights only. Accordingly, black contents projected on the screen become transparent while other-colored contents will be normally shown (c.f. Figure 2(a)). With appropriate lights, stage design and seat arrangement, the audience will not notice the existence of the screen and the immersive experience is created. The advantages of gauze screen are flexibility and portability. For our purpose to "hide" the screen, its size should be as large as possible. Fabricating large gauze screen is easier than making large electronic screen e.g. OLED (Organic light emitting diode). Furthermore, the gauze screen is easy to be decomposed. Its frame and its gauze can be detached (c.f. Figure 2(c)) and the frame can be further divide into short rods (c.f. Figure 2(b)). This makes it easier to be moved among different theatres and more suitable for tour performances.

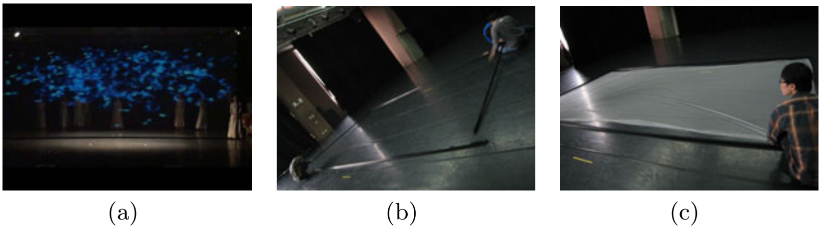

(a) (b) (c)

Fig. 2. (a) The gauze screen coated with special painting materials reflecting non-black lights only. (b)(c) Snapshots of our staff's assembling the gauze screen on the stage.

2.2 Interaction

With the assistance of interaction technologies, we can make the visual effect in accord with the rhythms of performers' actions detected from drumming sound. For example, "the time to appear", "the speed of movement", and "the size" of visual effects may correspond to sound events like "timings of hit", "speed of hits", and "strengths of hits", respectively. We regard each visual effect as a virtual character, relative to the real character: the performer. Each virtual character has its own reaction to the performer. Combining all these virtual and real characters forms a story intensity curve (c.f. Section 2.4) that may increase the comprehensibility and attractiveness for the audience.

2.3 Visual Content Design

The visual content are designed through an iterative process. Initially, our visual design artists proposed several designs to the U-Theatre artists. Then, they re-designed the contents according to U-Theatre artists' feedbacks. The above process iterated several times until the both sides have reached a common agreement. As a result, all the visual contents are joint creations of both U-Theatre and our visual design artists. During the design process, we found that the U-Theatre artists prefer plain, abstract, or natural contents, instead of fancy ones. The designed contents are listed below: geometric symbols (Figure 3), Buddhist styled symbols (Figure 4), and simulated natural phenomena video (Figure 5).

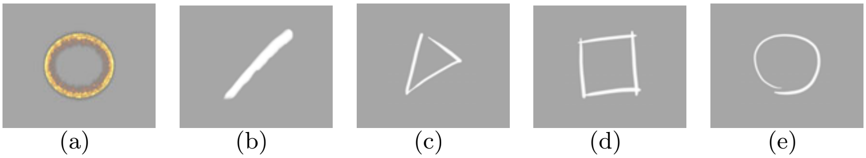

(a) (b) (c) (d) (e)

Fig. 3. Geometric Symbols: (a) Energy Wave(EW), (b) Stroke(ST), (c) Triangle(TR), (d) Rectangle(RE), (e) Circle(CR)

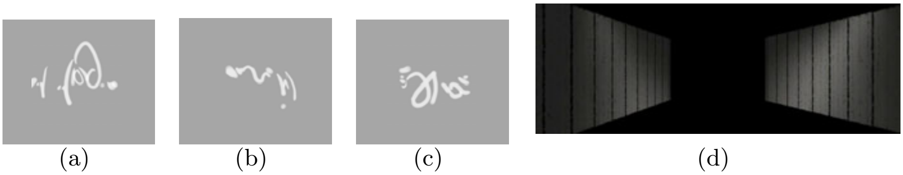
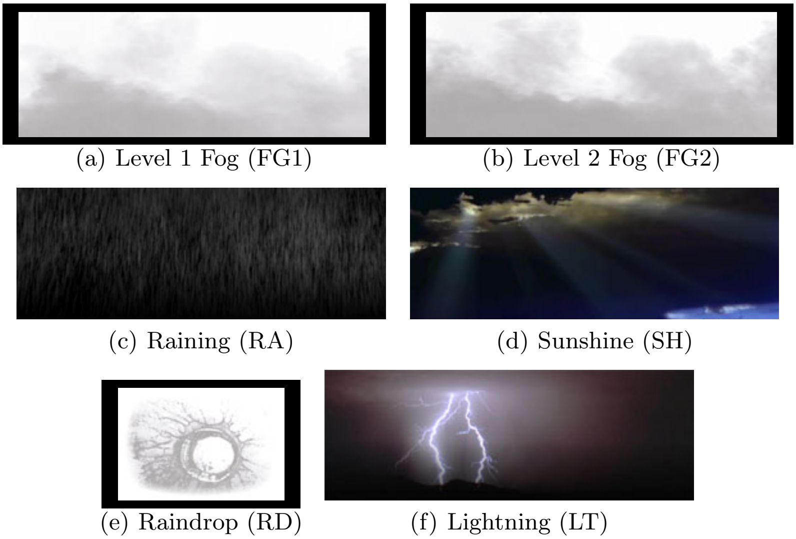

(a) (b) (c) (d)

Fig. 4. Buddhist Styled Symbols: (a)(b)(c) Part of Abstract Sanskrit Totems (TT), (d) Sutra Video Screenshot (SU)



(a) Level 1 Fog (FG1)

(b) Level 2 Fog (FG2)

(c) Raining (RA)

(d) Sunshine (SH)

(e) Raindrop (RD)

(f) Lightning (LT)

Fig. 5. Screenshots of Simulated Natural Phenomena Videos

2.4 Story Intensity Curve

The term “story intensity” refers to the degree of tension that the audience experienced[5]. A good story that people would enjoy often contains three stages: exposition (EX, beginning), conflict (CO)/climax (CX) and resolution (R)[5]. The story intensity during these stages often follows the shapes like Figure 6(a). In real world, the curve is often jagged (c.f. Figure 6(b)), but the overall shape will still conform to the standard one [5].

To increase the comprehensibility and attractiveness of the performance, both the U-Theatre artist and we carefully arranged the performer’s action and the appearing order of the visual contents in order to conform to the standard story intensity curve (c.f. Figure 6(c)). First, the U-Theatre artist adds an exposition section in the beginning of the original pure-drumming to introduce the “interaction” in this show. In the exposition, the performer will act with his motions similar to the projected visual effects, just like he is playing around with the drum. For example, he may use his drum sticks to draw a triangle above the drum with a triangle (cf. Figure 3(c)) showing on the screen at the same time. After that, the performer normally drums loud or soft according to an internal story intensity in his mind. As illustrated in Figure 6(c), the visual contents are also arranged to conform the story intensity curve (including the one hidden in the performer’s mind). First, the geometric symbols accompany the exposition section. Then, the Sanskrit totems appear as a bridge between geometric symbols and the sutra book video. As the performer drumming to the climax and suddenly drumming soft, we switch the visual contents to the category of natural phenomena. In this section, we let the worst weather (FG2+RA+LT) match the

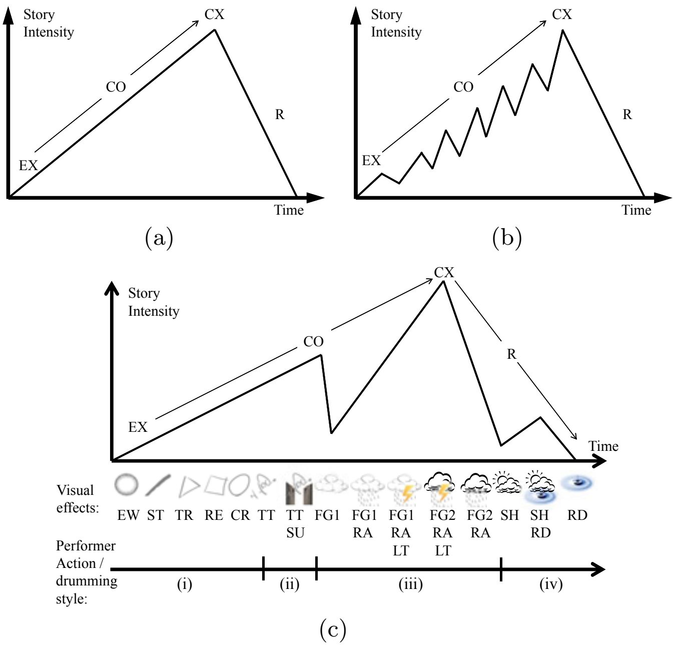


Fig. 6. Standard Story Intensity Curve: (a) Theoretic story intensity curve (b) Real world story intensity curve (c) The Story Intensity Curve of U-Drumwave (i): Testing and playing around with the drum (introducing interactivities) (ii): Drumming to the climax, then suddenly drumming soft (iii): Drumming from soft to loud (climax), and vice versa (iv): Drumming very softly, then stopping drumming, slowly going off stage

climax of drumming. After it clears up and the performer stops drumming, the raindrop effects are shown to decrease the intensity.

3 System Implementations

3.1 Physical Installation

Figure 7 illustrates the physical installation of U-Drumwave. We regard the combination of the blending PC (the red PC) and two projectors (Hitachi CP-A200 3 LCD) as a virtual projector. The main programs running on the control PC (the white PC) analyzes the signals from the audio mixer and then outputs the corresponding visual effects to the virtual projector. During the run-time, the working screen will display the control GUI.

Virtual Projector. Since the gauze screen is wider than the maximum range that one projector can cover, we use two projectors in charge of each half of the projected contents. To reduce the effort of calibration, a blending program

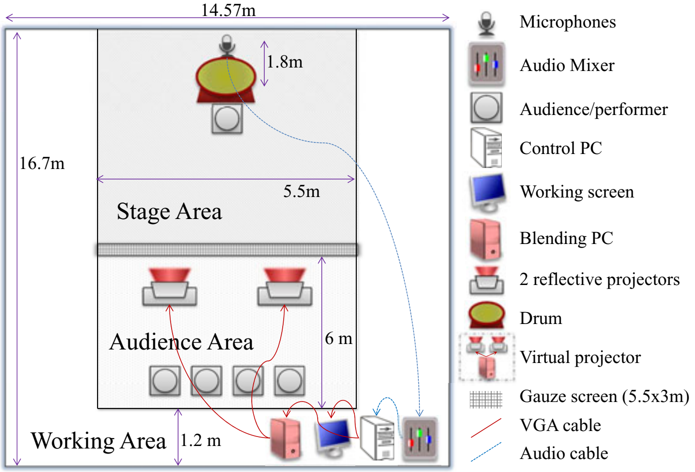
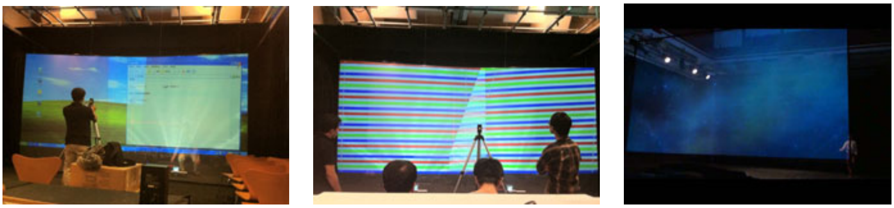


Fig. 7. Semantic Illustration of the Physical Installation

running on the blending PC is embedded as a plug-in of KMPlayer[5], a free video playing software available on the web. There are also tools for helping adjust the overlap ratio of the two halves and tuning the affine transformation parameters of each half (c.f. Figure 8). After adjusting, every video played by the KMPlayer will apply these parameters and show on the gauze screen seamlessly. Since KMPlayer supports real-time playback of frame grabber cards, we let the control PC just output VGA signals as for normal projectors, and KMPlayer on the blending PC plays frames captured from frame grabber card as playing a normal video. For the control PC, the combination of the blending PC and 2 projectors forms a virtual projector.



(a) Before Adjusting (b) During Adjusting (c) After Adjusting

Fig. 8. Adjusting the projection parameters on the stage

[5] http://www.kmplayer.com/

3.2 Software Architecture on the Control PC

The software architecture on the control PC is described in Figure 9. We implemented it on the Cycling'74 Max/MSP/Jitter[6] platform. The system can be divided into three parts: audio processing, virtual character, and mixer-like control interface.

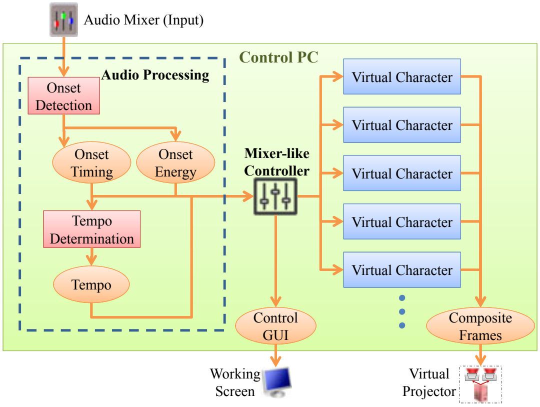


Fig. 9. Control Software Architecture

3.3 Audio Processing

In the audio processing part, we tend to detect the onsets of the input signal. For drumming performance, the onsets can be approximately regarded as beats. Thus, we then determine the tempo information using the timings of the onsets.

Onset Detection. The goal of onset detection is to find the timing and the strength of every hit to the drum. By observation, a hit on the Chinese drum produces a vertical line on the sound spectrogram. As a result, our approach is similar to the "spectra difference" method mentioned in [1] except that we use bark-scaled spectrogram instead and a different peak picking method. The bark-scaled (25 subbands) spectrogram is computed using Tristan Jehan's MSP external "`bark~`" [8]. The onset detection function $f(n)$ is calculated by the sum of the rectified energy difference of each frequency band between successive audio frames. We modified the peak picking method to adapt to this live application. That is, the determination of onsets only depends on previous frames. If $f(n)$ exceeds a certain threshold, we report an onset happened at time n. However, when the performer drums fast, $f(n)$ becomes less discriminative. As a result, two thresholds are used. The total energy in the previous frame $E(n-1)$ determines which threshold to use. As shown in Equation 1, the indicator function $O(n)$ represents the detected

[6] `http://cycling74.com/`

onsets. When $O(n) = 1$, an onset is detected, we output a "bang" message and the corresponding $f(n)$ value reflecting its strength. That is,

$$O(n) = \begin{cases} 1\ , E(n-1) > T_E^{high} \text{ and } f(n) > T_f^{low} \\ 1\ , E(n-1) \in (T_E^{low}, T_E^{high}] \text{ and } f(n) > T_f^{high} \\ 0\ , \text{otherwise.} \end{cases} \tag{1}$$

In addition, the widely used spectral brightness feature[9] is used to reduce false detection caused by other sounds (e.g. human voice). We regard an onset happens if $O(n) = 1$ and $Brightness(n) < 0.08$ because the drum usually sounds bass. The core of the onset detection algorithm is implemented in JavaScript.

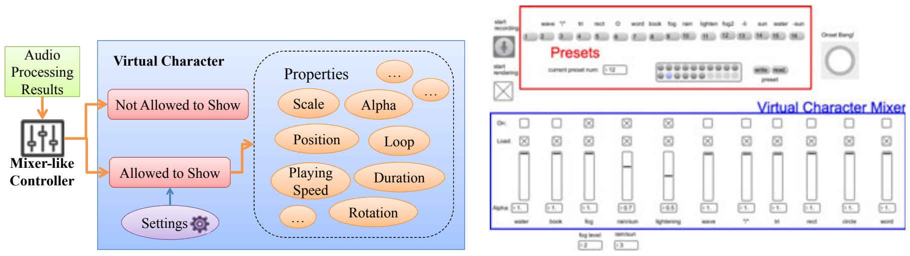
Tempo Determination. After obtaining the timing of each onset, we further estimate the tempo (in BPM, beat per minute) of the drumming performance through the reciprocal of onset intervals. An MSP built-in object `sync~` supports the above BPM calculation. However, using `sync~`, the BPM value only updates when it receives "bang" message. As a result, we implement our own patch to immediately update the tempo once the performer starts to slow down. We not only calculate BPM value upon receiving a "bang", but also check the elapsed time from the last onset to now. Once the elapsed time exceeds the last onset intervals, we replace the onset interval as this elapsed time. Subsequently, the output tempo value will gradually decrease when the performer start to slow down until he hits the drum again.

3.4 Virtual Character

The virtual character part deals with the response actions of the visual contents to the audio events. Figure 10(a) shows the structure of one virtual character and its relationships to other components. Users will decide whether a virtual character is allowed to show through the control interface. Once being allowed, the virtual character starts to receive audio processing results. Then, it appears according to the properties of showing based on its settings. For example, the size may be full-screen or varying with the onset energy. The vanish time may be receiving "not allowed to show" or automatic vanishing upon last video frame. The virtual character is implemented as a wrapper object of "`jit.qt.movie`" and "`jit.gl.videoplane`". We choose this combination because it is easier to composite multiple images and videos with varying properties. Besides, the alpha channel works better with "`jit.gl.*`" objects. This makes it easier to show irregular shape objects without revealing annoying rectangle frame and blocking other objects beneath them.

3.5 Control Interface

Figure 10(b) shows a screenshot taken from the control interface of U-Drumwave system. Motivated by audio mixer devices, we design 2 check boxes and one slider



(a) (b)

Fig. 10. (a) The structure of Virtual Character (b) Control Interface Screenshot

bar for each virtual character (the bottom part). One check box controls loading the virtual characters into the memory. The other one determine whether the characters are allowable to show. The slider bar controls the transparency of the character's appearance. we also design a presetting part to record the appearance settings, so the users can switch numerous settings of virtual characters simultaneously. To avoid sharp changes between settings, the slider bar will smoothly slides to the target value. "Fade-in/out" effects between the characters are then resulted. The "onset bang" button at the right hand side will light up each time while an onset is detected. If an important event is miss detected, clicking the button to send an "bang" message is a remedy.

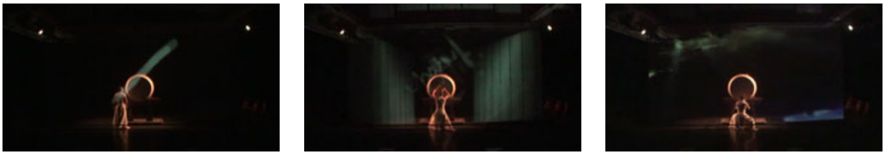
4 Live Demo

The U-Theatre artists combined U-Drumwave with their other drum songs to create a 40-minutes show and presented it to the committees of Taiwan's Council for Cultural Affairs on Nov. 17, 2010. Figure 11 presents some snapshots taken during the performance[7]. The government-appointed reviewers highly appreciated the show. They commented like, "Very splendid! Fortunately this is not another tech-show! The technology and the art mix just right and seamless." After the show, the reviewers also enjoy the interactive system. The exposition of U-Drumwave acquired the most interest. Finally, they granted financial support for U-Theatre to run the formal version on August, 2011 in Taiwan.

5 Discussion

The most difficult but also the most interesting development challenge of the U-Drumwave project is the communications between engineers and artists. Our crew comes from various professional domains. We have the drum master, the artistic director, the visual design artists, the gauze screen technicians, the recording technicians and the interactive program engineers. The U-Theatre

[7] Demo video: `http://www.cmlab.csie.ntu.edu.tw/~known/ud/`



(a) (b) (c)

Fig. 11. Snapshots captured from the live U-Drumwave performance

artists often describe things via feelings. These descriptions are vague for the engineers to write programs. Besides, the U-Theatre artist used to impromptus and often change the plot when rehearsing. It is a challenge for the engineers to change the programs as fast according to the artists' requests. The control UI helps a lot to these issues. The artists and the engineers can discussed the flow and the design of the contents with viewing the results on the gauze screen. Besides, it is easily to immediately modify virtual characters' actions, just like a real man changing his gestures. Thus, the resulting performance nicely harmonizes the art and the technology.

6 Related Work

6.1 Spatial Augmented Reality

Spatial Augmented Reality (SAR) is one of AR's sub-domain. SAR does not request users to equip with eye-worn or head-mount display. As Bimber and Raskar[4] mentioned, existing SAR techniques can be categorized into 3 groups according to the display techniques used: "screen-based video see through display", "spatial optical see-through display" and "projection-based spatial dispaly". The first one mixes captured image and virtual object as video and displays on regular monitors. The second one overlays synthetic image on real object through optical techniques, such as optical combiners [3], transparent screen ([10] and our approach) or optical hologram[2]. The last one directly projects image seamlessly on physical object's surface instead of a projection screen[7].

6.2 Interactive Technology for Performance

The development of interactive technology for performance has long been explored. The first representative work was Gordon Pask and Robin McKinnon-Wood's electromechanical system MusiColour in 1953 [6]. Motivated by the concept of synaesthesia, Pask makes color lights adjusted in response to live music. The responded color lights are treated as another performer on the stage. There are many follow up systems. Sparacino et al. [11] gave a widely survey of interactive spaces of performing art. A recent approach was Wei et al.'s Ozone[12], a state-based interactive system for live performance.

7 Conclusions and Future Work

In this paper, we share our experience of harmonizing the technology and the art to produce a show. The artistic intentions are realized through the development of a technology enabled interactive system on the performing stage, including the combinations of a transparent screen, the sound-driven interactive technology, iterative visual content design and the concept of story intensity curve. It is our belief that U-Drumwave has created a new style of performance and set a working example. As for the further improvements, possible directions include enhancing the onset detection accuracy, incorporating timbre recognition, real-time rendered particles, and human action detection with IR or depth cameras.

Acknowledgments. The rest crew of U-Drumwave: Ruoyu Liu, ChihChun Huang, Tai-Huei Lee, MAKA Inc., Chris Wu, Vivi Chen, Eddie Lin, Jade Cheng, Bala Tsai, Chien-Tse Yang, Hao-Ming Chen, and Edward Chang.

References

1. Bello, J.P., Daudet, L., Abdallah, S.A., Duxbury, C., Davies, M., Sandler, M.B.: A Tutorial on Onset Detection in Music Signals. IEEE Trans. Audio, Speech, Lang. Process. 13(5), 1035–1047 (2005)
2. Bimber, O.: Combining Optical Holograms with Interactive Computer Graphics. IEEE Computer 37(1), 85–91 (2004)
3. Bimber, O., Fröhlich, B., Schmalstieg, D., Encarnação, L.M.: The Virtual Showcase. IEEE Comput. Graph. Appl. 21(6), 48–55 (2001)
4. Bimber, O., Raskar, R.: Spatial augmented reality: Merging real and virtual worlds. A K Peters, Ltd. (2005)
5. Block, B.: The Visual Story: Seeing the Structure of Film, TV and New Media. Focal Press (2001)
6. Haque, U.: The Architectural Relevance of Gordon Pask. Architectural Design 77(4), 54–61 (2007)
7. Jacquemin, C., Chan, W., Courgeon, M.: Bateau Ivre: an Artistic Markerless Outdoor Mobile Augmented Reality Installation on a Riverboat. In: ACM Multimedia, pp. 1353–1362. ACM Press (2010)
8. Jehan, T., Schoner, B.: An Audio-Driven Perceptually Meaningful Timbre Synthesizer. In: International Computer Music Conference (2001)
9. Lartillot, O., Toiviainen, P.: MIR in Matlab (II): A Toolbox for Musical Feature Extraction from Audio. In: The International Society for Music Information Retrieval Conference, pp. 237–244 (2007)
10. Laser Magic Productions: Holograms, Transparent Screens, and 3D Laser Projections (2003), http://www.laser-magic.com/transscreen.html
11. Sparacino, F., Davenport, G., Pentland, A.: Media in Performance: Interactive Spaces for Dance, Theater, Circus, and Museum Exhibits. IBM Syst. J. 39(3), 479–510 (2000)
12. Wei, S., Fortin, M., Navab, N., Sutton, T.: Ozone: Continuous State-based Media Choreography System for Live Performance. In: ACM Multimedia, pp. 1383–1392. ACM Press (2010)

Real-Time Visualizations of Gigapixel Texture Data Sets Using HTML5

Charles-Frederik Hollemeersch, Bart Pieters, Aljosha Demeulemeester,
Peter Lambert, and Rik Van de Walle

Multimedia Lab, Ghent University - IBBT,
Gaston Crommenlaan 8, B-9050 Ledeberg-Ghent, Belgium
charlesfrederik.hollemeersch@ugent.be
http://multimedialab.elis.ugent.be/

Abstract. With the recent standardization of WebGL as part of HTML5, new possibilities have arisen for graphically intensive web-based applications. This paper presents our gigapixel texture visualization system which runs entirely within the limitations of a standards-compatible browser. Compared to existing approaches, our system offers high-performance 3D texture visualization and streaming without any dedicated plugins. We show that real-time performance can be achieved (less than 12ms render time per frame) on current-generation desktop hardware for texture data sets of at least 15 gigapixels.

Keywords: Streaming, WebGL, Visualization.

1 Introduction

The introduction of the new HTML5 standard has enabled a new generation of web applications. Web applications are no longer constrained to DOM based user interfaces or proprietary plugins to generate rich visual output. Standardized technologies such as the 2D canvas and 3D WebGL allow graphics operations to be scripted directly from within JavaScript. By generating graphics entirely on the client side, the latency and bandwidth of the web application can also be significantly reduced compared to systems generating graphics in the cloud and streaming the results to the clients as a video stream.

Visualizing high-resolution image and texture data sets is a challenging problem which has many practical uses. For example, GIS, biology, archeology, heritage, and educational applications all have benefited from efficiently aquiring and accessing large image data sets [1]. In this paper we will show how accessing large texture data sets can be made possible within the framework of the HTML5 standard by using WebGL and other HTML5 features.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 621–623, 2012.

Table 1. Per-frame render times of our application in different browsers. Results were measured on a 2.4GHz Intel Core2 Quad CPU and a NVIDIA Geforce GTX 480.

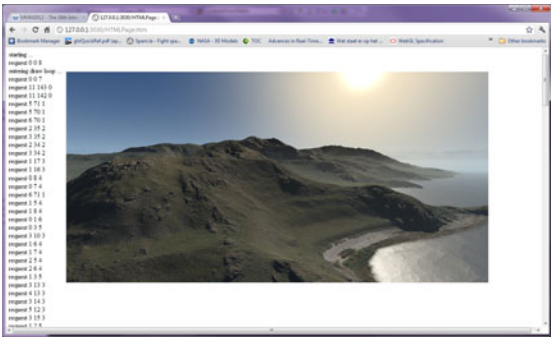
		Google Chrome 12	Mozilla Firefox 5	Opera 11 Preview
Frame (ms)	Rendering	12	2.5	3.1*

* Note, the Opera 11 preview currently generates invalid results.

2 Visualizing Gigapixel Texture Data Sets Using HTML5

There are currently several web-based technologies that allow acessing GigaPixel images over the web[1],[2]. However most of these technologies rely on the use of the Adobe Flash plugin to do the most graphics-intensive parts of their visualization work. In addition to this, all these visualizations are limited to 2D images. Furthermore, they rely on analytical approaches to determine the set of required data[2] which do not extend well to 3D visualizations. By adopting concepts and ideas from the high-performance computing world [3] it becomes possible to visualize large texture data sets applied on 3D models and geometry in real time on the latest versions of most major browsers. Our method works by offloading the most computationally expensive operations to the GPU using WebGL and uses algorithmic optimization to accelerate the remaining steps in JavaScript. Currently we have not done any extensive code-level optimizations.

Figure 1 shows a screenshot of our demo application. This application visualizes a large texture data set (122880 × 122880 pixels, around 75 gigabytes of uncompressed data) in real time in a standard browser without any custom plugins. Table 1 shows the average time it takes to draw the scene in the browser. These times include updating internal cache data-structures and generating the



```
starting ...
request 0 0 8
entering draw loop ...
request 0 0 7
request 11 143 0
request 11 142 0
request 5 71 1
request 5 70 1
request 6 70 1
request 2 35 2
request 3 35 2
request 2 34 2
request 3 34 2
request 1 17 3
request 1 16 3
request 0 8 4
request 0 7 4
request 6 71 1
request 1 5 4
request 1 8 4
request 0 1 6
request 0 3 5
request 3 10 3
request 1 6 4
request 1 7 4
request 2 5 4
request 2 6 4
request 1 3 5
request 3 13 3
request 4 13 3
request 3 14 3
request 5 12 3
request 3 15 3
request 1 2 5
```

Fig. 1. A screenshot of the demo running in the Google Chrome browser. This is a visualization of orthopotography and height data made available by Utah's State Geographic Information Database (http://gis.utah.gov/).

[1] http://gigapan.org/

[2] http://www.yosemite-17-gigapixels.com/

necessary requests for image data. Note that Opera 11 Preview[3] currently gives invalid output. However on both Chrome and Firefox we can easily achieve real time framerates while still leaving enough CPU time available for other browser tasks. The speed difference between Chrome and Firefox are probably due to the fact that chrome sandboxes WebGL calls and translates all commands to DirectX, while firefox runs WebGL in the same process. However, a more detailed investigation is needed to confirm this.

3 Conclusions

We have shown that it is possible to create computational and data intensive applications using the latest generation of HTML5-based browsers. In particular in our demo we have shown that 3D visualizations using high-resolution textures are no longer limited to native applications requiring high-end computers. In the near future this technology will allow making large data sets available to a larger and less technically skilled public across a wide range of computing devices.

In the future we want to further optimize our system and extend it so it becomes a fully functional frontent for our collaborative editing tool [4]. This will allow a large number of users to not only visualize the data set but also modify and annotate it.

Acknowledgments. The research activities that have been described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWOFlanders), the Belgian Federal Science Policy Office (BFSPO), and the European Union.

References

1. Frenkel, K.A.: Panning for science. Science 330, 748–749 (2010)
2. Kopf, J., Uyttendaele, M., Deussen, O., Cohen, M.F.: Capturing and viewing gigapixel images. ACM Transactions on Graphics 26 (2007)
3. Hollemeersch, C., Pieters, B., Lambert, P., Van de Walle, R.: Accelerating virtual texturing using cuda. In: Engel, W. (ed.) Gpu Pro:Advanced Rendering Techniques, ch. 10.2, pp. 623–641. A K Peters (2010)
4. Hollemeersch, C.-F., Pieters, B., Demeulemeester, A., Cornillie, F., Van Semmertier, B., Mannens, E., Lambert, P., Desmet, P., Van de Walle, R.: Graphics for serious games: Infinitex: An interactive editingsystem for the production of large texture datasets. Comput. Graph. 34, 643–654 (2010)

[3] http://labs.opera.com/news/2011/02/28/

Enhancing the User Experience with the Sensory Effect Media Player and AmbientLib

Markus Waltl, Benjamin Rainer, Christian Timmerer, and Hermann Hellwagner

Alpen-Adria-Universität Klagenfurt, Inst. of Information Technology
Multimedia Communication Group, Universitätsstraße 65-67
Klagenfurt, Austria
firstname.lastname@itec.uni-klu.ac.at

Abstract. Multimedia content is increasingly used in every area of our life. Still, each type of content only stimulates the visual and/or the hearing system. Thus, the user experience depends only on those two stimuli. In this paper we introduce a standard which offers the possibility to add additional effects to multimedia content. Furthermore, we present a multimedia player and a Web browser plug-in which uses this standard to stimulate further senses by using additional sensory effects (i.e., wind, vibration, and light) to enhance the user experience resulting in a unique, worthwhile sensory experience.

Keywords: MPEG-V, User Experience, Sensory Experience, Media Player, Ambient, World Wide Web.

1 Introduction

Each day a vast amount of multimedia content is consumed through distribution channels such as Blu-Ray discs or the Internet. All traditional multimedia content (i.e., combinations of video, audio, text, and image) has in common that it only stimulates the human visual and/or hearing system. Thus, research commenced which introduces and evaluates the enhancement of multimedia content with additional stimuli (e.g., olfaction, mechanoreception, termoreception) [1, 2].

As this research area gained a lot of interest, the Motion Picture Experts Group (MPEG) initiated the work on the MPEG-V – Media Context and Control [3] standard. Part 3 of this standard is referred to as Sensory Information (SI) and provides the possibility to enrich available multimedia content with additional effects (e.g., wind, vibration, light, scent) for enhancing the user experience. Such effects are described by so-called *Sensory Effect Metadata* (SEM) descriptions providing information such as the type of effect, intensity of the effect, playback time, etc.

We [4-6] and also others [7, 8] started using and evaluating MPEG-V in various application areas (e.g., multimedia playback, broadcasting, World Wide Web). In the remainder of this paper we describe our implementation (Section 2) and demonstrator (Section 3).

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 624–626, 2012.

2 Sensory Effect Media Player and AmbientLib

The *Sensory Effect Media Player* (SEMP) is a Windows-based media player which offers the possibility to load videos and SEM descriptions. These descriptions are processed, synchronized with the video, and rendered on appropriate devices (e.g., ambient lights, vibration devices, fans). Currently the player supports the amBX system [9] consisting of two fans, two "light" speakers (left and right) with a subwoofer, a wall-washer unit, and a vibration panel. We used the freely available SDK to program the amBX system for enhancing the user experience while watching videos. The architecture of SEMP is illustrated in Fig. 1 (a). SEMP can automatically extract color information from the currently rendered frame and display the color on the amBX lights.

Since more and more multimedia content is available on the Internet through different portals (e.g., YouTube and Vimeo), we started to work on a browser plug-in (called *AmbientLib*) which enables the sensory experience in the Web browser. The current version of the plug-in supports all major browsers (i.e., Opera, Google Chrome, Safari, Mozilla Firefox, Internet Explorer) and is easy to install. *AmbientLib* handles light effects the same way as SEMP but with the difference that the videos are embedded in Web sites. The plug-in is able to handle Flash videos and videos provided through the HTML5 video tag. Furthermore, if the plug-in detects an available SEM description, it also provides wind and vibration effects accordingly. Fig. 1 (b) presents the architecture of *AmbientLib*.

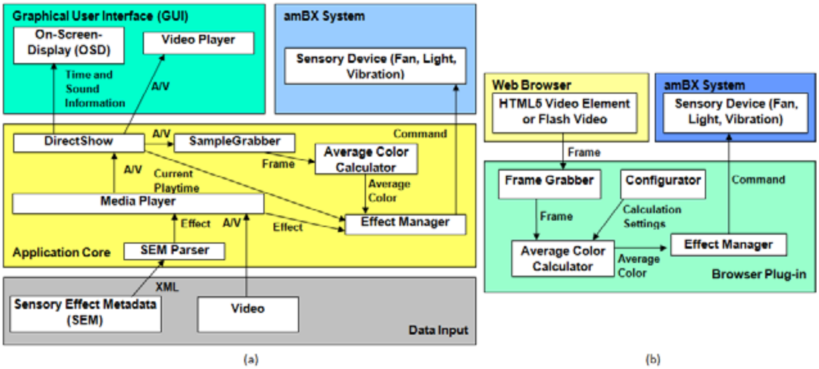


Fig. 1. Architectures of the Sensory Effect Media Player (a) and the AmbientLib Plug-in (b)

3 Demonstration

In our demonstration, we will present both SEMP and *AmbientLib*. First, we will present videos with and without sensory effects (i.e., light, vibration, and wind) by using SEMP. Second, we will demonstrate a number of videos accompanied by SEM descriptions using *AmbientLib*.

Participants will be able to try SEMP and *AmbientLib*. We believe that the possibility to experience the sensory effects first hand will trigger discussions on this research. A small example of the demonstration is depicted in Fig. 2 (the fan in action is depicted on the upper right corner).

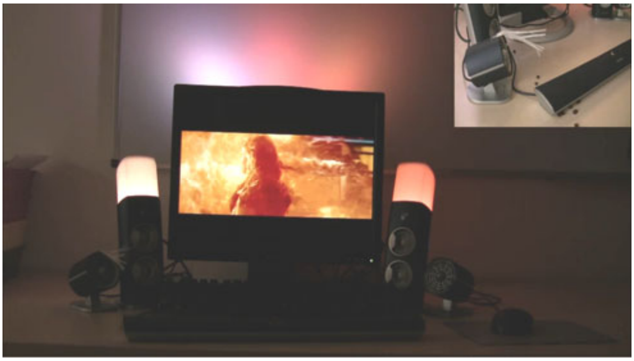


Fig. 2. Enhanced Video Playback with SEMP

Acknowledgments. This work was supported in part by the European Commission in the context of the ALICANTE project (FP7-ICT-248652).

References

1. Chang, A., O'Sullivan, C.: Audio-haptic feedback in mobile phones. In: ACM Conference on Human Factors in Computing Systems (CHI 2005), CHI 2005 Extended Abstracts on Human Factors in Computing Systems, Portland, OR, USA, pp. 1264–1267 (2005)
2. Ghinea, G., Ademoye, O.A.: Olfaction-enhanced multimedia: perspectives and challenges. In: Multimedia Tools and Applications, pp. 1–26 (August 2010)
3. ISO/IEC 23005-3: Information technology – Media context and control – Part 3: Sensory information (June 2011)
4. Waltl, M., Timmerer, C., Hellwagner, H.: A Test-Bed for Quality of Multimedia Experience Evaluation of Sensory Effects. In: 1st Int'l. Workshop on Quality of Multimedia Experience (QoMEX 2009), San Diego, USA (2009)
5. Waltl, M., Timmerer, C., Hellwagner, H.: Increasing the User Experience of Multimedia Presentations with Sensory Effects. In: 11th Int'l. Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2010), Desenzano del Garda, Italy (2010)
6. Waltl, M., Timmerer, C., Hellwagner, H.: Improving the Quality of Multimedia Experience through Sensory Effects. In: 2nd Int'l. Workshop on Quality of Multimedia Experience (QoMEX 2010), Trondheim, Norway (2010)
7. Choi, B.S., Joo, S., Lee, H.: Sensory Effect Metadata for SMMD Media Service. In: 4th Int'l Conf. on Internet and Web Applications and Services (ICIW 2009), Venice/Mestre, Italy, pp. 649–654 (2009)
8. Yoon, K., Choi, B., Lee, E., Lim, T.: 4-D Broadcasting with MPEG-V. In: 12th Workshop on Multimedia Signal Processing (MMSP) 2010, Saint-Malo, France, pp. 257–262 (2010)
9. amBX UK Ltd, `http://www.ambx.com`

EMIR: A Novel Music Retrieval System for Mobile Devices Incorporating Analysis of User Emotion

Lijuan Zhou[1], Hongfei Lin[1], and Cathal Gurrin[2]

[1] School of Computer Science and Technology, Dalian University of Technology
[2] School of Computing, Dublin City University, Ireland
marissa.zhou.cn@gmail.com, hflin@dlut.edu.cn,
cgurrin@computing.dcu.ie

Abstract. We present an Emotional Music Information Retrieval system for mobile devices that utilizes a machine learning approach to detect latent emotion from within both user queries (non-descriptive queries) and the lyrics of songs and uses both elements to develop an effective Music Information Retrieval system. Emotion is extracted from the songs and queries and mapped into a high-dimensional emotion space, which allows for the employment of conventional text retrieval techniques to calculate the similarity between a user query and the latent emotion in song lyrics, thereby producing a ranked list of songs for playback.

Keywords: Music Information Retrieval, Emotion Detection, Machine Learning, Emotion Space.

1 Introduction

Music is inherently expressive of emotional meaning, however many music search and recommendation systems (*Google Music, Last.fm* and *Xiami Music* etc.) rely on search using descriptive keywords (*title, album, artist,* etc.), or rely on a recommendation engine using past history. It is our conjecture that by integrating the emotion of the user, that the utility provided by a Music Information Retrieval (MIR) system can be significantly enhanced. Such a system is presented in this work, which allows a user, with no specific piece of music in mind, to still get recommendations that match their stated mood.

A basic underlying premise of this work is an understanding that a user query may contain no descriptive keywords of specific music; rather the query may simply indicate the current emotional state of the user. In this paper and in prior work [1,2], we define such queries as Non-Descriptive Queries. We exploit the emotional context of such queries with the aim of providing a ranked set of songs for the user that matches their stated mood. In prior work on emption detection in the text domain, emotion can be categorized into six basic categories (ANGER, DISGUST, FEAR, JOY, SADNESS and SURPRISE) [3]. In addition, contextual text information from lyrics has been proven effective in music emotion recognition [4]. This work on contextual text information has had significant impact on music emotion recognition, but very few researchers have explored the application of emotion detection in for

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 627–629, 2012.

music information retrieval. At the same time, collaborative social tagging has become an essential part of the solution to many MIR problems, including employing lyric-based search [5,6]. In this work, we extend previous work mapping the emotional context of both queries and songs in a high-level emotion space and calculating similarities within this space, whereas previous work mostly focused on original key attribute matching method.

2 Approach to Emotion Detection

With the consideration of integrating emotion into the MIR system, we designed and implemented a demo EMIR System, which is now described. For complete details of this technique, see our previous work [1.2].

2.1 Detecting Emotion from Lyrics

In order to create an effective emotional music IR system, there are a number of core challenges that need to be addressed:

1) A model of music representation, in our case, into a six-dimension model for plotting songs into an emotion space, needs to be created.
2) Analysis of the structure of sentences in lyrics to discover potential relationships between emotion and sentence structure.
3) Selection of the best performing features for emotion detection and subsequent emotion detection.
4) Computation of emotion similarity betweens songs and queries to create a ranked list of results for playback.

To solve these problems, we employ machine learning classify the emotion of the music and the queries in the same six dimension emotion space. Due to the complexity of computing emotion-based similarity (explicit keyword terms to implicit emotions), we use an increasing saturation to give weights the emotions in queries. Finally, to support the generation of ranked lists, we use a revised BM25 retrieval model, to modify the traditional *tf-idf* weighting method, as was proposed in our previous work [1, 2]. An architecture of this demonstration system is shown in Fig. 1.

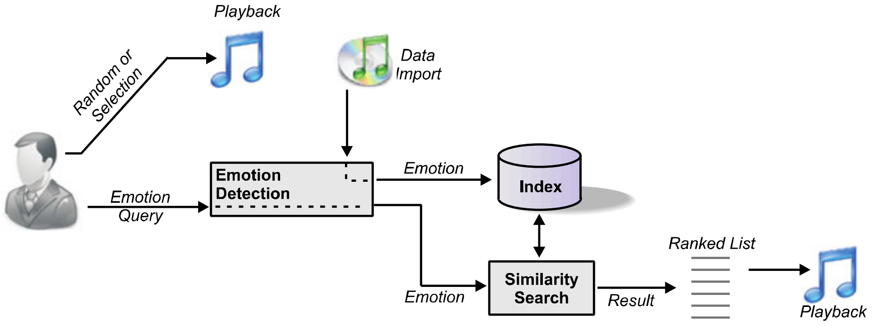
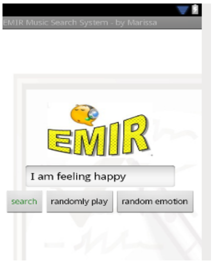


Fig. 1. Overview of EMIR system

2.2 User Interaction with the EMIR Demonstrator

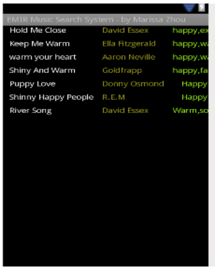
Our EMIR system mainly deals with user music needs by analyzing the emotion of users according to their queries. For demonstration purposes, the system is implemented on an Android mobile phone. As shown in Fig(s). 2-4, users after logging into the system are presented with a search interface (Fig. 2), or they may select the random options. Assuming search, users enter a textual query that describes their emotional state (e.g. 'having a bad day'), and the emotion detection engine detects the emotion of the query, and the similarity search (retrieval engine) generates a ranked list of music according to the emotion of the query (Fig. 3). In the result interface, users can view more detailed information of the song or click to listen, as well as add it into personal playlist. The EMIR system also collects user clickthrough data and queries to maintain a mood curve of the user over time (Fig 4).

Future work includes exploring fusion of mood and descriptive textual queries as well as a real-time recommendation engine based on the users general textual input to a computer (i.e. constant mood monitoring) or social networking status updates.



I am feeling happy

Fig. 2. Search



Hold Me Close	David Essex	happy,ex
Keep Me Warm	Ella Fitzgerald	happy,wa
warm your heart	Aaron Neville	happy,wa
Shiny And Warm	Goldfrapp	happy,fa
Puppy Love	Donny Osmond	Happy
Shinny Happy People	R.E.M	Happy
River Song	David Essex	Warm,so

Fig. 3. Search Results

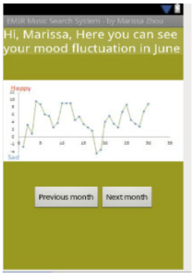

EMIR Music Search System - by Marissa Zhou
Hi, Marissa, Here you can see your mood fluctuation in June
Happy
12
10
8
6
4
2
0
-2
-4
Sad
0
5
10
15
20
25
30
35
Previous month
Next month

Fig. 4. Mood Curve

References

1. Zhou, L.J., Lin, H.F., Liu, W.F.: Enriching Music Information Retrieval Using Emotion Detection. In: SIGIR 2011 Workshop on Enriching Information Retrieval (ENIR 2011), Beijing, China, July 28 (2011)
2. Li, J., Lin, H., Zhou, L.: Emotion tag based music retrieval algorithm. In: Proceedings of The Sixth Asia Information Retrieval Symposium, Taipei, Taiwan, December 1-3, pp. 599–609 (2010)
3. Picard, R.: Affective Computing. MIT Press, Cambridge (2000)
4. Zaanen, M.V., Kanters, P.: Automatic mood classification using tf*idf based on lyrics. In: The 11th International Society of Music Information Retrieval Conference, Utrecht, Netherlands, pp. 75–80 (August 2010)
5. Lamere, P.: Social Tagging and Music Information Retrieval. In: WWW 2010, Raleigh, NC, USA (April 2010)
6. Laurier, C., Sordo, M., Serrá, J., Herrera, P.: Music Mood Representations from Social Tags. In: 10th International Society of Music Information Retrieval Conference, Kobe, Japan, pp. 381–386 (October 2009)

Summarization and Presentation of Real-Life Events Using Community-Contributed Content

Manfred Del Fabro and Laszlo Böszörmenyi

Institute of Information Technology, Klagenfurt University, Austria
{manfred,laszlo}@itec.aau.at

Abstract. We present an algorithm for the summarization of social events with community-contributed content from Flickr and YouTube. A clustering algorithm groups content related to the searched event. Date information, GPS coordinates, user ratings and visual features are used to select relevant photos and videos. The composed event summaries are presented with our video browser.

1 Introduction

Twenty years ago people were informed about a big social event, such as a royal wedding, essentially through a few, authorized, professional camera teams and journalists of printed press. Nowadays, a vast amount of additional photos, videos and corresponding metadata are uploaded to social platforms, such as Flickr and YouTube. If we use these social platforms to get an overview of a specific social event, we receive a - usually extremely long - list of photos or videos. However, long lists are not suited to get a good overview. A compact presentation of a predefined length, which gives us a summary of the event would be desirable. Such a summary should consist of content of good technical quality, high diversity and high coverage [3].

We present an algorithm that summarizes real-life events based on community-contributed content. In a summary photos and videos can be mixed up. The aim is to provide a rich view of the original event that consists of the different views of different people that witnessed the event.

2 Summarization and Presentation of Events

Our algorithm has six input parameters: (1) Search terms that are passed directly to Flickr and YouTube as text queries. (2) The number of streams to be shown in parallel. (3) The maximum duration of the event summary. (4) The name of the location of the event. (5,6) The dates of the lower and the upper bound of the timespan when the content must have been produced.

A summary may consist of more than a single sequence of photos and videos. While videos have a natural length we define a default duration – currently 7 seconds – for still images in the summary. In a single sequence an image needs a rather short time, but as soon as more than one sequence is shown in parallel the viewers need more time to look at all photos shown in parallel.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 630–632, 2012.

The flow chart in Figure 1 illustrates the single steps of our event summarization algorithm. On Flickr we search for photos that were taken within the specified timespan. The YouTube API, unfortunately, does not allow to state a capturing or uploading date for the query. Therefore, we perform a post-processing step where we eliminate those videos that do not fit into the given timespan. The performance penalty for this is still acceptable, as we use only metadata for the post-processing.

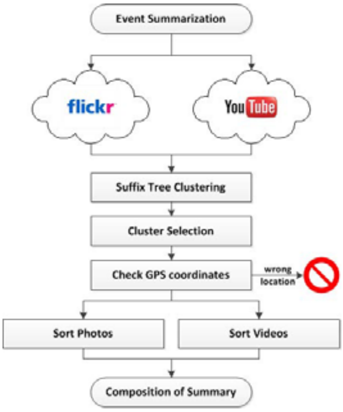


Fig. 1. Flow chart of algorithm

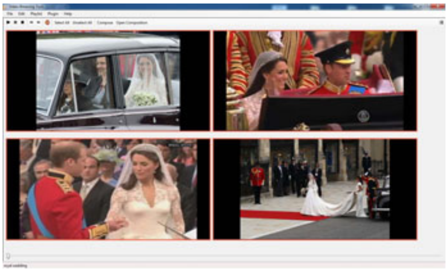


Fig. 2. Screenshot of event summary

We use a text suffix tree clustering algorithm [4] to group the content based on the textural descriptions. This algorithm has already successfully been applied to web document clustering. It is fast, separates relevant from irrelevant content and high quality clusters are produced even if only text snippets are available, which is indeed the case for the metadata of multimedia content. A dominant phrase is generated for each cluster. For the content selection we choose the largest cluster of which the dominant phrase includes the search terms of the query.

The textual descriptions of photos and videos are often misleading. We try to eliminate content produced in a wrong location by investigating the GPS coordinates of the content. The location indicated in textual form is translated into GPS coordinates using the Google Geocoding API[1] and matched against all photos and videos that have associated GPS data.

The selection of photos is based on the number of how frequently a photo has been viewed on Flickr. The selection of videos is based on the user ratings (up to 5 stars), the number of views and the number of "likes" a video has on YouTube. In general, the overall duration when photos are shown is approximately equal to the duration of the videos in the summary. This ratio is automatically adapted if the number of either the photos or the videos is too low. If the cluster selected

[1] Google Geocoding API: `http://code.google.com/apis/maps/documentation/geocoding/`

for the summary contains no videos or if the length of all videos exceeds the maximum duration, no videos are included.

To avoid redundancy we extract the Color and Edge Directivity Descriptor (CEDD) [1] from each candidate image to compare it with all images, which are already in the summary. If the distance to a photo in the summary is too low the candidate image will not be added. For videos the textual descriptions are sufficient to detect redundancy. Finally, when all photos and videos are selected we sort the whole content based on the timestamps.

Our own video browser [2] is used for the presentation of event summaries. The screenshot in Figure 2 shows a summary consisting of four parallel sequences. As parallel audio playback is not desirable, the audio stream of one of the videos is selected (either per default or by pointing at a sub-window).

To demonstrate our results we composed summaries of four well-known social events, which took place in the last three years: (1) the inauguration of Barack Obama, (2) the Royal Wedding of William and Kate, (3) the FIFA World Cup Final 2010 and (4) the UEFA Champions League Final 2011. Screen captures of the four composed event summaries are available online[2].

3 Conclusion and Future Work

In this paper we presented an algorithm for the summarization of real-life events based on community-contributed multimedia content. We used photos from Flickr and videos from YouTube to compose four summaries of events that attracted the attention of a lot of people.

The major future challenge is the temporal alignment of the content. The timestamps from the camera metadata are not sufficient. In our future work we are going to incorporate additional sources of information, like textual descriptions of the events, for the selection and the temporal alignment of content.

References

1. Chatzichristofis, S.A., Boutalis, Y.S.: CEDD: Color and Edge Directivity Descriptor: A Compact Descriptor for Image Indexing and Retrieval. In: Gasteratos, A., Vincze, M., Tsotsos, J.K. (eds.) ICVS 2008. LNCS, vol. 5008, pp. 312–322. Springer, Heidelberg (2008)
2. Del Fabro, M., Schoeffmann, K., Böszörmenyi, L.: Instant Video Browsing: A Tool for Fast Non-sequential Hierarchical Video Browsing. In: Leitner, G., Hitz, M., Holzinger, A. (eds.) USAB 2010. LNCS, vol. 6389, pp. 443–446. Springer, Heidelberg (2010)
3. Sinha, P., Mehrotra, S., Jain, R.: Summarization of personal photologs using multidimensional content and context. In: Proc. of the 1st ACM International Conference on Multimedia Retrieval, pp. 4:1–4:8. ACM, New York (2011)
4. Zamir, O., Etzioni, O.: Web document clustering: a feasibility demonstration. In: Proc. of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 46–54. ACM, New York (1998)

[2] Demo videos: http://soma.lakeside-labs.com/?page_id=279

Tracking Persons in Ultra-HD Panoramic Video

Marcus Thaler, Rene Kaiser, Werner Bailer, and Andreas Kriechbaum*

Institute for Information and Communication Technologies,
Joanneum Research, Graz, Austria
firstname.lastname@joanneum.at

Abstract. We present a demo for person detection and tracking in high-resolution panoramic video streams, obtained from a panoramic camera stitching video streams from 6 HD resolution tiles. The AV content analysis uses a CUDA accelerated feature point tracker, a blob detector and a CUDA HOG person detector, which are used for region tracking in each of the tiles. The results of each tile are then fused for the entire panorama to track persons over multiple tiles.

1 Introduction

The FascinatE system[1] aims to create an innovative end-to-end system for immersive and interactive TV services. The production process operates for a range of viewing devices in parallel, covering anything from a mobile handset to an immersive panoramic display. FascinatE uses a *format agnostic* approach [1], which proposes a paradigm shift towards capturing a format agnostic representation of the whole scene from a given viewpoint, rather than the view selected by a cameraman based on assumptions about the viewer's screen size and interests. In addition to several broadcast cameras, FascinatE uses the Omnicam [1], a high-resolution panoramic camera, which allows cropping interesting regions. The Omnicam is a collection of 6 HD cameras for obtaining a 180° panoramic video sequence stichted together from the 6 tiles.

In order to take reasonable decisions what is currently happening in the scene and which camera streams are capturing that action, automatic content analysis is used. In a sports scenario for example an option could be to follow a certain athlete the majority of time on close-up unless an important action takes place elsewhere in the scene. For such functionality, automatic detection and tracking of objects is necessary.

2 Real-Time Person Tracking

The tracking algorithm has to detect and track persons within real-time constraints over six static and rectified HD image-sequences from the OmniCam. Instead of using the ultra high definition image, each video tile is separately

* The author is now with Austrian Institute of Technology, Vienna, Austria.

[1] http://www.fascinate-project.com/

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 633–635, 2012.

analyzed by different workstations to enable real-time analysis. The two main steps of the person tracking algorithm are implemented by the *region tracker* that detects and tracks persons in the image sequence form one tile, and the *multi-tile tracker* that merges the tracking results from the different tiles. The region tracking algorithm integrates three video analysis methods: a CUDA HOG person-detector, a blob detector and a CUDA point tracker.

To avoid false positives at locations outside the soccer field and to further reduce the calculation time, masks have been created for all camera sequences. For person detection *fastHOG*, a real-time GPU implementation of the HOG [2] algorithm was selected as a basis due to its favorable performance. We modified the scale ratio of the original *fastHOG* implementation to meet our real-time requirement and to extend the scale range in order to detect persons as small as 32 pixels in height. Additionally, to reduce the amount of false positive results, only persons within certain height thresholds at a certain distance to the camera are detected.

To overcome missed detections for situations of sudden movement changes (esp. under presence of motion blur), a blob detector is added to the region tracking. In our system the *OpenCV* blob detector[2] is used. The detector is based on a foreground/background discrimination with a subsequent grouping of adjacent, foreground labeled pixels. The feature point tracker [3] we use for person tracking is a very fast Lucas Kanade algorithm based GPU-feature point tracker. To take full advantage of the capabilities of recent NVIDIA GPUs the feature point tracker is implemented for GPUs based on the recent NVIDIA GPU architecture Fermi [4].

The feature point tracker, the improved *fastHOG* and the blob detector work in parallel, which provides stable tracked feature points and region detections. The results of the person and blob detector for each image of the different image sequences provide the regions of detected persons for further processing. The extracted feature points and regions are combined as a region tracker, as shown in Figure 1. The algorithm linking person IDs to the appropriate combined regions with their corresponding feature points is described in [4]. The algorithm enables to find missed person detections respectively regions by distance clustering.

The tracking system is demonstrated on a single PC with appropriate graphics board, processing a full HD stream in real-time. Results from several streams are collected using a light-weight Web based protocol and integrated, including re-identification of persons moving from one tile to the other. The tracked players are visualised in a custom player that provides functionalities for selecting and followign regions of interest.

3 Discussion

The person multi-tile tracking algorithm for 6 parallel HD image sequences we present performs very close to real-time. For quantitative evaluation we compared the bounding boxes of the ground truth data with the bounding boxes

[2] `http://opencv.willowgarage.com/wiki/VideoSurveillance`

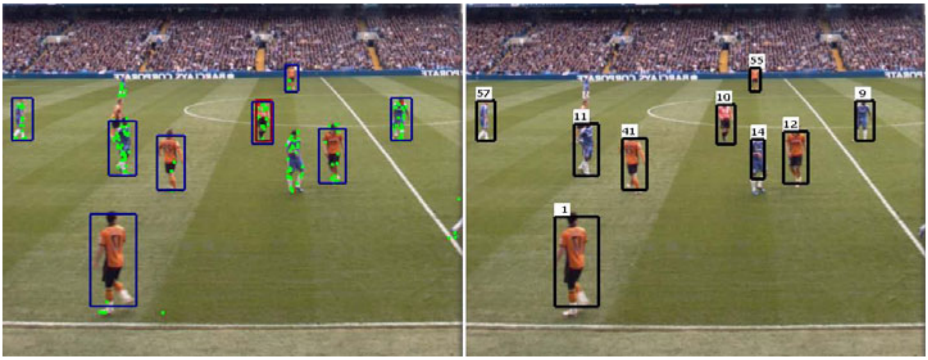


Fig. 1. The left image shows detected person regions and tracked feature points. The resulting tracked persons with their IDs are shown on the right.

of person regions obtained from content analysis. With a bounding box overlap threshold of 25% we calculated an average precision of 95.57% and an average recall of 58.84% for all test sequences. The high precision indicates that the combination of both region detectors and the feature point trackers operates sufficiently for scenes with fewer rapid movements. The lower recall is caused by situations of sudden movement changes and turns of the players. Due to missed detections of the *fastHOG* in situations of sudden turns, the performances of the blob-tracker needs to be enhanced by further modifications. The remaining occlusion issues could be solved by trajectory analysis and verification by color-features.

Acknowledgements. The research leading to this paper has been partially supported by the European Commission under the contract number FP7-248138, "FascinatE – Format-Agnostic SCript-based INterAcTive Experience" (`http://www.fascinate-project.eu/`).

References

1. Schäfer, R., Kauff, P., Weissig, C.: Ultra high resolution video production and display as basis of a format agnostic production system. In: Proceedings of International Broadcast Conference, IBC 2010 (2010)
2. Prisacariu, V., Reid, I.: FastHOG - a real-time GPU implementation of HOG. Technical report, Department of Engineering Science, Oxford University (2009)
3. Fassold, H., Rosner, J., Schallauer, P., Bailer, W.: Realtime KLT Feature Point Tracking for High Definition Video. In: Skala, V., Hildebrand, D. (eds.) GraVisMa 2009 - Computer Graphics, Vision and Mathematics for Scientific Computing (2010)
4. Kaiser, R., Thaler, M., Kriechbaum, A., Fassold, H., Bailer, W., Rosner, J.: Real-time person tracking in high-resolution panoramic video for automated broadcast production. In: Proceedings of CVMP 2011 (2011)

A Real-Time Life Experience Logging Tool

Zhengwei Qiu, Cathal Gurrin, Aiden R. Doherty, and Alan F. Smeaton

CLARITY: Centre for Sensor Web Technologies, School of Computing,
Dublin City University, Glasnevin, Dublin 9, Ireland
{zqiu,cgurrin,adoherty,asmeaton}@computing.dcu.ie

Abstract. E-memories attempt to digitally encode all life experiences in an archive for later search and real-time recommendation. In this paper we describe a prototype real-time e-memory gathering infrastructure and system, that uses smartphones to gather and organise semantically rich e-memory.

1 Introduction

An e-memory is a new concept in digital information management and refers to the digital gathering of life experiences, whether through photos of what we see, videos of what we experience, audio recording of what we hear, or the digital capture of our real-world interactions (e.g. locations, people or actions). Maintaining an e-memory has been alluded to as early as 1945 by Vannevar Bush who envisaged a person wearing a forehead mounted camera [1] to gather life experience. Today, the equivalent device is a SenseCam, a small wearable device that passively captures a person's day-to-day activities as a series of photographs [6]. It is typically worn around the neck and therefore is oriented towards the majority of activities which the user is engaged in. The device incorporates on-board sensors to determine when is appropriate to take a photo. Wearing a device like a SenseCam, a wearer can very quickly build large and rich visual e-memories of millions of photos and hundreds of millions of sensor readings per year.

There has been recent research activity in e-memories with the MyLifeBits project at Microsoft gathering and making searchable, a long-term e-memory (incorporating SenseCams) for one individual [5]. Doherty et al. have developed an event segmentation technique and browsing interface [3] for Sensecam archives. However, one drawback of these systems is that the sensing technology is not real-time because the user needs to upload the content periodically to the e-memory archive before it is processed to support search and retrieval. A real-time system would allow for context-based push retrieval from the e-memory, thereby being more suitable as a real-time memory-aid. De Jager et al. [2], have developed a hardware device to enable real-time capture and feedback of life-experiences.

2 A Real-Time e-Memory Prototype

In this work, we extend prior research by developing a prototype e-memory solution that operates on smartphones to gather data which is then semantically

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 636–638, 2012.

enriched using both physical and virtual sensors, thereby providing effective search and retrieval facilities in real-time. The prototype system incorporates a smartphone for data capture (worn like a SenseCam passively capturing photos every minute), software for the segmentation and annotation of life experiences, a server for storage of e-memories and a WWW front end to the server. There are a number of key elements that are needed to achieve the e-memory functionality, from life-experience capture using wearable sensors, to experience segmentation and semantic experience annotation through to search support and user interaction. We will discuss each of these elements of our prototype individually below.

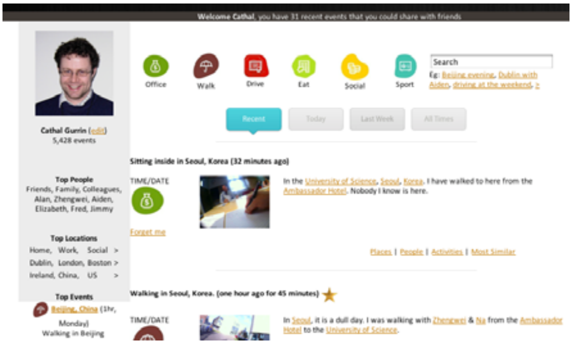
Sensor Capture from Wearable Sensors. A smartphone includes sensors that can capture a rich life-experience archive (just like a SenseCam). We mine data constantly from onboard physical sensors; accelerometer, compass, camera, GPS, Bluetooth, microphone, WiFi and communication/media activity sensors. Used alone, such raw readings do not provide much semantic value, but the semantic analysis (below) enriches the collection for enhanced search functionality.

Experience Segmentation. Typically, in a full day, we know that a person encounters more than 20 individual *events*, with each event lasting 30 minutes on average [4]. Prior work on event segmentation analyses the sensor streams from SenseCams to generate a segmentation of life-experience into events, post-capture [4]. In this work we take this approach of mining events from sensor streams, but migrate the processing to the smartphone to operate in real-time and upload events to the e-memory archive as they happen.

Semantic Analysis Engine for Sensor Streams. The output of the sensor capture consists of raw sensor streams, as described above. To support real-time analysis, both server-side and smartphone semantic analysis tools are needed; these act as virtual (software) sensors to enrich the raw sensor streams with semantically meaningful annotations. For example, using raw accelerometer values, we can identify the physical activities of a user [7]. We utilise the following virtual sensors; semantic date/time, meaningful location, personal physical activity, social interactions, environmental context, semantic visual concepts automatically identified from the photos and personal context of the user's life pattern. Using these sources, we semantically enrich the annotations of events and construct a narrative to describe each event, needed for both search and presentation.

Indexing, Retrieval and Presentation. In order to retrieve life experiences for search or recommendation, either later or in real-time, the experiences and their annotations need to be indexed. In this work we employ a standard search engine to index the annotations for every life-experience data and provide keyword search through the e-memory archive, ranking and presenting the multimedia rich life experience data to the user through a WWW interface.

The prototype utilises a smart-selection algorithm to upload events to the server (for indexing and retrieval) in real-time across a wifi or 3G network. If a known wifi network is not available, a minimal event representation is uploaded using 3G, with the remaining data uploaded via wifi later. The WWW interface supports multimodal access and provides search and interaction functionality, as shown in Figure 1.



Welcome Cathal, you have 31 recent events that you could share with friends

Sitting inside in Seoul, Korea (32 minutes ago)

In the University of Science, Seoul, Korea. I have walked to here from the Ambassador Hotel. Nobody I know is here.

Walking in Seoul, Korea. (one hour ago for 45 minutes) ★

In Seoul, it is a dull day. I was walking with Zhengwei & Na from the Ambassador Hotel to the University of Science.

Fig. 1. WWW interface to an E-memory Archive

3 Conclusions

We have presented a real-time e-memory prototype that gathers life-experiences using mobile devices and stores them in a server-based e-memory which supports search and retrieval. The software is operational and gathers data for the e-memory using android smartphones. Work is ongoing to develop context-aware push e-memory recommendations and real-time search from the mobile device.

Acknowledgements. The authors thank Science Foundation Ireland (grant No. 07/CE/I1147) and (AD) the Irish Health Research Board (grant MCPD/2010/12).

References

1. Bush, V.: As We May Think. The Atlantic Monthly (1945)
2. de Jager, D., Wood, A.L., Merrett, G.V., Al-Hashimi, B.M., O'Hara, K., Shadbolt, N.R., Hall, W.: A low-power, distributed, pervasive healthcare system for supporting memory. In: Proceedings of the First ACM MobiHoc Workshop on Pervasive Wireless Healthcare, MobileHealth 2011, Paris, France (2011)
3. Doherty, A.R., Moulin, C.J., Smeaton, A.F.: Automatically assisting human memory: A SenseCam browser. Memory, 1 (2010)
4. Doherty, A.R., Smeaton, A.F.: Automatically segmenting lifelog data into events. In: Ninth International Workshop on Image Analysis for Multimedia Interactive Services, pp. 20–23. IEEE (2008)
5. Gemmell, J., Aris, A., Lueder, R.: Telling Stories with MyLifeBits. In: IEEE International Conference on Multimedia and Expo, ICME 2005, vol. 06(06), pp. 1536–1539 (July 2005)
6. Hodges, S., Williams, L., Berry, E., Izadi, S., Srinivasan, J., Butler, A., Smyth, G., Kapur, N., Wood, K.: SenseCam: A Retrospective Memory Aid. In: Dourish, P., Friday, A. (eds.) UbiComp 2006. LNCS, vol. 4206, pp. 177–193. Springer, Heidelberg (2006)
7. Qiu, Z., Gurrin, C., Doherty, A.R., Smeaton, A.F.: Mining user activity as a context source for search and retrieval. In: International Conference on Semantic Technology and Information Retrieval, Kuala Lumpur, Malaysia (June 2011)

AAU Video Browser: Non-Sequential Hierarchical Video Browsing without Content Analysis

Manfred Del Fabro and Laszlo Böszörmenyi

Institute of Information Technology, Klagenfurt University, Austria
{manfred,laszlo}@itec.uni-klu.ac.at

Abstract. We participate in the Video Browser Showdown with our easy-to-use video browsing tool. It can be used for getting a quick overview of videos as well as for simple Known Item Search (KIS) tasks. It offers a parallel and a tree-like browsing interface for navigating through the content of single videos or even small video collections in a hierarchical, non-sequential manner. We want to validate whether simple KIS tasks can be completed without a time consuming content analysis in advance.

1 Introduction

We have already introduced our tool for instant video browsing that requires no content analysis at all [1]. This tool is well suited for scenarios where users just quickly want to get an overview of a video or to find specific, already known segments in it. The latter is the case at the Video Browser Showdown.

Usually, video browsing solutions are based on content analysis of the underlying video. Almost all presented solutions perform a shot segmentation first and provide browsing mechanisms based on the shot structure. While such tools typically perform better in retrieval tasks, the content analysis step requires a lot of processing time.

With our participation in the Video Browser Showdown we want to show that in Known Item Search (KIS) scenarios it is an overkill to perform a deep content analysis. For such scenarios it is better to provide quick, yet powerful, interactive navigation means.

2 The AAU Video Browser

At the moment our video browser offers two different views: a *parallel* and a *tree-based view*. In both views every video is divided into n parts of equal length. The number of parts (n) can be increased or reduced with a single click.

The parallel view is shown in Figure 1, where an example video is divided into twelve parts of equal length. The user can navigate one level down in the browsing hierarchy by clicking with the right mouse button on one of the video windows. The selected part of the video is divided into n parts of equal length

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 639–641, 2012.

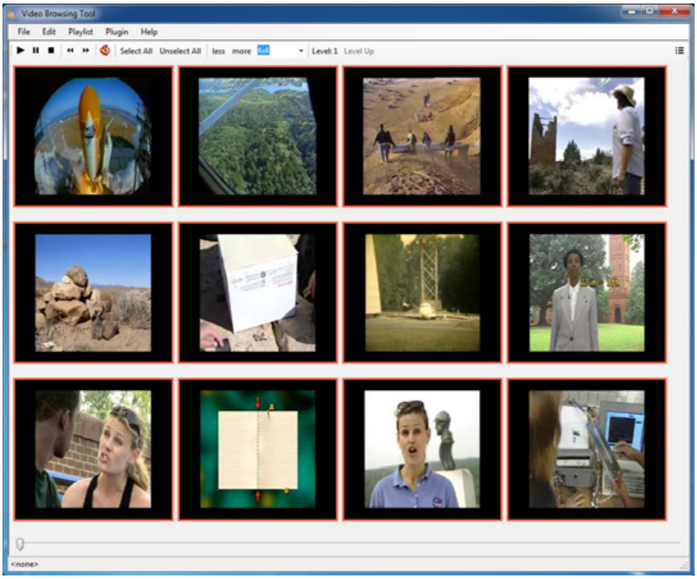


Fig. 1. Parallel View

again. To get a coarser view again, it is possible to go back to a higher level. The parallel view can only show one level of the browsing hierarchy at a glance.

In contrast the tree-based view presents all levels simultaneously in a tree-like structure. The context of the shown parts is better preserved. Figure 2 shows a screenshot of the tree-based view. Each row represents one level of the browsing hierarchy. The browsing history from the top to the bottom level is preserved by coloring the selected parts on each layer with a green border. Therefore, alternative browsing paths can be found quickly. If a part is selected with the right mouse button, a new row showing only that part is added to the view. Browsing through a video this way can be compared with navigating through a tree structure. In Figure 2 the first part of each level has been selected.

In both views it is possible to traverse the content of the video in a hierarchical way down, until the frame level is reached, and up again.

Beyond hierarchical browsing, our video browser can also be used for parallel playback of the content. All parts of a video or only selected ones can be watched in parallel. The playback speed can be adjusted, thus allowing a fast forward of all parts shown. The slider at the bottom of the video browser can be used to scroll through selected parts in parallel. Users can get an impression of the whole video in a fraction of the overall duration. The audio playback is only enabled for one selected video window (where the mouse points at). The ability to play only the audio stream of the part regarded to be interesting, helps the users in getting a better browsing experience.

Both introduced views are not only limited to single video files. They can be used to explore small video collections as well. Opening a folder that contains

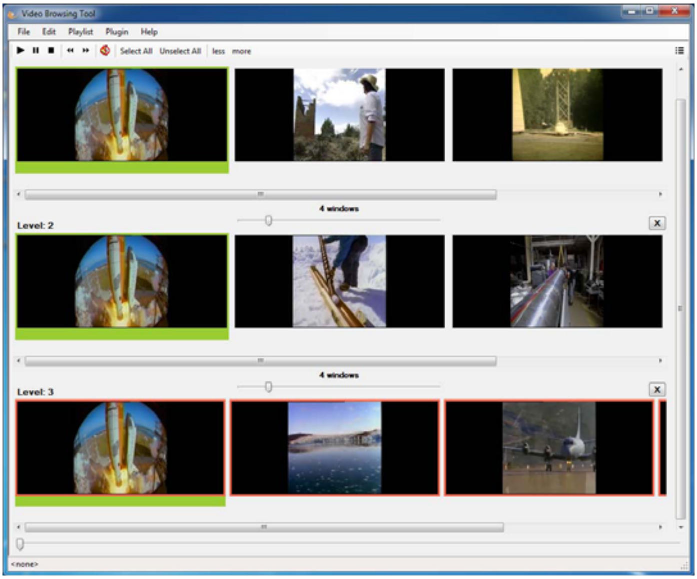


Fig. 2. Tree-Based View

several videos adds an additional level to the browsing hierarchy. On the top level all videos of the selected folder are shown, serving as starting point for a hierarchical exploration of the whole video archive.

3 Conclusion

Our tool provides easy-to-use interaction means for non-sequential hierarchical video browsing. The parallel view can be used to get an overview of the content of a video by using parallel playback or parallel scrolling. The tree view provides mechanisms for quickly exploring different search paths within a video and thus it is better suited for Known Item Search (KIS) tasks. Our video browser is suggested particularly for situations in which video analysis is not adequate (e.g. due to lack of rich semantics) or would take too much time. At the Video Browser Showdown we want to validate that. Nevertheless, as our video browser implements a plug-in architecture, it is possible to extend it in future with additional views or with video analysis plug-ins, e.g. for video segmentation.

Reference

1. Del Fabro, M., Schoeffmann, K., Böszörmenyi, L.: Instant Video Browsing: A Tool for Fast Non-sequential Hierarchical Video Browsing. In: Leitner, G., Hitz, M., Holzinger, A. (eds.) USAB 2010. LNCS, vol. 6389, pp. 443–446. Springer, Heidelberg (2010)

Video Browser Showdown by NUS

Jin Yuan, Huanbo Luan, Dejun Hou, Han Zhang,
Yan-Tao Zheng, Zheng-Jun Zha, and Tat-Seng Chua

School of Computing, National University of Singapore, Singapore
{yuanjin,zhazj,chuats}@comp.nus.edu.sg,
{luanhuanbo,houdejun214,zhanghan8788,yantaozheng}@gmail.com

Abstract. The known item search task (KIS) aims to retrieve a unique video or video clip in the video corpus. This paper presents a novel interactive video browsing system for KIS task. Our system integrates visual content-based, text-based and concept-based search approaches. It allows users to flexibly choose the search approaches. Moreover, two novel feedback schemes are employed: first, users can specify the temporal order in visual and conceptual inputs; second, users can label related samples with respect to visual, textual and conceptual features. Adopting these two feedback schemes greatly enhances search performance.

1 Introduction

Recently, a new video search task, named the "*Known Item Search*" (KIS), has been proposed to simulate a real-world video search scenario [1]. In KIS task, users aim to find a desired video or video clip that has been seen and known before by mining their memory. Compared to the typical video search task, finding relevant sample in KIS task is more difficult for two reasons: first, there is only one right answer for each user's query; and second, users may not be able to completely describe the content of the desired video or video clip according to their memory, especially when it happened a long time ago.

This paper presents a multifunctional and friendly video browsing system for KIS task. The system assembles the visual content-based, text-based and concept-based approaches. Users can flexibly select one or more approaches to search video results according to their memory. Moreover, our system supports the use of temporal sequence and related samples to enhance the search performance.

2 Video Browsing System

2.1 Framework

Often, a user only remembers a part of visual scenes or a part of textual information in the desired video. Therefore, it is necessary to provide a multi-modality search platform as shown in Figure 1. Users can input a visual, textual, or conceptual query. These different queries are respectively used by a visual model,

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 642–645, 2012.

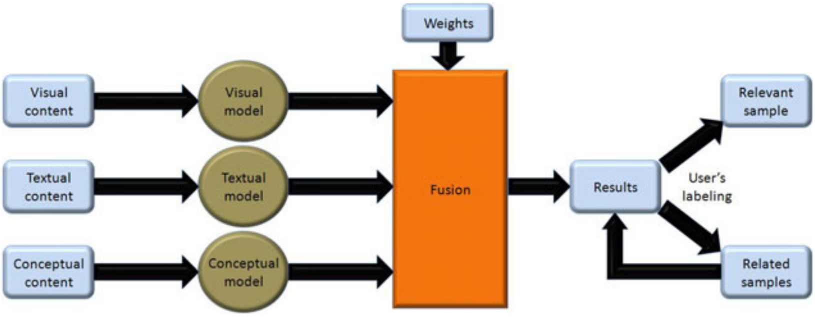


Fig. 1. The framework of our video browsing system

textual model, or conceptual model to retrieve the results. The final search results are then generated by fusing the individual results from these models, where the fusion weights are manually set by the users. Browsing the search results, users can label "*related samples*" [2] (the samples that are similar to the relevant samples), or renew queries to find the desired video.

2.2 Visual Content-Based Search

When users remember some visual scenes of the desired video in their memory, they can draw visual content on a sketchpad to form a visual query Q^v. For each visual query Q^v and each keyframe K_i in the dataset, it is divided into multiple visual blocks, where each block B_j^v (B_j^i) is represented as a feature vector $f(B_j^v)$ ($f(B_j^i)$) based on Color feature. The visual matching score $R(Q^v, K_i)$ between Q^v and K_i is calculated based on cosine distance as:

$$R(Q^v, K_i) = \frac{1}{J} \sum_{j=1}^{J} Cos(f(B_j^v), f(B_j^i)) \qquad (1)$$

where J is the number of the blocks in Q^v drawn by the users.

To enhance search performance, users can draw multiple visual contents and specify the temporal order among them. This temporal relationship can be used to enhance the prediction accuracy. Let $\mathcal{Q} = \{Q_1^v, Q_2^v, \ldots, Q_M^v\}$ be the sequence of visual contents drawn by the users, where Q_m^v occurs before Q_{m+1}^v. The relevance score of K_i with respect to $\mathcal{Q}$ is calculated by considering two factors: the visual matching score of K_i to one of the visual contents ($R(Q_m^v, K_i)$ in Eq (2)); and the visual matching scores of its temporal neighboring keyframes to the other visual contents. Here, for the other $M-1$ visual contents, we select $M-1$ keyframes from the $2W$ neighboring keyframes to maximize the value of relevance score. Moreover, we require the temporal order of visual contents to be consistent with that of the selected keyframes (see the constraints in Eq (2)).

$$\begin{aligned} R(\mathcal{Q}, K_i) = & \max_{1 \le m \le M} \{ \max_{p_b} \prod_{b=1}^{m-1} R(Q_b^v, K_{p_b}) \cdot R(Q_m^v, K_i) \cdot \max_{p_a} \prod_{a=m+1}^{M} R(Q_a^v, K_{p_a}) \} \\ s.t. \quad & p_b \in \{i-W, i-W+1, \ldots, i-1\} \quad p_a \in \{i+1, i+2, \ldots, i+W\} \\ & p_1 < p_2 < \ldots < p_{m-1} < p_{m+1} < p_{m+2} < \ldots < p_M \end{aligned} \tag{2}$$

2.3 Text-Based Search

The system allows users to select the categories for the desired video. Through the user interface, user can specify whether the desired video clip contains speech or subtitle. In addition, users can also indicate the semantic category which the desired video clip belongs to. We define seven semantic categories $\{C_k\}_{k=1}^{7}$ ("*Music*", "*Entertainment*", "*Education*", "*Science*", "*Comedy*", "*News*", "*Cartoon*") suggested by the YouTube website. For each category C_k, we downloaded the tag files of the top 100 videos from YouTube. These tag files were merged and expressed as a normalized text vector T_k^c. Furthermore, for each video clip V_n in the dataset, we extracted textual words by ASR and OCR. After filtering stop words, V_n is represented as a normalized text vector T_n^v. Finally, we calculate a relevance score between V_n and C_k by considering two factors: The semantic closeness of V_n to C_k which is measured by the Google distance between T_n^v and the category name C_k; and the co-occurrence between the text words from V_n and C_k which is measured by the cosine distance between T_n^v and T_k^c. We balance these two factors with a weight parameter. When users select one category, the system will rank the results according to the relevance scores.

2.4 Concept-Based Search

Users can express their query as a sequence of concept bundles [3], and specify the temporal order among them. For example, the query {("*car*", "*sky*"), ("*lady*", "*singing*")} describes the desired video containing at least two shots: one containing the concepts "*car*" and "*sky*", that occurs before the other one containing both "*lady*" and "*singing*". For each concept bundle in the query, we calculate a relevance score for each keyframe by the concept-based video search approach [3]. The relevance score of a keyframe to the query (a sequence of concept bundles) is calculated by the same approach as Eq (2).

2.5 Related Sample-Based Search

While browsing the search results in the interface, users can label "*related samples*" with respect to three features (visual content, text, or concept). For example, user can label a keyframe as visually related sample if he thinks that it is visually similar with one of the keyframes in the correct video. To distinguish different kinds of related samples, we allocate three areas in the interface, and users can drag one kind of related samples from search results to the corresponding area. Furthermore, for each keyframe in the dataset, we previously found its K nearest samples according to the three features respectively. Once the users click a related sample, its K nearest samples are shown in the interface.

References

1. Chen, X.Y., Yuan, J., et al.: TRECVID 2010 Known-item Search by NUS. In: TRECVID Workshop (2010)
2. Yuan, J., Zha, Z.-J., et al.: Utilizing Related Samples to Enhance Interactive Concept-based Video Search. IEEE Transactions on Multimedia (2011)
3. Yuan, J., Zha, Z.-J., et al.: Learning Concept Bundles for Video Search with Complex Queries. In: Proc. of ACM Int. Conf. on Multimedia (2011)

Clipboard: A Visual Search and Browsing Engine for Tablet and PC

David Scott, Jinlin Guo, Hongyi Wang, Yang Yang,
Frank Hopfgartner, and Cathal Gurrin

Dublin City University,
Glasnevin, Dublin 9, Ireland
{dscott,jguo,yang.yang,frank.hopfgartner,cgurrin}@computing.dcu.ie,
hongyi.wang3@mail.dcu.ie

Abstract. In this work, we present a handheld video browser that utilizes two methods of search; Concept Search and Keyframe Similarity. Concept Search allows a user to define a query using selected visual concepts and presents the user with a cluster of video segments based on extracted image features using OpponentSIFT. Keyframe Similarity has a dependance on the previous search for input criteria, allowing a user to select a keyframe for similarity search, returning three types of results; local keyframes from the current scene, global shot similarity based on visual features and text similarity of shots, based on frequently occurring words generated from ASR transcripts.

Keywords: Multi-modal Access, tablet pc, visual concept, keyframe similarity.

1 Introduction

Having been involved in TRECVid since its beginnings, participating in Ad-hoc search, Instance Search and most recently in the Known Item Search task [2], we have developed many video browser systems for evaluation. In this work, we present a handheld video search and browsing engine that integrates shot boundary detection, optimal keyframe extraction, scene detection, concept-based querying, keyframe browsing and three-way similarity search to allow a user to locate a known video item with minimum input and in as short a time as possible.

In this work our chosen platform is a tablet PC, which can be either an iPad or any Android tablet. The interface is developed in HMTL 5, thereby allowing cross-platform deployment. The user is presented with an interface which has a title bar (top of the screen, permanently visible) containing a set of concepts that help to partition the collection. Users select concepts to build a visual query, this visual query returns a ranked list of shots which are displayed in descending order of relevance. The user can either select a keyframe to determine if it is correct, select other concepts to refine the search or do a similarity search on the selected keyframe to obtain items with similar visual, textual features or keyframes from the same video segment. At the point when the user has found

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 646–648, 2012.

the required video segment, s/he will tag the video segment and move onto process another information need. There are a number of key search/browsing techniques that our tablet search engine implements:

- **Concept Search:** Models are trained to recognize real-world entities such as *Person, Vehicle, Building* etc, from the source video. These extracted keyframes are compared to these models and given a probability of containing query defined concept features and the engine ranks the results on this probability in descending order.
- **Keyframe Browsing:** The results returned to the user are in the form of a keyframe browser, in ranked order. The user may browse through these results attained through searching as they will contain the entire collection of keyframes.
- **Similarity Search:** Upon selection of a keyframe that seems similar to the user information need, the user is presented with a tabular view of keyframes within the same video scene, a list of keyframes containing comparative low-level features and a list of keyframes which share similar textual features based on ASR keywords.

In the following section we will discuss in more detail the underlying technologies that support the segmentation, searching and browsing functionality.

2 Technical Components

2.1 Constituent Video Segmentation

In order to facilitate easy browsing through the video, we have implemented a shot boundary detection algorithm and a scene segmentation algorithm. Keyframes are selected to represent each video shot by calculating the most average frame by determining the average vector representation of the MPEG-7 descriptors and finding the closest frame to it. Scene segmentation enables the browsing through an entire scene associated with any selected video shot, which a user can do when exploring shot similarity. This is achieved by making the assumption that unrelated scenes have a significantly different representation of the audio signal. Therefore, we determine the Mel-Frequency Cepstrum Coefficients (MFCCs) of the video's audio layer and segment the video by identifying the neighbouring coefficients which have a delta above a threshold.

2.2 Concept Search

Concept-based search is a effective method to bridge the gap between low-level features and high-level semantics. A series of concept detectors were trained by using a SVM framework and the popular Bag-of-Visual-Word (BoVW) model for keyframe-visual-content representation. More specifically, in the BoVW model, we extract OpponentSIFT feature; then k-means clustering is used for construction of a visual vocabulary with 1024 visual words. Finally, for each keyframe, a

1024-dimension histogram is generated by summing all the occurrences of each cluster (visual word), using the nearest neighbour centroid for each extracted feature. In the SVM, the χ^2 kernel is used since it achieves better performance when comparing with other kernels for concept detection. By employing concept search, a ranked list is generated for each user query that ranks the *entire collection of keyframes* for rapid browsing, hence potentially shots will be highlighted by pre-calculated using the OpponentSIFT features visual descriptors mentioned above. The ranked list is likely to be long, so the user is able to navigate through the keyframes by using swipe gestures.

2.3 Similarity Search

Similarity search is a secondary search technique, which is activated once a user selects a keyframe from the ranked list as form of relevance feedback. Similarity search returns a tabbed view of:

- **Local keyframes** within the video segment in temporal order.
- **Globally similar keyframes** based on a fusion of MPEG-7 descriptors; edge, color histogram and scalable color, which produces a ranked list of keyframes in decreasing order of similarity.
- **Textual similarity** based on precomputed similarity of all shots, calculated using conventional text IR techniques operating on the ASR transcripts of the video shots.

3 Conclusion

In conclusion, this paper presents a visual search and browsing engine for hand-held devices. This engine has been designed for situations in which a user needs to locate known items in a short time period. Key features of the engine are shot/scene boundary detection, concept-based search, keyframe browsing and three-way similarity search.

Acknowledgements. The research was funded by iAD - information Access Disruptions, a centre for research-based innovation with CRI number: 174867, funded in part by the Norwegian Research Council.

References

1. van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluating Color Descriptors for Object and Scene Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(9), 1582–1596 (2010)
2. Foley, C., Guo, J., Scott, D., Ferguson, P., Gurrin, C., Smeaton, A.F.: TRECVid 2010 Experiments at Dublin City University. In: TRECVid 2010 - Text REtrieval Conference TRECVid Workshop, Gaithersburg, MD (2010)

OVIDIUS: A Web Platform for Video Browsing and Search

Andrei Bursuc[1,2], Titus Zaharia[1], and Françoise Prêteux[3]

[1] Institut Télécom, Télécom SudParis, ARTEMIS Department, UMR CNRS 8145 MAP5,
9 rue Charles Fourier, 91011 Evry Cedex, France
{Andrei.Bursuc,Titus.Zaharia}@it-sudparis.eu
[2] Alcatel-Lucent Bell Labs France, route de Villejust, 91620 Nozay, France
[3] Mines ParisTech, 60, Boulevard Saint-Michel, 75272 Paris Cedex, France
Francoise.Preteux@mines-paristech.fr

Abstract. This paper presents the interactive web-based video retrieval platform so-called OVIDIUS (*On-line VIDeo Indexing Universal System*). OVIDIUS makes it possible to browse and access video content in a fine, per segment basis. The hierarchical metadata exploits the MPEG-7 approach for structural description of video content by decomposing the video in scenes, shots and keyframes. The user has the possibility to select objects and regions of interest from the video and retrieve them. The developed approach shows all its pertinence within a multi-terminal (both fixed and mobile) context due to its implementation based open web technologies.

Keywords: MPEG-7 visual descriptors and descriptions schemes, video indexing/retrieval, multiple instance detection, partial matching, web services.

1 Introduction

A successful video web platform should integrate effectively navigation, search and retrieval features within an ergonomic and user-friendly interface. The key issue relates to the way in which the interactivity between user and system is handled. Notably, the navigation and search functionalities should be centered on the user, displaying at any time a sufficient amount of content such that the user's attention span is not overwhelmed. In addition, the querying process should be performed in an intuitive manner, by letting the user select areas of arbitrary size and shape, according to his personal elements of interest. The OVIDIUS platform [1] achieves a good balance between such features in an accessible web-based environment.

2 Video Browsing

The video browsing process is based on the MPEG-7 structural approach for video description [2], which relies on the abstract concept of *segment*, defined as a generic piece of an audio-visual document. Individual, specialized segments are then defined with the help of an inheritance mechanism in order to create description schemes (DS) adapted to each type of media. Five MPEG-7 DSs are currently supported: Video Segment DS, StillImage DS, AudioSegment DS, StillRegion DS, and MovingRegion

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 649–651, 2012.

DS. In combination with a recursive decomposition mechanism (which may be temporal, spatio-temporal or spatial), such an approach makes it possible to create a hierarchical and multi-granular video content description, adapted to both navigation/browsing and search functionalities.

The OVIDIUS GUI makes it possible to obtain a visual and interactive representation of the MPEG-7 descriptive structure and integrates the following components: video player, selector of the hierarchical level of each segment, rows of iconic representations of segments including the display of basic information such as (identifier, time stamp, order, type), navigation buttons and a timeline scrollbar for instant access to scenes from different parts of the video (Fig. 1). A color code is used in order to inform the user about the current hierarchical level of access. This menu is aiming to offer a complete user experience when searching and accessing media. While visualizing the video, the user can simultaneously skim through neighboring scenes and shots on different hierarchical levels.

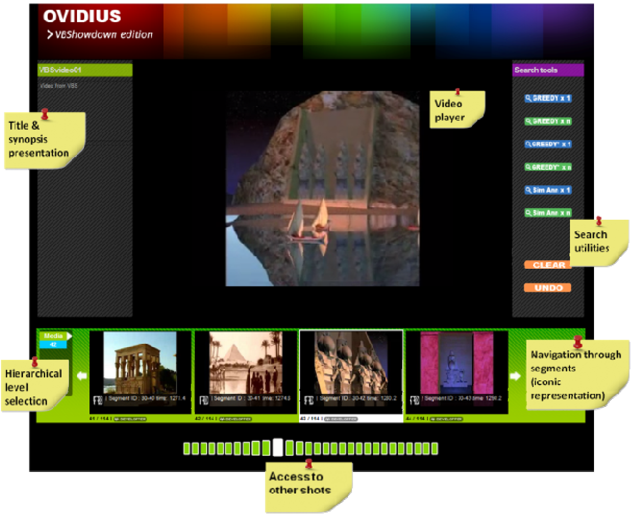


Fig. 1. Navigation interface of the OVIDIUS platform

The user can also add another navigation row of iconic representation from a different hierarchic level of segmentation; similar with the navigation threads introduced in [3]. One thread contains the scene previews, while the other one displays the previews of the shots corresponding to the current scene.

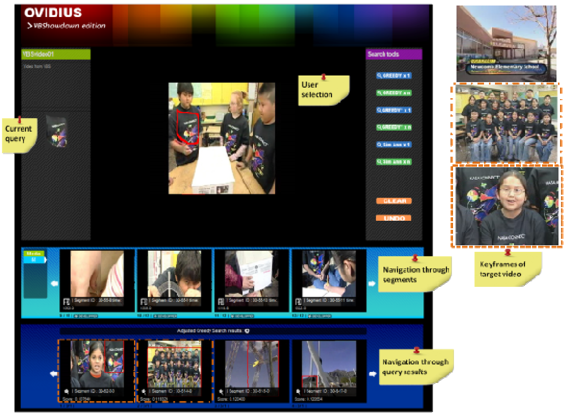
3 Search Functionalities

In addition to global content-based retrieval functionalities, which exploit the set of MPEG-7 visual descriptors, OVIDIUS provides a region-based, partial search module [4]. Regions and objects of interest can be interactively selected by the user and retrieved inside the video. The retrieval results are displayed in decreasing similarity order in a dedicated thread. Let us note that the user can continue his navigation on the other threads, while the system processes the query.

The search mechanism is based on an over segmentation of the keyframes giving for each keyframe up to 200-300 regions of homogeneous colors. The set of colors, together with their percentage of occupation in the image are regrouped into a visual representation, which extends the MPEG-7 dominant color descriptor. A dynamic region construction algorithm generates for each frame different configurations of color regions aiming to obtain candidate objects the most similar to the given query model. The different region construction algorithms are described in details in [4].

4 Video Browser Showdown Use-Case

Given the short video sequence to find in the current video, the user can start browsing the scenes and shots looking for the query video or for objects or textures similar with the ones in the query (*e.g.* the T-shirts of the children in the example presented in Fig. 2). Once such a frame is found, the user can select the area of interest and ask the system to retrieve it. When the query processing ends, keyframes from different moments of the video, containing similar patterns or objects can be accessed instantly or can be used as additional queries to narrow the area of search.



Current query

User selection

Navigation through segments

Navigation through query results

Keyframes of target video

Fig. 2. OVIDIUS video retrieval use-case

References

1. Bursuc, A., Zaharia, T., Prêteux, F.: Mobile Video Browsing and Retrieval with the OVIDIUS Platform. In: Proc. ACM Multimedia 2010 International Conference, Florence, Italy, pp. 1659–1662 (October 2010)
2. ISO/ IEC 15938-5:2003, Information technology - MultimediaContent Description. Interface - Part 5: Multimedia Description Schemes (2003)
3. De Rooij, O., Snoek, C.G.M., Worring, M.: Balancing thread based navigation for targeted video Search. In: Proceedings of the ACM International Conference on Image and Video Retrieval, Niagara Falls, Canada, pp. 485–494 (July 2008)
4. Bursuc, A., Zaharia, T., Prêteux, F.: Retrieval of Multiple Instances of Objects in Videos. In: Schoeffmann, K., et al. (eds.) MMM 2012. LNCS, vol. 7131, pp. 358–369. Springer, Heidelberg (2012)

Hierarchical Navigation and Visual Search for Video Keyframe Retrieval

Carles Ventura, Manel Martos, Xavier Giró-i-Nieto,
Verónica Vilaplana, and Ferran Marqués

Technical University of Catalonia (UPC), Barcelona, Catalonia, Spain
{carles.ventura,xavier.giro,veronica.vilaplana,ferran.marques}@upc.edu

Abstract. This work presents a browser that supports two strategies for video browsing: the navigation through visual hierarchies and the retrieval of similar images. The input videos are firstly processed by a keyframe extractor to reduce the temporal redundancy and decrease the number of elements to consider. These generated keyframes are hierarchically clustered with the Hierachical Cellular Tree (HCT) algorithm, an indexing technique that also allows the creation of data structures suitable for browsing. Different clustering criteria are available, in the current implementation, based on four MPEG-7 visual descriptors computed at the global scale. The navigation can directly drive the user to find the video timestamps that best match the query or to a keyframe which is globally similar in visual terms to the query. In the latter case, a visual search engine is also available to find other similar keyframes, based as well on MPEG-7 visual descriptors.

Keywords: Video browser, Hierarchical Navigation, Image retrieval.

1 Introduction

As a consequence of recent technology development, large amounts of video data are generated and stored. Accessing these rich portions of data in terms of audiovisual and semantic content is an open research issue with several solutions depending on the user needs. This work proposes a hybrid interface that combines both navigation and search tools to provide users with a high degree of flexibility for choosing the most appropriate strategies according to the query nature. Figure 1 shows a screenshot of GOS (Graphic Object Searcher), the GUI that exploits the presented techniques, after a query search based on color.

2 Hierarchical Cellular Tree

The Hierarchical Cellular Tree (HCT) algorithm [1] was designed to bring an effective solution for indexing large multimedia databases. The elements are partitioned and stored within cells on the basis of their similarity. As its name implies, HCT is a hierarchical structure, which consists of one or more levels,

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 652–654, 2012.

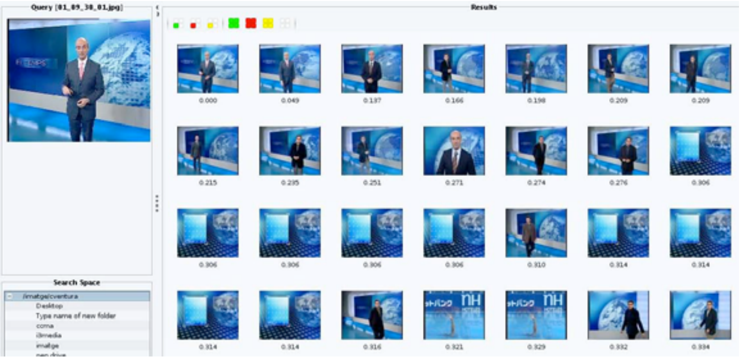


Fig. 1. Screenshot of GOS, the graphical user interface

and each level in turn holds one or more cells. Each cell has an element as a nucleus, which is contained in a cell in the upper level except for the cell held by the top level. These representative elements are used during the top-down search for item insertions and query requests. Another dynamic cell feature is the cell compactness, which quantifies how focused or compact the clustering for the items within the cell is.

The hierarchical structure of the HCT allows a multiscale visual navigation since the nucleus of each cell is representative of the underlying elements. Given a level in the HCT, the set of thumbnails built from the nucleus provides the user with visual and intuitive data to decide what path offers greater chances to find the keyframes that better match the query.

In [2], we have implemented the HCT in order to index image databases based on the following MPEG-7 visual descriptors: (*i*) Color Structure Descriptor, (*ii*) Dominant Color Descriptor, (*iii*) Color Layout Descriptor, and (*iv*) Texture Edge Histogram Descriptor. Therefore, the similarity of the elements are computed according to their respective dissimilarity functions also defined in the MPEG-7 standard [3]. Moreover, we have introduced some modifications to the original HCT which have resulted in a better performance of the retrieval system. Specifically, we have redefined the covering radius cell parameter and we have designed a retrieval scheme based on the Preemptive Cell Search algorithm, which is the technique on which the insertion operation is based.

3 Visual Search

The proposed system contains a second tool for keyframe search based on visual similarity. Any keyframe can be selected to formulate a query-by-example among the rest of keyframes. This tool helps users to find similar portions of the video.

The same HCT structure used for navigation can also be used in visual search at global scale as an index for fast retrieval. This way the same data structure can be exploited manually, through navigation, or by an automatic visual search engine.

4 Video Browser Approach

The problem we aim to solve consists in finding a preselected segment of interest in a video file by interactive search. We have to consider that the preselected segment of interest, i.e. the query clip, is not available for the retrieval system, so the query clip cannot be processed. Therefore, we have to find it by navigating through the video file to which the query clip belongs to.

With this purpose, we have adopted the following strategy, where the first 1-3 steps are performed offline to speed up the retrieval process:

1. The test video is previously processed by a keyframe extractor in order to work with a lower number of frames which represent the video.
2. Global features are extracted for each keyframe, in particular, the following MPEG-7 visual descriptors [3]: (*i*) Color Structure, (*ii*) Dominant Colors, (*iii*) Color Layout, and (*iv*) Texture Edge Histogram.
3. A HCT is built over each of the extracted visual descriptors, considering as a similarity metric the visual distances recommended in MPEG-7.
4. Depending on the query clip, the user decides on which visual descriptor the navigation must start. The GUI shows to the user all the elements belonging to the level of the hierarchy whose thumbnails fill the screen. Then, the user selects in which element (actually, in which subtree associated to that element) is interested. As a consequence, the elements belonging to the level below the selected one are shown to the user, who decides whether descending through that subtree or coming back to the upper level.
5. If the navigation drives the user to a keyframe which he decides is globally similar to any of the frames in the segment of interest, the user can launch a visual search process by selecting *Search for similar images* on a pop-up menu which appears with a right-click.
6. If the search drives the user to a keyframe of the segment of interest, the timestamp of the keyframe found is used to retrieve their neighbouring keyframes in time by selecting *Go to temporal segment* on a pop-up menu.
7. The user finally selects the first and final keyframe of the segment of interest.

Acknowledgements. This work was partially founded by the Catalan Broadcasting Corporation through the Spanish project CENIT-2009-1026 BuscaMedia, by TEC2010-18094 MuViPro project of the Spanish Government, and by FI-DGR 2011 Research Fellowship Program of Generalitat de Catalunya.

References

1. Kiranyaz, S., Gabbouj, M.: Hierarchical Cellular Tree: An Efficient Indexing Scheme for Content-Based Retrieval on Multimedia Databases. IEEE Transactions on Multimedia, 102–119 (January 2007)
2. Ventura, C.: Tools for image retrieval in large multimedia databases. Master Thesis at UPC (September 2011), `http://hdl.handle.net/2099.1/13011`
3. Manjunath, B.S., Salembier, P., Sikora, T.: Introduction to MPEG-7, Multimedia Content Description Interface. John Wiley and Sons, Ltd. (June 2002)

A Key-Frame-Oriented Video Browser

Mario Taschwer

Institute of Information Technology (ITEC),
Klagenfurt University, Austria
mt@itec.aau.at

Abstract. We propose a video browser facilitating known-item search in a single video. Key frames are presented as four images at a time and can be navigated quickly in both forward and backward directions using a slider. Alternatively, key frames can be displayed automatically at different frame rates. The user may choose between three mappings of key frames to the four key frame widgets based on video time stamps and color similarity.

1 Introduction

Video browsing is the task of efficiently navigating through a video in order to quickly arrive at one or more video segments of interest. A typical use case is a person trying to find a particular scene in a movie she had watched a few months ago. This *known-item search* task is different from the equally named problem of finding a particular video within a large video collection [2].

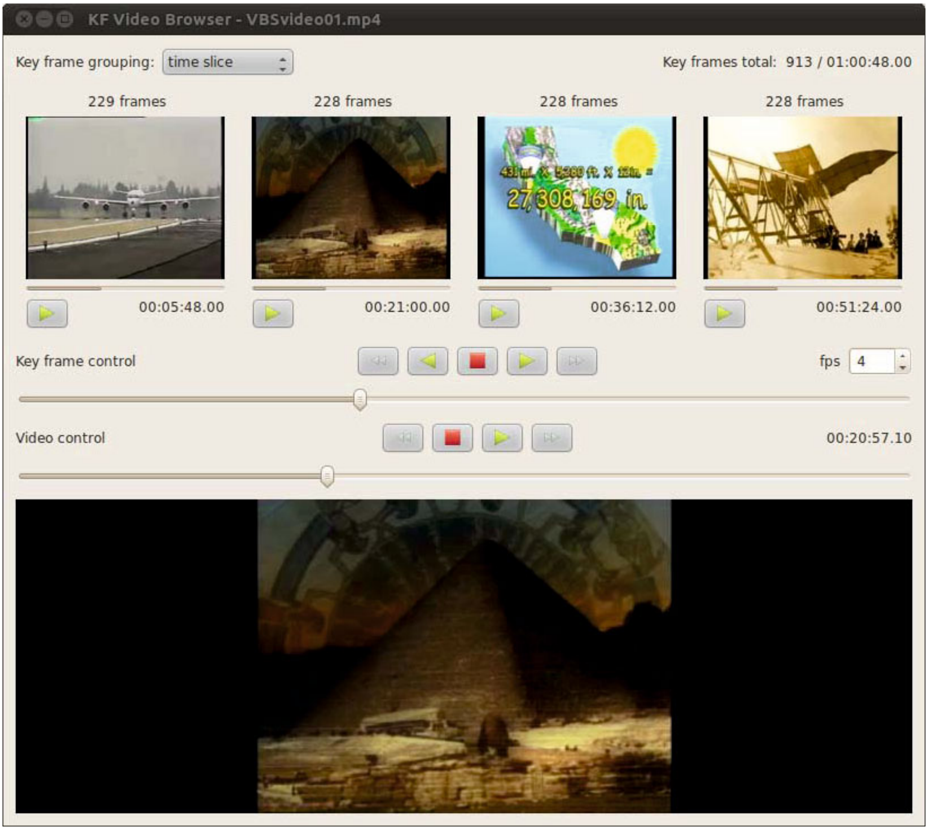
Common digital media players provide VCR-like controls only, making the known-item search task often a time-consuming and wearisome effort. Efficient navigation through a one-hour video requires alternative representation and interaction models. Meaningful reduction of presented visual data and a more flexible way of browsing them are needed.

We propose a video browser tool supporting these needs in a simple, but hopefully efficient way: visual data are reduced to key frames, which are presented to the user as four images at a time. Because all key frames are kept in memory, navigation through them happens quickly both in forward and backward order. The efficiency of this approach relies on the human ability to quickly recognize visual patterns in parallel and on the technical ability of immediate key frame display control.

2 User Interface

The user interface of the proposed video browser is shown in Figure 1. Each of the four image widgets near the top displays a subsequence of key frames ordered by time stamp. The combo box above them allows to choose between three construction methods of these subsequences: (a) *time slice:* the sequence of all key frames is partitioned into four subsequences of consecutive key frames

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 655–657, 2012.



KF Video Browser - VBSvideo01.mp4

Key frame grouping: time slice

Key frames total: 913 / 01:00:48.00

fps 4

Fig. 1. User interface of the proposed video browser. The top four image widgets display key frames only, the bottom widget represents a traditional video player.

of approximately equal length. That is, the first key frame widget shows only key frames taken from the first quarter of the video, the second widget shows only key frames taken from the second quarter, and so on. (b) *sequential:* each key frame widget displays all key frames, but with different starting offsets such that four consecutive key frames are visible at the same time. When sliding through key frames, the same key frame virtually moves over all four widgets, giving it a higher chance to be recognized by the user. (c) *color similarity:* the set of all key frames is partitioned into four k-means clusters with respect to 13-dimensional color feature vectors. Within each cluster, key frames are sorted by time stamp and the resulting subsequence is assigned to a single key frame widget. Obviously, the number of key frames per widget may vary significantly.

The upper one of the two sliders is the key frame control slider, which simultaneously affects all four key frame widgets. Dragging the slider to the right gradually displays all key frames assigned to each widget – possibly at different frame rates depending on the number of key frames per widget. In addition to manual slider dragging, the slider can be operated automatically using the VCR-like controls above. The maximal display frame rate per widget can be chosen from a range between 2 and 20 fps. The *step backwards* and *step forward* buttons (double arrows) move the slider by 5% of the slider length.

The *play* button below each key frame widget allows to start video playback at the time stamp of the key frame, using the traditional video player in the lower part of the user interface.

3 Known-Item Search in Video

Depending on the particular video segment that is to be found, different key frame grouping methods may be preferred: (a) If the video segment contains a striking color structure, the *color similarity* grouping will be most effective, because the segment will be contained in a rather small cluster. (b) If the video segment is represented by only a few key frames, the *sequential* grouping may be beneficial, because the same key frame will be displayed in all four key frame widgets at different times. (c) Otherwise, the *time slice* grouping provides a good starting point as all key frames can be displayed by scrolling through only a quarter of them.

From a user's perspective, bidirectional navigation in the key frame sequences is important, because human reaction to recognizing a key frame of interest usually involves a certain delay.

4 Key Frame Preprocessing

Key frames are extracted after shot boundary detection [1]. To facilitate searching for short video segments, the extraction rate is adapted to shot length, varying between 1 fps for shots shorter than 4 seconds, and 1/6 fps for shot lengths of at least 20 seconds.

Feature extraction for color similarity clustering is based on color histograms with 12 bins. Every pixel is assigned a probability vector of 11 well-known color names [3] and its lightness value in CIELAB color space. The normalized sum of these vectors over all pixels of an image constitutes a key frame's feature vector.

References

1. Fang, H., Jiang, J., Feng, Y.: A fuzzy logic approach for detection of video shot boundaries. Pattern Recognition 39(11), 2092–2100 (2006)
2. Smeaton, A., Over, P., Kraaij, W.: Evaluation campaigns and trecvid. In: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, pp. 321–330. ACM (2006)
3. van de Weijer, J., Schmid, C., Verbeek, J., Larlus, D.: Learning color names for Real-World applications. IEEE Transactions on Image Processing 18(7), 1512–1523 (2009)

A Video Browsing Tool for Content Management in Media Post-Production

Werner Bailer, Wolfgang Weiss, Christian Schober, and Georg Thallinger

JOANNEUM RESEARCH Forschungsgesellschaft mbH
DIGITAL – Institute for Information and Communication Technologies
Steyrergasse 17, 8010 Graz, Austria
firstname.lastname@joanneum.at

Abstract. We propose an interactive video browsing tool for supporting content management and selection in post-production. The tool enables users to iteratively cluster the content set by different features, and restrict the content set by selecting a subset of clusters. In addition, similarity search is supported. A desktop and Web-based user interface, including temporal preview is available.

1 Architecture and Workflow

The application scenario of the proposed tool is content management in the post-production phase of film and TV production. Users typically deal with large amounts of audiovisual material with a high degree of redundancy and need to select a small subset for use in a production. Newly shot material is typically sparsely annotated, thus the browsing tool has to rely on automatically extracted features.

When content is ingested into the tool automatic content analysis is performed. Currently, camera motion estimation, visual activity estimation, extraction of global color features and estimation of object trajectories are implemented. The extracted features are represented in MPEG-7 and indexed in an SQLite database.

The tool implements an iterative content selection process, where users cluster content by one of the automatically extracted features and can then select relevant clusters to reduce the content set. Further clustering by the same or other features can then be applied to the reduced set. A detailed description of the tool can be found in [1].

2 User Interface

We have implemented a desktop and web-based user interfaces, using the same backend. The central component of the browsing tool's user interface is a light table (cf. Figure 1). The light table shows the current content set and cluster structure using a number of representative key frames for each of the clusters. The clusters are visualized by colored areas around the images. By clicking on a key frame in the light table view, a video player is opened and plays the segment

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 658–659, 2012.

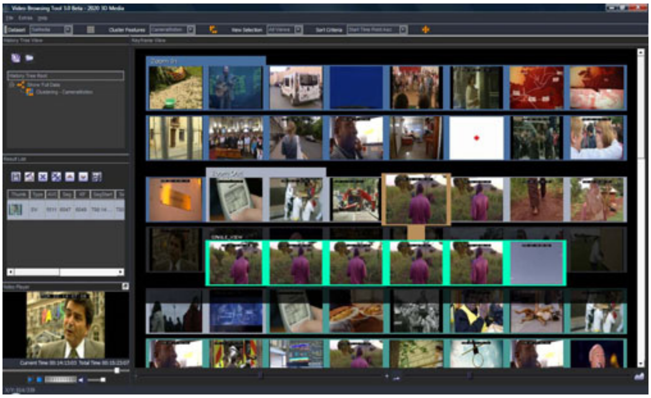


Fig. 1. Screenshot of the video browsing tool desktop application

of the video that is represented by that image. The temporal context of a key frame is shown by a time line of temporally adjacent key frames that appears when the user moves the mouse over a frame.

On the left side of the application window the history and the result list are displayed. The history window automatically records all clustering and selection actions done by the user. By clicking on one of the entries in the history, the user can go back to a previous point. Then users can choose to discard the subsequent steps and use other cluster/selection operations, or to branch the browsing history and explore the content using alternative cluster features. The result list can be used to memorize video segments and to extract segments of videos for further video editing, e.g. as edit decision list (EDL). Users can drag relevant key frames into the result list at any time, thus adding the corresponding segment of the content to it. The size of the images in the light table view can be changed dynamically so that the user can choose between the level of detail and the number of visible images without scrolling. Furthermore, the application allows to execute a similarity search based on following features: camera motion, motion activity, color layout, multi-view media item and multi-view camera.

Acknowledgements. The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 215475, "2020 3D Media – Spatial Sound and Vision".

Reference

1. Bailer, W., Weiss, W., Kienast, G., Thallinger, G., Haas, W.: A video browsing tool for content management in post-production. International Journal of Digital Multimedia Broadcasting (March 2010)

Video Browsing with a 3D Thumbnail Ring Arranged by Color Similarity

Klaus Schoeffmann, David Ahlström, and Laszlo Böszörmenyi

Klagenfurt University, Universitaetsstr, 65-67, 9020 Klagenfurt, Austria
{Klaus.Schoeffmann,David.Ahlstroem,Laszlo.Boeszoermenyi}@aau.at

Abstract. We propose a 3D arrangement of thumbnail images for the purpose of browsing a single video file. The thumbnail images are linearly extracted from the video and used as textures for bended screens in a 3D-ring arrangement, which act as links for the playback of the corresponding video segments. Furthermore, the thumbnail images in this 3D-ring are intuitively organized by their dominant colors according to the HSV color space. This color-based organization should help users to estimate the position of a known item in the 3D-ring.

1 Introduction

A 3D arrangement enables to visualize a large number of images with a relatively small screen estate. Such a visualization has the characteristic that images in the front are shown at high detail while images in the back are kept in view, although at low detail and with some distortion due to 3D projection. However, through interaction/navigation the user can bring images from the back to the front in order to inspect them in more detail. The low detail images are supposed to be informative enough for deciding whether a given image could be of interest at all.

Based on our previous research results [1] we propose using a horizontal *3D Ring* arrangement for the purpose of browsing a moderately large set of images, as it provides an intuitive and convenient way of navigation and produces minor distortion of images in comparison to other 3D arrangements. For the purpose of the Video Browser Showdown we linearly sample images from the video (e.g., with five seconds distance) and arrange them on the 3D Ring according to their dominant color. These thumbnails act as links for the playback of the corresponding segment, shown at larger size above the 3D Ring (see Figure 1).

2 User Interface

Images in the 3D arrangement are sorted by color according to an efficient and intuitive sorting algorithm, that allows for real-time (i.e., on-the-fly) processing. The basic idea is to sort images based on their dominant hue color in the HSV color space. Therefore, we classify pixels of images into a 16-bin hue histogram (each bin represents pixels belonging to a hue range of 22.5 degrees) and use the

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 660–661, 2012.

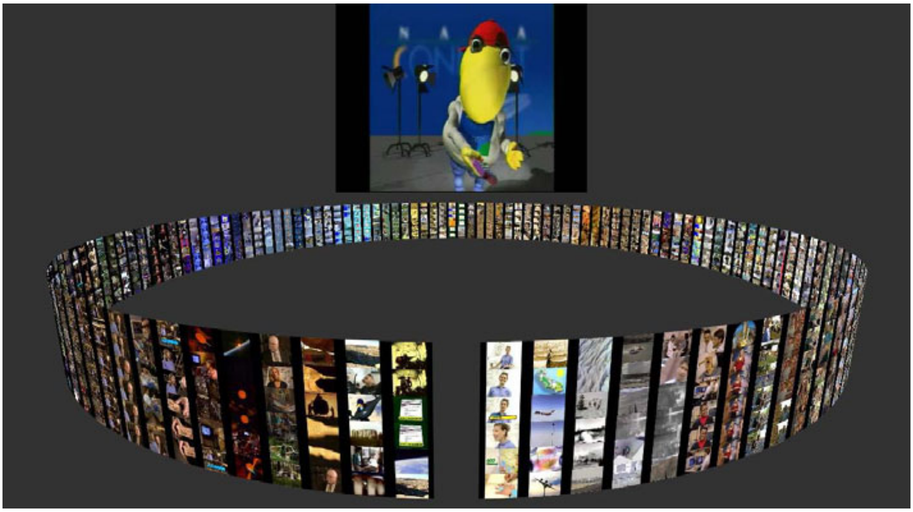


Fig. 1. Video Browsing with a 3D Thumbnail Ring Arranged by Color Similarity. The missing column indicates the begin/end of the list.

index of the dominant bin as a basic sorting criterion. To give a more consistent view, images belonging to the same dominant bin are sorted such that the Euclidian distance of an HSV histogram between adjacent images is minimal. Moreover, we perform a special treatment for bright and dark images; these are arranged at the beginning and the end of the list, respectively. The resulting sorted list is directly used for the cylindrical arrangement with a column-major order. Therefore, the images on the 3D Ring arrangement are always presented in the same 'color-sequence'. It starts with bright images, followed by gray, brown, orange, yellow, green, turquois, blue, pink, red, and finally dark images (if available, respectively). Please note that based on the available colors in the chosen image set, specific colors may be not contained while other colors may utilize a rather large area in the 3D Ring. The color-based arrangement should enable a trained user to roughly estimate the position of a known image in the 3D Ring and help him/her to more quickly find desired images.

The user can smoothly rotate the 3D Ring arrangement by using the mousewheel or quickly bring a particular area of the 3D arrangement into view (i.e., to the front) by using the right mouse button. A left click on a thumbnail image immediately starts playback for the corresponding segment in a playback window above the ring.

Reference

1. Schoeffmann, K., Ahlström, D., Böszörmenyi, L.: A user study of visual search performance of interactive 2d and 3d storyboards. In: Proceedings of the 9th International Workshop on Adaptive Multimedia Retrieval, AMR 2011 (2011)

Multimodal Cue Detection Engine for Orchestrated Entertainment

Danil Korchagin, Stefan Duffner, Petr Motlicek, and Carl Scheffler

Idiap Research Institute, Martigny, Switzerland
{danil.korchagin,stefan.duffner,petr.motlicek,
carl.scheffler}@idiap.ch

Abstract. In this paper, we describe a low delay real-time multimodal cue detection engine for a living room environment. The system is designed to be used in open, unconstrained environments to allow multiple people to enter, interact and leave the observable world with no constraints. It comprises detection and tracking of up to 4 faces, estimation of head poses and visual focus of attention, detection and localisation of verbal and paralinguistic events, their association and fusion. The system is designed as a flexible component to be used in conjunction with an orchestrated video conferencing system to improve the overall experience of interaction between spatially separated families and friends. Reduced latency levels achieved to date have shown improved responsiveness of the system.

Keywords: Multimodal signal processing, data analysis, sensor fusion.

1 Introduction

The TA2 (Together Anywhere, Together Anytime) project [1] tries to understand how technology can help to nurture family-to-family relationships to overcome distance and time barriers. This is something that current technology does not address well: modern media and communications are designed for individuals, as phones, computers and electronic devices tend to be user centric and provide individual experiences. Existing multiparty conferencing solutions available on the market, such as Microsoft RoundTable conferencing table [2], are not designed to be used in open, unconstrained environments.

In our previous work [3], we have developed a framework for just-in-time multimodal association and fusion for open, unconstrained environments with spatially separated multimodal sensors. It relies on score-level information fusion derived from spatially separated sensors. By placing the sensors at their individually optimal locations, we clearly obtain a better performance of low-level semantic information. Performance levels achieved on hand-labelled, echo-cancelled dataset have shown sufficient reliability at the same time as fulfilling real-time processing requirements with latency within 200-300 ms. In current work we evolve the previous system towards better responsiveness of the system and integration of additional components, which have been identified as important for the extraction of additional

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 662–670, 2012.

semantic cues to be used by an orchestration engine [4]. The orchestration engine produces then an orchestrated video chat by choosing at each point in time the perspective that best represents the social interaction based on decision-level rule-based fusion.

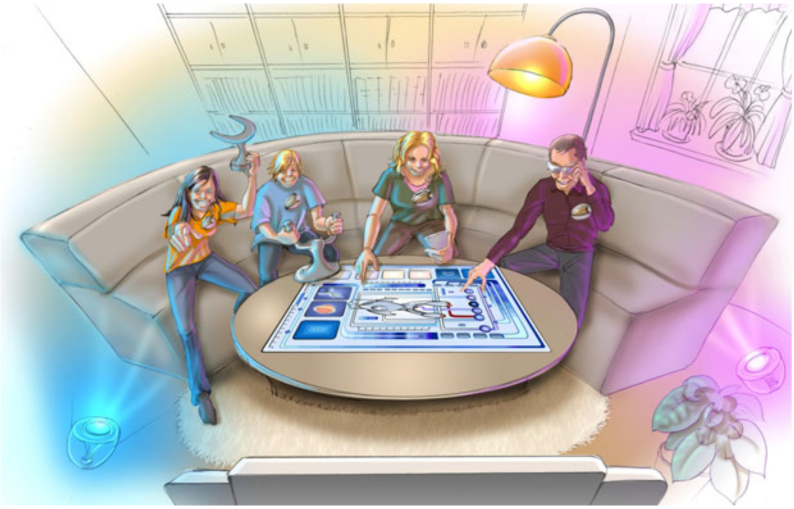


Fig. 1. Illustration of a family environment setup

In this context, TA2 presents several challenges: the results need to be computed in real-time with low affordable delay from spatially separated sensors (as opposed to other systems, such as [5, 6, 7], relying on collocated sensors) in open, unconstrained environment. Furthermore, the results are supposed to be localised in the image space to allow for a dynamic and seamless orchestrated video chat.

2 A Real-Time Architecture

The presented multimodal cue detection engine includes a face detector, a multiple face tracker, multiple person identification, head pose and visual focus of attention estimation, an audio real-time framework with spatial localisation, a large vocabulary continuous speech recognizer and keyword spotter, multimodal association and fusion (see Fig. 2). A face tracking algorithm has been developed to track a variable number of faces even when there is no face detection for a long period of time. Although the accuracy of far-field Automatic Speech Recognition (ASR) is not yet good enough to be exploited for obtaining an accurate real-time transcription, it is sufficient to augment the behaviour of an orchestration module. Words in the transcript are used to search for participants' proper names relevant to the group of people or keywords relevant to a given scenario. Furthermore, the orchestration (which is not part of the multimodal cue detection engine) will be able to reason and act upon these events together with other cues that could potentially come from a game engine, aesthetic or cinematic rules, making orchestrated video chat dynamic and seamless.

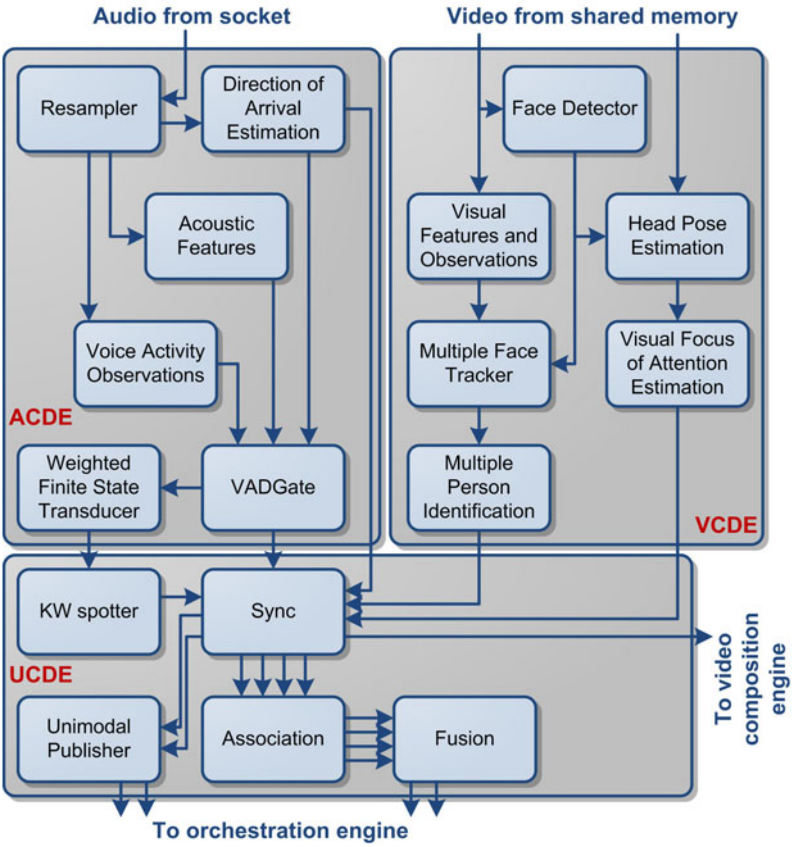

Audio from socket
Video from shared memory
Resampler
Direction of Arrival Estimation
Acoustic Features
Voice Activity Observations
ACDE
Weighted Finite State Transducer
VADGate
Face Detector
Visual Features and Observations
Head Pose Estimation
Multiple Face Tracker
Visual Focus of Attention Estimation
Multiple Person Identification
VCDE
KW spotter
Sync
UCDE
Unimodal Publisher
Association
Fusion
To video composition engine
To orchestration engine

Fig. 2. The system architecture is built around several modules comprising a so-called Video Cue Detection Engine (VCDE) with a face detector, a multiple face tracker, multiple person identification, head pose and visual focus of attention estimation; an Audio Cue Detection Engine (ACDE) with a direction of arrival estimator, a voice activity detector and a large vocabulary continuous speech recogniser; a Unified Cue Detection Engine (UCDE) with association, fusion and transmission of the results to external components (orchestration engine, video composition engine).

The audio input to the multimodal cue detection engine and the semantic output from it are implemented via sockets, while the video stream is transferred via shared memory. The core capture devices for the system are a Full HD video camera and an audio diamond array with four omnidirectional microphones [8]. Video frames from the shared memory of the video grabber server are retrieved every 40 ms at a resolution of 640x360 pixels, while audio packets are retrieved every 10 ms and contain interleaved 4 channel PCM audio in 16-bit at 48 kHz.

The multimodal processing operates in multi-framing mode with non-overlapping video frames, overlapping audio frames of 16 ms in step of 10 ms for voice activity detection and ASR, and overlapping audio frames of 32 ms in step of 16 ms for direction of arrival estimation.

2.1 Multiple Face Tracking

A multiple face tracking algorithm is automatically initialised and updated using outputs from a standard face detector [9]. The challenge for face tracking in this scenario is that face detections are not continuous and that the time between two successive detections can be very long (up to 30 s in our experiments). This is due to head poses that are difficult to detect by state-of-the-art algorithms, or partial occlusions caused by hands in front of the face (see Fig. 3). However, in the TA2 scenario it is necessary to know at each time instant where the people are in the video scene.

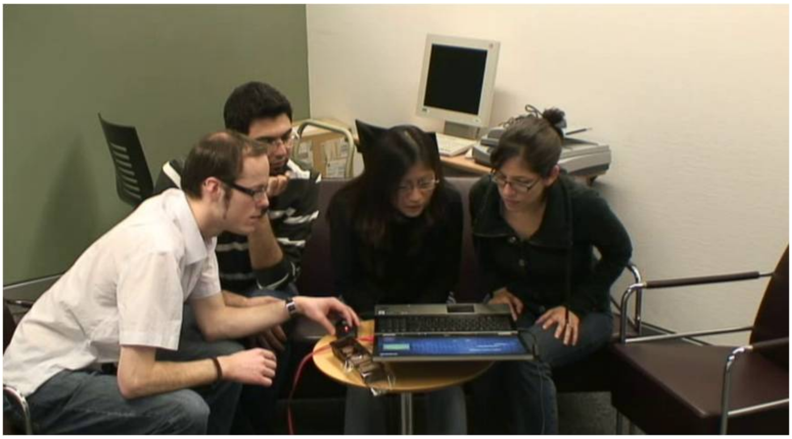


Fig. 3. An example of difficult to detect head poses and partial occlusions [10]

The solution employed in this work is based on a multi-target tracking algorithm using Markov Chain Monte Carlo (MCMC) sampling, similar to [11]. This is a Bayesian tracking framework using particles to approximate the current state distribution of all visible targets. At each time step, targets are added and removed using the output of an additional probabilistic framework that takes into account the output of the face detector as well as long-term observations from the tracker and image [12].

The state space is the concatenation of the states of all visible faces, where the state of each single face is a rectangle described by the 2D position in the image plane, a scale factor and the eccentricity (height/width ratio).

The dynamic model is the product of the models of each visible face and a Markov Random Field that prevents targets becoming too close to each other. The state dynamics of each single face are described by a first-order autoregressive model for the position and a zeroth-order model for scale and eccentricity.

Finally, the observation likelihood is the product of the observation likelihoods of each visible face, which in turn is calculated using the Bhattacharyya distance between the HSV (Hue-Saturation-Value) colour histograms over three horizontal bands on the face region and the respective reference colour histograms which are initialised when the face is detected.

2.2 Multiple Person Identification

Whenever a tracker loses a target and reinitialises it later on, or a person leaves the visual scene and comes back later, the tracking algorithm tries to recognise the respective person in order to associate it to a previously tracked target. This is not done inside the tracking algorithm but on a higher level taking into account longer-term visual appearance observations. Each person's appearance is modelled by three sets of HSV colour histograms calculated on face and shirt regions. Using multiple histograms per person copes for different appearances due to changes in body pose. However, only the most similar histogram of a person is used and updated at each time.

When identifying a "new" face, the current colour histograms are compared to the stored models of all previously seen people and if the similarity is above a certain threshold the corresponding ID is assigned, otherwise a new person model is created.

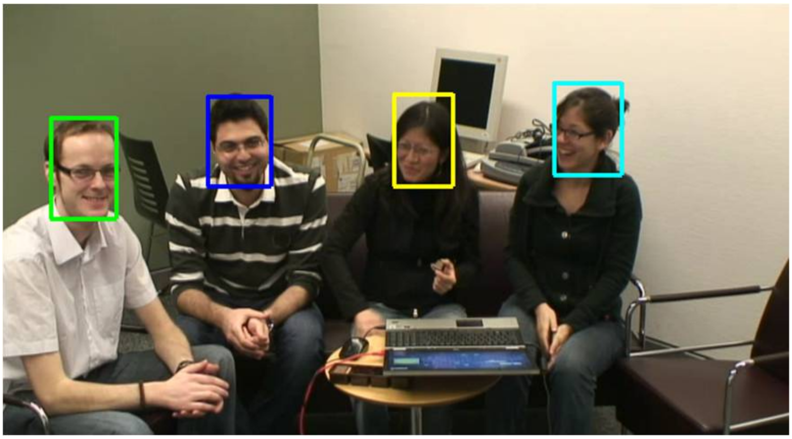


Fig. 4. Consistent person identification within the session (here indicated by different colours) is an important requirement to the multimodal cue detection engine

2.3 Head Pose Estimation

Based on the output of the face tracker, the head pose (i.e. rotation in 3 dimensions) of an individual is estimated. The purpose of computing head pose is the estimation of a person's visual focus of attention (see section 2.4).

Head pose is computed using visual features derived from the 2-dimensional image of a tracked person's head. The features used here are gradient histograms [13] and colour segmentation histograms. Colour segmentation is done by classifying each pixel around the head as either skin, hair, clothing or background based on colour models that are adapted to each individual being tracked [14].

To compensate for the variability in the output of the face tracker, the 2-dimensional face location is re-estimated by the head pose tracker. This serves to normalise the bounding box around the face as well as possible, while simultaneously using the visual features mentioned above to estimate pose. This joint estimation of head location and pose improves the overall pose accuracy [15].

2.4 Visual Focus of Attention

Given the estimated belief (probability distribution) over head pose, the visual focus of attention target is estimated. In the context of this work, the following targets are of interest: the video conferencing screen, the touch sensitive table, and any other person in the room.

The range of angles that correspond to each target is modelled using a Gaussian likelihood. This likelihood is derived from the known spatial locations of the targets within the conference room. The posterior belief over each target is computed with Bayes' rule using the method given in [16].

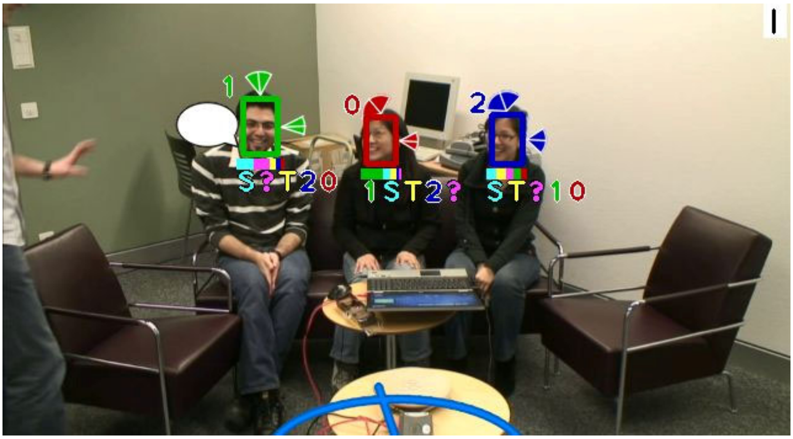


Fig. 5. Multimodal cue visualisation. For each person, it shows its ID (at the top-left of the face bounding box), its head orientation estimation, i.e. pan and tilt, with a variance indication (on the top and right side of the box), and the estimated distribution over targets where the person is looking at (at the bottom of the box), where the left-most target is the most likely one. The letter “S” means “screen”, “T” means “table”, “?” means “unknown”, and the numbers correspond to the IDs of the other persons. The blue line in the bottom of the image indicates the estimated direction of arrival of sound. The speech bubble indicates that a person is speaking, and the output of the keyword spotting is shown in the top-right of the image, here the word “I”.

2.5 Direction of Arrival Estimation

Speaker localisation is performed by the direction of arrival module (Fig. 2). The algorithm is based on spatio-temporal fingerprint processing [17] in steps of 6°, which represents a computationally efficient solution with low algorithmic delay compared to short-term clustering of generic sector-based activity measures [8, 18] used in our previous study [3]. It relies only on the geometry of the microphone array and does not depend on prior knowledge of the room dimensions. It can be effectively used to both detect and localise multiple sources in open, unconstrained environments.

2.6 Voice Activity Detection

Voice activity detection (VAD) covers both verbal and paralinguistic activities and is implemented as a gate. The gate segments the input stream in accordance to directional and voice activity / silence information from an algorithm based on silence models or trained multi-layer perceptrons (MLP) using traditional ASR features [19]. The association and fusion [3] of the detected voice activity events with person IDs from the video-based identification are performed by the time voice activity is confirmed and the corresponding audio-based directional cluster is estimated.

2.7 Keyword Spotting

The ASR component is represented by the Weighed Finite State Transducer (WFST) based token passing decoder known as Juicer [19]. The output from the decoder is used to perform the spotting, association and fusion [3] of proper names and keywords with person IDs from the video identification taking into account the estimated audio-based directional cluster for the corresponding time interval. More specifically, the spotting is performed based on the predefined list of participants and keywords relevant to the given scenario (e.g., orchestrated video chat).

3 Improvements and Results

During subjective evaluations of our previous version of multimodal cue detection engine, several bottlenecks have been experienced. To overcome these bottlenecks, several architectural and algorithmic changes have been applied and presented in this paper.

First of all, while the socket interface was allowing for a flexible software solution, the experienced latency for uncompressed video signal transmission from remote video grabber was resulting in additional latency of 30-300 ms. This clearly noticeable lag was successfully removed by switching to a shared memory interface for video input stream. While a shared memory interface could be potentially used for audio input stream as well, experienced latency of 12-20 ms for the audio transmission is on an acceptable level.

To reduce the latency of audio processing we have decided to reduce the algorithmic delays of both direction of arrival estimation and voice activity detection.

The algorithmic latency of both components has been reduced from 200 ms down to 128 ms. This is due to the replacement of the previous implementation based on a short-term clustering approach by the computationally more efficient spatio-temporal fingerprints processing and the reduction of corresponding temporal filters.

Exact clock synchronisation between separated audio and video grabbers was seen as another source of potential problems and during subjective evaluations we have found that the use of local timestamps results in more consistent multimodal association and fusion. Moreover, since the position of people does not significantly change within a few hundred milliseconds, predictive temporal association was finally employed within the system to further remove possible lags during the capturing of the video stream by hardware and video grabber.

We have found that it is beneficial to have acoustic tracking of the active acoustic sources as an additional input to the voice activity detection gate to properly treat barge-in events, which were not always detected in a former system.

Since the participants do not sit at predefined positions in the room, theoretically it can cause ambiguities in the association and fusion. Clearly, the same acoustic directional cluster can correspond to different positions in the image and vice-versa. However, since the participants are mainly located around a coffee table, such ambiguities occur rarely during evaluations.

Finally, head pose and visual focus of attention estimation have been identified as important semantic cues for the orchestration engine and have been successfully integrated into the multimodal cue detection engine. Head pose estimation is to be used for better selection of frontal/side views with respect to aesthetic and cinematic rules, while visual focus of attention can be beneficial for better modelling of social interactions (e.g. predictive turn estimation during grant-floor moments) and can have a direct impact on temporal filters within the aesthetic and cinematic rules.

Objective evaluations of involved components were performed, and their results can be found in [3, 12, 14, 17]. The corresponding annotated dataset has been made publically available [10]. The algorithmic latency within the multimodal cue detection engine stays within 130 ms, except for proper name and keywords spotting, which are transmitted by the end of acoustically separated utterances.

4 Conclusion

We have developed a low delay real-time multimodal cue detection engine for open, unconstrained environments with spatially separated multimodal sensors. We have described applied architectural and algorithmic changes to reduce an overall latency down to 130 ms and fulfil real-time processing requirements. The achieved results are promising for future wider evaluations and further development of the platform in several directions such as improvement of performance, reduction of the latency, and integration of additional components allowing richer multimodal cues.

Acknowledgments. The research leading to these results has received funding from the European Community's Seventh Framework Programme ICT Integrating Project “Together Anywhere, Together Anytime” (TA2, FP7/2007-2013) under grant agreement no. ICT-2007-214793. We are grateful to Philip N. Garner and Jean-Marc Odobez for their valuable help at various stages of this work.

References

1. Integrating project within the European research programme 7: Together anywhere, together anytime (2008), http://www.ta2-project.eu
2. Microsoft: Microsoft RoundTable conferencing table (2007), http://www.microsoft.com/uc/products/roundtable.mspx
3. Korchagin, D., Motlicek, P., Duffner, S., Bourlard, H.: Just-in-time multimodal association and fusion from home entertainment. In: Proc. IEEE International Conference on Multimedia & Expo (ICME), Barcelona, Spain (2011)
4. Falelakis, M., et al.: Reasoning for video-mediated group communication. In: Proc. IEEE International Conference on Multimedia & Expo (ICME), Barcelona, Spain (2011)
5. Bohus, D., Horvitz, E.: Dialog in the open world: platform and applications. In: Proc. of ICMI, Cambridge, USA (2009)
6. Otsuka, K., et al.: A realtime multimodal system for analyzing group meetings by combining face pose tracking and speaker diarization. In: Proc. of ICMI, Chania, Greece (2008)
7. Bernardin, K., Stiefelhagen, R.: Audio-visual multi-person tracking and identification for smart environments. In: Proc. of ACM Multimedia (2007)
8. Korchagin, D., Garner, P.N., Motlicek, P.: Hands free audio analysis from home entertainment. In: Proc. of Interspeech, Makuhari, Japan (2010)
9. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proc. of CVPR, Hawaii, USA (2001)
10. Duffner, S., Motlicek, P., Korchagin, D.: The TA2 database: a multi-modal database from home entertainment. In: Proc. of Signal Acquisition and Processing, Singapore (2011)
11. Khan, Z.: MCMC-based particle filtering for tracking a variable number of interacting targets. IEEE Trans. on Pattern Analysis and Machine Intelligence 27, 1805–1918 (2005)
12. Duffner, S., Odobez, J.-M.: Exploiting long-term observations for track creation and deletion in online multi-face tracking. In: Proc. IEEE Conference on Automatic Face & Gesture Recognition (2011)
13. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (2005)
14. Scheffler, C., Odobez, J.-M.: Joint adaptive colour modelling and skin, hair and clothing segmentation using coherent probabilistic index maps. In: Proc. of BMVC (2011)
15. Ba, S.O., Odobez, J.-M.: A probabilistic framework for joint head tracking and pose estimation. In: Proc. of the International Conference on Pattern Recognition (2004)
16. Ba, S.O., Odobez, J.-M.: Recognizing visual focus of attention from head pose in natural meetings. IEEE Transactions on System, Man and Cybernetics 39(1), 16–33 (2009)
17. Korchagin, D.: Audio spatio-temporal fingerprints for cloudless real-time hands-free diarization on mobile devices. In: Proc. of the 3rd Joint Workshop on Hands-Free Speech Communication and Microphone Arrays (HSCMA), Edinburgh, UK, pp. 25–30 (2011)
18. Lathoud, G., McCowan, I.A.: A sector-based approach for localization of multiple speakers with microphone arrays. In: Proc. of SAPA, Jeju, Korea (2004)
19. Garner, P.N., et al.: Real-time ASR from meetings. In: Proc. of Interspeech, Brighton, UK, pp. 2119–2122 (2009)

The Ultimate Immersive Experience: Panoramic 3D Video Acquisition

Christian Weissig, Oliver Schreer, Peter Eisert, and Peter Kauff

Fraunhofer Heinrich-Hertz Institute, Einsteinufer 37, 10587 Berlin, Germany
{Christian.Weissig,Oliver.Schreer,
Peter.Eisert,Peter.Kauff}@hhi.fraunhofer.de

Abstract. The paper presents a new approach on an omni-directional omni-stereo multi-camera system that allows the recording of panoramic 3D video with high resolution and quality and display in stereo 3D on a cylindrical screen. It has been developed in the framework of the TiME Lab at Fraunhofer HHI, an experimental platform for immersive media and related content creation. The new system uses a mirror rig to enable a multi-camera constellation that is close to the concept of concentric mosaics. A proof of concept has shown that the systematical approximation error related to concentric mosaics is negligible in practice and parallax-free stitching of stereoscopic video panoramas can be achieved with high 3D quality and for arbitrary scenes with depth ranges from 2 meters to infinity.

Keywords: Panoramic Imaging, Ultra-High Definition, 3D Video Panorama, Omni-Directional Cameras, Omni-Stereo Panorama, Concentric Mosaics.

1 Introduction

It is widely accepted that concept of immersive media is one of the most promising market segments of future technology. One feature of immersive media is panoramic imaging using large cylindrically or spherically curved screens, often in combination with multi-projection systems providing ultra-high resolution by exact juxtaposition and blending of multiple projector images [1][2][3][4][5]. Being lost in niche markets like theme parks for a long time, these applications are now migrating into new market segments like event and exhibition technology, training centers or even entertainment. Typical applications are dome projections (e.g. in planetariums), giant screen cinemas (e.g. re-opening of digital Cinerama theatres) or immersive 180° or 360° surround video (e.g. simulation and training centers) [6][7][8]. In future, they may even address immersive viewing in new types of cinema theatres or other public venues, or, to its end, in immersive home entertainment.

In February 2010, the Fraunhofer Heinrich-Hertz-Institute (HHI) in Berlin, Germany, has opened its 'Tomorrow's Immersive Media Experience Laboratory (TiME Lab)', an experimental platform for immersive media and related content creation. The TiME Lab uses up to 14 HD projectors for panoramic 2D and 3D projection at a cylindrical 180° screen with a resolution of 7k x 2k as well as a 'Wave Field Synthesis (WFS)' sound system with 128 loudspeakers [9].

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 671–681, 2012.

Apart from multi-projection, a further main challenge of panoramic imaging is to create live footage supporting these special video formats in combination with ultra-high resolution. One solution is, in analogy to multi-projection, to use multiple cameras where the single cameras look into different directions such the resulting images can be stitched seamlessly to large panoramic views. The technology of such omni-directional camera systems has a long tradition. First systems that use multiple cameras and mirrors to achieve full surround capture with high image resolution have already been used in the 60s by Ub Iwerks for Disney theme park productions [10]. Since then many further mirror-based system approaches have been proposed (e.g. [11]). Other approaches place a hyperboloid mirror in front of a single camera to capture panoramic views [12][13]. Today, the advances and ongoing miniaturization of digital video cameras enables more compact systems and several commercial companies offer omni-directional cameras for a wide range of applications [14][15][16][17][18][19][20][21]. Good overviews about different approaches on panoramic imaging are given in [22][23].

The term "3D panoramic video" is often used for 360° viewing capability in a 2D panorama. However, in this context, two video panoramas of the same scene but with different perspective are considered, one for the left and one for the right eye in order to allow stereoscopic 3D. Although the concept of omni-directional cameras for capturing 2D video panoramas is well understood and a lot of efficient systems are already available, capturing of 3D video panoramas is still a challenge and a partly unsolved problem. Against this background, this paper mainly presents a prototype system of a new 3D omni-directional camera, which allows an almost error-free panoramic 3D video acquisition by using a special mirror rig. After a short review of capturing systems for panoramic 2D video, the 3D approach is explained in more detail. Meanwhile, a proof of concept has been achieved by first test shootings and the content has been presented in HHI's TiME Lab during the "Berlinale" film festival in Berlin on February 2011.

2 Review of Omni-Directional Imaging and Panoramic 2D Video

As known from projective geometry, the optimal multi-camera arrangement for capturing panoramic videos requires that the focal points of all camera views coincide in a common point (see left drawing in Figure 1) [22][23]. In case of capturing static 2D panoramas, this condition is usually achieved by rotating a single camera at a tripod with a revolving camera head. For video, however, this approach is impractical due to the need of using multiple cameras on one hand and the physical dimensions of each camera on other hand. Hence, many commercial solutions capture video panoramas with the star-like approach from Figure 1 (right) [15][18][21]. In this case the focal points of all cameras are located on a common circle, while the optical axes are perpendicular to the arc. This approach works reasonably well as long as only far distant objects appear in the scene. However, the existence of a non-zero parallax angle does not allow seamless stitching in case of close objects in the overlap area.

A suitable approximation of the optimal solution from Figure 1 (left) can be achieved by using special mirror-rigs. If all cameras and mirrors are arranged correctly, it is possible to superimpose the virtual images of all focal points in one

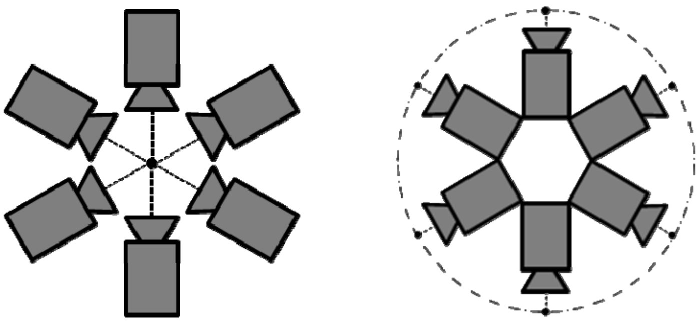


Fig. 1. Optimal camera arrangement (left) and star-like approach (right)

common central point behind the mirrors. Since the first applications in the 60s, many further system approaches have been proposed and have made a lot of progress, last but not least, due to the advent of digital TV and cinema cameras [10][22][23].

An interesting new approach has recently been presented by Fraunhofer HHI. The so-called OMNICAM is a scalable system, which can be equipped with up to 12 HD cameras for 360° shooting. In its current implementation (see Figure 2), it uses 6 HD cameras suitable to shoot 180° panoramas. The six cameras generate tiles of 1080x1920 pixels each, which can subsequently be stitched to one large panorama with a final resolution of 6984 x 1920 for 180°. As the cameras are used in portrait format, the vertical field-of-view is about 60°, a feature that is extremely useful for immersive media.

A special property of the OMNICAM is its very accurate calibration capabilities shown in Figure 3. This illustration depicts a horizontal section through the mirror pyramid at the plane where the optical axes intersect the mirror surfaces and, with it,

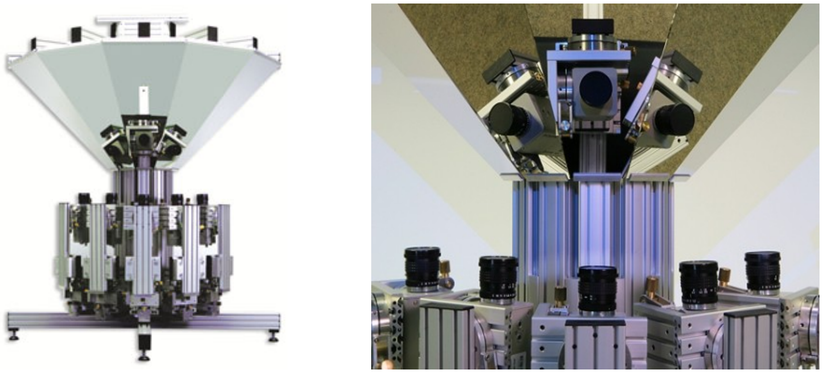


Fig. 2. OMNICAM with 6 HD cameras: long-shot (left), close-up (right)

how the virtual images of the focal points are located behind the mirrors. Note that the cameras look from bottom upwards and that the mirrors deflect the optical axes horizontally in radial direction (see also Figure 2).

In a first step the rig is calibrated such that all virtual images of the focal points coincide in the centre **C** of the mirror pyramid (see Figure 3, left). This initial state refers to the optimal camera arrangement from Figure 1 (left). It is obtained by very precise optical measurements in the laboratory.

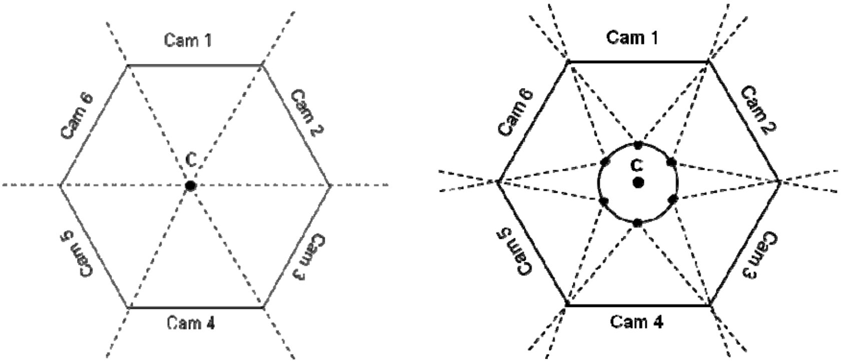


Fig. 3. Optimal mirror-based arrangement (left), radial off-centered arrangement (right)

Although this initial and optimal state allows a parallax-free stitching for scenes with a depth range from zero to infinity, it is not really suitable under real working conditions. If all cameras have a common focal point in the center of the mirror pyramid, there would be no overlap between the different tiles due to a hard cut at the mirror edges. Hence, there is no possibility to blend pixels between adjacent image tiles. In former applications like theme park productions this drawback has been concealed by segmented projection screens.

However, this is not acceptable any longer for seamless projection of video panoramas in future immersive media applications. Hence, at least some slight overlap between adjacent image tiles is needed. In order to obtain overlaps, the focal points of the cameras have to be moved symmetrically by precise actuators out of the center in radial direction (Figure 3, right). By this off-center adjustment of the focal points, it becomes possible to regulate a scene-adaptive trade-off between sufficient overlaps for blending and parallax-free stitching. In practice, the OMNICAM is usually operated with a radial-shift of about 5 mm, resulting in a blending area of about 10 pixels and a parallax-free stitching of scenes with a depth range from 2m to infinity. Figure 4 shows an example of the whole OMNICAM processing for a sports production with a particularly high depth range of the captured scene at the outer left and right blending areas. Meanwhile, the OMNICAM has been used under real working conditions for a couple of 2D panorama productions shown in the TiME Lab.

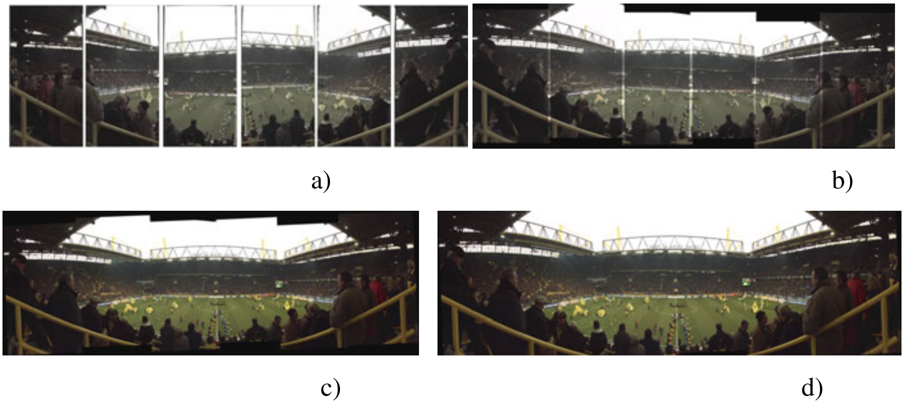


Fig. 4. Subsequent processing steps of panorama generation with OMNICAM: a) original camera views; b) geometrical correction and warping; c) photometrical correction, color matching and blending; d) final cut-out of panoramic view by cropping.

3 Extension to Omni-Stereo Imaging and Panoramic 3D Video

In principle, the above considerations can also be extended towards omni-directional recording of 3D panoramas. However, in the 3D case the situation is much more sophisticated. The main challenge is to solve a fundamental conflict between two competing requirements. On one hand, as in 2D, panoramic 3D imaging also requires a parallax-free stitching of the left- and right-eye panoramas. On the other hand, significant parallaxes are needed between the two stereo panoramas to obtain an adequate stereo impression.

Known solutions from literature that solve this problem are mainly suited for static scenes. The capture of static omni-stereo panoramas has already been investigated since more than 15 years. A nice overview on the major principles can be found in [24]. As already mentioned in the previous section, the optimal solution for static 2D panoramas is to rotate a single camera around its focal point (see also left drawing of Figure 5). As shown in Figure 5 (right), the straight forward extension to 3D is a rotation of a stereo camera around the center of its baseline.

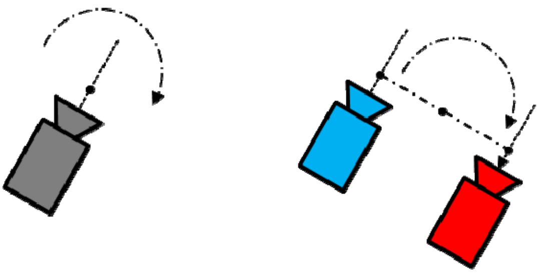


Fig. 5. Optimal solutions: omni-directional 2D (left) and 3D capture (right)

From literature, this concept is also known as concentric mosaics, a special version of the plenoptic function [25]. Unfortunately, it is not that easy to apply this solution to the acquisition of 3D video panoramas, especially not for the star-like approach from Figure 1 (right). Figure 6 shows a corresponding star-like arrangement for stereo cameras. As it can be seen from this drawing, the inter-axial distance **B** of one stereo sub-system is much smaller than the inter-axial distance **S** between adjacent panorama cameras. This situation perfectly explains the above mentioned conflict between unwanted parallax errors for the stitching process and the required parallax for stereo reproduction. Imagine that the stereo baseline **B** is well adapted to the near and far objects in the scene such that the resulting parallaxes between the left and right views produce a clearly visible and comfortable stereo effect. Hence, as the inter-axial distance **S** is in any case larger than **B**, the parallax error that appears while stitching the left- and right-eye panoramas is larger than the stereo parallax, if the same near and far objects are also present in the overlap area. Or in other words, if one wants to avoid visible parallax errors while stitching, the stereo effect is lost. The only situation where it works is given by a scene that has enough depth in the single stereo segments but remains flat in the stitching areas. Obviously, such a situation is difficult to control under real shooting conditions.

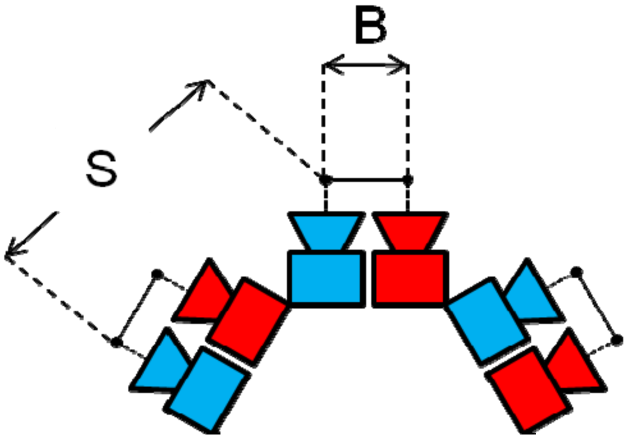


Fig. 6. Star-like arrangement for a panoramic 3D camera setup

But even the optimal solution from Figure 5 (right) is difficult to achieve with real video cameras. The concept of concentric mosaics requires that the stereo camera is rotated in very small angular increments respecting the plenoptic sampling theorem [26]. In the extreme case, the stereo rig even consists of vertical line cameras only and the rotation scans the stereo panorama column by column. Hence, in case of stereo panoramas with ultra-high resolution of several thousand pixels, the angular increment is significantly lower than one degree. It is self-evident that such a set-up cannot be realized with multiple video cameras.

Against this background, Fraunhofer HHI has developed the prototype of a mirror-based panoramic 3D camera that can be considered as an approximation of concentric mosaics by using video cameras [27]. Figure 7 shows a close-up view of the test system that has been used as for a proof-of-concept. It uses mirror segments of 24° and two cameras behind each mirror. The stereo cameras are toed-in such that the optical axes intersect at the mirror surface. The stereo baselines can be chosen in a range of 40 to 70 mm to control the depth budget. The 3D camera rig is highly modular and it allows acquisition of live 3D panorama footage up to 180° or even 360°. The vertical field of view is again 60°. For 360° panoramas the resulting resolution is 15,000 by 2,000 pixels per stereo view.

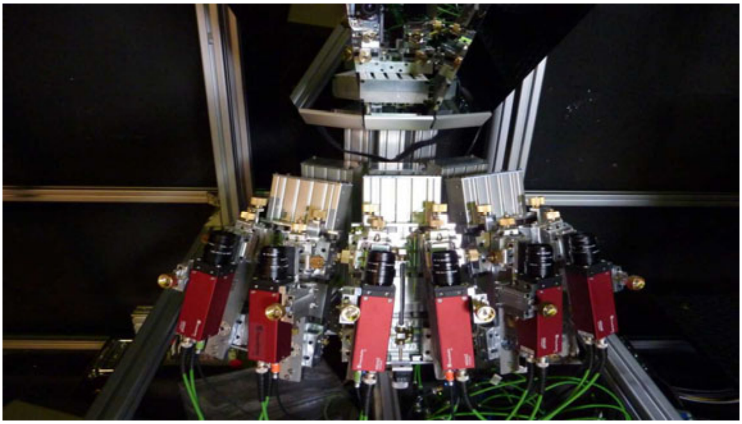


Fig. 7. Close-up view of the multi-stereo-camera arrangement of the 3D OMNICAM

An exact calibration takes care that the systematical approximation error that appears in comparison to the ideal situation of concentric mosaics is minimized. In this context Figure 8 shows the optimal arrangement of the stereo sub-systems. In analogy to Figure 3, the illustration again refers to a horizontal section through the mirror pyramid. The red and blue dots indicate the virtual focal points of the left and right cameras. The red and blue dashed line show the related fields-of-view and camera orientations defined by the mirrors. The black solid lines between the red and blue dots represent the inter-axial distance (baseline) of regular stereo pairs (i.e., both cameras are behind same mirror segment). In contrast, the dashed black lines describe the inter-axial distance between the virtual focal points of a crossed stereo pair (i.e., left and right cameras are from different but adjacent mirror segments).

The optimal state with the minimized systematical approximation error is reached if the regular baselines (solid black lines) are equal to the virtual baselines (dashed black lines). Note that the regular baselines are adjusted physically at stereo rigs themselves whereas the virtual baselines are mainly defined by the distance of the stereo rigs from the mirror surface. Hence, the same regular baseline for all stereo rigs has to be chosen first and then the distances from the mirror rigs to the mirrors have to be selected such that the virtual baselines are equal the regular ones. Finally, the cameras have to be toed-in such that the fields-of-view fit to the borders of the mirror segments.

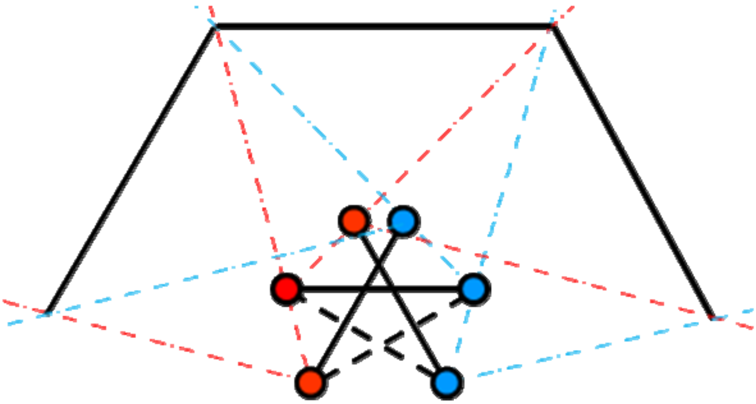


Fig. 8. Concentric arrangement of stereo sub-systems

It is worthwhile to notice that the baselines of the single stereo systems do not intersect at their center as one could assume from the optimal solution of concentric mosaics shown in Figure 5 (right). In fact, the baseline centers are again slightly shifted in radial direction such that they are located at a small circle around the center of the mirror pyramid. On one hand, similar to the 2D-case, this off-center shift ensures some overlap for stitching and seamless blending between neighbored views of the left- or right-eye panorama. But, on other hand, in contrast to the 2D case, it cannot be chosen arbitrarily and is strictly constrained by the above mentioned side-condition that virtual baselines have to equal regular ones. Interestingly, theoretical considerations have shown that the systematical approximation error compared to concentric mosaics is minimized when this more heuristic constraint is fulfilled. Moreover, the off-center shift decreases if the angle of the mirror segments decreases as well and becomes zero in case of infinitesimal mirror angles. Thus, concentric mosaics can be considered as the limiting case of the above constellation.

4 Experimental Results and Proof-of-Concept

Due to the special camera arrangement from Figure 8, the final composition of the 3D panorama consists of image parts from both, regular and virtual stereo pairs as shown in Figure 9 by the original camera images and Figure 10 by using an anaglyph overlay representation after stitching. Hence, the results in Figure 10 already show that panoramic stereo video can be captured with the presented novel multi-stereo camera setup based on mirrors.

The sheared rectangles with the solid white lines in Figure 9 show the effective image borders pruned by the mirror segments. The shearing is given by the fact that cameras are not positioned in the center of the mirror any longer but are moved horizontally by half a baseline to the left or right, respectively, and are additionally toed-in to compensate the shift. Note that the shearing has opposite directions in left and right views due to opposite horizontal movements and toe-ins.

Furthermore, the left image pair in Figure 9 shows the views of a regular stereo pair behind one particular mirror segment. It can be noticed that, in contrast to standard stereo applications, the overlap between the two views is considerably limited. The remaining parts have to be taken from related views in neighbored mirror segments or, in other words, from virtual stereo pairs as discussed in the previous section (see example in the right image pair in Figure 9). These circumstances also explain why the virtual baseline has to be equal to the regular one. Otherwise, the depth scaling would permanently change throughout the entire 3D video panorama.

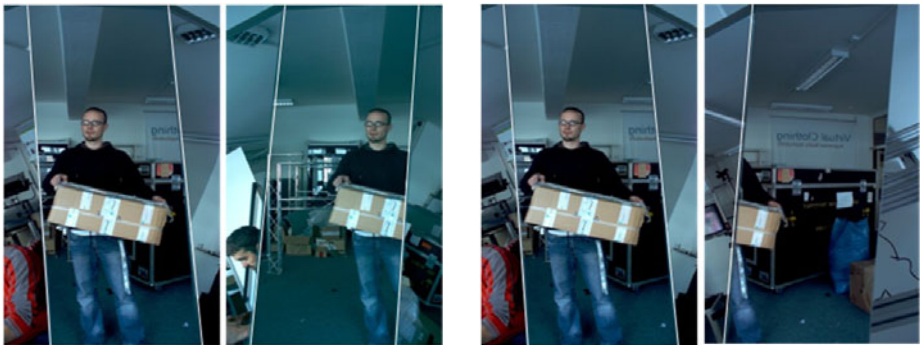


Fig. 9. Stereo content captured by 3D-OMNICAM: left and right view from regular stereo pair at same mirror segment (left) and stereo content from virtual stereo pair across neighbored mirror segments (right)

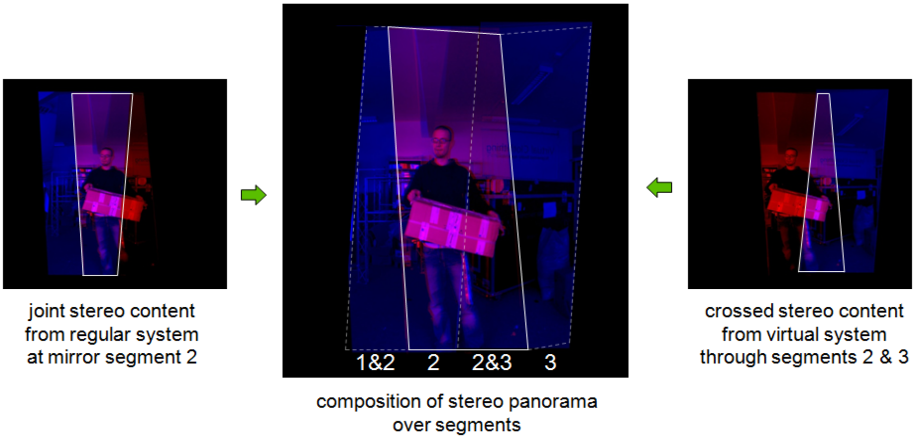


Fig. 10. Composition scheme of 3D OMNICAM to generate 3D video panorama

Fig. 10 shows how these images are overlaid and stitched to obtain the final 3D video panorama. An anaglyph representation has been used for this purpose. The sheared rectangle with the solid white lines in the center image refers to the left view of the stereo rig at mirror segment 2. The dashed white lines in the center image show the right views from the stereo rigs in mirror segments 2 and 3, respectively.

As a consequence, the size of overlapping areas between views with crossed content of a virtual stereo pair (e.g., left view from the stereo rig in mirror segment 2 and right view from the stereo rig in mirror segment 3, see trapeze with white border lines in right image of Figure 10) is almost the same as the one between the views of a regular stereo pair (e.g., both views from the stereo rig in mirror segment 2, see corresponding trapeze in left image of Figure 10).

5 Conclusion

The development of the 3D-OMNICAM is based on long experiences of Fraunhofer HHI in the fields of panoramic video production. Several test productions with ultra-high-resolution panoramic video have been made since opening the TiME Lab in February 2010. They have proven the robustness and practicability of the OMNICAM under real working conditions. The need of a 3D version appeared end of 2010 when the TiME Lab was upgraded to stereo projection using 14 HD beamers and an Infitec system for stereo separation. Following the concept of concentric mosaics, a tricky solution for a 3D-OMNICAM could be found using a mirror rig in combination with a sophisticated arrangement and calibration of the stereo sub-systems. Meanwhile, the 3D-OMNICAM has been implemented and the proof finished successfully. First results have already been screened and evaluated in the TiME Lab.

Acknowledgments. The development of core technologies and key component for the OMNICAM and the TiME Lab has been supported by the German Federal Ministry of Economics and Technology (BMWi).

References

1. Majumder, A.: Intensity seamlessness in multi-projector multi-surface displays. Technical Report, Univ. of North Carolina, Chapel Hill, US (1999)
2. Li, K., Chen, Y.: Optical blending for multi-projector display wall system. In: IEEE Proc. 12th Laser and Electro-Optics Society, vol. 1, pp. 281–282 (1999)
3. Weissig, C., Feldmann, I., Schüssler, J., Höfker, U., Eisert, P., Kauff, P.: A Modular High-Resolution Multi-Projection System. In: Proc. 2nd Workshop on Immersive Communication and Broadcast Systems, Berlin, Germany (October 2005)
4. Gotz, D.: The Design and Implementation of PixelFlex: A Reconfigurable Multi-Projector Display System. Technical Report, Univ. of North Carolina, Chapel Hill, US (2001)
5. Bimber, O.: Multi-Projector Techniques for Real-Time Visualizations in Everyday Environments. In: Proc. IEEE Virtual Reality Conference, Workshop on Emerging Display Technologies (2006)
6. Lantz, E.: A Survey of Large-Scale Immersive Displays. In: Proc. Emerging Display Technology Conference, ACM SIGGRAPH (2007)
7. Fraunhofer IFF. The Elbe Dome: Immerse in Virtual Worlds. VDTC (2011), `http://www.vdtc.de/allg/elbe-dom-eng-fraunhofer-iff.pdf`

8. HPC Market Watch, Seattle Cinerama Grand Reopening (2011), `http://markets.hpcwire.com/taborcomm.hpcwire/news/read?GUID=15456683&ChannelID=3197`
9. Fraunhofer HHI, Official Opening of the HHI TiME Lab. In: Symposium Tomorrow's Cinema – The Future of Content (February 2010), `http://www.hhi.fraunhofer.de/en/events/trade-fairs-and-events-archive/official-opening-of-the-hhi-time-lab/time_ov/official-opening-of-the-hhi-time-lab/`
10. Iwerks, U.: Panoramic Motion Picture Camera Arrangement. Canadian Patent Publication, no. CA 673633 (1963)
11. Majumder, A., Gopi, M., Seales, B., Fuchs, H.: Immersive teleconferencing: A new algorithm to generate seamless panoramic video imagery. In: Proc. of the 7th ACM International Conference on Multimedia, pp. 169–178 (1999)
12. Rees, D.W.: Panoramic television viewing system. United States Patent No. 3, 505, 465 (April 1970)
13. Baker, S., Nayar, S.: A theory of single-viewpoint catadioptric image formation. Int. Journal of Computer Vision 35, 175–196 (1999)
14. MegaVision, The Mega Vision System Overview (October 2004), `http://www.megavision.co.jp/eng/solution/index.html`
15. Point Grey, Spherical Vision, `http://www.ptgrey.com`
16. Carmagus, Endzone, `http://www.camargus.com/maxx-zoom.html`
17. Journal Sentinel, ESPN offers a closer view with Maxx Zoom technology, `http://www.jsonline.com/sports/103677489.html`
18. Immersive Media, Dodeca 2360 Camera System, `http://www.immersivemedia.com/products/capture.shtml`
19. Full View, FC-1005 Camera & FC-110 Camera, `http://www.fullview.com/products.html`
20. Remote Reality, OmniAlert360, `http://www.remotereality.com/omnialert360-productsmenu-121`
21. iMovie Inc, GeoView-3000-LB3, `http://www.imoveinc.com/geoview.php`
22. Sturm, P., Ramalingam, S., Tardif, J.-P., Gasparini, S., Barreto, J.: Camera Models and Fundamental Concepts Used in Geometric Computer Vision. Foundations and Trends in Computer Graphics and Vision, vol. 6(1-2), pp. 1–183 (2010)
23. Tan, K.A., Hua, H., Ahuja, N.: Multiview Panoramic Cameras Using Mirror Pyramids. Trans. on Pattern Analysis and Machine Intelligence 26(7), 941–946 (2004)
24. Peleg, S., Ben-Ezra, M., Pritch, Y.: Omnistereo: panoramic stereo imaging. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(3), 279–290 (2001)
25. Shum, H.-Y., He, L.-W.: Rendering with Concentric Mosaics. In: Proc. SIGGRAPH 1999. ACM, Los Angeles (1999)
26. Chai, J.-H., Tong, X., Chan, S.-C., Shum, H.-Y.: Plenoptic Sampling. In: Proc. SIGGRAPH 1999. ACM, New Orleans (2000)
27. Schüssler, H.-J., Weissig, C., Kauff, P., Eisert, P.: 3D OmniCam. US Provisional Application, No. 61/473595 (April 2011)

The FascinatE Production Scripting Engine

Rene Kaiser, Wolfgang Weiss, and Gert Kienast

Institute of Information and Communication Technologies
Joanneum Research
Graz, Austria
{firstname.lastname}@joanneum.at

Abstract. In the realm of a format agnostic live event broadcast system, the FascinatE Scripting Engines are software components that automate taking decisions on what is visible and audible at each playout device and prepare the audiovisual content streams for display. Essentially, they act together as a Virtual Director with the production team possibly steering it via a backend user interface. We present an architecture for this real-time system and describe interfaces to other production components. Details of subcomponents of the distributed engine, design decisions and technology choices are discussed.

1 Introduction

The FascinatE project aims to create an innovative end-to-end system for immersive and interactive TV services. It allows users to navigate in an ultra-high resolution video panorama, showing live or recorded content, with matching accompanying audio. The output is adapted to the viewing device, covering anything from a mobile handset to an immersive panoramic display with surround sound, delivering a personalized multi-screen experience.

FascinatE is using a panoramic camera with high enough resolution for cropping interesting regions, the OmniCam [1], depicted in Figure 1. It is a collection of 6 HD cameras sharing a single optical centre for obtaining a 180° panoramic video sequence stitched together in real-time from the 6 tiles. The vertical field of view is 60 degrees. The HD cameras are placed on their side and point upwards to a reflecting mirror to maximize the resolution such that when the video sequences are stitched together, the resolution of the final panorama is usually 6984 x 1920 pixels. This resolution allows to capture even distant objects in good quality so that e.g. persons at the other end of a sports field can be detected automatically. The minimum distance of objects to the camera depends on the accuracy of the camera mounting and is roughly two meters. Besides the OmniCam, a range of broadcast cameras such as the ALEXA[1] are used.

In this paper we describe our architecture for an automatic view selection component, the FascinatE Production Scripting Engine (PSE). The PSE takes decisions on what is visible and audible at each playout device and prepares the

[1] http://www.arri.com/camera/digital_cameras/cameras.html

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 682–692, 2012.

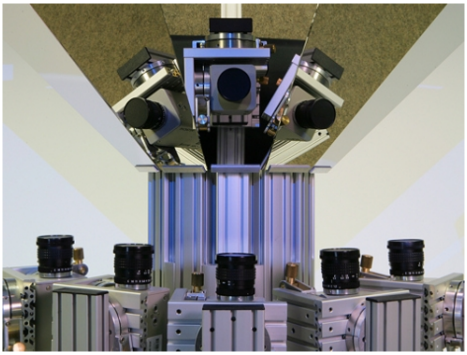


Fig. 1. The OmniCam [1]

audiovisual content streams for display. Such components are commonly referred to as a *Virtual Director*. In order to take reasonable decisions, the engine needs knowledge about what is currently happening in the scene and which camera streams are capturing that action.

The rest of this document is structured as follows. Section 2 elaborates on related work and Section 3 discusses requirements of different scenarios and special features the PSE enables compared to traditional passive broadcast viewing. Section 4 presents relevant components of the FascinatE system that interact with the PSE. The detailed architecture of the PSE follows in Section 5 before we discuss the state of development and discuss future extensions in Section 6.

2 Related Work

A number of approaches using recorded content has been proposed, e.g. the interactive storytelling system proposed by Kim et al. [2] or the narrative structure language (NSL) [3] developed within the NM2 project[2]. However, our research problem here is a significantly different one due to the real-time decision making requirement, the lack of comprehensive high level annotations and the lack of a thoroughly authored story frame. In the following, we discuss and compare a number of innovative systems.

The Production Scripting Engine processes a stream of events and makes decisions based on the events utilizing temporal reasoning. Event processing has been identified as a feasible solution to implement this as it can be integrated well with a rule engine. A comprehensive overview of this research field has recently been compiled in [4]. A survey of methods for automatic interpretation and understanding video events is presented in [5]. Methods of abstracting video data and how to model events are investigated. For event modeling, the authors cite Finite-State machines, Bayesian Networks, Hidden Markov Models, Dynamic Bayesian Networks, Conditional Random Fields, Grammar models, Petri Nets, Constraint Satisfaction and Logic approaches.

[2] http://www.ist-nm2.org/

A Virtual Director commonly refers to an intelligent software system that attempts to automatically frame and select between multiple video camera views, essentially replacing a human director and camera operator crew. The Virtual Director in the TA2[3] [6] system aims at taking such decisions in a different domain. The *Orchestration Engine*, as it is called, selects between a range of streams in an attempt to support group-to-group communication in scenarios of social video conferencing. Thereby low-level event streams from audiovisual content analysis are processed based on a rule set that defines how to abstract from it for the detection of higher-level events. Both this and the decision making process are realized using the JBoss Drools event processing engine.

The Apidis[4] project aims at automatically producing personalized multimedia content. Personalized in this context means that the user's preferences such as the preferred team or player as well as the user's profile, history and device capabilities are considered for selections. The autonomous generation of team sports video content is presented in [7]. Players are detected, their movements are tracked and the scoreboard is permanently analyzed where events and states of the game are extracted. One of the principles they follow is called "smoothness" [8] which includes generating a smooth moving sequence of cameras and viewpoints based on their individual optima. Thereby statistical inference is used to recover noise-distorted camera input and two Markov Random Fields with statistical physics are used to model viewpoint and camera smoothing.

The My eDirector 2012[5] [9,10] project aims to create context-aware and personalized media but in real-time streaming environments for large scale broadcasting applications. This allows end users to direct their own coverage of large athletic events and to create their own personal Virtual Director. Raw video content is enriched with annotations generated by scene analysis, person tracking and other sensor data. This annotated media stream is used to make camera decisions based on the user's profile.

NoTube[6] [11] brings TV to a community enhanced platform by connecting TV content and user data using Semantic Web technologies. A user profile models user activities is generated by aggregating data from Social Network activities, e.g. on Facebook or delicious. TV metadata is enriched with structured data from the Linked Data cloud. User recommendations are then generated with the average semantic distance between the interests of the user and potentially identified TV program items.

Within BBC's Automated Coverage project[7] another example of an approach to automate camera selection was investigated. A generic spring model was applied for camera framing such that moving objects were covered with smooth pans.

[3] http://www.ta2-project.eu/

[4] http://www.apidis.org/

[5] http://www.myedirector2012.eu/

[6] http://notube.tv/

[7] http://www.bbc.co.uk/rd/projects/virtual/automated-coverage/

Overall, there is a number of activities researching into *beyond HD*. NHK's Super Hi-Vision[8] system for example offers an even larger resolution than the OmniCam, though a flat image. NHK targets several features such as program customization, recommendation and Social TV services. However, it's not only the higher resolution but the wide range of potential new features that are noteworthy. A high degree of production automation is not only interesting for economic reasons but enables a range of features for the benefit of the viewer, parallelizing personalization capabilities and more.

3 Scenarios and Features

Our system supports a set of features for the viewer that go beyond traditional broadcast viewing (e.g. multi-language audio) and achieve a higher degree of personalization through interaction. The FascinatE system is format agnostic which means that it produces live content streams for different playout device types in parallel, effectively reducing production cost.

Viewers are given increased freedom to interact with the system in order to individually select what they want to see — directly or via more abstract options. In one extreme case, a viewer would not interact with the system at all and watch the default coverage, on the other side heavy interaction goes as far as free view navigation within the high-resolution content on dedicated devices. We are currently evaluating intuitive use of gesture control for such purposes, using the Kinect sensor (compare this concept from the videoconferencing domain [12]). As a human production team could not cater for a large audience in parallel, the PSE is needed to automate content selection as far as possible. The format agnostic production system reasons for different viewer groups in parallel. There is at least one group per playout device type, and possibly more implied by the options available in the viewer interface.

One key feature is the the ability to closely follow an object such as a person, or to follow groups such as athletes from a certain country. The PSE's intelligent behavior aims to ensure that, despite these selections, actions of a high priority are overruling specific user preferences to the benefit of the viewer experience. As an example, in a football situation where a touchdown (very high priority event) is likely to happen next, the closeup-view of a player apart from the scene will be left to make sure the viewer doesn't miss the more important action. This of course requires basic prediction functionality that can also lead to improper behavior at times.

Another feature is the viewer-specific replay generation. Based on information derived from the user profile, the PSE selects events in which individuals are specifically interested. Saving on production cost, the duration, i.e. the length of the replay before and after the low-level annotation, is determined automatically based on the type of the event according to the scenario specific event (ontology) model.

[8] `http://www.nhk.or.jp/digital/en/super_hi/index.html`

FascinatE will implement a range of scenarios of increasing complexity. Options are events of considerable public interest suited for live broadcast and include sports events as well as music concerts and parades. An example is represented by the confrontation of two different use cases:

- A soccer match usually shows a single main event, generally determined by the presence of the ball and its position on the field, moving at a relatively slow speed. All actors are present on the same playfield at the same time, participating at the same actions and interacting with each other.
- During an athletics meeting, actors are distributed on a much wider area, that cannot be framed within a single shot of the OmniCam as occlusions occur and distort the viewing experience. Actors are grouped over different areas of the playfield, where they take part in different actions. Different events of interest may occur at the same time. Interaction happens mainly between athletes active in the same area.

4 The FascinatE System

The following will introduce components of our system which have relevant interfaces to the PSE. In our context we define a *shot* as a certain camera view that has a different framing size per playout device. A shot can be static or e.g. follow a moving object as a closeup.

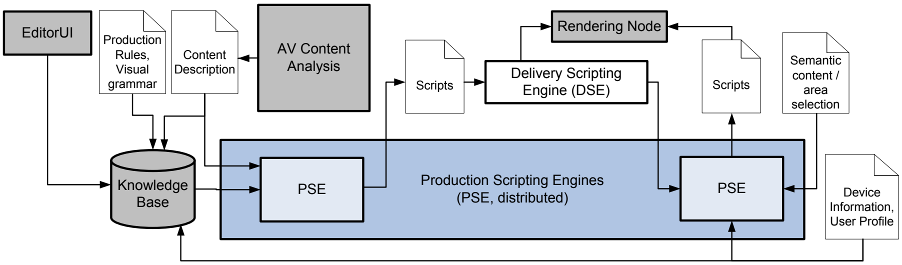

EditorUI
Production Rules, Visual grammar
Content Description
AV Content Analysis
Scripts
Rendering Node
Delivery Scripting Engine (DSE)
Scripts
Semantic content / area selection
Knowledge Base
PSE
Production Scripting Engines (PSE, distributed)
PSE
Device Information, User Profile

Fig. 2. Detail of the FascinatE system architecture

4.1 AV Content Analysis

So far, AV content analysis has only been applied to the panoramic image, not the broadcast cameras. Independent of the scenario a person tracking module as described in [13] is informing the PSE of the location of persons in the scene. Additional audiovisual analysis components include the extraction of a saliency measure for regions and the detection of scenario-specific events, implemented using a machine learning based classification approach.

4.2 EditorUI Tools

Further, the PSE is tightly integrated with an user interface for the professional production team, the EditorUI tools[9] (see screenshot in Figure 3 and also Figure 4 for information flow). They allow to create live annotations for concepts not covered by AV content analysis. The main tasks are identity assignment for a subset of the person tracks (e.g. most interesting persons in a concert, sports match), live action annotation, steering the PSE through commands and selection between shot options provided by the PSE (effectively a re-prioritization). They could further serve as a tool for validating and correcting the results of content analysis.

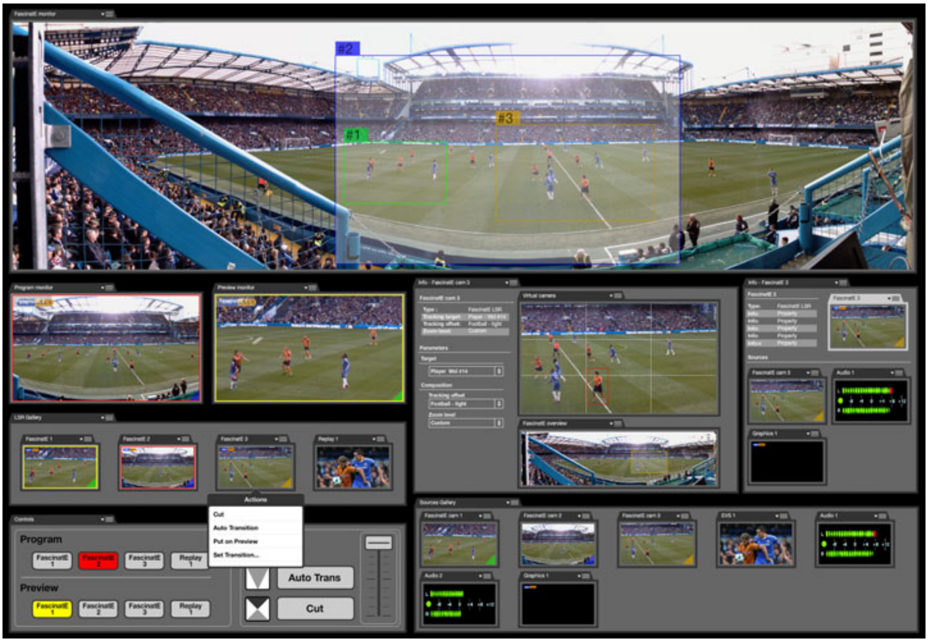


Fig. 3. Sketch of the EditorUI interface. The upper part renders the whole panorama and displays views that are defined by the PSE as color overlay boxes.

4.3 Delivery Scripting Engine (DSE)

The PSE is closely working together with another type of Scripting Engine, the Delivery Scripting Engine (DSE). The DSE is taking instructions from the PSE to prepare content streams and makes sure needed content is available for the renderers, optimizing bandwidth management.

[9] The toolset was designed by The Interactive Institute, Sweden, http://www.tii.se/.

5 The Production Scripting Engine

The Production Scripting Engine (PSE) is responsible for decision making on content selection. The key feature is to automatically select a suitable area within the OmniCam panorama image, in addition to cuts to human operated broadcast cameras. Selection behavior is based on pragmatic (cover most interesting actions) and cinematographic (ensure basic aesthetic principles) rules, comparable to the approach proposed by Falelakis et al. [6]. The decision making process is not always fully automatic but can involve supervision by a human-in-the-loop, a production team member deciding between prepared options using the EditorUI tools. The PSE is a distributed component with at least one instance at the production site and one at the terminal end[10], which is also reflected in the PSE architecture diagram in Figure 4.

The primary PSE is consuming real-time low-level event streams as extracted by AV analysis and generated by the EditorUI (near real-time annotations). Low-level means that the information bits are per se not directly usable for the decision making process, as they are very narrow facts/statements about details of the scene. What the PSE really needs to have in order to take quality scripting decisions is a more abstract understanding of the current situation, though. As an example, a sequence of coordinates of players and a ball are relatively meaningless. However, a certain constellation might indicate a very high probability that a basket/goal is to be scored in the upcoming seconds, and the PSE might want to react to such a situation in a specific way.

The output of the PSE is called a *script*, which consists of a combination of content selection options and decisions, renderer instructions, user interface options a.s.o. Scripts (custom XML format) are passed to subsequent PSE components from the production site towards the terminal, where final instructions are given to a device-specific renderer. The Terminal PSE processes scripts received from previous PSE instances. When the rules allow to do so[11], it determines a single shot update per viewing group and sends final decisions as instructions to the renderer.

All PSE instances keep a local state to keep track of their own decisions, which is required for a number of features, e.g. a certain shot variety as an aesthetic principle. Different shots (fixed, dynamic as e.g. following an object) and shot types will be modeled as a OWL2 ontology model. Event types of this controlled vocabulary are specific to a scenario. In the example of soccer content they include for example different zoom level side views, player closeups, audience cheer pans, fixed goal area shots etc.

Most of the PSE's subcomponents will use JBoss Drools[12] which is a unified and integrated platform for rules, workflows and event processing. It is a hybrid chaining rule engine implementing a forward and a backward chaining inference

[10] More complex scenarios will require additional levels of decision making in between, taking into account licensing and content rights, targeted advertisements, event audience privacy issues and such.

[11] Aesthetic rules ensure e.g. that cuts between shots don't happen too often.

[12] http://www.jboss.org/drools

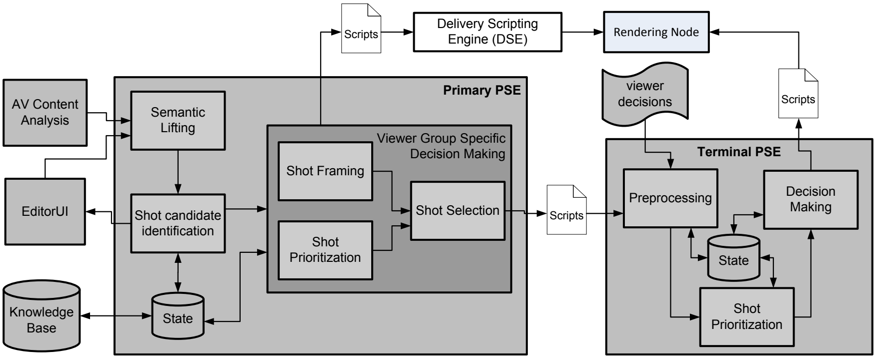


Fig. 4. The PSE architecture with the production site component left and the terminal end component right

engine. The forward chaining rules are processed using an extended version of the Rete [14] algorithm called ReteOO for efficient pattern matching. The business logic, the rules, can be expressed in a declarative way by using the native Drools Rule language, a XML rule language or a self defined domain specific language.

Summarizing, the EditorUI and the AV content analysis are the main metadata (knowledge) sources for decision making within the PSE. The following sections will describe the individual subcomponents of the PSE.

5.1 Semantic Lifting

The Semantic Lifting component is designed to receive low-level events from both AV content analysis and the EditorUI component. It is the first interface of the PSE subcomponents and processes event streams, therefore it must be able to handle a high volume of input events. Low-level events from the AV analysis are sent in MPEG-7 format[13] and include bounding boxes of detected persons.

The purpose of the Semantic Lifting component is to filter and enrich (metadata such as confidence values) the incoming information. Further, higher-level events are extracted from the low-level event stream. This can be achieved in various forms, e.g. by observing trends and sudden changes, by detecting predefined patterns in the event stream, by fusing audio and video events, by doing fuzzy classification or by a combination of these methods. As a result the Semantic Lifting component outputs events of a higher semantic level that can be directly used to determine shot candidates.

After defining the first set of rules and sketching the system's architecture we identified following requirements — the Semantic Lifting component must be able to:

[13] Multimedia Content Description Interface, ISO/IEC 15938:2001

- Process event streams in real-time.
- Handle possible out of order events.
- Reason with uncertain information (i.e. confidence values of low-level events).
- Express and reason on First Order Logic statements, as predicate logic is not sufficient.
- Perform spatial, spatiotemporal and temporal reasoning including Allen's temporal operators [15].

5.2 Shot Candidate Identification

The shot candidate identification aims to determine suitable candidates based on the high level event information as provided by Semantic Lifting. The output are not final decisions, but options that subsequent components use to take decisions for each individual user group based on a range of factors prioritizing different types of shots and the event/action types assigned to them. The component determines at least one candidate, the actual number depends on the scenario. We have yet to determine the optimal number given further parameters such as the EditorUI integration. In that frontend for the production team, options are likely to be visualized as rectangular color overlays on the panoramic image. This seems to work well for visualization and to assist decision, but only for a limited number of views at the same time.

For the identification itself a number of domain-specific rules are executed by JBoss Drools to immediately and properly react to actions in the scene. As previously mentioned, we are using our event/action ontology for classification of interesting actions, so that we can determine the most interesting ones. Even though the dynamic occurrence of unexpected events might require additional views to be added at times, one strong design principle is to ensure continuity, to work with a rather constant number of views. The initial design of this component will be revisited as soon as practical lessons learned allow to derive its ideal behavior.

5.3 Shot Framing

For each shot candidate the Shot Framing component computes a suitable framing bounding box. Size and aspect ratio differ per viewing device and the decision gives the coordinates of the surrounding box. The component deals with both fixed shots and such covering moving objects for which *smooth* camera pans are needed. The calculation employs a spring model for smoothing out minor movements and avoiding rough stops. It takes the object type, direction and speed of movement into account. As an example, a horizontally moving athlete is positioned side of the image center so that more of the running direction area is seen. Further, the bounding box size depends on the distance of the object to the camera, i.e. more distant objects are covered by smaller boxes so that they appear larger.

5.4 Shot Prioritization and Shot Selection

Naturally, shot prioritization and selection have to operate per viewer group as well. In our current architecture, the Shot Prioritization component is a preparation step within the workflow which (re-)assigns priorities to shots based on a range of factors. The Shot Selection component consumes that input and either takes a final decision or computes a set of suitable options to be sent to the next PSE instance in the network. The terminal PSE for example has to take final decisions, other PSE instances may pass on a list of options. Options might but not necessarily have to correspond to preferences in the viewer's menu. Shot selection takes the model from the viewer profile and of the shot's associated metadata into account. The recent viewing history is also a factor e.g. to ensure variety in types of shots ensured by cinematographic rules. At the terminal end final decisions have to be made, therefore device specific transition commands are included in the scripts as well. Decision scripts are passed to the renderer.

6 Discussion

We have presented an architecture for a *Virtual Director* implementation that supports a format-agnostic live event broadcast production system and is able to frame shots within a high-resolution panoramic video stream. The architecture design and technology choice has been informed by lessons learned in a number of related activities (cp. [6]). We are at the beginning stage of the implementation of the Production Scripting Engine and plan to test and evaluate event processing and rule evaluation performance, system delay and viewer satisfaction.

Developing a rule-base that realizes the desired behavior is not a straightforward engineering task. There is little related work on the formal representation of the principles that broadcast teams operate on. Such principles are typically expressed as event-condition-action (ECA) rules that consider their context in the condition, but are still expressed independent from each other. However, the challenge is to achieve a sound behavior by the *sum* of the rules, their interplay. As we target different scenarios (mainly sports events and music concerts), effective re-use of a subset of rules is desired. One example in that regard should be the ability to smoothly follow an object moving through the panoramic image.

These issues will mainly influence the implementation of the Semantic Lifting and Shot candidate identification components and likely reveal interesting research questions. FascinatE further aims for intelligent audio orchestration. As an example for sports broadcast, if there is an audience shot after a successful score, the audio should correspond (loud cheers). More general, a viewer may want to hear the fans of his/her favorite team more than those of the opposing — the visibility of objects should correspond to their audibility. Objects that are currently not visible may not be audible at all or at a dimmed level. Audio close-ups might become problematic though, as they may interfere with the privacy of a player-coach discussion, or may cover inappropriate language of players, etc.

Acknowledgements. The research leading to this paper has been partially supported by the European Commission under the contract FP7-248138, “FascinatE – Format-Agnostic SCript-based INterAcTive Experience” (http://www.fascinate-project.eu/).

References

1. Schäfer, R., Kauff, P., Weissig, C.: Ultra high resolution video production and display as basis of a format agnostic production system. In: Proceedings of International Broadcast Conference, IBC 2010 (2010)
2. Kim, S., Moon, S., Han, S., Chang, J.: Programming the story: Interactive storytelling system. Informatica 35(2) (2011)
3. Ursu, M.F., Kegel, I., Williams, D., Thomas, M., Mayer, H., Zsombori, V., Tuomola, M.L., Larsson, H., Wyver, J.: Shapeshifting tv: interactive screen media narratives. Multimedia Systems 14(2), 115–132 (2008)
4. Etzion, O., et al.: The event processing manifesto. In: Dagstuhl Seminar Proceedings (10201), pp. 1–60 (2011)
5. Lavee, G., Rivlin, E., Rudzsky, M.: Understanding video events: a survey of methods for automatic interpretation of semantic occurrences in video. Trans. Sys. Man Cyber Part C 39(5), 489–504 (2009)
6. Falelakis, M., Kaiser, R., Weiss, W., Ursu, M.: Reasoning for Video-Mediated Group Communication. In: Proceedings IEEE International Conference on Multimedia & Expo (July 2011)
7. Chen, F., Delannay, D., Vleeschouwer, C.D.: Multi-sensored vision for autonomous production of personalized video summaries. In: 2nd International ICST Confrence on User Centric Media, Spain, Palma de Mallorca (September 2010)
8. Chen, F., Vleeschouwer, C.D.: Autonomous production of basketball videos from multi-sensored data with personalized viewpoints. In: WIAMIS, pp. 81–84 (2009)
9. Patrikakis, C., Pnevmatikakis, A., Chippendale, P., Nunes, M., Santos Cruz, R., Poslad, S., Zhenchen, W., Papaoulakis, N., Papageorgiou, P.: Direct your personal coverage of large athletic events. IEEE Multimedia (2010)
10. Patrikakis, C., Papaoulakis, N., Stefanoudaki, C., Voulodimos, A., Sardis, E.: Handling multiple channel video data for personalized multimedia services: A case study on soccer games viewing. In: 2011 IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops (2011)
11. Schopman, B., Brickley, D., Buser, V., Miller, L., van Aart, C., Aroyo, L., Linguerri, S., Malaisé, V., Minno, M., Mostarda, M., Nixon, L., Palmisano, D., Raimond, Y., Siebes, R.: Notube: making the web part of personalised tv. In: WEBSCI 2010 (2010)
12. DeVincenzi, A., Yao, L., Ishii, H., Raskar, R.: Kinected conference: Augmenting video imaging with calibrated depth and audio. In: Human Factors, pp. 621–624 (2011)
13. Kaiser, R., Thaler, M., Kriechbaum, A., Fassold, H., Bailer, W., Rosner, J.: Real-time person tracking in high-resolution panoramic video for automated broadcast production. In: Proceedings of CVMP 2011 (2011)
14. Forgy, C.L.: Expert systems, pp. 324–341. IEEE Computer Society Press, Los Alamitos (1990)
15. Allen, J.F.: Maintaining knowledge about temporal intervals. Commun. ACM 26, 832–843 (1983)

Linking User Generated Video Annotations to the Web of Data

Michiel Hildebrand[1] and Jacco van Ossenbruggen[1,2]

[1] VU University Amsterdam,The Netherlands
[2] CWI Amsterdam, The Netherlands

Abstract. In the audiovisual domain tagging games are explored as a method to collect user-generated metadata. For example, the Netherlands Institute for Sound and Vision deployed the video labelling game *Waisda?* to collect user tags for videos from their collection. These tags are potentially useful to improve the access to the content within the videos. However, the uncontrolled tags allow for multiple interpretations, preventing long term access. In this paper we investigate a semi-automatic process to define the interpretation of the tags by linking them to concepts from the Linked Open Data cloud. More specifically, we investigate if existing web services are suited to find a number of candidate concepts, and if human users can select the most appropriate concept from these suggestions. We present a prototype application that supports this process and discuss the results of a user experiment where this application is used with different data sources.

Keywords: User-generated metadata, video annotation, linked data, tag reconciliation.

1 Introduction

Recently, the Netherlands Institute for Sound and Vision collected a large number of user-generated annotations with the video labelling game *Waisda?* [9]. With these time-based user tags the institute aims to improve access to the content within their videos. Currently, the professional annotations only describe videos as a whole, while their users predominately place orders for fragments within a video [7]. However, the uncontrolled nature of the user tags make it difficult to guarantee long term access. In general, the tags allow for multiple interpretations: named entities, such as persons, typically contain only part of the name (e.g. only the first or last name), and (ii) the subject terms are only available in the vocabulary used by the players of the game, which might not coincide with the vocabulary of a searcher.

A typical solution on the Semantic Web to define the interpretation of a textual value is by linking it to a concept defined by a vocabulary publicly available on the web. This process is also known as “reconciliation”[1]. Typically,

[1] The term was coined by the authors of the data cleaning tool Freebase Gridworks, now Google Refine.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 693–704, 2012.

this is a semi-automatic process where a reconciliation service suggests a number of candidate concepts and the user selects the most appropriate one. In this paper, we investigate if this approach can be applied to the tags of the user-generated video annotations from *Waisda?*.

More specifically, our research questions are:

1. what is the coverage of different reconciliation sources with respect to *Waisda?* tags?, and,
2. to what extent can human users use these services to quickly and correctly select the most appropriate concept in context of a video?

To investigate these questions, we focus on two services. First, Freebase[2] provides a reconciliation service for their community build database, including structured information from Wikipedia. Second, the semantic layer of Europeana[3] provides a reconciliation service for several controlled vocabularies of (cultural) institutions, including the in-house vocabulary of the Netherlands Institute for Sound and Vision. To test reconciliation for the Waisda tags against these datasources, we implemented a user interface that allows the user to (i) select a data source to reconcile against, and (ii) select a concept from this source that is appropriate in context of the video. We then conducted a small user experiment where four participants used this interface to reconcile the *Waisda?* tags associated with a historic newsreel, and analyzed the results.

In the remainder of this paper we first describe in Section 2 the *Waisda?* video labelling game, discuss the nature of the collected tags and demonstrate how reconciliation of these tags could enable long term access. Next, in Section 3, we briefly review related work on tag reconciliation. Our approach for reconciliation of user-generated video annotations is explained in Section 4. We describe the user experiment with our prototype implementation in Section 5, and discuss the lessons learned from this experiment in Section 6. Finally, we wrap-up the paper in Section 7 by discussing the limitations of our approach and directions for future work.

2 *Waisda?* Video Labelling Game

Waisda? is a multiplayer game where users describe videos by entering tags [9]. The development of the game was initiated and guided by the Netherlands Institute for Sound and Vision and developed by a Web development company. In the first pilot project the game was used to annotate digitized historic newsreels as well as more recent TV episodes from a Dutch broadcaster. The homepage of *Waisda?* contained four channels, each continuously streaming videos from a predefined category. A player starts a game by selecting one of the channels, and plays against other players that joined the same channel. Figure 1 shows a screenshot of the game with a video from a Dutch TV episode. During the game

[2] http://www.freebase.com

[3] http://semanticweb.cs.vu.nl/europeana/

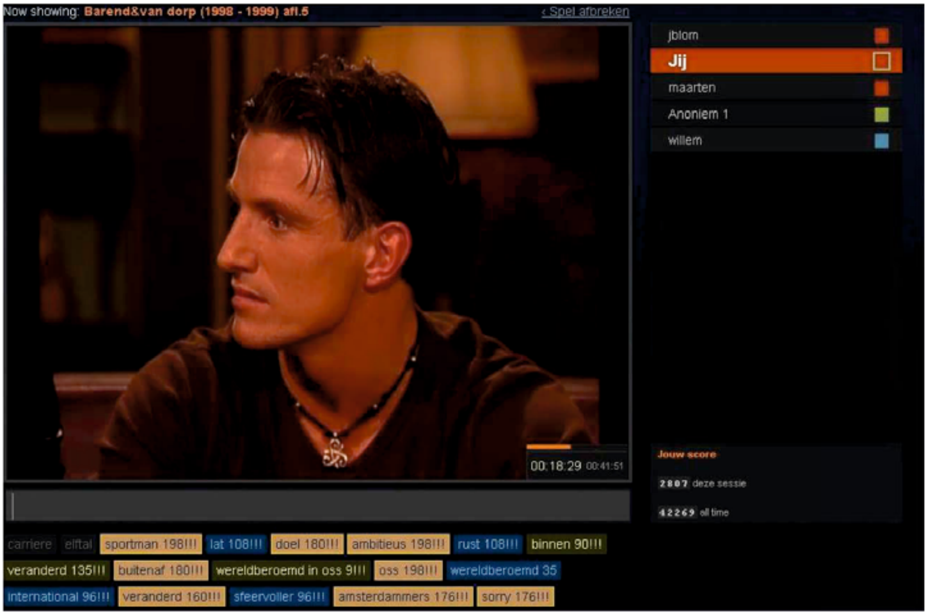


Fig. 1. Screenshot of the *Waisda?* labelling game, where a user is tagging an episode of a Dutch TV talk show

the user can score points by entering tags in the textfield, shown below the video. Based on the principles of Louis van Ahn's ESP image labelling game [14], points are scored when two players enter the same tag. As in the Yahoo! Video Tag game [13] this notion of the same tag is extended to a 10 second time-interval. The tags added by the user are shown below the textfield, and the points scored with a tag are indicated by the different colors. The ranking of the players in the current game are displayed to the right of the video.

In the first pilot with *Waisda?* more than 420,000 tags were added to 612 videos, an average of almost 700 tags per video. Table 1 shows a list of tags assigned to a newsreel about the 1937 visit of the Dutch royal family to the coronation of George VI. We will use this example throughout the rest of the paper. As the game targeted a Dutch audience the tags are primarily in Dutch. They contain person names, such as "Elisabeth" and "Juliana", locations, such as "England", and subject terms, such as "paarden" (horses in Dutch) and "beefeater".

Searching Within Videos. Based on previous work [4] we expect that the time-based annotations collected in *Waisda?* can be used to support users with the task of finding objects within a video. For example, the time-based tags allow the user to directly navigate to the point in the video showing "beefeaters", or find specific content by searching through the tags. By reconciling the tags

Table 1. Example list of *Waisda?* tags assigned to the George VI coronation newsreel from 1937, containing person names, locations and subject terms

gouden koets, wegrijden, elisabeth, god save the king, hye park, westminster abbey, abbey, priester, geestelijken, Hyde, millitairen, kanonnen, beefeater, regen, hek, paarden, zwart, straat, tocht, aankomst, kerk, intocht, stoet, koets, kroning, mensenmassa, parade, rust, juliana, koning, kroon, niets, engeland, bernhard, park, troon

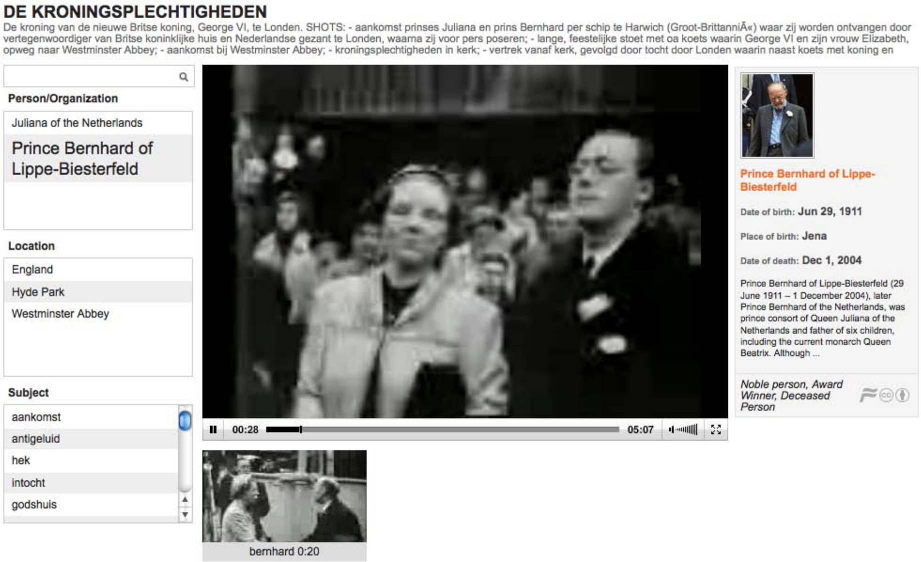
against concepts this functionality can be enhanced in several ways. The screenshot in Figure 2 shows a prototype that supports search and browsing within a video, which is using the tags that were reconciled in the user experiment described in Section 5. In the middle it contains a video player and to the left the reconciled tags. The reconciliation has uniquely identified the tags, as shown by the full name for the persons and locations. For example, the concept with label "Prince Bernhard of Lippe-Biesterfeld" instead of the tag "Bernhard". The type information available for the concepts enables a categorization of the tags, as shown on the left side of the screenshot. The rich information of the concepts extends the possible ways to find tags. For example, by reconciliation to Freebase the subject term "beefeater" can also be found by the synonym "yeamon warders". The mappings from the Dutch version of WordNet, Cornetto, to the English version enable multilingual access. Finally, the links to the concepts provide rich background information for each tag, as shown to the right of the video in the screenshot of Figure 2.

3 Related Work

This work is inspired by Google Refine, a tool to clean and transform tabular data[4]. Among other operations it allows the user to reconcile data values to the concepts from an external source. By default Google Refine suggests topics (locations, persons, movies etc.) from Freebase. The task of the user is then to select the most suited suggestion in context of the table row in which the value occurs. For our task, Google Refine is not directly suited as the *Waisda?* tags have to be reconciled in context of the video.

Google Refine requires a reconciliation service to be able to return a ranked list of candidates for a given string. Most Linked Data providers do not yet support this service. Therefore, Maali et.al. investigated how this reconciliation can be supported by existing Linked Data services [8], such as the standard query language for the Semantic Web SPARQL [11] or the Semantic Web search engine Sindice [12]. Their evaluation showed that it is feasible to use these sources for reconciliation, and they implemented extensions for Google Refine to access these services. It should, however, be noted that reconciliation interfaces require high precision results because they can show only a limited number (typically

[4] http://code.google.com/p/google-refine/



DE KRONINGSPLECHTIGHEDEN

De kroning van de nieuwe Britse koning, George VI, te Londen. SHOTS: - aankomst prinses Juliana en prins Bernhard per schip te Harwich (Groot-BrittanniÃ«) waar zij worden ontvangen door vertegenwoordiger van Britse koninklijke huis en Nederlandse gezant te Londen, waarna zij voor pers poseren; - lange, feestelijke stoet met oa koets waarin George VI en zijn vrouw Elizabeth, opweg naar Westminster Abbey; - aankomst bij Westminster Abbey; - kroningsplechtigheden in kerk; - vertrek vanaf kerk, gevolgd door tocht door Londen waarin naast koets met koning en

Person/Organization

- Juliana of the Netherlands
- Prince Bernhard of Lippe-Biesterfeld

Location

- England
- Hyde Park
- Westminster Abbey

Subject

- aankomst
- antigeluid
- hek
- intocht
- godshuis

Prince Bernhard of Lippe-Biesterfeld

Date of birth: Jun 29, 1911

Place of birth: Jena

Date of death: Dec 1, 2004

Prince Bernhard of Lippe-Biesterfeld (29 June 1911 – 1 December 2004), later Prince Bernhard of the Netherlands, was prince consort of Queen Juliana of the Netherlands and father of six children, including the current monarch Queen Beatrix. Although ...

Noble person, Award Winner, Deceased Person

Fig. 2. Screenshot of prototype with enhanced access to tagged video. To the left of the video the reconciled tags are categorized by their type. To the right of the video background information of the current content is shown.

in the range from three to ten) of suggestions to the user to choose from. In general, such a high precision is only reached by using additional restriction on the type of the results, e.g. persons. In our task this information is not available.

There are several approaches to link tags to concepts fully automatically. For example, for pictures on `Flickr.com` [1,10] or videos on `Youtube.com` [3]. We believe that such automatic techniques could also be applied to link the user tags from *Waisda?*. However, it is unlikely that the precision of such algorithms will reach the quality standards of an archive for long term preservation. Therefore, the need for human assessment remains. Furthermore, this work focuses on the use of existing reconciliation services to find candidate concepts. We hope our use case inspires the integration of advanced ranking algorithms into such services. Finally, the interactive approach presented in this paper also gives us the opportunity to collect a golden standard, which we can use for the evaluation of more advanced suggestion algorithms in future work.

4 Linking Video Annotations to Concepts

In this section we present our semi-automatic approach to link the tags collected in *Waisda?* to concepts from the Linked Data cloud. At the backend we use existing reconciliation services to collect a number of candidate concepts, and at a front-end we provide a graphic user interface that allows the user to select the appropriate candidate.

4.1 Reconciliation API

The reconciliation API, as introduced in Google Refine, is a web service that links textual values to database identifiers. The service is intended to be used semi-automatically, where the service suggests a number of candidate concepts and the human user selects the concepts appropriate for her case. The algorithm to match a query to a concept is not specified in the API. Typically, it performs a fuzzy string match between the query and the textual labels of the concepts, for example the name of a location or person.

The main parameters of the service are straightforward: a query object with one or more textual values, an optional specification of the type of the results, and a limit on the number of results returned[5]. The service returns for each query an identifier, the name, all available types, the score produced by the algorithm and a boolean that is true when the service is confident enough to indicate the match as the right candidate.

4.2 Linked Open Data Sources

The Linked Open Data initiative has inspired many data providers to publish their datasets on the Web [2]. Popular sources are DBPedia[6], containing structured data from Wikipedia, and Geonames[7] as a source for geographic locations. We expect that these sources are useful for reconciliation of the *Waisda?* tags. However, the data providers do not provide a reconciliation service. Recently, Talis launched an initiative to provide several services for several datasets from the Linked Data cloud, including reconciliation[8]. However, after exploration we observed that the ranking of the results was not sufficient for our purposes. Freebase provides an alternative to these data sources, is also available as Linked Data and provide a public reconciliation service.

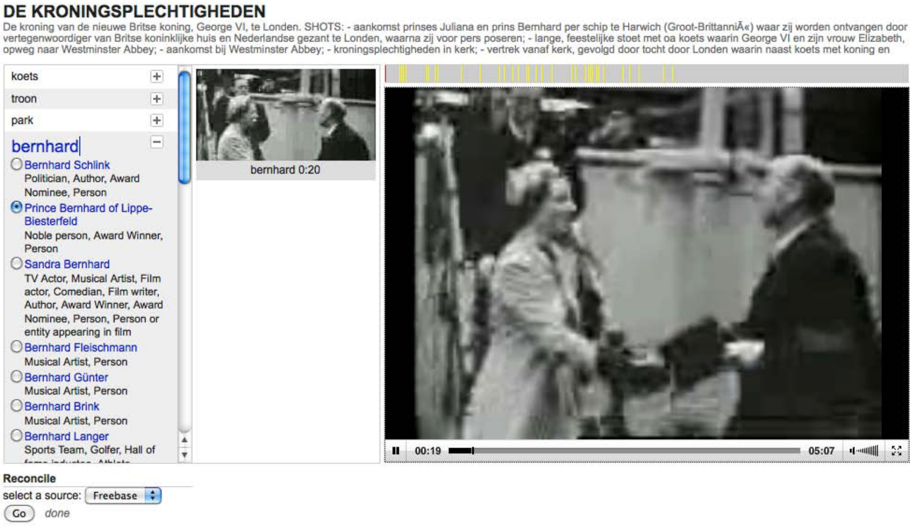
Europeana provides another useful Linked Data source for our tag set. The semantic layer of Europeana contains a large number of controlled vocabularies from different (cultural) institutes. Within the reconciliation service each datasource can be accessed separately. In particular, we consider two sources to be relevant for the tags in *Waisda?*. First, the Netherlands Institute for Sound and Vision uses an in-house thesaurus for the documentation of audiovisual content. This so-called GTAA thesaurus (Dutch acronym for Common Thesaurus Audiovisual Archives) contains approximately 160,000 terms in six facets: subjects, locations, person names, organization names, maker names and genres. Second, Cornetto, a WordNet-like lexical semantic database of Dutch that contains 70,000 synsets [15].

[5] The full specification is available at: http://code.google.com/p/google-refine/wiki/ReconciliationServiceApi

[6] http://www.dbpedia.org

[7] http://www.geonames.org

[8] http://www.kasabi.com



DE KRONINGSPLECHTIGHEDEN

De kroning van de nieuwe Britse koning, George VI, te Londen. SHOTS: - aankomst prinses Juliana en prins Bernhard per schip te Harwich (Groot-BrittanniÃ«) waar zij worden ontvangen door vertegenwoordiger van Britse koninklijke huis en Nederlandse gezant te Londen, waarna zij voor pers poseren; - lange, feestelijke stoet met oa koets waarin George VI en zijn vrouw Elizabeth, opweg naar Westminster Abbey; - aankomst bij Westminster Abbey; - kroningsplechtigheden in kerk; - vertrek vanaf kerk, gevolgd door tocht door Londen waarin naast koets met koning en

koets +
troon +
park +
bernhard −

○ Bernhard Schlink
Politician, Author, Award Nominee, Person

● Prince Bernhard of Lippe-Biesterfeld
Noble person, Award Winner, Person

○ Sandra Bernhard
TV Actor, Musical Artist, Film actor, Comedian, Film writer, Author, Award Winner, Award Nominee, Person, Person or entity appearing in film

○ Bernhard Fleischmann
Musical Artist, Person

○ Bernhard Günter
Musical Artist, Person

○ Bernhard Brink
Musical Artist, Person

○ Bernhard Langer
Sports Team, Golfer, Hall of

bernhard 0:20

00:19 05:07

Reconcile

select a source: Freebase

Go *done*

Fig. 3. Screenshot of tag reconciliation interface

4.3 User Interface

The screenshot in Figure 3 shows the user interface of the prototype application. On the right it contains the current video, on the left it contains the list of tags assigned to this video and in between them a list of frames for the currently selected tag. The user starts a session by selecting a reconciliation source, shown below the tag list. The reconciliation is performed by sending a request to the corresponding service. When the service returns results for a tag an icon is added to the right of the tag. Selecting such a tag then shows these concepts below the tag. To support the user in identifying the concepts, they are represented by their label and their types. The selection of a tag also brings up the frames that are annotated by this tag, and clicking a frame will play the video starting at this frame. Finally, when the user selects a concept a request is sent to the server to update the database and the tag is highlighted in the interface.

5 User Experiment

We performed a user experiment to test the coverage of the different data sources for the *Waisda?* tags and to what extent users are able to select the most appropriate concept in context of the video. Our assumption is that the users are somewhat familiar with the vocabularies used, and know what kind of concepts they can expect. In this experiment we focus on a quick and simple method to link tags to concepts. Therefore, we expect the user to select one of the concepts suggested by a reconciliation service, and we do not include functionality to first

Table 2. Distribution of reconciled tags over the different sources

	tags	**unique for source**	**concepts**
Freebase	13	3	15
GTAA	12	3	12
Cornetto	22	15	25
total	33		

fix the spelling of tags or manually search within a vocabulary in case no suited concept is found. Furthermore, we expect that a suited concept should be part of the first 10 suggestions provided by the service.

5.1 Setup

For the experiment we used the video presented in Section 2, a newsreel about the visit of the Dutch royal family to the coronation of George VI. In total the video contained 36 tags. We asked four colleagues from our department to use the prototype interface to select the most appropriate concept for each tag. They were instructed to only select a concept when an appropriate one was among the suggestions. They could use the frames related to a tag and play the video add that time point, but were not forced to do so. The users reconciled the tags against three different sources: GTAA, Cornetto and Freebase. A session started with a short explanation of the interface using a different video. In the experiment the participants performed the following steps: select a datasource, start the reconciliation service by clicking the "Go" button, and once results returned start selecting concepts. When all tags were considered the user continued with the next datasource.

5.2 Results

For 33 (out of 36) tags a concept was selected by one or more participants. The three tags for which no concept was selected all contained a spelling error that prevented the reconciliation services to find the appropriate concept: "hye park", "elisabeth" instead of "elizabeth", and "millitairen" instead of "militairen". Table 2 shows the distribution of these tags over the different data sources. Using Cornetto as the reconciliation source most concepts were selected (22), while for GTAA and Freebase less than half of the tags could be linked.

Table 3 shows the number of tags that were reconciled per participant. These numbers are comparable to the total number of tags reconciled by all participants. The low number of tags selected from GTAA by P3 can be explained by the fact that this participant only selected entities from this source and no subject terms, while the others did select the subject terms. Further manual examination showed that for each source most of the tags were reconciled by 3

Table 3. Number of tags reconciled by the four participants. The final column shows inter-rater agreement using Krippendorff's alpha.

participants:	**P1**	**P2**	**P3**	**P4**	α
Freebase	11	10	9	8	0.78
GTAA	9	12	5	10	1.00
Cornetto	21	20	20	18	0.92
total	32	32	27	29	0.91

or 4 users. For GTAA and Cornetto only one tag was reconciled by only one user, and for Freebase 2 tags. This already gives some intuitive indication that the participants, to a large extent, agreed upon which tags to reconcile.

To formally compute the agreement between our participants, we need a suitable metric. Because we have more than four raters, and missing data for some of the tags, we use Krippendorff's alpha (α) as a reliability coefficient [5]. Note that $\alpha = 1.0$ indicates perfect agreement (e.g. in the GTAA cases) while $\alpha = 0.0$ indicates a level of agreement that can be completely attributed by chance. The $\alpha = 0.78$ for Freebase concepts can be interpreted as moderately high agreement, but is quite low compared to the other data sources. This can be partly explained by the fact that we treated both truly distinct concepts and closely related concepts as disagreement in the computation of α. All other αs are above 0.9, which is generally interpreted as a high level of agreement. We conclude that for this small experiment, with a coverage of 33 out of 36 tags and a total reliability coefficient of 0.91, users can effectively reconcile tags using the selected services.

To explain the disagreements among the users we manually assessed the selected concepts. For Freebase the participants selected different concepts for three tags. However, on close examination we considered the selected concepts valid alternatives. For example, two users selected for the tag "abbey" the specific location "Westminster Abbey", while another selected the general concept for "abbeys". For Cornetto the differences between the selected concepts were very subtle. For example, the tag "hek" is in Dutch used for "gate" as well as "fence". Both these interpretations were also applicable to the video. Although the participants agreement on all selected concepts from GTAA we found one error. For the tag "bernhard" (the former prince of the Netherlands) three of the four users selected a Dutch interviewer with the name "Prins, Bernhard" ("prins" is Dutch for "prince"). The participants overlooked the type shown below this name indicating that this was an interviewer.

We also analysed how how many of the tags were only found in a single source, shown in Table 2 in the column labeled *unique*. This shows that most tags that were reconciled against Freebase and GTAA could also be found in another source. More specifically, the general subject terms, for example "park", were also selected from Cornetto. Most entities, such as persons and locations were,

Table 4. Ranks of the selected tags

rank	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
#concepts	32	9	2	6	0	1	2	1	2

however, only found in GTAA or Freebase. All entities selected from GTAA were also found in Freebase. The majority of the tags selected from Cornetto were only found in this source. However, for some tags there were subject terms selected from GTAA that were not found in Cornetto. All these tags were plural forms of regular words missed by the reconciliation service of Cornetto.

Finally, Table 4 shows for the selected concepts at which position in the list of candidate concepts they were ranked. We excluded the falsely selected concept mentioned above. More than half of the selected concepts were ranked first (32), while 23 concepts were ranked second or lower. Several (6) selected concepts were even ranked sixth or lower.

6 Lessons Learned

Although the experiment involved only one video and 36 tags, we can make several observations about the data sources, services and user interface.

- Cornetto is a promising source for subject terms. For most of the subject tags in the experiment the participants could select a concept from Cornetto. Only tags with irregular plural forms were not found. More flexibly string matching techniques, or a better stemming algorithm for Dutch, could help to also find concepts in these cases. However, selecting the most suited concept from Cornetto in the user interface can be time consuming, as the differences between interpretations can be very subtle. In contrast, for GTAA there is usually only one concept used for all interpretations.
- For all the named entities the participants could select a concept from Freebase as well as GTAA. However, the additional information in GTAA is sparser than in Freebase, making it more difficult to identify the right concept. The coverage of Freebase worked well for the video used in the experiment, as the persons and locations are well known. For specific Dutch content Freebase might, however, be less suited.
- Full coverage was prevented by spelling errors and variations in the tags. However, in some cases the video fragment also contained the correctly spelled tag.
- When users know what to expect they adapt their behavior. For example, they quickly noticed that Freebase was most suited for persons and locations, making them only briefly scan the list for subject terms.
- For most tags the context in the video was already clear from the video description or the other tags. However, the video and the frames corresponding to the tags were used in several occasions, and the participants remarked that they were sometimes essential to determine the correct interpretation.

Based on these observations we derive three recommendations for configuring a reconciliation interface for video annotations. First, we recommend to reconcile against multiple data sources at the same time, as the best coverage is acquired by a combination of sources. To effectively use the combined results we recommend that duplicates are merged into a single suggestion. This can, for example, be done using alignments between the concepts in different sources [6]. In addition, the results are best presented in different categories, for example persons, locations and subjects. This allows the user to quickly ignore suggestions from a wrong type. Second, to deal with spelling variations and errors we recommend a preprocessing step that merges similar tags. Third, to simplify the selection of the most suited concept the results from sources that provide high precision should be presented first. The results found in sources that provide high recall should be presented as an alternative. The same approach can be used for a single datasource to distinguish the results found by high precision or high recall string matching techniques.

7 Discussion

We showed that it is feasible to semi-automatically reconcile the user tags collected in *Waisda?* against open data available on the Web. We acknowledge that the small scale of the user experiment prevents us from making more general conclusions. Therefore, we are planning a large scale experiment in the context of a second pilot with *Waisda?*. Based on the lessons learned in this research we will improve our prototype and make it suitable for large scale use. We hope that the integration with *Waisda?* attracts users to reconcile their own annotations or those made by others. We believe, that the data collected in such a large scale experiment can be valuable for the Netherlands Institute for Sound and Vision to improve access to their collection. In addition, such data can be valuable for the research community as a golden standard for concept suggestion algorithms and automatic concept detection.

Acknowledgments. This research was partially supported by the PrestoPRIME project, funded by the European Commission under ICT FP7 (Seventh Framework Programme, Contract No. 231161).

References

1. Angeletou, S., Sabou, M., Motta, E.: Semantically enriching folksonomies with flor. In: Proc of the 5th ESWC. Workshop: Collective Intelligence and the Semantic Web (2008)
2. Bizer, C., Heath, T., Berners-Lee, T.: Linked data - the story so far. Int. J. Semantic Web Inf. Syst. 5(3), 1–22 (2009)
3. Choudhury, S., Breslin, J.G., Passant, A.: Enrichment and Ranking of the YouTube Tag Space and Integration with the Linked Data Cloud. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 747–762. Springer, Heidelberg (2009)

4. Gligorov, R., Hildebrand, M., Van Ossenbruggen, J., Schreiber, G., Aroyo, L.: On the role of user-generated metadata in audio visual collections. In: Proceedings of the International Conference on Knowledge Capture (K-CAP), pp. 145–151. ACM Press (June 2011)
5. Hayes, A.F., Krippendorff, K.: Answering the call for a standard reliability measure for coding data. Communication Methods and Measures 1(1), 77–89 (2007)
6. Hildebrand, M., van Ossenbruggen, J.R., Hardman, L., Jacobs, G.: Supporting subject matter annotation using heterogeneous thesauri, a user study in web data reuse. International Journal of Human-Computer Studies 67(10), 888–903 (2009)
7. Huurnink, B., Hollink, L., van den Heuvel, W., de Rijke, M.: Search behavior of media professionals at an audiovisual archive: A transaction log analysis. Journal of the American Society for Information Science and Technology 61(6), 1180–1197 (2010)
8. Maali, F., Cyganiak, R., Peristeras, V.: Re-using cool uris: Entity reconciliation against lod hubs. In: Proceedings of the 4th Linked Data on the Web (LDOW) Workshop at the World Wide Web Conference, WWW (2011)
9. Oomen, J., Belice Baltussen, L., Limonard, S., van Ees, A., Brinkerink, M., Aroyo, L., Vervaart, J., Asaf, K., Gligorov, R.: Emerging practices in the cultural heritage domain - social tagging of audiovisual heritage. In: Proceedings of the WebSci 2010: Extending the Frontiers of Society On-Line, Raleigh, US (April 2010)
10. Overell, S.E., Sigurbjörnsson, B., van Zwol, R.: Classifying tags using open content resources. In: Baeza-Yates, R.A., Boldi, P., Ribeiro-Neto, B.A., Cambazoglu, B.B. (eds.) WSDM, pp. 64–73. ACM (2009)
11. Prud'hommeaux, E., Seaborne, A.: SPARQL Query Language for RDF. W3C Recommendation (2008)
12. Tummarello, G., Delbru, R., Oren, E.: Sindice.com: Weaving the Open Linked Data. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 552–565. Springer, Heidelberg (2007)
13. van Zwol, R., Garcia, L., Ramirez, G., Sigurbjornsson, B., Labad, M.: Video tag game. In: Proceedings of the 17th International World Wide Web Conference (WWW 2008), Beijing, China (April 2008)
14. von Ahn, L., Dabbish, L.: Labeling images with a computer game. In: CHI 2004: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 319–326. ACM Press, New York (2004)
15. Vossen, P., Maks, I., Segers, R., van der Vliet, H.: Integrating lexical units, synsets and ontology in the Cornetto database. In (ELRA), E.L.R.A. (ed.) Proceedings of the Sixth International Language Resources and Evaluation, LREC 2008 (2008)

Challenges in Storing Multimedia Data for the Future - An Overview

Britta Meixner[1], Michael Ettengruber[2], and Harald Kosch[1]

[1] University of Passau,
94032 Passau, Germany
meixner@fim.uni-passau.de, harald.kosch@uni-passau.de
[2] Business Development Manager - HP Storage,
94315 Straubing, Germany
ettengruber@t-online.de

Abstract. Preserving access to multimedia data over time may prove to be the most challenging task in all things concerning multimedia. Preserving access to data from previous technical generations has always been a rather difficult endeavor, but multimedia data with an almost endless succession of encoding and compression algorithms sets the stakes even higher, especially when not only considering migrating the data from one generation earlier to a current technology but from decades ago. The time to start thinking and developing techniques and methodologies to keep data accessible over time is right now because the first challenges become visible on the horizon: How to archive the ever growing (and growing exponentially so) amounts of data without major manual intervention as soon as a storage media runs out of free space. Is there such a thing as "endless storage capacity"? Would an "endless storage capacity" really help? Or do we need totally new ways of thinking in regard to archiving digital data for the future?

Keywords: Multimedia storage, Encoding, Software, Hardware, Privacy.

1 Introduction

It is difficult to estimate how big the amount of data that have to be stored will be in ten, twenty or fifty years. Based on current figures of growth of photos on Flickr (www.flickr.com), Facebook (www.facebook.com) ([13], [14], [16] [34]) and on Internet videos [39] we made a conservative estimate of the amount of image and video data in terabyte in the next fifty years. There could be about 12 billion terabyte of images in 2035 and about 580 billion terabyte in fifty years. The amount of video data appears to be a little smaller, but reaches about 8 billion terabyte in 2035 and 410 billion terabyte in fifty years. This would mean, that an amount of about 1000 billion terabyte of image and video data have to be stored and maintained in 2060 only in the area of photo- and video platforms and social web.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 705–715, 2012.

With regards to a growing number of smartphones and cheap end user video cameras, the numbers might even be higher. Future technologies might also bring a high amount of data with them. Future technologies will most likely create even more data than is apparent today. Interestingly, one of the most underestimated fields of imaging is health care[28]. MRIs and similar diagnostic tools create imaging data whose amount rises exponentially with the improvement of resolution (see [5], [6]). But not only traditional ways of imaging add to the challenge, the current trend to incorporate the third dimension in consumer imaging multiplies the amount of data required as well ([18], [31]).

This leads to the following questions: How can we deal with the growing amount of data? How can we assure that the multimedia data produced from the past until now will still be available and accessible in fifty years?

2 Related Work

A huge amount of scientific work can be found on various aspects of storing, organizing and searching data in the future. One major aspect of data storage is on cloud computing. Agrawal et al. stated in [1] that a "single perfect data management solution for the cloud is yet to be designed". They also determine that "ensuring the security and privacy of the data outsourced to the cloud is an important problem [...] for data management systems in the cloud". Bein et al. describe two algorithms for the calculation of a minimum number of servers and a minimum total cost given a sequence of online storage requests for a cloud in [4]. They consider requests for larger amounts of data than one server has storage space.

Domain specific approaches can be found in different areas of research, like space agencies of the database sector. Albani describes a project called CASPAR which deals with the "preservation of knowledge associated with the data." His work describes the usage of the system for sensor-data from the ESA [2]. The NASA examines how its planetary data can be stored and preserved in the cloud [27]. Rahman et al. migrate databases from a relational to a dimensional model and calculate information which is embedded in the code to preserve databases for the future. Thereby the original database is affected, but the result is independent of DBMS details and application logic [36].

Different concepts and systems designed to preserve each type of data are developed. One concept is described by Becker et al. in [3]. They describe "a systematic approach for evaluating potential alternatives for perservation actions and building thoroughly defined, accountable preservation plans for keeping digital content alive over time." Thereby they do not investigate a special kind of data, but research the whole process when data have to be stored for future usage. An open, fully distributed system called DISTARNET is presented by Subotic et al. in [42]. It is designed to "reliably execute pre-defined workflows for long-term preservation." It provides "dynamic replication, automated consistency checking and recovery of the archived digital objects" in an autonomous way.

Work focusing on media or 3D data can be found varying from standards to implemented systems. Plagemann and Goebel state in [35] that concepts emerging from projects with network centric view can be used to develop content centric systems like a peer-to-peer video on demand streaming system. They start with future Internet concepts and build a multimedia system based on these. Holmqvist et al. tested "virtualization as a strategy for maintaining future access to multimedia content" in [17]. They found out, that the current state of the art of virtualization and emulation cannot be recommended for ensuring future access to multimedia data and applications. Doyle et al. present a framework for the preservation of 3D data which consists of an emulation and a metadata part to ensure authentical and usable digital objects [11]. Harada et al. describe a standardized packaging format for digital media files called the MPEG-A Professional Archival Application Format (PA-AF) [15]. It implements the information package which is described by the Open Archival Information System (OAIS) reference model [20]. "Archiving is accomplished through a hierarchical file structure and rich contextual information, while preservation is realized by enabling portability in the structure and file attributes." (Vetro in [15])

This small overview shows efforts in storing and managing data for the future either on servers with large storage capacities or in the cloud. This work considers problems and challenges dealing with future hardware, software and security and privacy concerns.

3 Hardware

Let us start with the basic component for storing digital data. There needs to be some type of hardware that is capable of storing digital data for an unforeseeable amount of time. This is one of the major obstacles in archiving and preserving digital content - finding the right type of hardware and storage media. "Classic" or "regular" storage media can be classified into two major groups, magnetic media and optical media.

3.1 Magnetic Media

While magnetical media may be readily available, their most prominent member is the regular and omnipresent hard disc drive. In terms of longevity and for archival purposes it is quickly becoming apparent that regular hard disk drives may not be the means of choice for preserving digital content. Not only do they suffer from mechanical failure far too often, its basic design has not changed in almost fifty years from a mechanical perspective, but the interface technology with which the data is transported from the drive to a computer system underwent dramatic changes on a regular basis ([8], [10]). Deng states in [10] that the hard disc drive will be preferably used in the near future because of its cost efficiency (compared to solid-state drives (SSDs)), but advancements like a dual actuator [7] or multiple spindles [41] can make it more efficient in performance and energy consumption. Even if the basic protocol for moving data to and from the drive has been kept at least compatible, the mechanical and electrical interface has not. From the eighties where 8-bit narrow-SCSI was state of the art [43] over to Fibre-Channel in the 90's and 00's [12], [25] to the current implementations of SAS (Serial Atached SCSI) [25] and SATA (Serial ATA) [25] the electrical and mechanical interfaces of hard drives have changed and are unfortunately incompatible. External data storage technologies like eSata, flash drives or storage area networks (SAN) which are mainly built on the technologies described above suffer from the same disadvantages as their fixed counterparts.

One example of how fast the available technology can (and will in the future) change was the transition from parallel- to serial-SCSI: From the 1980s when parallel-SCSI (pSCSI) was developed [38] until the year 2005 it was the predominant interconnect technology for hard drives in server systems. 2004 the transition to serial-SCSI (SAS) started and was finished in 2006 for most hardware vendors [40]. Now in 2011 it is beginning to become a major problem getting new parallel-SCSI controllers, cables and spare parts since the official support from the original vendor ends usually five years after the product has become obsolete and been removed off the price list. Though environments tend to "live" longer than predicted by the original vendor, they will not live "endlessly" and factual replacement problems will lead to a forced transition to a new technology. Moving data from one generation to the directly subsequent generation of hardware technology is more likely to be feasible than when one or more generations of hardware have been developed in between. But it is still a tedious and manual job - moving data from older to newer hard drives. It needs manual intervention and - most importantly - very precise control from the person doing the actual copying/moving/migration of data. So considering hard drives or fixed media technology for long term archival purposes is most likely a technological cul de sac and probably not the best way.

3.2 Optical Media

Removable optical media have long been a favorite in the domain of long term archiving. But recent developments have shown that even removable media do have some severe drawbacks when it comes to long term availability of data. Removable media are one of the very few technologies that can actually withstand the required time periods of up to hundred years without elaborate climate control where the media are stored [32]. Although they still retain and deliver the information that has been written on them a long time ago, their reading devices e.g. optical CD-ROM, DVD or UDO drives suffer from the same vagaries of interface technology and economic obsolescence like their fixed media counterparts. When an interface technology goes obsolete and is replaced by the more modern, faster, cheaper to manufacture and thus more interesting technology, not all data on previous technologies are transported to the new interface technology. Optical media have been around for more than 30 years [29] and it is one of the very few technologies where the initial media format called Compact Disk is still readable by all common reading devices whether they have been technologically upgraded to read the more modern formats like DVD oder BluRay or not: The Compact Disc is currently the only technology that is even remotely capable of withstanding the required time spans for archival purposes and a working reading device may exist even 50 years from now. Although without a joint effort from a majority of manufacturers economic necessities may well end this predominance in the market over night when it is no longer profitable implementing the capability of reading the original CD format into the current and future models of optical drives.

4 Software and Encoding

The encoding algorithms used to store digital multimedia content plays a similarly important role in keeping data accessible over long periods of time. Encoding and compression technologies and algorithms for multimedia content tend to be quite elaborate and sophisticated as well as specialized for the specific media format. There are many codecs out there for numerous media playing applications which all need to be maintained, updated and patched as well as improved and migrated to new platforms, operating systems and applications. Creating and playing multimedia content on a wide dissemination is a relatively new capability of a computer since it requires computing power not previously available to everyone. The field of multimedia applications is as of 2011 still undergoing major changes and has not yet established its stable market players and reaching a level of predictability that comes with a set of established market players.

One of the major open source video players of our time VLC player lists a vast amount of currently supported video formats on its web site [44], but for example lists the previously widely available Indeo Video 5 [19] as not supported. So digital content that has been encoded using Indeo Video 5 is no longer being played by the current software version of VLC player - nor is it supported by the more recent operating systems making these videos inaccessible when not supported by a more specialized video playing application (of which there still are a few available but for how long?). The threat of multimedia data becoming inaccessible although physically present is not only real, but needs to be taken into consideration when thinking about long term archival and storage of multimedia data. When considering technologies and methodologies for long-term archival of multimedia data, there needs to be a common approach from software and hardware developers and vendors.

The software side needs to come up with a set of encoding and compression algorithms which may not necessarily be the best of their breed but more a kind of smallest common denominator type of technology. Although this may well result in much larger amounts of data or less functionalities in multimedia content, the biggest obstacle is getting the decoding schemes implemented in unforeseeable future generations of software products in a standardized fashion. This is only likely to happen when product developers and software vendors have an own vested interest in implementing this technology and a market exists that is actively requiring this feature/codec/algorithm being implemented. Approaches to provide such standardized video delivery were made by the Moving Pictures Expert Group which developed the video formats MPEG-1 [21], MPEG-2 [22] and MPEG-4 [23]. It may well be required for digital content to be converted into an "archival data format" after its active lifecycle where a certain amount of information loss may be acceptable (compression artifacts, loss of interactivity) in order to ascertain accessibility of the majority of content for the unforeseeable future.

In many cases the decoding algorithms tend to be moved from a software only solution into a at least hardware-assisted if not hardware-based decoder for multimedia content [33]. Although software-only solutions tend to be implemented quicker and easier since the development of specialized hardware components like application-specific integrated circuits (ASICs) is not necessary, long term support is in many cases determined by hardware support. Standardized hardware components are a reliable and - after initial development - cheap way of ensuring long term support as well as wide spread usage since the actual platform the content is run on does not matter any more. This is extremely important for long term archival of data since the future of information technology is not only unclear, but its development is so quick that todays predictions may probably be tomorrow's old news.

But one thing will have an ever increasing impact on our daily lives and how we use computer technology: The mobile phone. In less than a decade every person will probably only carry what we call today a smart phone where all his information requirements as well as all computing power will be residing and providing its user with everything he may need for working or recreational purposes. Of course, any successful archiving strategy must take these changing behaviors in using computer technology into account and be as "always on" as the rest of the digital universe.

When the major players in the continuum of digital archive could standardize on a set of features which needs to be implemented we could well be on our way to a world where the dangers of losing digital content might be a thing of the past compared to now, where the loss of digital content over time is certain.

5 Organization and Finding of Contents

More than one person has advocated a kind of "universal electronic world archive" in the past [26]. While this may sound like a good idea - we think it is not! The archival of data in the future, especially when thinking of multimedia content, must be kind of content specific. Not all data can be treated equally. Data may have different archival requirements e.g. bandwidth for retrieving, searching and querying capabilities and many more. These must be taken into consideration when thinking of long term archival of multimedia content. The archival of multimedia data is space challenging. Multimedia data tends to grow much faster than non-media related data. This is in part due to the technical specifications of devices creating multimedia content - digital cameras ever increasing amount of pixels per image grows the amount of storage space required - as well as the ever more widespread use of multimedia technologies in everyday life e.g. digital TV or mobile media. Strategies for storing and archiving multimedia data need to take all this into consideration and provide a common set of features as well as searching capabilities on a unprecedented scale:

When a user archives all his digital images, images he may have created over a time span of many years, he may upload the images "as they are". This means that he may have named the directories where the images are stored with references to the content or he may even name the images individually. Regardless of which way, it is a time-consuming and tedious job and the results are rather unpredictable. The user may never find the right image again when looking for one specific image. Similarly this may happen to video or audio content. Thus a long-term archiving solution is not only about creating the environment for storing and retrieving information, it is even more about finding the right information in the least possible amount of time and without annotations added by the user. Highly sophisticated and intelligent search algorithms must be developed which allow looking into multimedia content and indexing and/or referencing multimedia content on a much higher and wider scale than anything available today or archival will prove to be a black hole for multimedia content. For example must a multimedia content searching algorithm be able to identify human faces when "shown" a face - even in case the image of the "shown" face is of different age, or an archival system for images might not prove useful over the proposed time span of more than 50 years.

Similarly for any other multimedia content, the archival system must be capable of recognizing patterns with which it can interact with the user. Being it graphic descriptions of content ("movie where actor xx wears a funny hat") or descriptions of "logical" content ("movie where actor xx plays Dr. Yes"). But the information must solely come from the content itself, not added manually during the archiving process or otherwise the archival system will not be "self-sufficient" and rely on (possibly incorrect) information entered manually for later retrieval of multimedia data. Later correction of previously manually entered data may not be possible anymore - e.g. when the person who entered the data manually is deceased and was the only possible source for the correct information. Different annotation standards for various areas of digital data already exist. Multimedia data can be described via MPEG-7 [24]. Digital ressources can be described via Dublin Core [9] or the Resource Description Framework (RDF) [45]. The open standard Digital Imaging and Communications in Medicine (DICOM) [30] can be used to exchange medical information.

Being able to ascertain the correctness of information in a multimedia archival system is especially important since multimedia content in form of film or "moving images" is regarded as a highly credible source of information by most people. Therefore, it is an essential requirement of any archival system to make sure only valid and correct information is entered into its databases - preferably autonomically and automatically.

6 Security and Privacy Concerns

Any archival system must not only be secured from unauthorized access but additionally from tampering or changing the stored information - even from the potential "owner" of the content. Thus a multimedia archival system must not allow altering of any content that has ever been entered into the system. While it may well provide for means of version control and updating, the content itself may never be altered, only stored again with a new version.

Whether content is allowed to be deleted or not needs to be discussed. Both positions seem to have valid arguments - the "non-deleters" will have to face a theoretically and practically ever increasing archive with the subsequent technical challenges while the "deleters" may postpone the technical challenges but may face incalculable loss of content by allowing deletion of content by its individual contributors.

Regardless of any deletion strategy, the privacy concerns may be even more important. Who decides which content may be accessed by whom? In archiving applications for individual users this may seem self-explanatory but still there are issues that need to be addressed: Who has access to the data after the owner is deceased? Is there a "right of succession" for archived multimedia data? Will it be possible for individual personal data to be deleted although deletion of data may not seem desirable? Does entering personal multimedia data into an archival system alter the individual rights of the owner of the data (relinquishing ownership)?

An archival system for multimedia data may have potentially a huge influence how we may in the future perceive our past. Up until now the multimedia information about our past has been provided by analog film, its content was and is - in its original form - rather hard to forge. So information derived from motion picture oder still picture has traditionally been credited with much more credibility and plausibility than a personal witness or verbal recollection. "Images do not lie"! This is not any longer true for digital data. Digital multimedia content can be forged, altered, changed - professional movie studios use Computer-generated imagery (CGI) to create complete feature films [37] - and still possesses a lot of credibility. This credibility makes altering or forging of motion or still images quite interesting for rather unsavoury elements of human society. Any digital multimedia archive needs to be protected to the full technical extent from forgeries and alterations of its archived content. Its usability and public perception as a reliable source of information may depend on it but even more the public itself depends on the reliability of the information stored in a multimedia archive.

7 Conclusion

An archival system for multimedia content does not only face the challenges of traditional or "regular" archives but faces some pretty interesting additional challenges which are unique to the storage and retrieval of digital multimedia content. Any widely usable digital multimedia archive must have addressed all issues, not only a part or subset of it. In order to attain a level of usability which might make this a viable way of storing digital multimedia data there is still a lot of work to do. Not only need some universally recognized and usable algorithms for archiving and retrieving digital multimedia data be developed, there is still the need for a common hardware platform which might be supported by several manufacturers/vendors on long term basis - sufficient not for years but for decades. The time to start is now - or we might lose a substantial amount of data mankind has created already.

References

1. Agrawal, D., Das, S., El Abbadi, A.: Big data and cloud computing: current state and future opportunities. In: Proceedings of the 14th International Conference on Extending Database Technology, EDBT/ICDT 2011, pp. 530–533. ACM, New York (2011)
2. Albani, S.: The ESA Approach to Long-Term Data Preservation using CASPAR. Website. ERCIM News (visited October 1, 2011), `http://ercim-news.ercim.eu/en80/special/the-esa-approach-to-long-term-data-preservation-using-caspar`
3. Becker, C., Kulovits, H., Guttenbrunner, M., Strodl, S., Rauber, A., Hofman, H.: Systematic planning for digital preservation: evaluating potential strategies and building preservation plans. Int. J. on Digital Libraries 10(4), 133–157 (2009)
4. Bein, D., Bein, W., Phoha, S.: Efficient Data Centers, Cloud Computing in the Future of Distributed Computing. In: Third International Conference on Information Technology: New Generations, pp. 70–75. IEEE Computer Society, Los Alamitos (2010)
5. Bourne, R.: Fundamentals of Digital Imaging in Medicine. Springer, London (2009) ISBN: 9781848820869
6. Bryan, R.N.: Introduction to the Science of Medical Imaging. Cambridge University Press (2009) ISBN: 9780521747622
7. Chandy, J.A.: Dual actuator logging disk architecture and modeling. Journal of Systems Architecture 53(12), 913–926 (2007)
8. Coughlin, T.M.: Digital storage in consumer electronics: the essential guide. Embedded technology series. Elsevier/Newnes, Burlington (2008) ISBN: 9780750684651
9. Dublin Core Metadata Initiative: Dublin Core Metadata Initiative - Making it easier to find information. Website, `http://dublincore.org/` (visited October 1, 2011)
10. Deng, Y.: What is the future of disk drives, death or rebirth? ACM Comput. Surv. 43(3), 23:1–23:27 (2011)
11. Doyle, J., Viktor, H., Paquet, E.: Long-term digital preservation: preserving authenticity and usability of 3-D data. Int. J. Digit. Libr. 10(1), 33–47 (2009)
12. Fibre Channel Association: Fibre channel: connection to the future. Elsevier Science (1998) ISBN: 9781878707451
13. Flickr Photos Growth Chart. Website (May 5, 2008), `http://www.flickr.com/photos/rexguo/2467112209/sizes/o/` (visited July 30, 2011)
14. A picture a day: Flickrs storage growth. Website (October 16, 2006), `http://blog.forret.com/2006/10/a-picture-a-day-flickrs-storage-growth/` (visited July 30, 2011)
15. Harada, N., Kamamoto, Y., Moriya, T., Sabirin, H., Sabirin, H., Kim, M.: Archive and Preservation of Media Content Using MPEG-A. IEEE Multimedia 17, 94–99 (2010)

16. Hird, J.: 20+ mind-blowing social media statistics revisited. Website (January 29, 2010), http://econsultancy.com/uk/blog/5324-20+-mind-blowing-social-media-statistics-revisited(visited July 30, 2011)
17. Holmqvist, K., Halbach, T., Kristoffersen, T.: Virtualization as a Strategy for Maintaining Future Access to Multimedia Content. In: First International Conference on Advances in Multimedia, MMEDIA 2009, pp. 29–32 (2009)
18. Hopping, R.: 3D mobile phones: All you need to know, Website, (February 7, 2011), http://www.knowyourmobile.com/features/761306/3d_mobile_phones_all_you_need_to_know.html (visited October 1, 2011)
19. Intel Corporation: Intel Indeo®Video 5. Website (1997), http://www.siggraph.org/education/materials/HyperGraph/video/codecs/indeo_v5/overview.htm (visited July 30, 2011)
20. ISO: ISO 14721:2003: Space data and information transfer systems - Open archival information system - Reference model (2003)
21. ISO/IEC, Ed. MPEG: MPEG-1. Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s. Website (June 1996), http://mpeg.chiariglione.org/standards/mpeg-1/mpeg-1.htm (visited October 1,2011)
22. ISO/IEC, Ed. MPEG: MPEG-2. Generic coding of moving pictures and associated audio information. Website (October 2000), http://mpeg.chiariglione.org/standards/mpeg-2/mpeg-2.htm (visited October 1, 2011)
23. Koenen R.: ISO/IEC, Ed. (MPEG): Overview of the MPEG-4 Standard. Website (March 2002), http://mpeg.chiariglione.org/standards/mpeg-1/mpeg-1.htm (visited October 1, 2011)
24. Martínez, J.M.: ISO/IEC, Ed. MPEG-7 Overview (version 10). Website (October 2004), http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm (visited October 1, 2011)
25. Jacob, B., Ng, S., Wang, D.: Memory systems: cache, DRAM, disk. Elsevier/Morgan Kaufmann, Burlington (2007) ISBN: 9780123797513
26. Kahle, B., Prelinger, R., Jackson, M.E.: Public Access to Digital Material. D-Lib Magazine, 7(10), Corporation for National Research Initiatives, Website (October 2001), http://www.dlib.org/dlib/october01/kahle/10kahle.html (visited July 30, 2011)
27. Mattmann, C., Crichton, D.J., Hart, A.F., Kelly, S.C., Hughes, J.S.: Experiments with Storage and Preservation of NASA's Planetary Data via the Cloud. IT Professional 12(5), 28–35 (2010)
28. McNeill, R.: At PhysOrg.com: Medical imaging's growth leaves standards in the dust, critics say, Website (September 23, 2009) http://www.physorg.com/news172862921.html (visited October 1, 2011)
29. Meinders, E.R., Mijritskii, A.V., Pieterson, L.V., Wuttig, M.: Optical data storage: phase-change media and recording. Philips Research. Springer, Heidelberg (2006) ISBN: 9781402042164
30. NEMA: DICOM - Digital Imaging and Communications in Medicine. Website, http://medical.nema.org/ (visited October 1, 2011)
31. Nintendo of America Inc.: Nintendo 3DS - Real 3D Graphics. No Glasses Needed. Website, http://www.nintendo.com/3ds/hardware (visited October 1, 2011)
32. Null, L., Lobur, J.: The essentials of computer organization and architecture. Jones and Bartlett Learning, Sudbury (2006) ISBN: 9780763737696

33. NVIDIA Corporation: Adobe Flash 10.1. Website, http://www.nvidia.com/object/adobe_flashplayer_plus_nvidia.html (visited July 30, 2011)
34. Online Marketing Trends. Website (March 1, 2011), http://www.onlinemarketing-trends.com/2011/03/facebook-photo-statistics-and-insights.html (visited July 30, 2011)
35. Plagemann, T., Goebel, V.: The future internet and its prospects for distributed multimedia systems and applications. In: Proceedings of the 17th ACM International Conference on Multimedia, MM 2009, pp. 919–920. ACM, New York (2009)
36. Rahman, A.U., David, G., Ribeiro, C.: Model Migration Approach for Database Preservation. In: Chowdhury, G., Koo, C., Hunter, J. (eds.) ICADL 2010. LNCS, vol. 6102, pp. 81–90. Springer, Heidelberg (2010)
37. Rodriguez, E.: Computer Graphic Artist. Global Media (2007) ISBN: 9788189940423
38. Schulz, G.: Resilient storage networking: designing flexible scalable data infrastructures. Digital Press storage technology series. Elsevier Digital Press, Burlington (2004) ISBN: 9781555583118
39. Scott, J.: The Rise Of Online Video Will Break The Internet. Website (June 2011), http://www.reelseo.com/rise-online-video-break-internet/ (visited July 30, 2011)
40. SCSI Trade Association: Serial Attached SCSI Master Roadmap. Website (June 1, 2011), http://www.scsita.org/sas_library/2011/06/serial-attached-scsi-master-roadmap.html (visited July 31, 2011)
41. Stiles, E.M., Calderwood, R.C.: Hard disk drive with multiple spindles. United States Patent Application 20060044663, Website (March 2006), http://www.freepatentsonline.com/y2006/0044663.html (visited July 31, 2011)
42. Subotic, I., Schuldt, H., Rosenthaler, L.: The DISTARNET Approach to Reliable Autonomic Long-Term Digital Preservation. In: Yu, J.X., Kim, M.H., Unland, R. (eds.) DASFAA 2011, Part II. LNCS, vol. 6588, pp. 93–103. Springer, Heidelberg (2011)
43. Thompson, R.B., Thompson, B.F.: PC hardware in a nutshell. O'Reilly, Sebastopol (2003) ISBN: 9780596005139
44. VideoLAN Organization: VLC playback Features. Website, http://www.videolan.org/vlc/features.php?cat=video, (visited July 30, 2011)
45. W3C/RDF working Group: Resource Description Framework (RDF) Website (2004), http://www.w3.org/RDF/ (visited October 1, 2011)

Optimizing Multimedia Retrieval Using Multimodal Fusion and Relevance Feedback Techniques

Apostolos Axenopoulos, Stavroula Manolopoulou, and Petros Daras

Centre for Research and Technology Hellas,
Informatics and Telematics Institute,
1st Km Thermi - Panorama, 57001 Thessaloniki, Greece
{axenop,manolop,daras}@iti.gr

Abstract. This paper introduces a novel approach for search and retrieval of multimedia content. The proposed framework retrieves multiple media types simultaneously, namely 3D objects, 2D images and audio files, by utilizing an appropriately modified manifold learning algorithm. The latter, which is based on Laplacian Eigenmaps, is able to map the mono-modal low-level descriptors of the different modalities into a new low-dimensional multimodal feature space. In order to accelerate search and retrieval and make the framework suitable even for large-scale applications, a new multimedia indexing scheme is adopted. The retrieval accuracy of the proposed method is further improved through relevance feedback, which enables users to refine their queries by marking the retrieved results as relevant or non-relevant. Experiments performed on a multimodal dataset demonstrate the effectiveness and efficiency of our approach. Finally, the proposed framework can be easily extended to involve as many heterogeneous modalities as possible.

Keywords: Multimodal Search, Multimedia Indexing, Relevance Feedback.

1 Introduction

The ever-increasing amount of multimedia content, which is available in the Internet, intensifies the need for effective search through the various online media databases. Towards this direction, a lot of research has been conducted on developing methods for content-based multimedia retrieval. Moving beyond traditional text-based retrieval approaches, content-based media search is based on the extraction of low-level features (e.g. color, texture, shape, etc.) automatically from the content. While the problem of retrieving one single modality at a time, such as 3D objects, images, video or audio has been extensively covered, retrieval of multiple modalities simultaneously has yet to yield significant results.

Cross-media retrieval comprises all multimedia search methods that use a query of one modality to retrieve results of another modality. Moving beyond cross-media retrieval, multimodal retrieval allows users to enter multimodal

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 716–727, 2012.

queries and retrieve multiple types of media simultaneously. Thus, users will be able to search and retrieve content of any type using a single unified retrieval framework and not a specialized system for each separate media type.

Most of the cross-modal retrieval methods [2] are based on a well-known technique called Canonical Correlation Analysis (CCA) [1], which constructs an isomorphic subspace (CCA subspace) in order to learn multi-modal correlations of media objects. In [3], the intra- and inter-media correlations of text, image and audio modalities are investigated in order to produce a Multi-modality Laplacian Eigenmaps Semantic Subspace (MLESS). In [4], a structure called Multimedia Document (MMD) is introduced to define a set of multimedia objects (images, audio and text) that carry the same semantics. After creating a Multimedia Correlation Space (MMCS), a ranking algorithm is applied, which uses a local linear regression model for each data point and it globally aligns all of them through a unified objective function.

Despite the significant progress of content-based multimedia retrieval in terms of retrieval accuracy, even the most accurate methods may fail to return results that fully satisfy the end-users. This is due to the fact that the above methods do not take into account user's subjectivity. In order to enable personalized retrieval, Relevance Feedback (RF) techniques have been extensively used. Some of the most common RF approaches involve query refinement [5], where the initial query is moved so as to get closer to the relevant objects. Techniques based on query expansion [4] usually replace the query point with multiple query points, which are then given as input to an appropriate ranking algorithm. Another category of RF approaches uses different weights on the objects' low-level features, when computing their pairwise dissimilarity measure. Re-weighting [9] enhances the importance of those dimensions of a descriptor vector that help in retrieving the relevant objects and reduces the importance of those dimensions that hinder this process.

In this paper, a unified framework for search and retrieval of multimedia content is proposed. The framework achieves retrieval of multiple media types simultaneously, such as 3D objects, images and sounds, using as query any of the above types or combinations of them. The method is novel in the sense that queries may consist of multiple modalities and the retrieved results can have multiple modalities as well. The method can be applied even to very large multimedia databases, by exploiting an appropriate large-scale indexing scheme. Finally, a relevance feedback method is chosen among several state-of-the-art approaches to improve the retrieval performance, while at the same time it is combined with the above indexing scheme to achieve improved retrieval even in large-scale. The proposed framework can be easily extended in order to address a wider variety of media types and application paradigms.

The rest of the paper is organized as follows: In Section 2, the multimodal descriptor extraction procedure is analyzed, while in Section 3, a description of the large-scale indexing technique, which is used to accelerate multimodal retrieval, is given. In Section 4, an overview of several Relevance Feedback techniques is available. These techniques were tested in terms of improvement the retrieval accuracy as well as in terms of adaptability to the indexing scheme and the results are shown in Section 5. Finally, conclusions are drawn in Section 6.

2 Creating a Multimodal Feature Space

2.1 Basic Concepts and Overview

In multimodal search and retrieval problems, it is much more convenient to enclose multiple media types, which share the same semantics, into a media container, and label the entire container with the semantic concept, instead of labelling each media instance separately. This approach has been already followed in both [4] and [3], where authors introduced new structures to organize data based on their semantic correlations, namely Multimedia Documents (MMDs) and Multimedia Bags, respectively.

Following the same concept, the framework proposed in this paper is based on a multimedia structure called "*Content Object (CO)*". A CO can span from very simple media items (e.g. a single image or an audio file) to highly complex multimedia collections (e.g. a 3D object accompanied with multiple 2D images and audio files). Moreover, a CO may include additional metadata related to the media, such as textual information, classification information, real-world data (location or time-based), etc. When a user refers to a CO, s/he directly refers to all of its constituting parts. In the current work, 3D objects, 2D images and sounds are considered as the constituting modalities of COs. Further extensions of the proposed framework, in order to include other modalities, are planned for future work.

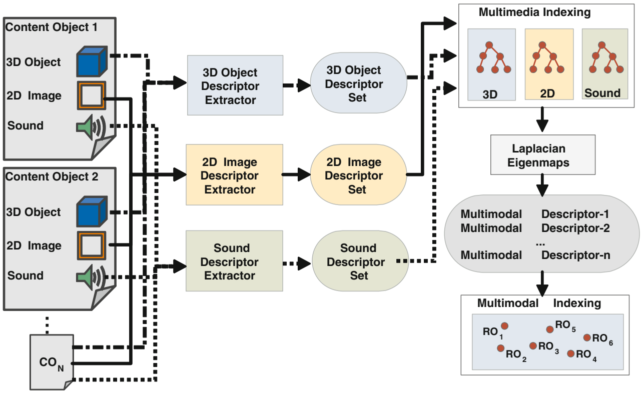


Fig. 1. Multimodal descriptor extraction and indexing

The procedure for the creation of the multimodal feature space is depicted in Figure 1. Given a dataset of Content Objects, low-level descriptors are extracted for each of their constituting modalities. By using the proposed manifold ranking method, which is based on Laplacian Eigenmaps (LE), the low-level descriptors are mapped to a new low-dimensional feature space. In this new space, semantically similar COs, irrespective of their constituting modalities, are described by multimodal descriptor vectors close to each other, in terms of Euclidean distance. The LE-based method requires the computation of an adjacency matrix, whose non-zero elements correspond to pairs of neighboring COs. In order to accelerate the computation of neighbors, a multimedia indexing scheme is applied to each separate modality. After the creation of the new multimodal feature space, content-based search of COs is performed by directly matching their new multimodal descriptors. For faster retrieval, when it comes to large-scale datasets, an appropriately selected indexing scheme is adopted to index the multimodal descriptors.

2.2 Multimodal Feature Space Creation

During the construction of the multimodal feature space, all COs of a dataset, irrespective of their constituting modalities, are represented as l-dimensional points in a new feature space. In this feature space, semantically similar COs lie close to each other with respect to a common distance metric (such as the L-2 distance). The methodology, which will be followed in this paper, is known as manifold learning. This has been already used for non-linear dimensionality reduction in vector spaces, but for one singe modality only. Such an approach is applied for the first time on multimodal data in this paper.

Let a multimedia dataset of N COs and p different modalities. The fist step of the LE-based method is to compute a $N \times N$ adjacency matrix $\mathbf{W}$, where each item W_{ij} has a non-zero value only when COs i and j are neighbors. In the original version of the Laplacian Eigenmaps algorithm, the non-zero elements W_{ij} are exponential functions of the distance between i and j. However, in our case, computing distances between COs is not trivial, since it requires merging descriptors of heterogeneous modalities into one unified distance measure. Therefore, in this paper, the following modification is proposed: when items i, j are neighbors, the item W_{ij} of the adjacency matrix is assigned the value 1 instead of the distance between i and j. Since the items of the adjacency matrix are COs, the neighborhood criterion is determined as follows: two COs, i and j are neighbors if and only if at least one pair of their constituting items of the same modality are neighbors. If the two COs do not have items of common modality they are not considered as neighbors. Neighborhood among single-modality items is determined by ranking these items with respect to their mono-modal distance.

In order to proceed, let us assume, for simplicity, that each CO_i consists of exactly one item per modality. In the general case, it is possible to have only few modalities in CO_i as well as more than one items of the same modality. Let a media item within CO_i of m-th modality ($1 \leq m \leq p$) be represented by the descriptor vector $\mathbf{x}_i^m$. For the m-th modality, a distance measure is defined as $d^m(\mathbf{x}_i^m, \mathbf{x}_j^m)$ to calculate the mono-modal dissimilarity. The k_m-nearest neighbors of $\mathbf{x}_i^m$ are retrieved by ranking all the media items of m-th modality

($\mathbf{x}_j^m$) within the database, with respect to their mono-modal distances d^m. The ranked list of k_m-nearest neighbors of $\mathbf{x}_i^m$ is defined as:

$$\mathbf{NeighList}_{CO_i}^m = \{index_{CO_i}^m(1), index_{CO_i}^m(2), \cdots, index_{CO_i}^m(k_m)\} \tag{1}$$

where $index_{CO_i}^m(1)$ is the index of the CO which corresponds to the media item of m-th modality, ranked as the first nearest neighbor of $\mathbf{x}_i^m$. $index_{CO_i}^m(2)$, $\cdots$,$index_{CO_i}^m(k_m)$ are the indices of the COs corresponding to the $2^{nd}, \cdots, k_m^{th}$ ranked items, respectively. Similarly, p lists of nearest neighbors are extracted, one for each modality. The final k-nearest neighbors of CO_i are computed by taking equal number of first neighbors from each list $\mathbf{NeighList}_{CO_i}^m$, $1 \leq m \leq p$, i.e. k/p neighbors, with $(k/p) < k_m$. In case a CO_j appears in the k/p neighbors of more than one lists $\mathbf{NeighList}_{CO_i}^m$, this CO_j is counted only once. The remaining positions in the k-nearest neighbors list are then filled with the next closest COs.

In the general case that a CO consists of less than p modalities, more nearest neighbors are taken from each modality, in order to keep the number k of the neighboring COs the same. Finally, a $N \times k$ matrix, $\mathbf{NN}_{CO}$, is created, where each row i represents the k-nearest neighbors of CO_i. The $\mathbf{NN}_{CO}$ matrix is taken as input to create the $N \times N$ adjacency matrix $\mathbf{W}$, where:

$$W_{ij} = \begin{cases} 1, & \text{if } CO_j \text{ belongs to } k\text{-neighbors of } CO_i. \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Our LE-based method uses the matrix $\mathbf{W}$ as input to create a multimodal feature space of low dimension, where every CO is represented as a l-dimensional vector, that is the $N \times d$ matrix $\mathbf{Y}$. Except for the creation of the adjacency matrix, the remaining steps of the LE method are the same as in [10], where a detailed description is available. It is stressed once again that the motivation for choosing binary values (1, 0) to construct $\mathbf{W}$ (instead of using actual distances) was to overcome the heterogeneity of descriptors from multiple modalities. Different descriptors require different distance metrics that cannot be put together.

3 An Indexing Scheme for Large-Scale Multimodal Retrieval

It is obvious that for the creation of the $N \times N$ adjacency matrix $\mathbf{W}$, p $N \times N$ distance matrices are required, which store the pair-wise distances among the p modalities of all database's COs. However, when it comes to really large multimedia datasets, both calculation and storage of all-to-all distance matrices becomes prohibitive. Consequently, the distance matrix does not provide an efficient solution in real-life problems. On the other hand, multimedia indexing is a widely used method to speed up the nearest-neighbor search in large databases. Through indexing, there is no need to compute one-to-all distances of the query with all database objects. In the present work, an algorithm for large-scale multimedia indexing, which was introduced in [11], has been adopted to avoid computation of large distance matrices. The main idea of the method is that when two objects are very similar (close to each other in a metric space) their view of the surrounding world is similar as well. Thus, instead of using the distance between two objects, their similarity can be approximated by comparing their ordering of similarity according to some reference points.

Let $\mathbf{S} = \{o_1, o_2, \ldots, o_M\}$ be a set of M media objects and d a distance function between objects of S. Let $\mathbf{RO} \subset \mathbf{S}$ be a set of reference objects chosen from $\mathbf{S}$. An object $o_i \in \mathbf{S}$ can be represented as the ordering $\bar{o}_i$ of the reference objects RO according to their distance d from o_i, as follows: $\bar{o}_i \in O^{RO}_{d,o_i}$, where O^{RO}_{d,o_i} is the ordered list containing all objects of $\mathbf{RO}$, ordered according to their distance d from o_i. The position in O^{RO}_{d,o_i} of a reference object $ro_j \in \mathbf{RO}$ is denoted as $O^{RO}_{d,o_i}(ro_j)$. The distance between two objects in the transformed domain is given by $\bar{d}(\bar{o}_1, \bar{o}_2) = SFD(O^{RO}_{d,o_1}, O^{RO}_{d,o_2})$, where SFD is the Spearman Footrule Distance, which is used as a measure to compare ordered lists:

$$SFD(O^{RO}_{d,o_1}, O^{RO}_{d,o_2}) = \sum_{ro \in RO} \mid O^{RO}_{d,o_1}(ro) - O^{RO}_{d,o_2}(ro) \mid \qquad (3)$$

The distance between the two objects in the transformed domain can be used to perform approximate similarity search, instead of using the classical distance metric d. The approximate distance can be easily computed by representing (indexing) the transformed objects with inverted files, as follows: Entries of the inverted file are the objects of $\mathbf{RO}$. The posting list associated with an entry $ro_j \in \mathbf{RO}$ is a list of pairs $(o_i, O^{RO}_{o_i}(ro_j)), o_i \in S$, that is a list where each object o_i of the dataset $\mathbf{S}$ is associated with the position of the reference object ro_i in $\bar{o}_i$. In other words, each reference object is associated with a list of pairs each referring an object of the dataset and the position of the reference object in the transformed objects representation. A more detailed description of the algorithm is available in [11]. By using the above indexing structure, search within the dataset $\mathbf{S}$ is much faster than using the classical distance metric d to calculate dissimilarity between descriptor vectors.

The multimedia indexing scheme described above is applied a) to the mono-modal descriptors, to avoid computation of large distance matrices during the creation of the multimodal feature space; b) to the multimodal descriptors, to facilitate faster multimodal retrieval in large scale. In the first case, the indexing algorithm is applied for each modality separately, thus, the dataset $\mathbf{S}$ is the set of media items o of the same m-th modality and d is the distance metric $d^m(\mathbf{x}^m_i, \mathbf{x}^m_j)$ that computes the dissimilarity between the mono-modal descriptors $\mathbf{x}^m$ of the m-th modality. Similarly, in the case of the multimodal descriptors, the dataset $\mathbf{S}$ is the set of COs and d is the distance metric that computes the dissimilarity between their corresponding l-dimensional descriptors, which were extracted by using the LE method (Section 2).

4 Improving Retrieval Using Relevance Feedback

Multimodal search and retrieval may not always be accurate enough to bring the most relevant results to the user. This is due to the following reasons: a) the discriminative power of the low-level descriptors of one or more modalities is low; b) the system does not take into account the users' subjectivity, i.e. different users may regard different items as relevant or irrelevant. In order to overcome the above issues, Relevance Feedback has been exploited to ensure delivery of more accurate and personalized results. Since there is already extensive research on RF for content-based multimedia retrieval, several methods were tested not only in terms of improvement of retrieval accuracy but also in terms of applicability to the proposed framework. A brief overview of the most representative ones is given in the sequel.

4.1 Relevance Feedback Methods

Let N_p, N_n the number of COs marked as relevant and non-relevant to a query CO$_q$, respectively, $\mathbf{p}^i = \{p^i(1), \ldots, p^i(l)\}$ be the multimodal descriptor of the i^{th} relevant CO ($i = 1, \ldots, N_p$) and $\mathbf{n}^i = \{n^i(1), \ldots, n^i(l)\}$ is the multimodal descriptor of the i^{th} non-relevant CO ($i = 1, \ldots, N_n$), where l is the dimensionality of the multimodal descriptor vectors, N_p and N_n the number of COs marked as relevant and non-relevant, respectively. Let also $\bar{\mathbf{p}} = \{\bar{p}(1), \ldots, \bar{p}(l)\}$ and $\bar{\mathbf{n}} = \{\bar{n}(1), \ldots, \bar{n}(l)\}$, where $\bar{p}(j)$ and $\bar{n}(j)$ are the mean values of the N_p relevant and N_n non-relevant COs along the j^{th} coordinate ($j \in \{1, \ldots, l\}$). A query refinement RF method, which was introduced in [5], is based on modifying the initial query as follows:

$$\mathbf{q}' = \alpha \cdot \mathbf{q} + \beta \cdot \bar{\mathbf{p}} - \gamma \cdot \bar{\mathbf{n}} \tag{4}$$

where $\mathbf{q}'$ is the descriptor vector of the refined query, $\mathbf{q}$ the initial query descriptor and α, β, γ are appropriately selected constants ($\alpha = 1 - \beta + \gamma$). Instead of using the above equation, the following can be used:

$$\mathbf{q}' = \alpha \cdot \mathbf{q} + \beta \cdot \tilde{\mathbf{p}} - \gamma \cdot \bar{\mathbf{n}} \tag{5}$$

where $\tilde{\mathbf{p}} = \{\tilde{p}(1), \ldots, \tilde{p}(l)\}$. $\tilde{p}(j)$ is computed as follows: let the set $\mathbf{Y}_p(j) = \{p^i(j) \mid |p^i(j) - \bar{p}(j)| \leq 3\bar{\sigma}(j)\}$, for all i, j, where $\bar{\sigma}(j)$ is the standard deviation of the N_p relevant COs with respect to the j^{th} feature. Then, $\tilde{p}(j)$ is given by:

$$\tilde{p}(j) = \frac{1}{|\mathbf{Y}_p(j)|} \sum_{p^i(j) \in \mathbf{Y}_p(j)} p^i(j) \tag{6}$$

In other words, for the computation of the j^{th} feature of the refined query's multimodal descriptor vector we keep only those relevant COs that have almost similar values $p^i(j)$, while we discard those relevant COs that have values $p^i(j)$ far from the average $\bar{p}(j)$. In this case, we avoid taking into account outliers, thus, producing more accurate refined queries ([6]).

Another approach for RF we tested is based on query expansion. Query expansion replaces the initial query with multiple queries, which correspond to the items marked as relevant by the user. The multiple query points are given as input to the system and an appropriate ranking algorithm merges the multiple ranked lists. Such an approach has been recently adopted in several multimedia retrieval frameworks [4], [8].

Re-weighting in RF is applied to the similarity measure. The idea is to analyze the relevant objects in order to understand which features (dimensions) of the descriptor vector are more important than others in determining "what makes an object relevant". The general framework of re-weighting can be described as follows: let $\mathbf{P} = \{\mathbf{p}^1, \mathbf{p}^2, \ldots, \mathbf{p}^{N_p}\}$ the set of relevant objects' multimodal descriptors and $\mathbf{p}^i = \{p^i(1), \ldots, p^i(l)\}$ the descriptor of the i^{th} object. The refined distance measure d_w between the query descriptor vector $\mathbf{q}$ and the descriptor $\mathbf{m}$ of a database CO is given by:

$$d_w(\mathbf{q}, \mathbf{m}) = \frac{1}{l} \sum_{j=1}^{l} w_j \cdot |q(j) - m(j)| \tag{7}$$

where w_j is the weight of the j^{th} feature that is estimated as $w_j \propto 1/\bar{\sigma}_j^2$ and $\bar{\sigma}_j^2$ is the variance of the N_p relevant COs along the j^{th} feature (coordinate).

RF can be further enhanced if the most informative objects of the dataset are presented to the user for feedback. This technique is known as active learning and it has been used for RF in [7],[8]. More specifically, let N be the number of COs of the dataset and $\mathbf{W}$ the $N \times N$ dissimilarity matrix of all COs. After the first RF iteration, two vectors of size N are defined: $\mathbf{B}^+$, where the i^{th} coordinate is assigned the value 1, if CO_i is a positive result, and the value 0 otherwise; $\mathbf{B}^-$, where the i^{th} coordinate is assigned the value -1, if CO_i is a negative result, and the value 0 otherwise. Then the following scores are calculated:

$$\begin{aligned} A_i^+ &= \max(\mathbf{W}(i,:)\mathbf{B}^+) \\ A_i^- &= \min(\mathbf{W}(i,:)\mathbf{B}^-) \end{aligned} \tag{8}$$

During the second RF iteration, the system presents to the user for feedback a set of unlabeled objects, which are determined as follows: a) those COs with the largest $A_i^+ + A_i^-$, i.e. the most relevant ones; b) those COs with the largest $A_i^+ + |A_i^+ + A_i^-|$, i.e. the most inconsistent ones.

5 Experimental Results

For the experimental evaluation of the proposed method, a multimodal dataset was compiled by us, since, to the best of our knowledge, no benchmark dataset for multimodal retrieval is available. For the creation of the dataset, three different modalities were used, namely 3D objects, 2D images and sounds. A total number of 495 COs is created, classified into 10 categories. To create these COs, 266 3D objects, 370 2D images and 283 sounds were used. While the 2D images are in fact snapshots of the corresponding 3D objects, the selection of sounds was not a trivial task. It required collection of environmental sounds, which were freely available in the Internet, and manual classification of these sounds to the predefined 10 categories. Then, the sounds that were classified to a specific category were randomly attached to 2D images and/or 3D objects of the same category. The dataset can be downloaded from the following url: http://3d-test.iti.gr:8080/3d-test/Download/Multimodal_Database_1.zip

The 3D object descriptors were extracted using the combined Depth-Silhouette-Radialized Extent (DSR) descriptor [14]. The 2D image descriptors consist of 2D Polar-Fourier coefficients, Zernike moments and Krawtchouk moments [15]. Finally, the audio descriptors are extracted using the algorithm presented in [16].

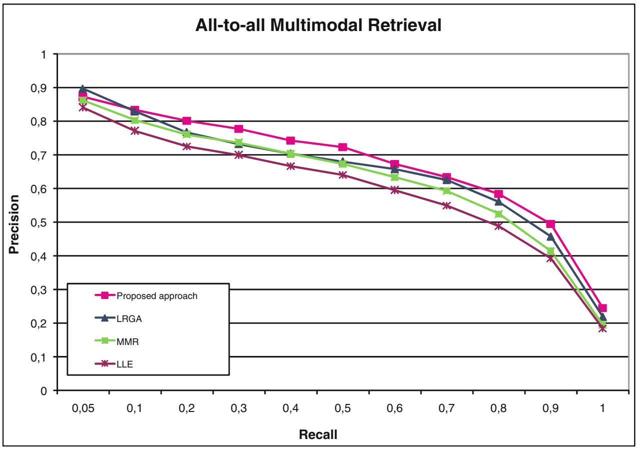


Fig. 2. Comparison of the proposed method against LRGA, MMR and LLE

In order to evaluate multimodal retrieval, each CO was extracted from the dataset and was used as query to retrieve the remaining COs in terms of similarity of the multimodal descriptor vectors. In Figure 2, the precision-recall diagrams of the proposed method against other similar multimodal retrieval approaches are presented. A definition of precision and recall values is available at [15]. The proposed method was compared in our experimental dataset with the Local Regression Global Alignment (LRGA) method [4], the Modified Manifold Ranking (MMR) method [12] and the Locally Linear Embedding (LLE) [13]. It is clear that our method outperforms the others in terms of retrieval accuracy.

In Figure 3, the improvement of the proposed multimodal retrieval method using RF is demonstrated. More specifically, *Query Expansion* is a method similar to the ones presented in [4], [8], *Re-Weighting* is based on equation (7), *Query Refinement* corresponds to the query reformulation method given in (4) and *Improved Query Refinement* is the modified version given in (5). Active learning using either the most relevant (*Query Refinement - L1*) or the most inconsistent (*Query Refinement - L2*) objects is also presented.

In this experiment, after retrieval of the initial ranked list, the user marks the top-P COs as relevant or non-relevant. In order to speed-up the evaluation process, instead of assigning users the task of marking the retrieved results, we took into account the dataset classification information: when a CO from the top-P retrieved results belongs to a category similar to the query CO, then it is automatically marked as relevant, otherwise, it is marked as irrelevant. From the diagram in Figure 3, it is obvious that all RF approaches improve the initial retrieval method. The best performance is achieved for the *Query Expansion*, the *Improved Query Refinement* and the methods based on Active learning. However, when it comes to integration with the large-scale indexing scheme presented in Section 3, most of the above RF methods cannot be easily adapted. The reasons are given below.

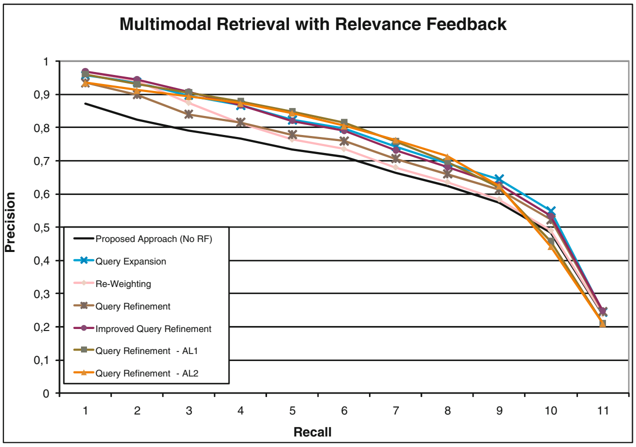


Fig. 3. Improvement of retrieval accuracy of the proposed method when several relevance feedback techniques are used

Query expansion, for example, requires retrieval of the entire dataset, not of the nearest neighbors only. This is essential for the ranking algorithm, which merges the ranked lists from multiple queries. Therefore, the collaboration of query expansion with indexing, which returns only a small subset of the dataset, is not feasible. Similar difficulties are observed in re-weighting methods. Reweighting is based on modifying the dissimilarity function at every RF iteration, since it dynamically changes the weights of each descriptor. However, in the indexing scheme proposed in this paper, the dissimilarities among all objects of the dataset have been already calculated during the pre-processing stage. During retrieval, the indexing algorithm does not calculate the dissimilarities of the query with the objects of the dataset. Thus, re-weighting would have no effect on retrieval of relevant objects. Finally, the use of active learning requires storing of a $N \times N$ dissimilarity matrix $\mathbf{W}$ of all objects of the dataset.

However, when it comes to really large datasets, where the use of large-scale indexing is recommended, storing of such large dissimilarity matrices becomes prohibitive.

Among the RF approaches presented above, query refinement seems to cooperate well with the proposed indexing scheme, without significant modifications. After the indexing algorithm returns a list of k-first results for a given query, the user marks the relevant and non-relevant objects. Then, the query refinement method produces a new query, which is given as input to the indexing algorithm.

Table 1. Retrieval accuracy of the proposed method combining the large-scale indexing with query-refinement-based relevance feedback

	Method	Tier-1	NN
1	No Indexing - No RF	0.735262	0.822222
2	No Indexing - RF (not keeping history)	0.791769	0.931313
3	Indexing - No RF	0.734725	0.818182
4	Indexing - RF (not keeping history)	0.805069	0.941414
5	Indexing - RF (keeping history)	0.821179	0.931313
6	Update Index after step 5	0.842486	0.935354

Results of the combination of RF with large-scale indexing are presented in Table 1. Here, only the *Improved Query Refinement* method is used for RF, since it is the only one that achieves both high improvement of retrieval accuracy and adaptability to large-scale indexing. Two alternatives of RF are tested: a) the system does not keep history of the previous RF sessions; b) the system keeps history of the previous RF sessions. The retrieval accuracy is measured in terms of Tier-1 precision and Nearest Neighbor [12] (precision-recall cannot be measured in the case of indexing because only a small subset of the dataset is retrieved for each query). Keeping history of the previous RF sessions means that the refined query is stored back to the database instead of the initial query to be used in the next retrieval sessions. This modification achieves even higher improvement of retrieval accuracy. Finally, if the modified queries of the entire dataset are used to update the multimodal index, the maximum retrieval accuracy is achieved.

6 Conclusions

In this paper, a new framework for multimodal search and retrieval was presented. The searchable items are rich media objects, namely the Content Objects (COs), which consist of multiple modalities. Multimodal search is realized by creating a new multimodal feature space, where all COs, irrespective of their constituting modalities can be mapped. Thus, each CO can be represented by a multimodal descriptor. Moreover, a multimedia indexing scheme is utilized to index these multimodal descriptors so as to accelerate search and retrieval and make the proposed framework suitable even for large-scale applications. Finally, a relevance feedback technique, which was particularly selected to be adapted to the multimodal indexing framework, improved the accuracy of the retrieved results. Although it was tested on a small multimodal dataset, the proposed method can deal even with real-life large-scale datasets and it can be easily extended in order to address a wider variety of media types and application paradigms.

Acknowledgments. This work has been supported by the EC-funded project ***I-SEARCH*** (http://www.isearch-project.eu/isearch/).

References

1. Lai, P.L., Fyfe, C.: Canonical correlation analysis using artificial neural networks. In: Proc. European Symposium on Artificial Neural Networks, ESANN (1998)
2. Li, D., Dimitrova, N., Li, M., Sethi, I.K.: Multimedia Content Processing through Cross-Modal Association. In: Proceedings of the Eleventh ACM International Conference on Multimedia (MM 2003), USA (2003)
3. Zhang, H., Weng, J.: Measuring Multi-Modality Similarities Via Subspace Learning for Cross-Media Retrieval. In: Zhuang, Y.-T., Yang, S.-Q., Rui, Y., He, Q. (eds.) PCM 2006. LNCS, vol. 4261, pp. 979–988. Springer, Heidelberg (2006)
4. Yang, Y., Xu, D., Nie, F., Luo, J., Zhuang, Y.: Ranking with Local Regression and Global Alignment for Cross Media Retrieval. ACM MM, Beijing, China (2009)
5. Salton, G., Buckley, C.: Improving retrieval performance by relevance feedback. Journal of the American Society for Information Science 41, 288–297 (1990)
6. Schettini, R., Ciocca, G., Gagliardi, I.: Content-based color image retrieval with relevance feedback. In: International Conf. on Image Processing, Kobe, Japan (1999)
7. Zhang, H., Meng, F.: Multi-modal Correlation Modeling and Ranking for Retrieval. In: Muneesawang, P., Wu, F., Kumazawa, I., Roeksabutr, A., Liao, M., Tang, X. (eds.) PCM 2009. LNCS, vol. 5879, pp. 637–646. Springer, Heidelberg (2009)
8. He, J., Li, M., Zhang, H.J., Tong, H., Zhang, C.: Manifold-Ranking Based Image Retrieval. ACM MM, New York USA (2004)
9. Rui, Y., Huang, T.S.: Optimizing learning in image retrieval. In: IEEE Conf. Computer Vision and Pattern Recognition, South Carolina (2000)
10. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput. 15(6), 1373–1396 (2003)
11. Amato, G., Savino, P.: Approximate similarity search in metric spaces using inverted files. In: Proceedings of the 3rd International Conference on Scalable Information Systems (InfoScale 2008), pp. 1–10. ICST (2008)
12. Vanamali, T.P., Godil, A., Dutagaci, H., Furuya, T., Lian, Z., Ohbuchi, R.: SHREC 2010 Track: Generic 3D Warehouse. In: Proceedings of the Eurographics/ACM SIGGRAPH Symposium on 3D Object Retrieval (2010)
13. Saul, L.K., Roweis, S.T.: Think Globally, Fit Locally: Unsupervised Learning of Low Dimensional Manifolds. Journal of Machine Learning Research (2003)
14. Vranic, D.: 3d model retrieval. Ph.D. Dissertation, University of Leipzig (2004)
15. Daras, P., Axenopoulos, A.: A 3D Shape Retrieval Framework Supporting Multimodal Queries. International Journal of Computer Vision (July 2009), doi:10.1007/s11263-009-0277-2
16. Wichern, Xue, Thornburg, Mechteley, Spanias: Segmentation, Indexing, and Retrieval for Environmental and Natural Sounds. IEEE Transactions on Audio, Speech and Language Processing (March 2010)

Context-Aware Querying for Multimodal Search Engines*

Jonas Etzold[1], Arnaud Brousseau[2], Paul Grimm[1], and Thomas Steiner[2]

[1] Erfurt University of Applied Sciences, Germany
{jonas.etzold,grimm}@fh-erfurt.de
[2] Google Germany GmbH, ABC-Str. 19, 20354 Hamburg, Germany
{arnaudb,tomac}@google.com

Abstract. Multimodal interaction provides the user with multiple modes of interacting with a system, such as gestures, speech, text, video, audio, etc. A multimodal system allows for several distinct means for input and output of data. In this paper, we present our work in the context of the I-SEARCH project, which aims at enabling context-aware querying of a multimodal search framework including real-world data such as user location or temperature. We introduce the concepts of *MuSeBag* for multimodal query interfaces, *UIIFace* for multimodal interaction handling, and *CoFind* for collaborative search as the core components behind the I-SEARCH multimodal user interface, which we evaluate via a user study.

Keywords: Multimodality, Context Awareness, User Interfaces.

1 Introduction

The I-SEARCH project aims to provide a unified framework for multimodal content indexing, sharing, search and retrieval. This framework will be able to handle specific types of multimedia and multimodal content, namely text, 2D images, hand-drawn sketches, videos, 3D objects and audio files), but also real world information that can be used as part of queries. Query results can include any available relevant content of any of the aforementioned types. This is achieved through Rich Unified Content Annotation (RUCoD), a concept that we have introduced in [19] with the main idea being the consistent description of all sorts of content using a common XML-based description format. It becomes clear that a framework like I-SEARCH faces specific challenges with regards to the user interface (UI). Not only does it have to allow for the combination of multimodal queries, but it also has to do so on different devices, both desktop and mobile. This research being conducted in the context of a European research project, we have time constraints to take into account, hence, we cannot afford

* This work is partly funded by the EU FP7 I-SEARCH project under project reference 248296. We would like to thank all of the partners in the I-SEARCH project for their support.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 728–739, 2012.

to develop two separate UI stacks for desktop and mobile. Instead, we show how using newly added features in the markup language HTML we can kill these two birds with one stone.

The remainder of this paper is structured as follows: Section 2 presents related work, Section 3 introduces our chosen methodology, Section 4 goes into implementation details, Section 5 presents the evaluation of a user study that we have conducted, and finally Section 6 ends the paper with an outlook on future work and provides a conclusion.

2 Related Work

Many have been involved in research to improve user interfaces (UI) for search tasks in the last few years. They widely found evidence for the importance and special demand on the design of search UIs in order to achieve an effective and usable search [11,21,14]. Especially with the emerge of the so-called Web 2.0 and the vast amount of user generated content, the raise of the big search engines like Google and Bing continued, and search became one of the main tasks in our daily Internet usage [22]. This trend further increases the importance of the interaction with, and the design of search engines, and also raises the need for extending search tasks beyond textual queries on desktop systems. In this manner, Hearst [12] describes emerging trends for search interface design, which include that interfaces have to be more device-independent (i.e. also support mobile devices), and be able to support the creation of multimodal search queries where text can be enriched with multimedia and real-world data in order to deliver more precise results. With the development of multimodal search interfaces, also concepts for multimodal interaction, as defined by Nigay et al. [18], become an important aspect to distribute all features of this new type of search interfaces to the user. Rigas [23] also found evidence that the use of multimodal features of a search interface, e.g. speech or graphs can support the usability of the whole search engine. In order to combine the efforts towards multimodal interaction, the World Wide Web Consortium (W3C) follows an approach to create a framework that is described by the W3C Multimodal Interaction Working Group with its work-in-progress specification of the “Multimodal Architecture and Interfaces” [3]. Therein, the framework is used to describe the internal structure of a certain interaction component, including the in- and outputs of the various interaction types based on XML. Serrano et al. further created the open interface framework [25], which allows for the flexible creation of combined interaction pipelines using several input channels (e.g. speech and touch). Other approaches to provide frameworks for multimodal interaction and interfaces are described by Sreekanth [26], who uses a *Monitor Agent* to collect events from different modalities and Roscher [24], who uses the Multi-Access Service Platform (MASP), which implements different user interface models for each input modality and is able to combine them to more complex multimodal user interfaces including the synchronization of all inputs along the user interface models.

The possibility to generate more complex, but also more effective search queries with multimodal search interfaces, as well as the nature of the Internet as an environment where people can assist each other, make the integration of collaborative interaction approaches for search engines interesting. Mainly the work of Morris [17] and Pickens [20] described interesting ways of collaborative search approaches. They make use of a search session and state variables in user profiles to transfers changes made in the interface of one user to all other collaborating users and vice versa. Further, the survey about collaborative Web search practices done by Morris [16] as well as the status quo practices presented by Amershi [2] prove the need and practicability of collaborative search methods.

3 Methodology

In this Section, we present our methodology for context-aware querying of multimodal search engines, split up in three sub-tasks: *MuSeBag* for our multimodal query interfaces, *UIIFace* for our multimodal interaction framework, and *CoFind* for our collaborative search framework.

3.1 Multimodal Query Interfaces – MuSeBag

In order to create a visual platform for multimodal querying between user and search engine, the concept of *MuSeBag* was developed. *MuSeBag* stands for **Mu**ltimodal **Se**arch **Bag** and designates the I-SEARCH UI. It comes with specific requirements linked with the need for users to use multiple types of input: audio files or stream, video files, 3D objects, hand drawings, real-world information such as geolocation or time, image files, and of course, plain text. This part of the paper shows the approach chosen to create *MuSeBag*.

Multimodal search engines are still very experimental at the time of writing. When building *MuSeBag*, we tried to look for a common pattern in search-related actions. Indeed, *MuSeBag* remains a search interface at its core. In order for users to interact efficiently with I-SEARCH, we needed a well-known interface paradigm. Across the Web, one pattern is used for almost any and all search related actions: the text field, where a user can focus, enter her query, and trigger subsequent search actions. From big Web search engines such as Google, Yahoo!, or Bing, to intranet search engines, the pattern stays the same. However, I-SEARCH cannot directly benefit from this broadly accepted pattern, as a multimodal search engine must accept a large number of query types at the same time: audio, video, 3D objects, sketches, etc. Some search engines, even if they do not have the need for true multimodal querying, still do have the need to accept input that is not plain text.

First, we consider TinEye [27]. TinEye is a Web-based search engine that allows for query by image content (QBIC) in order to retrieve similar or related images. The interface is split in two distinct parts: one part is a text box to provide a link to a Web-hosted image, while the second part allows for direct file upload (Figure 1). This interface is a good solution for a QBIC search engine like TinEye, however, the requirements for I-SEARCH are more complex.

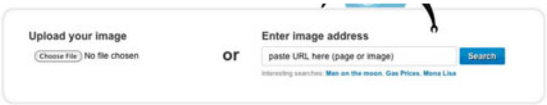
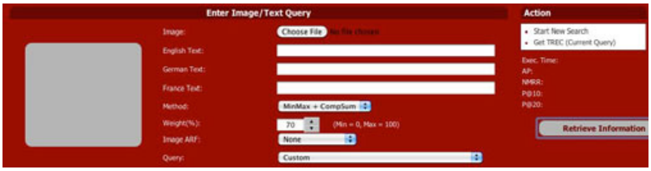

Upload your image
Choose file
No file chosen
or
Enter image address
paste URL here (page or image)
Search
Interesting searches: Man on the moon, Gas Prices, Mona Lisa

Fig. 1. Screenshot of the TinEye user interface

As a second example, we examine MMRetrieval [29]. It brings image and text search together to compose a multimodal query. MMRetrieval is a good showcase for the problem of designing a UI with many user-configurable options. For a user from outside the Information Retrieval field, the UI seems not necessarily clear in all detail, especially when field-specific terms are used (Figure 2).



Method: MinMax + CompSum
Weight(%): 70 (Min = 0, Max = 100)
Image ARF: None
Query: Custom

Fig. 2. Screenshot of the MMRetrieval user interface

Finally, we have a look at Google *Search by image* [10], a feature introduced in 2011 with the same UI requirements as MMRetrieval: combining text and image input. With the *Search by image* interface, Google keeps the text box pattern (Figure 3), while preventing any extra visual noise. The interface is *progressively disclosed* to users via a contextual menu when the camera icon is clicked.

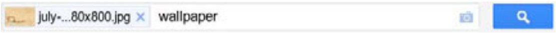
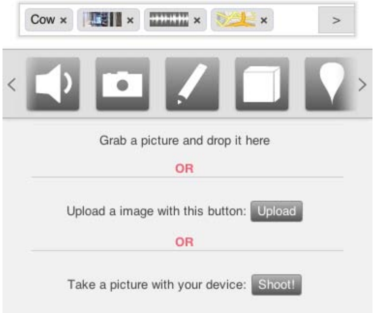


Fig. 3. Input for the *Search by image* user interface

Even if the *Search by image* solution seems evident, it is still not suitable for I-SEARCH since the interface would require a high number of small icons: camera, 3D, geolocation, audio, video, etc. As a result, we decided to adapt a solution that can be seen in Figure 4. This interface keeps the idea of a single text box. It is enriched with text auto-completion as well as "tokenization". By the term "tokenization" we refer to the process of representing an item (picture, sound, etc.) with a token in the text field, as if it was part of the text query. We also keep the idea of *progressive disclosure* for the different actions required by the various modes, e.g. uploading a picture or sketching something. The different icons are grouped together in a separated menu, close to the main search field.



Grab a picture and drop it here

OR

Upload a image with this button: Upload

OR

Take a picture with your device: Shoot!

Fig. 4. First version of I-SEARCH interface showing the *MuSeBag* concept

3.2 Multimodal Interaction Handling – UIIFace

Interaction is an important factor when it comes to context-awareness and multimodality. In order to deliver a Graphical User Interface (GUI) that is able to facilitate all the possibilities of a multimodal search engine, a very flexible approach with a rich interaction methodology is needed. Not only the way search queries are build should be multimodal, also the interaction to generate and navigate in such a multimodal interface should be multimodal. To target all those needs, we introduce the concept of *UIIFace* (**U**nified **I**nteraction **I**nter**face**) as general interaction layer for context-aware multimodal querying. *UIIFace* describes a common interface between these interaction modalities and the graphical user interface (GUI) of I-SEARCH by providing a general set of interaction commands for the interface. Each input modality provides the implementation for parts of the commands or all commands defined by *UIIFace*.

The idea of *UIIFace* is based on the open interface framework [25], which describes a framework for the development of multimodal input interface prototypes. It uses components that can represent different input modalities as well as user interfaces and other required software pieces in order to create and control a certain application. In contrast to this approach, *UIIFace* is a Web-based approach implemented on top of modern HTML5 [15] functionalities. Furthermore, it provides a command line interface to the Web-based GUI, which allows for the creation of stand-alone applications outside of the browser window. For the set of uni- and multimodal commands that can be used for I-SEARCH interfaces, the results of Chang [5] as well as the needs derived from the creation of multimodal search queries are used.

Figure 5 depicts the internal structure of *UIIFace* and shows the flow of events. Events are fired by the user's raw input. Gesture Interpreter determines defined gestures (e.g. zoom, rotate) found in the raw input. If no gestures were found, the Basic Interpreter routes Touch and Kinect[1] events to basic cursor and keyboard events. Gestures, speech commands and basic mouse and keyboard events are then synchronized in the Interaction Manager and forwarded as Combined

[1] A motion sensing input device by Microsoft for the Xbox 360 video game console.

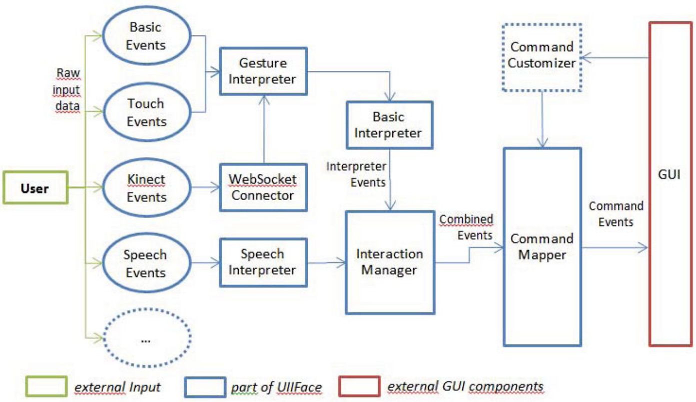


Fig. 5. Schematic view on the internal structure of *UIIFace*

Events to the Command Mapper which maps the incoming events to the defined list of interaction commands that can be registered by any Web-based GUI. The Command Customizer can be used to rewrite the trigger event for commands to user specific gestures or other input sequences (e.g. keyboard shortcuts). This is an additional feature that is not crucial for the functionality of *UIIFace*, but that can be implemented at a later stage in order to add more explicit personalization features.

3.3 Collaborative Search – CoFind

Another part of our methodology targets the increased complexity of search tasks and the necessity to collaborate on those tasks in order to formulate adequate search queries, which lead faster to appropriate result. The increased complexity is primarily caused by the vast amount of unstructured data on the Internet and secondly by situations where the expected results are very fuzzy or hard to describe in textual terms. Therefore the *CoFind* (**Co**llaborative **Find**ing) approach is introduced as a collaborative search system, which enables real-time collaborative search query creation on a pure HTML interface. Real-time collaboration is well-known in the field of document editing (e.g. EtherPad [7], Google Docs [9]); *CoFind* applies the idea of collaborative document editing to collaborative search query composition.

CoFind is based on the concept of shared search sessions in which HTML content of the participants' local clients is transmitted within this session. In order to realize collaborative querying, the concept provides functions for activating collaborative search sessions, joining other online users' search sessions and managing messaging between participants of the search session. Figure 6 shows how the parts listed in the following interact during the search process in order to create a collaborative search session:

Session Manager. Controls opening / closing of collaborative search sessions.
Content Manager. Broadcast of user interfaces changes to all participants.
Messaging Manager. Broadcast of status / user messages to all participants.

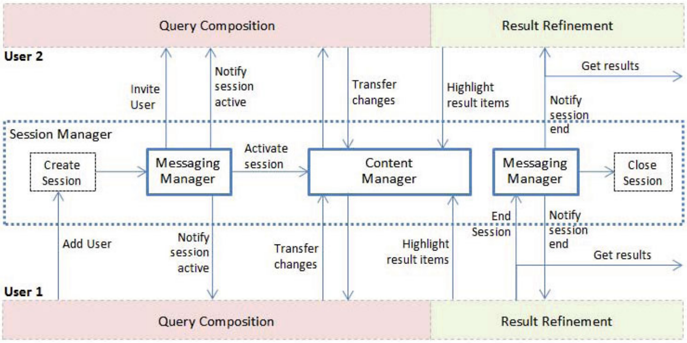


Fig. 6. Schematic diagram of interaction between parts of *CoFind*

The main flow of a collaborative search session can be described as follows: to join a collaborative search session initiated by a user A, a user B must supply the email address of user A. If user A is online and logged in, she receives an on-screen notification and needs to accept the collaboration request of the user B. Upon acceptance, a new session entry is created that stores all participants. Every time a change on the query input field or result set occurs, the changed state is transferred to all participants. Each participant is able to search and navigate through the result set independently from the others, but selected results can be added to collaborative result set. The search session is closed after all users have left the session or have logged out from the system.

4 Implementation Details

The I-SEARCH GUI is built using the Web platform. HTML, CSS, and JavaScript are the three main building blocks for the interface. The rationale behind this choice is that I-SEARCH needs to be cross-browser and cross-device compatible, requirements fulfilled by CSS3 [6], HTML5 [15] and the therein defined new JavaScript APIs that empower the browser in truly novel ways. However, our strategy also includes support for older browsers. When browsing the Web, a significant part of users do not have access to a cutting-edge Web browser. If a feature we use is not available for a certain browser version, two choices are available: either drop support for that feature if it is not important (e.g. drop visual shims like CSS shadows or border-radius), or provide alternate fallback solutions to mimic the experience. We would like to highlight that CSS and HTML are two standards that natively enable *progressive enhancement* thanks to a simple rule: when a Web browser does not understand an HTML attribute, a CSS value or selector, it simply ignores it. This rule is the guarantee that we can build future-proof applications using CSS and HTML. Web browsers render the application according to their capabilities: older browsers render basic markup and styles, while modern browsers render the application in its full glory. Sometimes, however, we have to ensure that all users can access a particular feature. In this case, we use the principle of *graceful degradation*, i.e. use fallback solutions when the technology stack does not support our needs in a certain browser.

4.1 CSS3 Media Queries

The I-SEARCH project needs to be compatible with a large range of devices: desktop browsers, phones, and tablets. Rather than building several versions of I-SEARCH, we use CSS3 media queries [6] to dynamically adapt the layout to different devices.

4.2 Canvas

The `canvas` element in HTML5 [15] allows for dynamic, scriptable rendering of 2D shapes and bitmap images. In the case of I-SEARCH, we use `canvas` for user input when the query requires a user sketch, and also to display results in novel ways. The `canvas` element being a core element of I-SEARCH, it is crucial to offer a fallback solution for older browsers. We plan to do so by using FlashCanvas [8], a JavaScript library, which adds the renders shapes and images via the Flash drawing API.

4.3 HTML5 Audio and Video

The HTML5 `audio` and `video` elements make multimedia content a first class citizen in the Web browser, including scriptability, rotation, rescale, controls, CSS styles, and so forth. For I-SEARCH, this flexibility allows us to create interesting and interactive visualizations of search results. If `audio` and `video` are not available, we fall back to Adobe Flash [1] to display media items to users.

4.4 File API

The HTML5 file API provides an API for representing file objects in Web applications, as well as programmatically selecting them and accessing their data. This is interesting in the case of I-SEARCH, since users are very likely to compose their query with local files, like audio files, pictures, etc. The file API allows for a new paradigm to deal with files, such as native support for dragging and dropping elements from the desktop to the I-SEARCH interface. This convenience feature is not crucial, an HTML file upload form serves as a fallback.

4.5 Geolocation

Context-aware search is one of the features of the I-SEARCH framework. This is particularly useful in the case of a user searching on a mobile device, as many mobile queries are location-based. HTML5 includes the geolocation JavaScript API that, instead of looking up IP address-based location tables, enables Web pages to retrieve a user's location programmatically. In the background, the browser uses the device GPS if available, or computes an approximate location based on cell tower triangulation. The user has to agree for her location to be shared with the application.

4.6 Sensors

Another important aspect for context-awareness is the use of hardware sensors integrated or attached to different device types. These sensors are capable of retrieving the orientation and acceleration of a device or capturing the movements of a user in 3D space. With that knowledge the system is able to make assumptions about the user's direct environment or to detect gestures, which further increases the overall context-awareness. Many of today's mobile devices have accelerometers and gyroscopes integrated that can be accessed through device-specific APIs. HTML5 supports events that target those sensors and defines unified events in the specification for the `deviceorientation` event [4]. Desktop sensors like the Kinect provide depth-information for tracking people in 3D space. These sensors do not yet have a common standard for capturing their data in a browser environment. For those sensors we have created a lightweight WebSocket-based [13] abstraction library.

4.7 Device API

With the Device API [28] the W3C currently creates the next standard related to HTML5. It is mainly targeted to give Web browsers access to attached hardware devices of the client computer. Therefore the Media Capture API, which is a part of the Device API, will enable access to the microphone and the Web camera of the user. We use this API in combination with appropriate fallback routines in order to create audio queries as well as image queries captured on-the-fly.

5 Evaluation

To validate our interface design choices with real multimodal search tasks, we have conducted a user study. We went for a comparative study design to explore how usage of different media types would look like and how they would influence the success rate of search queries. As this user study was mainly focused on the user interface and user interaction parts of I-SEARCH, we assumed that the system always had a correct answer to the (limited) set of permitted queries, even if the real search back-end was not yet in operation at the time of writing.

We therefore set the following hypotheses: (H1) Most users will start a search query with just one media type. (H2) Search refinements will be done by adding or removing other media types. (H3) All media types will be handled similarly.

For the user study we recruited seven participants (six male and one female) aged between 20 and 35. All participants were familiar with textual Web-based search. We asked all study participants to find three different items (sound of a tiger, 3D model of a flower, image of a car). The participants were shown these items beforehand in their original format and were then instructed to query the system in order to retrieve them via I-SEARCH. For the study a Windows laptop with a capacitive touch display was used. Each participant was interviewed after the completion of the study. Our goal was to validate our interface design as well as to measure the impact of the possibility of multimodal search. In general, we observed that the concept of multimodal search was new and unfamiliar to all participants. Actually, before the user study all participants considered Web search equal to text-based search, and only by using I-SEARCH they became aware of the possibility to use different media types and of multimodal interaction at all. Our hypothesis (H1) was statistically not supported. It depends highly on the behavior of each individual person whether one or more search items or media types are used. In combination with (H2), one obvious conclusion of the participant interviews was that adding search items as well as customizing them has to be as easy as possible. The participants did not hit obstacles in using one special query modality, however stated that if a query modality was difficult to use, they would replace it by using different query modalities, even if this implied that the search query would become complicated and challenging. The same conclusion applies to hypothesis (H3). In order to allow for multimodal search queries, the following recommendations can be derived from our user study:

1. No query modality should be privileged.
2. The handling of all search modalities should be as consistent as possible.
3. Search refinement should be possible in the result presentation.

6 Conclusion and Future Work

In this paper, we have presented relevant related work in the fields of search engine interface design, multimodality in the context of search, and collaborative search. Second, we have introduced our methodology with the concepts of *MuSeBag* for multimodal query interfaces, *UIIFace* for multimodal interaction handling, and *CoFind* for collaborative search as the core components behind the I-SEARCH multimodal user interface, together with their implementation details. Finally, we have briefly discussed first results of a user study on the I-SEARCH user interface.

Future work will focus on the following aspects: we will conduct more and broader user studies once the CoFind component is up and running, and once the search engine delivers real results and not mocked-up results as in the current study. We will also focus on user-placable tags for search queries, which will allow for the tracking of search results changes over time. From the hardware side we will work on supporting more input device modalities such as gyroscopes and compasses that are more and more common standard in modern smartphones. One of the main results from the user study was that consistency of the different input modalities both from a treatment and usage point of view needs to be improved. We will thus focus on streamlining the usability of the product, guided by to-be-conducted so-called A/B or also multivariate tests. This will allow us to fine-tune the user interface while the I-SEARCH search engine is already in real-world use.

Concluding, we feel that we are on a good track in the right direction towards an innovative multimodal search engine user interface design, however, have barely scratched the surface of what is still ahead. It is clear that our current user study can, at most, serve to detect overall trends, however, in order to retrieve statistically significant results we need to scale our tests to more users. Given our team composition of both academia (University of Applied Sciences Erfurt, Centre for Research & Technology Hellas) and industry (Google), we are in an excellent position to tackle the challenges in front us.

References

1. Adobe Flash Platform, http://www.adobe.com/flashplatform/
2. Amershi, S., Morris, M.R.: Co-located Collaborative Web Search: Understanding Status Quo Practices. In: Proceedings of the 27th International Conference on Human Factors in Computing Systems, CHIEA 2009, pp. 3637–3642. ACM, New York (2009)
3. Barnett, J., Group, M.I.W.: Multimodal Architecture and Interfaces (July 2011) http://www.w3.org/TR/mmi-arch
4. Block, S., Popescu, A.: DeviceOrientation Event Specification, Editor's Draft (2011), http://dev.w3.org/geo/api/spec-source-orientation.html
5. Chang, J., Bourguet, M.-L.: Usability Framework for the Design and Evaluation of Multimodal Interaction. In: Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction, BCS-HCI 2008, vol. 2, pp. 123–126. British Computer Society (2008)
6. Etemad, E.J.: Cascading Style Sheets (CSS) Snapshot (2010), W3C Working Group Note (May 12, 2011), http://www.w3.org/TR/CSS/
7. EtherPad, Collaborative Text Editor, https://github.com/ether/pad
8. FlashCanvas, JavaScript Library, http://flashcanvas.net/
9. Google Docs, Collaborative Document Editing, https://docs.google.com/
10. Google Search by image, Blog Post, http://googleblog.blogspot.com/2011/06/knocking-down-barriers-to-knowledge.html
11. Hearst, M.A.: Search User Interfaces, 1st edn. Cambridge University Press, New York (2009)
12. Hearst, M.A.: Emerging Trends in Search User Interfaces. In: Proceedings of the 22nd ACM Conference on Hypertext and Hypermedia, HT 2011, pp. 5–6. ACM, New York (2011)
13. Hickson, I.: W3C – The WebSocket API Editor's Draft (2011), http://dev.w3.org/html5/websockets
14. Huang, S.-T., Tsai, T.-H., Chang, H.-T.: The UI issues for the search engine. In: 11th IEEE International Conference on ComputerAided Design and Computer Graphics, pp. 330–335 (2009)

15. Hickson, I.: HTML5, A Vocabulary and Associated APIs for HTML and XHTML, W3C Working Draft (May 25, 2011), http://www.w3.org/TR/html5/
16. Morris, M.R.: A Survey of Collaborative Web Search Practices. In: Proceedings of the 26th Annual SIGCHI Conference on Human Factors in Computing Systems, CHI 2008, pp. 1657–1660. ACM, New York (2008)
17. Morris, M.R., Horvitz, E.: SearchTogether: an interface for collaborative web search. In: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, UIST 2007, pp. 3–12. ACM, New York (2007)
18. Nigay, L.: Design Space for Multimodal Interaction. In: IFIP Congress Topical Sessions, pp. 403–408 (2004)
19. Daras, P., Axenopoulos, A., Darlagiannis, V., Tzovaras, D., Le Bourdon, X., Joyeux, L., Verroust-Blondet, A., Croce, V., Steiner, T., Massari, A., et al.: Introducing a Unified Framework for Content Object Description. International Journal of Multimedia Intelligence and Security, Special Issue on "Challenges in Scalable Context Aware Multimedia Computing (2010), http://www.inderscience.com/browse/index.php?journalID=359
20. Pickens, J., Golovchinsky, G., Shah, C., Qvarfordt, P., Back, M.: Algorithmic Mediation for Collaborative Exploratory Search. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, pp. 315–322. ACM, New York (2008)
21. Quesenbery, W., Jarrett, C., Roddis, I., Allen, S., Stirling, V.: Designing for Search: Making Information Easy to Find. Technical report, Whitney Interactive Design (June 2008)
22. Quesenbery, W., Jarrett, C., Roddis, I., Stirling, V., Allen, S.: The Many Faces of User Experience (Presentation), Baltimore, Maryland, USA (June 16-20, 2008) http://www.usabilityprofessionals.org/
23. Rigas, D., Ciuffreda, A.: An Empirical Investigation of Multimodal Interfaces for Browsing Internet Search Results. In: Proceedings of the 7th International Conference on Applied Informatics and Communications, pp. 194–199. World Scientific and Engineering Academy and Society, Stevens Point, Wisconsin, USA (2007)
24. Roscher, D., Blumendorf, M., Albayrak, S.: A Meta User Interface to Control Multimodal Interaction in Smart Environments. In: Proceedings of the 14th International Conference on Intelligent User Interfaces, IUI 2009, pp. 481–482. ACM, New York (2009)
25. Serrano, M., Nigay, L., Lawson, J.-Y.L., Ramsay, A., Murray-Smith, R., Denef, S.: The Open Interface Framework: A Tool for Multimodal Interaction. In: CHI 2008 Extended Abstracts on Human Factors in Computing Systems, pp. 3501–3506. ACM, New York (2008)
26. Sreekanth, N.S., Pal, S.N., Thomas, M., Haassan, A., Narayanan, N.K.: Multimodal Interface: Fusion of Various Modalities. International Journal of Information Studies 1(2) (2009)
27. TinEye Image Search Engine, http://www.tineye.com/
28. W3C. W3C – Device APIs and Policy Working Group (2011), http://www.w3.org/2009/dap
29. Zagoris, K., Arampatzis, A., Chatzichristofis, S.A.: www.MMRetrieval.net: A Multimodal Search Engine. In: Proceedings of the 3rd International Conference on SImilarity Search and Applications, SISAP 2010, pp. 117–118. ACM, New York (2010)

A Novel Multi-modal Integration and Propagation Model for Cross-Media Information Retrieval

Wanxia Lin[1], Tong Lu[1, 2,*], and Feng Su[1]

[1] State Key Laboratory for Novel Software Technology,
Nanjing University, Nanjing 210093, China
[2] Jiangyin Institute of Information Technology of Nanjing University, China
linwanxia828@yahoo.com.cn, lutong@nju.edu.cn, suf@nju.edu.cn

Abstract. In this paper, we present a novel Probabilistic Latent Semantic Analysis-based (PLSA-based) aspect model and turn cross-media retrieval into two parts of multi-modal integration and correlation propagation. We first use multivariate Gaussian distributions to model continuous quantity in PLSA, avoiding information loss between feature-instance versus real-world matching. Multi-modal correlations are learned in an asymmetrical manner, giving a better control of the respective influence of each modality in the latent space. Then we propose a new propagation pattern to refine multi-modal correlations by efficiently taking the complementary from multi-modalities. Experimental results demonstrate that our method is accurate and robust for cross-media information retrieval.

Keywords: Multi-modal, PLSA, Asymmetric Learning, Propagation.

1 Introduction

With the production of large multimedia collections favored by cheap digital recording devices, there is a clear need for efficient multimedia content analysis systems. Correlations between interrelated data have been proven helpful in video analysis [1][2], image retrieval [3] or automatic annotation [4]; however, in most systems that handle digital media, different modalities are treated separately without combination at a higher level [5].

Actually, multi-modal data originated from the same source tend to be correlated [6]. It means that different modalities can take a complementary role on solving multimedia content analysis tasks and the presence of one modality can help understand certain semantics of others. For example, a video retrieval system that exploits both modalities of audio and image may achieve better performance in both accuracy and efficiency than one which exploits either one or the other [1][3], due to the fact that each modality may compensate for weakness of the other. As a result, multi-modal correlation has been given a growing attention in multimedia content analysis

[*] Corresponding author.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 740–749, 2012.

research in the recent years [5][7]. [18] uses Short-term Audio-Visual Atom (S-AVA), automatic detection of semantic concepts in unconstrained videos, by joint analysis of audio and visual content.

The state-of-the-art techniques can be roughly classified into two categories: multi-modal correlation analysis and cross-media index. *Multi-modal correlation analysis* [8][9] approach explores statistic correlations between cross modalities by analyzing their co-occurrence relationship. For example, after extracting visual and auditory features, the known canonical correlation (CC) can be analyzed between the feature matrices to learn their correlations [5] and then use a hierarchical manifold space to make the correlations more accurate [10]. However, difficulties still exist due to the heterogeneous feature space and the un-corresponding visual or auditory contexts.

Unlike multi-modal correlation analysis methods, *cross-media index* approach focuses on automatically labeling un-annotated multimedia data using textual models [11][12]. These methods first represent a visual or auditory feature cluster with the closet dictionary index, and then construct a linked representation to obtain image-text (or audio-text) translation results. Despite its success, this approach suffers from several weaknesses. First, representing each local visual or auditory feature by a dictionary index can result in severe loss of information. Second, cross-media index actually focuses on the annotation problem, ignoring semantics reasoning among multi-modal data.

In this paper, we propose a novel Multi-Modal Integration and Propagation (MMIP) model for cross-media retrieval. We consider multi-modal data can be better correlated in the higher level aspect space than the feature one. Our method has two stages. First, we extract quantized visual and auditory features and use multivariate Gaussian distributions to model continuous quantity. Model parameters are asymmetrically trained to learn multi-modal correlations. Then a new multi-modal propagation model is proposed to refine the correlations between images and audios, providing improved retrieval results by taking the complementary from multi-modalities.

2 Correlation Learning

In this section, we introduce how to represent a multi-modal document and model continuous quantity for multi-modal correlation learning.

2.1 Multi-modal Gaussian PLSA

Multi-modal Document. We consider the multi-modal data from the same category as a single *multi-modal document*. For example, a sound track of tiger roaring and some pictures of tigers are considered one multi-modal document. Given a collection of multimedia documents, such as audio clips and images, of N categories of topics or subjects $D = \{D_1,...,D_c,...,D_N\}$, a multi-modal document D_c of category c can be defined by

$$D_c = \{d_i^I \mid d_i^I \in \text{category } c\} \cup \{d_j^A \mid d_j^A \in \text{category } c\}, \tag{1}$$

where d_i^I is an image that is represented as a collection of various visual features and their occurrence counts:

$$S(d_i^I)=\{n(d_i^I,s_1),...,n(d_i^I,s_p),...,n(d_i^I,s_{N_I})\}\,, \tag{2}$$

where s_p is one specific visual feature vector and $n(d_i^I,s_p)$ is the number of s_p appearing in present d_i^I. Similarly, d_j^A is an audio clip of the same topic category, which is described as:

$$M(d_j^A)=\{n(d_j^A,m_1),...,n(d_j^A,m_q),...,n(d_j^A,m_{N_A})\}\,, \tag{3}$$

where m_q is one specific feature vector and $n(d_i^A,m_q)$ is the number of m_q present in d_j^A. Therefore, D_c can be represented by the following vector $SM(D_c)$ of size N_I+N_A:

$$SM(D_c)=\{n(D_c,s_1),...,n(D_c,s_p),...,n(D_c,s_{N_I}),n(D_c,m_1),...,n(D_c,m_q),...,n(D_c,m_{N_A})\}\,, \tag{4}$$

where $n(D_c,s_p)$ and $n(D_c,m_q)$ are defined as follows:

$$n(D_c,s_p)=\sum\nolimits_{d_i^I\in D_c} n(d_i^I,s_p)\,, \tag{5}$$

$$n(D_c,m_q)=\sum\nolimits_{d_j^A\in D_c} n(d_j^A,m_q)\,. \tag{6}$$

Model. Like standard PLSA, a latent aspect $z_k\ (k\in 1,...,K)$ is introduced in the generative process of each feature vector $x_j^*\ (j\in 1,...,N_*,\text{* stands for I or A})$ in $D_c\ (c\in 1,...,N)$. However, information may be lost between feature-instance versus real-world matching even N_* is relatively large [14]. For the unobservable variable z_k, we suppose x_j^* can be sampled from a multivariate Gaussian distribution. Due to the fact that our visual or auditory features in D_c are independent between different images or auditory clips, the learned multivariate Gaussian model can well predict the feature distributions of unknown images or audios. Fig. 1 shows the graphical model of our continuous PLSA, where two modalities, auditory and visual, are correlated by sharing the same distribution over the latent aspect $p(z_k\mid D_c)$.

In our model, the joint probability of an observed pair (D_c,x_j^*) is defined by

$$p(D_c,x_j^*)=\sum\nolimits_z p(D_c)p(z\mid D_c)p(x_j^*\mid z)=\sum\nolimits_z p(z)p(D_c\mid z)p(x_j^*\mid z)\,, \tag{7}$$

in which, each feature x_j^* is generated from the K Gaussian distributions, and each Gaussian distribution corresponds to one specific latent aspect z_k. For z_k, the conditional probability distribution of x_j^* is

$$p(x_j^* \mid z_k) = \frac{1}{(2\pi)^{Dim/2} \mid \Sigma_k \mid^{\frac{1}{2}}} \exp\{-\frac{1}{2}(x_j^* - \mu_k^*)^T \Sigma_k^{-1} (x_j^* - \mu_k^*)\} . \tag{8}$$

where Dim is the dimension of x_j^*, μ_k^* and Σ_k^* are the mean vector and the covariance matrix of x_j^* belonging to z_k.

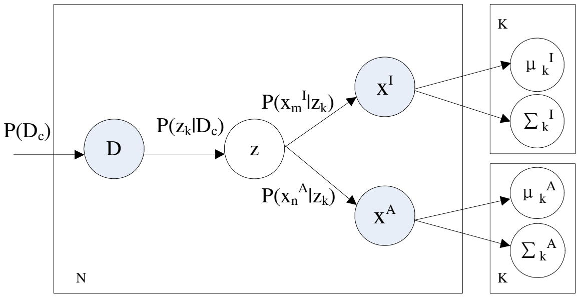


Fig. 1. Multi-modal Gaussian PLSA

2.2 Multi-modal Learning

In the training stage, we asymmetrically fuse multi-modal features by first constructing a latent space on one modality and then linking it with another one. Since the heterogeneous features in the same multi-modal document share the same latent semantics, the asymmetric learning is feasible and gives a better control of the respective influence of each modality in the latent space. The details of our multi-modal learning process are given as follows.

First, each D_c is processed and represented as $SM(D_c)$ and we first choose the visual modality to estimate the parameters of $p(z_k)$, $p(D_c \mid z_k)$, μ_k^I and Σ_k^I with the Expectation-Maximization (EM) algorithm on the training data set $\{D_1,..., D_c,..., D_N\}$. The algorithm is based on the likelihood of the observed data given the parameters of the distributions $p(z_k)$, $p(D_c \mid z_k)$, μ_k^I and Σ_k^I and iteratively looks for the maximum of this likelihood through the E and M step:

$$E(L^c) = \sum_c^N \sum_m^{N_I} n(D_c, x_m^I) \log \sum_{k=1}^K p(D_c \mid z_k) p(x_m^I \mid z_k) p(z_k) \tag{9}$$

E-step. Compute the conditional probability distribution of the latent aspect z_k given the observation pair (D_c, x_m^I) from the previous estimation of the model parameters:

$$p(z_k \mid D_c, x_m^I) = \frac{p(D_c \mid z_k) p(x_m^I \mid z_k) p(z_k)}{\sum_{k=1}^K p(D_c \mid z_k) p(x_m^I \mid z_k) p(z_k)} \tag{10}$$

M-step. Update the parameters of $p(z_k)$, $p(D_c \mid z_k)$, μ_k^I and Σ_k^I with the new $p(z_k \mid D_c, x_m^I)$.

Then, base on the parameters estimated from the visual modality, we adopt the folding-in method [15] to infer μ_k^A and Σ_k^A for the auditory modality, with the aspect distributions $p(z_k)$ and $p(D_c \mid z_k)$ kept fixed. Similarly, $p(x_m^I \mid z_k)$ and $p(x_n^A \mid z_k)$ can be inferred according to (8) after knowing the model parameters of $\mu_k^I, \Sigma_k^I, \mu_k^A$ and Σ_k^A. Note that the learned parameters remain valid for the images and audio out of the training set. Next, each d_i^I is represented as $S(d_i^I)$ and we once again use the folding-in method to infer $p(d_i^I \mid z_k)$ for the visual modality with μ_k^I, Σ_k^I and the aspect distributions $p(z_k)$ kept fixed, based on which $p(z_k \mid d_i^I)$ can be further inferred. $p(d_i^A \mid z_k)$ and $p(z_k \mid d_i^A)$ can be inferred in the same way.

3 Correlation Propagation

In this section, we propose a novel propagation model to reinforce multi-modal correlations for robust multimedia content analysis.

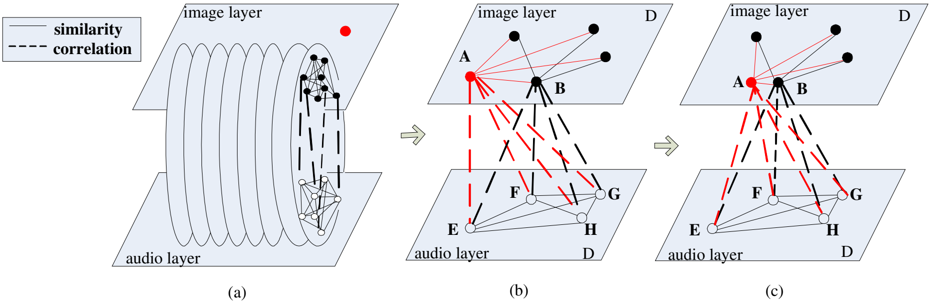


Fig. 2. Multi-modal propagation network

Given the training image-audio data set, the propagation model is initialized by calculating the correlations among every image and audio clip pair as follows:

$$Cor(d_i^{k_1}, d_j^{k_2}) = \frac{p(z \mid d_i^{k_1}) * p(z \mid d_j^{k_2})^T}{\mid p(z \mid d_i^{k_1}) \mid * \mid p(z \mid d_j^{k_2}) \mid} (k_1, k_2 = \mathrm{I} \text{ or } \mathrm{A}) \tag{11}$$

where $p(z \mid d_i^{k_1}) = \{p(z_1 \mid d_i^{k_1}), \ldots, p(z_j \mid d_i^{k_1}), \ldots, p(z_k \mid d_i^{k_1})\}$. The results are represented with four matrices of $C_{IA}, C_{AI}, C_{II}, C_{AA}$ which are described in $C_{k_1 k_2}$.

$$C_{k_1k_2} = [Cor(d_i^{k_1}, d_j^{k_2})] \quad (i, j \in 1,\ldots,N_d; k_1, k_2 \text{ stand for I or A}).$$

Fig. 2(a) illustrates the initial propagation model, in which each torus implies a multi-modal document composed either of "solid" image nodes or "hollow" auditory nodes, while a dotted line connects two nodes of different modalities with the length representing their inter correlation, and a solid line connects two nodes of the same modality with its length representing their intra similarity.

Suppose we need categorize a new input image or audio clip d_{new}^*. We first infer the aspect distribution of $P(z_k \mid d_{new}^*)$ using the folding-in method and learned the model parameters. Then, for each multi-modal document D_c, we calculate the correlation between d_{new}^* and each d_i from D_c by (11). As a result, we construct a new correlation model for d_{new} and the training data set. Next, inspired by [5], an intra similarity edge $Cor(d_{new}^X, d_i^X)$ or an inter correlation edge $Cor(d_{new}^X, d_j^Y)$ is updated by

$$\begin{aligned} Cor^{'}(d_{new}^X, d_i^X) &= \alpha Cor(d_{new}^X, d_i^X) + \\ &\quad \beta \sum_m \sum_n (Cor(d_{new}^X, d_m^Y) * C_{YY}(d_m^Y, d_n^Y) * C_{YX}(d_n^Y, d_i^X)) \\ & if\ Cor(d_{new}^X, d_m^Y) > \varepsilon_{XY}, C_{YY}(d_m^Y, d_n^Y) > \varepsilon_{YY}, C_{YX}(d_n^Y, d_i^X) > \varepsilon_{YX} \end{aligned} \tag{12}$$

$$\begin{aligned} Cor^{'}(d_{new}^X, d_i^Y) &= \alpha Cor(d_{new}^X, d_i^Y) + \\ &\quad \beta \sum_m \sum_n Cor(d_{new}^X, d_m^X) * C_{XY}(d_m^X, d_m^Y) * C_{YY}(d_m^Y, d_i^Y)) \\ & if\ Cor(d_{new}^X, d_m^X) > \varepsilon_{XX}, C_{XY}(d_m^X, d_m^Y) > \varepsilon_{XY}, C_{YY}(d_m^Y, d_i^Y) > \varepsilon_{YY} \end{aligned} \tag{13}$$

where α and β are propagation coefficients, controlling the weight factor of the initial correlation and the propagation, respectively. Note that a propagation path will be terminated in advance once an edge connecting two nodes $d_i^{k_1}$ and $d_j^{k_2}$ in (12) or (13) is less than the following correlation threshold, so as to be named as a *weak* correlation edge:

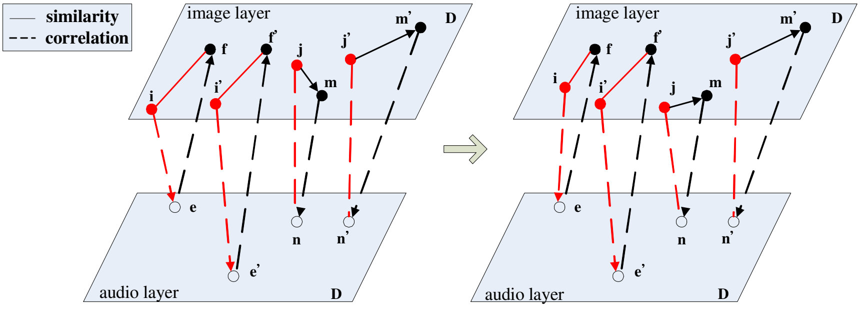
$$\varepsilon_{k_1k_2} = \frac{\sum_{i=1}^{N_d} \sum_{j \in P} Cor(d_i^{k_1}, d_j^{k_2})}{\sum_i^{N_d} N_i} = \frac{\sum_{i=1}^{N_d} \sum_{j \in P} C_{k_1k_2}(i,j)}{\sum_i^{N_d} N_i}, (k_1, k_2 = \text{I or A}) \tag{14}$$

where P is the shortest edge set connected with $d_i^{k_1}$ of size N_i. Inversely, an edge has a correlation value larger than (14) is named as a *strong* correlation one.

Finally, we decide the category of d_{new}^* as follows:

$$\begin{gathered} d_{new}^* \in \{category\ c \mid Max(Aver(D_c))\}, \\ Aver(D_c) = (\sum_{d_x^I \in D_c} Cor^{new}(d_{new}^*, d_x^I) + \sum_{d_y^A \in D_c} Cor^{new}(d_{new}^*, d_y^A)) / (\sum_{d_x^I \in D_c} \sum_{d_y^A \in D_c} 1). \end{gathered} \tag{15}$$

Fig. 3 illustrates some multi-modal propagation examples. In Fig. 3(a), new images d_i^I, $d_{i'}^I$, d_j^I and $d_{j'}^I$ are added into the existing multi-modal documents of (d_f^I, d_e^A), $(d_{f'}^I, d_{e'}^A)$, (d_m^I, d_n^A) and $(d_{m'}^I, d_{n'}^A)$, respectively. Suppose $Cor_{II}(d_i^I, d_f^I)$ is a weak correlation while $Cor_{IA}(d_i^I, d_e^A)$ and $Cor_{AI}(d_e^A, d_f^I)$ are strong ones, the final correlation between d_i^I and d_f^I can be reinforced after multi-modal correlation propagation with (14) (see the shortened edge of $Cor_{II}^{'}(d_i^I, d_f^I)$ in Fig. 3(b)). Similarly, propagation results for correlations between other nodes are also illustrated in Fig. 3.



(a) (b)

Fig. 3. Examples of multi-modal correlation propagation

4 Experiments and Discussions

Our experiment data set consists of 10 Image-Audio categories, each containing 100 auditory clips collected from movies and 150 images from the Corel image dataset and internet. For each category, we select 200 training multi-modal samples and 50 test ones. Every image and auditory clip in the same category comes from the same source and is used as the ground truth for our multi-modal analysis. The selected visual feature is a 128-dimentional Scale-Invariant Feature Transform (SIFT) [16] descriptor, while the auditory feature is 21-dimensional Mel-Frequency Cepstral Coefficients (MFCC) [17] descriptor.

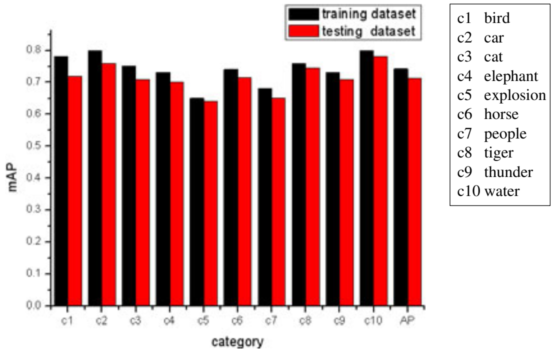


Fig. 4. Multi-modal retrieval using MMIP

In the first experiment, we randomly select 100 samples from the training set and test set to retrieve their most similar ones, respectively. The average results are shown in Fig. 4 (with 10-fold cross validation). It is encouraging that when the number of the returned images is 50, the mean average precision (mAP) averagely reaches 65%-78% for multi-modal retrieval. Fig. 4 also shows the effectiveness of the continuous PLSA modeling.

Table 1. Performance evaluation of different models (MMIP[1]: before multi-modal propagation; MMIP[2]: after multi-modal propagation)

		PLSA[13]	MMIP^{1*}	[5]	MMIP^{2*}
$d_i^I \to d_j^I$	*path*	$d_i^I \to d_j^I$	$d_i^I \to D_c$	$d_i^I \to ..., d_k^*, ... \to d_j^I$	
	mAP	0.62	0.70	0.58	0.75
$d_i^I \to d_j^A$	*path*	$d_i^I \to d_j^A$	$d_i^I \to D_c$	$d_i^I \to ..., d_k^*, ... \to d_j^A$	
	mAP	-	0.67	0.61	0.735
$d_i^A \to d_j^A$	*path*	$d_i^A \to d_j^A$	$d_i^A \to D_c$	$d_i^A \to ..., d_k^*, ... \to d_j^A$	
	mAP	0.65	0.71	0.55	0.76
$d_i^A \to d_j^I$	*path*	$d_i^A \to d_j^I$	$d_i^A \to D_c$	$d_i^A \to ..., d_k^*, ... \to d_j^I$	
	mAP	-	0.69	0.53	0.72

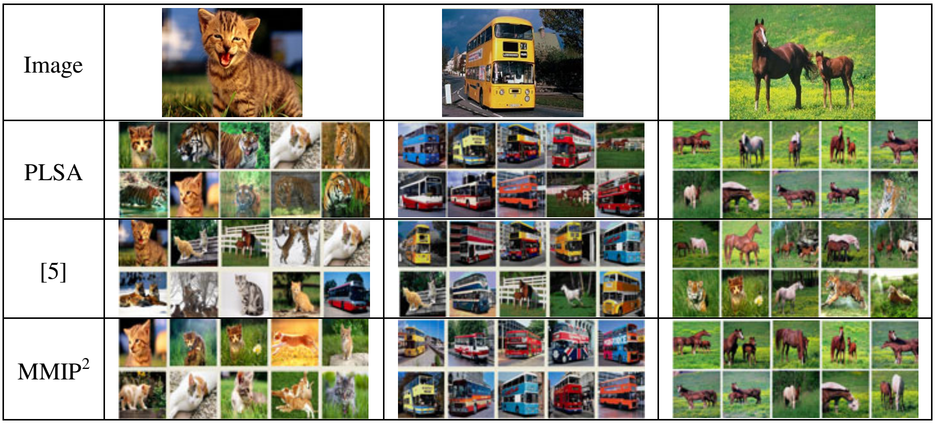


Fig. 5. Examples of the comparison results by different correlation methods

Next, we compare our model with PLSA-WORDS [13] and the correlation learning [5]. The results are shown in Table. 1. PLSA-WORDS is actually a discrete cross media model for automatic image annotation, without correlation propagations between multi-modalities. Therefore, PLSA-WORDS does not work well in image-audio retrieval in our experiments. Table. 1 also shows that the mean retrieval accuracy of MMIP after multi-modal propagation is nearly 17% higher than [5], showing that multi-modal data can be better correlated in the higher level aspect space than the low level feature space. Fig. 5 shows some examples of the comparison results.

Finally, Fig. 6 shows the mAP values for different numbers of latent aspects. On our training data set, $z = 50$ provides a balanced performance between the accracy and computational cost in the testing stage.

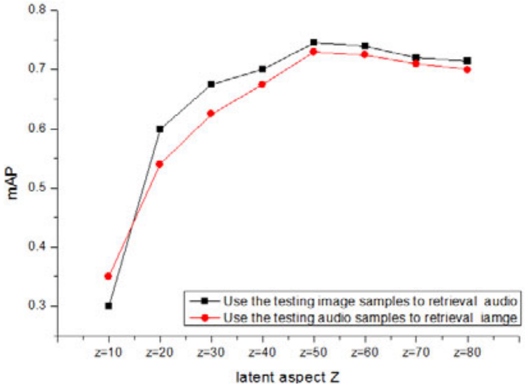


Fig. 6. Influence of different numbers of latent aspect *z* in the proposed model

5 Conclusion

This paper proposes a novel MMIP model for cross-media retrieval. Experimental results illustrate that our model improves multi-modal retrieval performance and e ffectively reduce information loss between feature-instance versus real-world matching. Further work includes improvement of the efficiency for large scale image-audio data set, test more low-level feature combinations, and give the benchmark for multimodal retrieval.

Acknowledgments. The work described in this paper was supported by the Natural Science Foundation of China under Grant No. 60723003 and 61003113, the 973 Program of China under Grant No. 2010CB327903, and the Natural Science Foundation of JiangSu under Grant No BK2009082.

References

1. Yu, B., Ma, W.Y., Nahrstedt, K., Zhang, H.J.: Video Summarization Based on User Log Enhanced Link Analysis. ACM Multimedia, 382–391 (2003)
2. Feng, S.L., Manmatha, R., Lavrenko, V.: Multiple Bernoulli Relevance Models for Image and Video Annotation. In: Proc. IEEE CVPR, vol. 2, pp. 1002–1009 (2004)
3. Datta, R., Li, J., Wang, J.Z.: Content-Based Image Retrieval - Approaches and Trends of the New Age. In: Proceedings of the 7th ACM SIGMM International Workshop on Multimedia Information Retrieval, Singapore, pp. 253–262 (2005)

4. Chang, E., Goh, K., Sychay, G., Wu, G.: CBSA: Content-Based Soft Annotation for Multimodal Image Retrieval Using Bayes Point Machines. IEEE Trans. on Circuits and Systems for Video Technology 13, 26–38 (2003)
5. Zhang, H., Zhuang, Y.T., Wu, F.: Cross-Modal Correlation Learning for Clustering on Image-Audio Dataset. ACM Multimedia, 273–276 (2007)
6. Beal, M.J., Attias, H., Jojic, N.: Audio-Video Sensor Fusion with Probabilistic Graphical Models. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2350, pp. 736–750. Springer, Heidelberg (2002)
7. Zhuang, Y.T., Yang, Y., Wu, F.: Mining Semantic Correlation of Heterogeneous Multimedia Data for Cross-Media Retrieval. IEEE Trans. on Multimedia 10, 221–229 (2008)
8. Wang, J.D., Zeng, H.J., Zheng, C., Lu, H.J., Li, T., Ma, W.Y.: ReCoM: Reinforcement Clustering of Multi-Type Interrelated Data Objects. In: ACM SIGIR, Canada, pp. 274–281 (2003)
9. Wang, X.J., Ma, W.Y., Xue, G.R., Li, X.: Multi-Model Similarity Propagation and its Application for Web Image Retrieval. ACM Multimedia, 944–951 (2004)
10. Yang, Y., Zhuang, Y.T., Wu, F., Pan, Y.H.: Harmonizing Hierarchical Manifolds for Multimedia Document Semantics Understanding and Cross-media Retrieval. IEEE Transactions on Multimedia 10, 437–446 (2008)
11. Blei, D.M., Jordan, M.I.: Modeling Annotated Data. In: Proc. ACM SIGIR, Toronto, Canada, pp. 127–134 (2003)
12. Barnard, K., Duygulu, P., Freitas, N.D., Forsyth, D., Blei, D.M., Jordan, M.I.: Matching Words and Pictures. J. Machine Learning Research 3, 1107–1135 (2003)
13. Monay, F., Perez, D.G.: Modeling Semantic Aspects for Cross-Media Image Indexing. IEEE Trans. on PAMI 29, 1802–1817 (2007)
14. Li, Z.X., Shi, Z.P., Liu, X., Shi, Z.Z.: Automatic Image Annotation with Continuous PLSA. In: Proceedings of ICASSP, pp. 806–809 (2010)
15. Hofmann, T.: Unsupervised Learning by Probabilistic Latent Semantic Analysis. In: Proceedings of Machine Learning, vol. 42, pp. 117–196 (2001)
16. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60, 91–110 (2004)
17. Foote, J.: Content-Based Retrieval of Music and Audio. In: Multimedia Storage and Archiving Systems II, Proc. of SPIE, vol. 3229, pp. 138–147 (1997)
18. Jiang, W., Cotton, C., Chang, S.F., Ellis, D., Loui, A.C.: Short-Term Audio-Visual Atoms for Generic Video Concept Classification. ACM Multimedia, 5–14 (2009)

RGB-D Based Multi-attribute People Search in Intelligent Visual Surveillance

Wu Liu[1,2], Tian Xia[1], Ji Wan[1,2], Yongdong Zhang[1], and Jintao Li[1]

[1] Institute of Computing Technology, Chinese Academy of Science,
Beijing 100190, China
[2] Graduate School of the Chinese Academy of Science, Beijing 100039, China
{liuwu,txia,wanji,zhyd,jtli}@ict.ac.cn

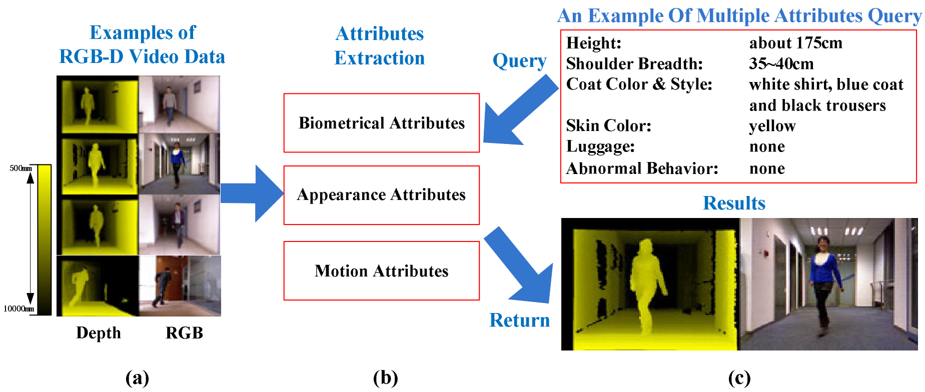
Abstract. Searching people in surveillance videos is a typical task in intelligent visual surveillance (IVS). However, current IVS techniques can hardly handle multi-attribute queries, which is a natural way of finding people in real-world. The challenges arise from the extraction of multiple attributes which largely suffer from illumination change, shadow and complicated background in the real-world surveillance environments. In this paper, we investigate how these challenges can be addressed when IVS is equipped with RGB-D information obtained by an RGB-D camera. With the RGB-D information, we propose methods that accurately and robustly segment human region and extract three groups of attributes including biometrical attributes, appearance attributes and motion attributes. Furthermore, we introduce a novel IVS system which is capable of handling multi-attribute queries for searching people in surveillance videos. Experimental evaluations demonstrate the effectiveness of the proposed method and system, and also the promising applications of bringing RGB-D information into IVS.

Keywords: IVS, RGB-D, Multi-Attribute Query, People Search.

1 Introduction

Over the last decade we have witnessed an explosive growth of surveillance video data. This drives the birth of intelligent visual surveillance (IVS), which mainly aims at automatically detecting targets in surveillance video via computer vision techniques [1]. Searching people in a long surveillance video is a typical task in IVS, and a natural way of finding people in real-world is through a multi-attribute based query. For example, finding the person whose height is between 175cm and 180cm, with white skin, green T-shirt, blue shorts and a big black luggage, running across the hall. Unfortunately, existing IVS techniques can hardly handle the above multi-attribute query due to the difficulties in the extraction of multiple attributes. The bottleneck is mainly derived from environment modeling and object segmentation in IVS, which lay the foundation of IVS but largely suffer from illumination change, reflection, shadow and complicated background in the real-world surveillance environments [1].

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 750–760, 2012.



Height:	about 175cm
Shoulder Breadth:	35~40cm
Coat Color & Style:	white shirt, blue coat and black trousers
Skin Color:	yellow
Luggage:	none
Abnormal Behavior:	none

Fig. 1. An example of our IVS system which is capable of handling multi-attribute queries for searching people in surveillance video. (a) shows some example videos in our RGB-D database which is captured by a RGB-D camera. (b) lists the multiple attributes extracted from the RGB-D video data. (c) illustrates an example of multiple attributes query for people search in surveillance video and the retrieval result . (Best seen in color)

In this paper, we focus on the extraction of multiple attributes, and build an IVS system which is capable of handling the above multi-attribute query for people search in surveillance video. The improvement is largely derived from the usage of RGB-D information.

RGB-D is a key terminology in RGB-D project [2] whose goal is to develop techniques enabling future use cases of depth cameras (we name it RGB-D camera). RGB-D camera can capture RGB image along with its per-pixel depth information, as shown in Fig. 1. The additional depth information brings an opportunity to address a range of challenging issues. For example, one work of RGB-D project [3] is to use RGB-D information in robotic manipulation and interaction, and Microsoft Kinect [4] used it in human-computer interaction. RGB-D information also brings great opportunities to IVS, particularly for the challenging issues of object segmentation. The discontinuity of depth information can be utilized to perform nearly perfect object segmentation which avoids environment modeling and is invariant to illumination changes. Moreover, RGB-D information is helpful for us to get the 3D information of the scene, and makes it possible to extract more useful attributes which are provide powerful support to multi-attribute query for people search.

In order to investigate and verify the benefits of bringing RGB-D into IVS, we extract three groups of attributes, i.e., biometrical attributes including height and shoulder breadth; appearance attributes including skin color, clothing color and style, and luggage; motion attributes including squatting, running and wandering. These attributes are widely used in describing a person in real-world people finding. Based on these attributes, a novel IVS system with RGB-D camera is built to provide an effective way of finding people via multi-attribute query. Some searching results of multi-attribute queries from our IVS system are illustrated in Fig. 1.

The previous arts we know in searching people via visual attributes are [5] and [6]. [5] mainly focuses on facial and clothing color attributes in surveillance environment.

Although facial information is very useful in personal identification, its reliability is limited by the requirement of people's close shot, which is only feasible in some particular surveillance environment. [6] proposes an approach for ranking and retrieval of images based on multi-attribute queries. Since it is for static portrait data, its attributes are quite different from ours for surveillance video. [7] also proposes a multiple attributes matching method for video retrieval, without introducing the method of multi-attribute extraction.

Our contributions are summarized as follows:

(1) Taking the advantages of RGB-D information in human segmentation and 3D scene establishment, multiple attributes facilitating for finding people in surveillance environments are proposed. And their robust extraction is implemented effectively and efficiently, which demonstrates the validity of bringing RGB-D information into IVS.

(2) A novel IVS system with RGB-D camera is established, which provides an effective way of searching people via multi-attribute query, following the natural way of finding people in real-world.

2 System Overview

In this section, we present an overview of the proposed framework for people searching in surveillance video with RGB-D information. As shown in Fig. 2, the framework consists of four components: RGB-D camera, analytic engine, data storage and user search interface.

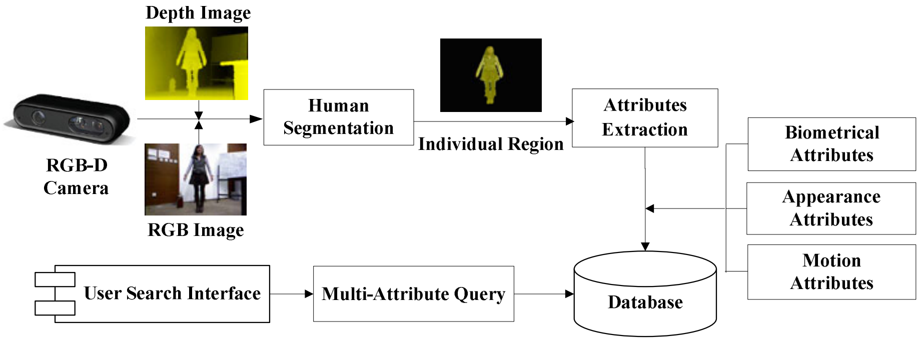


Fig. 2. The overview of our system architecture

2.1 RGB-D Camera

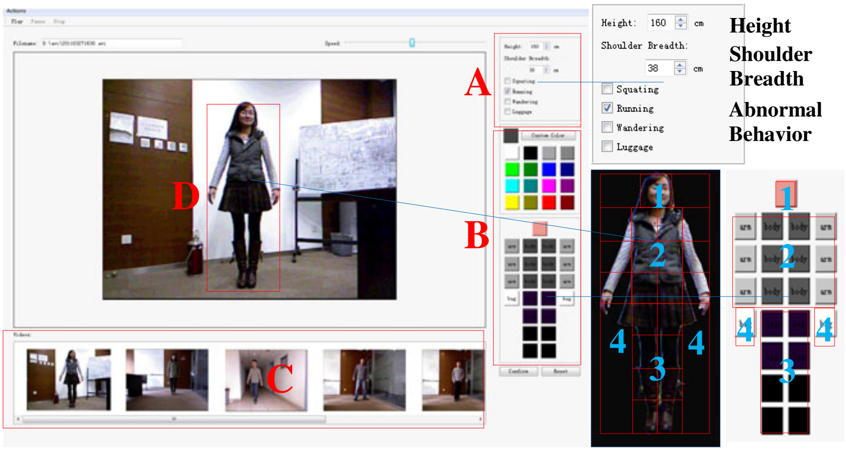
In our IVS system, the RGB-D camera uses Primesense [8] depth camera technology, it simultaneously captures both RGB and depth images at 640× 480 resolution with 30 fps. Its field of view is 58° H, 45° V and 70° D for horizontal, vertical and diagonal perspective separately. Using OpenNI [9], we can calibrate the RGB and depth images.

2.2 Attributes Extraction

This module performs all the computer vision analysis in IVS system, and consists of three steps: (1) segmenting each individual region from the background; (2) extracting biometrical and appearance attributes for each individual including height, shoulder breadth, skin color, clothing color and style, luggage; (3) extracting motion attributes by detecting the abnormal behaviors including squatting, running and wandering. The details of this module are presented in Section 3.

2.3 Data Storage and User Search Interface

The extracted attributes are indexed into a relational database to facilitate efficient multi-attribute search. A novel user search interface is also provided, where users can input multi-attribute queries. As shown in Fig. 3, region A and B are for inputting multi-attribute queries. In region A, users can express their desired attributes concerning height, shoulder breadth and abnormal behaviors. In region B, users can express their desired attributes concerning skin color, luggage, clothing color and style by two operations, selecting color and filling in the human template. Region C shows the search results and region D is a playback window for surveillance videos.



Height: 160 cm
Shoulder Breadth: 38 cm
□ Squating
☑ Running
□ Wandering
□ Luggage

1. skin color; 2. coat color and style; 3. trousers color and style; 4. luggage.

Fig. 3. The user search interface of our system. Region A and B are for inputting multi-attribute queries. Region C shows the search results and region D is a playback window for surveillance video. (Best seen in color)

3 Attributes Extraction

3.1 Human Segmentation

Our human segmentation is to take advantages of the RGB and depth information to obtain individual region as shown in Fig. 2. First, connected motion regions are obtained based on RGB-D information; then, the refined individual region is identified from these regions. The details are shown as below.

Supposing there are n persons in frame I and we define $I = B \cup \{P_i\}_{i=1}^{n} \cup NP$ where B is the background, P_i is the individual region of human i, and NP is the remaining region apart from background and human, such as some moving objects.

Let $V_{(x,y)}(c,d)$ denotes the RGB-D information of the pixel at (x, y) in frame I, c denotes the RGB values and d denotes the depth value. We can simply remove the majority of background by setting the threshold $d_{\min}$ and $d_{\max}$ according to real monitoring conditions and only considering the region within $d_{\min}$ and $d_{\max}$. Then we use the Split-and-Merge strategy to slice the foreground into depth connected regions R_i. Extended temporal differencing method considering color and depth changes is utilized to obtain the connected motion regions R_i'.

As $\{P_i\}_{i=1}^{n} \cup NP \in R_i'$, in order to remove *NP*, the actual area measure S_r of each R_i' in real world is calculated by Eq. (1)

$$S_r = (S_p / S) \times 4d^2 \times \tan(\frac{hor}{2}) \times \tan(\frac{ver}{2}) \quad , \tag{1}$$

where S_p is the number of pixels in region R_i' and S is the total number of pixels in depth image. d is the average depth of region R_i', hor and ver are the RGB-D camera's horizontal and vertical field of view. We suppose the horizontal bisecting line of the camera image is horizontal in the real world as well. Then we can remove some regions which are too larger or smaller than real human area with S_r value.

In order to remove the remaining NP whose actual area is similar with real human, a cascade of Adaboost classifier based on Haar-like features is used to detect the head and a Bayes color model [11] is used to detect skin in the region R_i'. The detected regions are considered as seed location to refine the whole body contours from the region R_i'.

In order to demonstrate that RGB-D information can enhance the performance of human segmentation, we conduct comparison to background subtraction method which uses EM and GMM models [1] to estimate the background from RGB images. We randomly select 10 sequences from the testing set, and find that our algorithm remarkably outperforms others on all the data. Due to the length limit, we only illustrate the results from one testing sequence in Fig. 4.

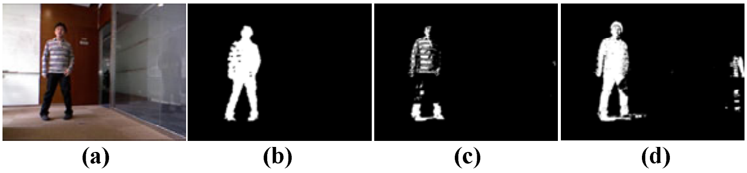


Fig. 4. Performance comparison of human segmentation: (a) original image; (b) result of our algorithm; (c) result of EM based algorithm; (d) result of GMM based algorithm

3.2 Details of Attributes Extraction

Biometrical Attributes. Intuitively, height and shoulder breadth are typical attributes in discriminating different people in surveillance system, which are also noted in some previous works [13, 14, 15]. However, existing height estimation work mainly relies on conventional RGB cameras, and the estimation is not robust to illumination changes. Taking the advantages of RGB-D information in 3D scene establishment, we propose a robust implementation to estimate height and shoulder breadth, and the details are shown as below.

Measuring points are important for height and shoulder breadth measurement. As shown in Fig. 5, we define four measuring points including V_{hh} and V_{hl} as the highest and lowest points for height measuring, V_{sl} and V_{sr} as the leftmost and rightmost points for shoulder breadth measuring. The measurement consists of two steps: detecting the measuring points in individual region and obtaining their location information in real world coordinate.

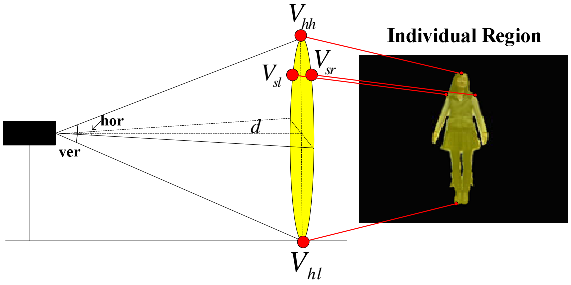


Fig. 5. Automatically measuring height and shoulder breadth

The four points are initialized at the intersection points of bounding rectangle and individual region. The situation of rectification is twofold: having head location or not. The head location is detected by a cascade of Adaboost classifier based on Haar-like features. As only the individual's silhouettes are fed into the classifier, the precision is guaranteed to be high. If head location is detected, V_{hh} will be the highest point of head region. Moving down from the head region, we can get the left shoulder V_{sl} and right shoulder V_{sr}. If there is no head detected, we calculate the depth distribution around V_{hh} to validate whether this point is on the head or not. V_{sl} and V_{sr} are located at the endpoint of the ellipse's horizontal axis, which can be got by the ellipse fitting of trunk. The four points' coordinate figure in pixel coordinates can be reconstructed in real world coordinates through Eq.(2),

$$\begin{cases} x_w = (x_p / X_{res} - \frac{1}{2}) \times d \times \tan(\frac{hor}{2}) \times 2 \\ y_w = (y_p / Y_{res} - \frac{1}{2}) \times d \times \tan(\frac{ver}{2}) \times 2 \end{cases} . \quad (2)$$

Here (x_p, y_p) and (x_w, y_w) are the location of V in pixel coordinate and real world coordinate, X_{res} and Y_{res} are the image width and height resolution, hor and ver are the horizontal and vertical field of view. We suppose the horizontal bisecting line of the camera image is horizontal in the real world as well. The height and shoulder breadth are calculated by $\overline{V}_{hl} - \overline{V}_{hh}$ and $\overline{V}_{sr} - \overline{V}_{sl}$ in world coordinates. $\overline{V}$ is the smoothing value of 5*5 neighborhood of V in individual region.

Appearance Attributes. The appearance attributes of an individual represent the identity of the person [16]. Clothing color and style, skin color and luggage are the most significant attributes for people tracking and identification, as they usually occupy large area in monitor screen and thus are more reliable [5]. The main challenges of extracting appearance attributes are derived from the changes in illumination, pose, and clothing appearance variation [16]. Considering the clothing style is no-rigid deformation with gestures, we just detect it when the people are facing the camera front and standing uprightly.

To compensate the influence caused by the illumination changes, the color of clothing region is transferred from RGB to HSV first. Then the HSV color space is quantified to 24 bins. The quantification method is that the H value is quantified into 6 bins, the S and V values are quantified into 2 bins each. The HSV color histogram is computed to represent the color information.

When two people are dressed up differently but with roughly the same amount of body surface covered with the same colors, they are likely to have similar histogram-based signatures, regardless of how the colors are distributed in space. This is a major limitation of all the holistic models based on histograms because it significantly reduces their discriminability. This issue is addressed here by a self-adapting blocks model covering the whole human body. As shown in Fig. 6, the individual region is divided into 23 blocks covering skin, coat, trousers and luggage. The self-adapting blocks model is structured as below:

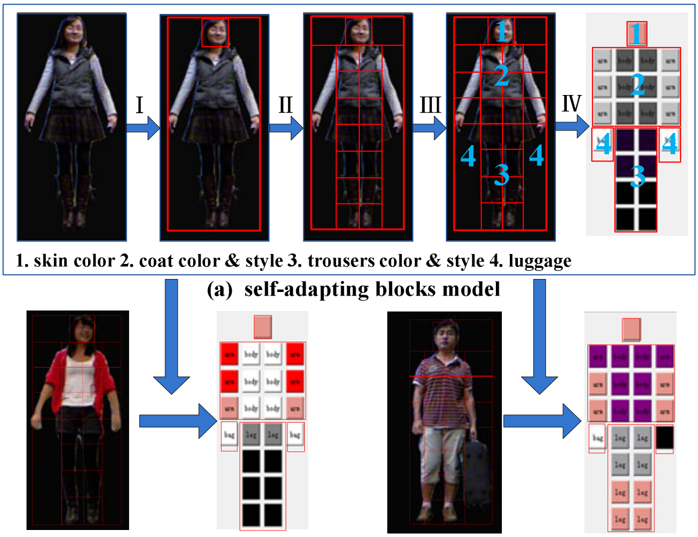
1. The external rectangle, individual region and head region have been detected in section 3.1. The head region is considered as one block and the skin color can be obtained by performing a Bayes color model [11] on this block.
2. The body trunk and leg region are detected under the head region and its width equals to shoulder breadth measured before. According to the theory of anthropometry [17], we can divide body trunk region into 14 equal blocks, the top 6 blocks belong to trunk and the button 8 blocks belong to leg.
3. The left and right regions abut against trunk are considered as arms and we divide them into 6 blocks. The two blocks under arms are considered as luggage regions.
4. For each coat, trousers and luggage blocks, we compute the HSV color histogram $H_j(k)$, where $j = 0,\ldots,21$ is block id and $k = 0,\ldots,23$ is the index of histogram bins. The clothing color and style is described by the combination of these blocks. Some examples are shown in Fig. 6.

All blocks' color histograms are indexed into the database. When users want to find people with these attributes, they can choose the color of each block by our interface. If no color is selected for a block, its weight is set to zero. Let $H'_j(k), j = 0,\ldots,22$,

$k = 0,\ldots,23$ is the query condition entered by users. For each people in the database, system will calculate the matching score by Eq.(3),

$$score = (\sum_{j=0}^{22} w_j \sum_{k=0}^{23} \min(H_j(k), H'_j(k))), \quad . \tag{3}$$

Here w_j is the weight of the block j which can be designated by users and $\sum_{j=0}^{22} w_j = 1$. We set the weights of body blocks twice the weights of other blocks by default as the body blocks' color histograms are more significant than others.


I
II
III
IV
1
2
3
4
4
1. skin color 2. coat color & style 3. trousers color & style 4. luggage

(a) self-adapting blocks model

(b) appearance attributes extraction examples

Fig. 6. Appearance attributes extraction. (a) shows the structured of self-adapting blocks model. (b) shows the appearance attributes extracted from two examples which use self-adapting blocks model. (Best seen in color)

Motion Attributes. Motion attributes are related to abnormal behaviors which are mostly concerned by users in visual surveillance [1]. Based on the multiple attributes extracted before, we detect three simple abnormal behaviors, i.e., wandering, running and squatting. The detection is performed as following. Firstly, the biometrical and appearance attributes extracted before are used to track human P_i. Then the stay time T_i of each P_i can be counted, which is used for wandering detection. Meanwhile, the gravity center $G_i(x, y, d, c)$ of each P_i is used to calculate the moving distance D_i in the real world coordinates. So the velocity of P_i is calculated by $S_i = D_i / T_i$, which is used for running detection. At last, the remarkable height changes are believed to be squatting, such as when $ht < \frac{1}{2}\overline{ht}$ suddenly, $\overline{ht}$ is the average height of P_i.

4 Experimental Results

In this section, we comprehensively evaluate the performance of attributes extraction and multi-attribute searching of people. The testing set is captured by a RGB-D camera (also known as Kinect-style camera): it consists of 100 RGB-D video surveillance sequences, captured at three different scenes including meeting room, corridor and entrance, with 50 different persons appearing in it, as shown in Fig. 1. All the people have height distribution of 150~185cm and shoulder breadth distribution of 35~55cm; wearing clothes with different styles and different colors; conduct actions including standing, squatting, jumping, walking, running, rotating and waving hands.

4.1 Performance of Attributes Extraction

Biometrical Attributes. Our system automatically measures the height and shoulder breadth of 38 people from the testing set, whose actual values have been measured manually. The results are shown in Fig. 7. The bias of height is -1.21 cm and standard deviation is 3.18 cm. The negative bias is due to when people standing in normal state, his height is generally lower than standing upright. The bias of shoulder breadth is -0.11 cm and standard deviation is 5.28 cm. The relatively large deviation is due to the interference brought by various poses of people.

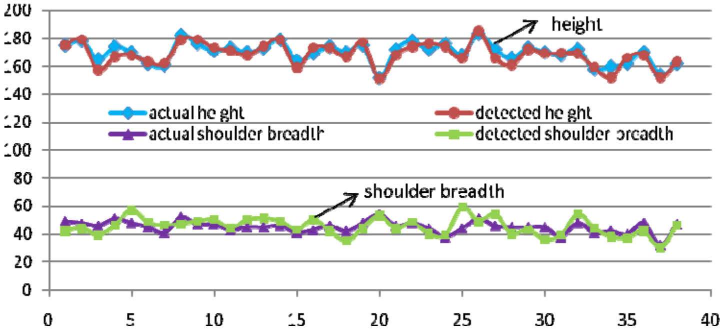


Fig. 7. Performance of height and shoulder breadth measurement. The X-axis is the people id and Y-axis is length unit in cm.

Motion Attributes. The ground-truth of three abnormal behaviors in testing set is labeled manually, and the results are shown in Table.1. Benefitting from the advantages of RGB-D information, wandering detection achieves satisfying performance, running and squatting detection can also meet searching requirements. The reason of the recognition rate of squatting and running is relatively lower than wandering is that the diversity of conduct actions may influence the obtaining of velocity and height changes.

Table 1. Performance of abnormal behavior detection

Abnormal behavior	Recall	Precision
Wandering	0.95	1
Squatting	0.94	0.738
Running	0.86	0.8

4.2 Evaluation of Multi-attribute People Searching

In order to evaluate the effectiveness of multi-attribute based people searching in our IVS system, we conduct a known-person search task which models the situation in which a user has seen a person in the testing set before, but doesn't know where to find it now. We invited 5 users, each user is required to search 10 different known-persons (seen before the searching by users for only once), therefore there are 50 known-person search tasks in all, and each task can be conducted in three rounds. From Fig. 8, we can find: (1) 28 tasks are completed successfully in the first round by appearance attributes including clothing color and style, skin color and luggage; by adding biometrical attributes, the number rises to 37; when all the attributes are used, the number rises to 40; (2) the number of hits rises if the user searches more than one time.

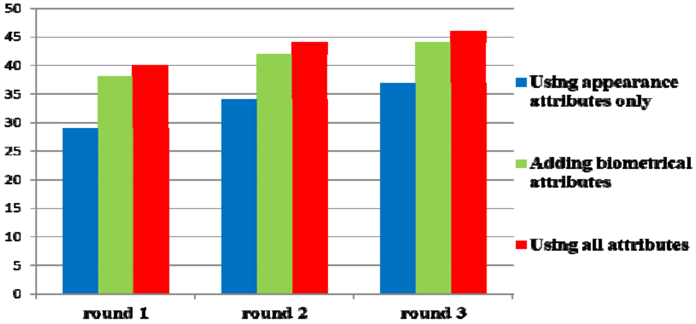


Fig. 8. Performance of 50 known-person search tasks conducted on our system by multi-attribute searching. The X-axis is search times and Y-axis is the number of hits.

5 Conclusions and Future Work

We innovatively bring RGB-D information into IVS system to assist multi-attribute based people searching. Taking the advantages of RGB-D information in human segmentation and 3D scene establishment, we have extracted biometrical attributes including height and shoulder breadth; appearance attributes including clothing color and style, skin color and luggage information; and also motion attributes including the detection of squatting, running and wandering. The comprehensive evaluations on our IVS system demonstrate the effectiveness of the extracted multiple attributes and their successful application in multi-attribute based people searching.

Our attempts in this paper indicate that RGB-D information has promising potential in IVS field. In the future, we will investigate how to further combine RGB and Depth information in a multiple graph framework [18][19] to discover more useful and challenging attributes in surveillance environments.

Acknowledgments. This work is supported by the National High Technology and Research Development Program of China (863 Program, 2009AA01A403); National Nature Science Foundation of China (61100087); Beijing Natural Science Foundation (4112055); Beijing New Star Project on Science & Technology (2007B071); Co-building Program of Beijing Municipal Education Commission.

References

1. Hu, W.M., Tan, T.N., Wang, L., Maybank, S.: A Survey on Visual Surveillance of Object Motion and Behaviors. IEEE Trans. on SMC (2004)
2. RGBD Projects, `http://ils.intel-research.net/projects/rgb`
3. Henry, P., Krainin, M., Herbst, E., Ren, X.F., Fox, D.R.-D.: Mapping: Using Depth Cameras for Dense 3D Modeling of Indoor Environments. In: ISER (2010)
4. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-Time Human Pose Recognition in Parts from a Single Depth Image. In: CVPR (2011)
5. Vaquero, D., Feris, R., Tran, D., Brown, L., Hampapur, A., Turk, M.: Attribute-Based People Search in Surveillance Environments. In: WACV (2009)
6. Siddiquie, B., Feris, R.S., Davis, L.S.: Image Ranking and Retrieval based on Multi-Attribute Queries. In: CVPR (2011)
7. Lin, C.H., Chen, A.L.P.: Indexing and Matching Multiple-Attribute Strings for Efficient Multimedia Query Processing. IEEE Trans. on Multimedia 8(2), 408–411 (2006)
8. PrimeSense, `http://www.primesense.com/`
9. OpenNI, `http://www.openni.org/`
10. Xia, L., Chen, C.C., Aggarwal, J.K.: Human Detection Using Depth Information by Kinect. In: International Workshop on Human Activity Understanding from 3D Data in conjunction with CVPR, HAU3D (2011)
11. Jones, M.J., Rehg, J.M.: Statistical Color Models with Application to Skin Detection. IJCV 46(1), 81–96 (2002)
12. Lai, K., Bo, L.F., Ren, X., Fox, D.: A Large-Scale Hierarchical Multi-View RGB-D Object Dataset. In: ICRA (2011)
13. Kispal, I., Jeges, E.: Human Height Estimation Using a Calibrated Camera. In: CVPR (2008)
14. Gallagher, A., Chen, T.: Jointly Estimating Demographics and Height with a Calibrated Camera. In: ICCV (2009)
15. Madden, C., Piccardi, M.: Height Measurement as a Session-based Biometric for People Matching Across Disjoint Camera Views. In: IAPR (2005)
16. Doretto, G., Sebastian, T., Tu, P., Rittscher, J.: Appearance-based Person Reidentification in Camera Networks: Problem Overview and Current Approaches. Journal of Ambient Intelligence and Humanized Computing 2(2), 127–151 (2010)
17. Anthropometry, `http://personal.cityu.edu.hk/~meachan/Online%20Anthropometry`
18. Wang, M., Hua, X.S., Hong, R.C., Tang, J.H., Qi, G.J., Song, Y.: Unified Video Annotation via Multi-Graph Learning. IEEE Trans. on Circuits and Systems for Video Technology 19(5) (2009)
19. Xia, T., Tao, D.C., Mei, T., Zhang, Y.D.: MultiView Spectral Embedding. IEEE Trans. on Systems, Man, and Cybernetics: Part B 40(6), 1438–1446 (2010)

Template Matching and Monte Carlo Markova Chain for People Counting under Occlusions

Jun-Wei Hsieh, Fu-Jiang Fang, Guo-Jin Lin, and Yu-Shi Wang

Dept. of Computer Science and Information Engineering,
National Taiwan Ocean University, Taiwan, R.O.C.
{shieh,m98570029,19957001}@ntou.edu.tw

Abstract. It is challenging to count and analyze people in crowds due to the changes of lighting, occlusions, shadows, backgrounds, and weather conditions. Especially for the occlusion problem, until now, it is still ill-posed. To deal with the occlusion problem, the MCMC (Monte Carlo Markova Chain) scheme is used in this paper to estimate all possible pedestrian positions across different frames. However, it requires good initial head positions for parameter searching and people counting. Thus, an intelligent head-shoulder-region detector is then developed for detecting all possible pedestrian candidates from videos. One key problem in head-shoulder detection is that the feature contrast between the objects and their background should be larger. To tackle this problem, a Linear Discriminant Analysis (LDA) approach is then used to enhance the boundaries between objects and features. Three contributions are made in this paper: (1) Intelligent head-shoulder-region detector; (2) People detection under occlusions; (3) Integrated people counting system using LDA. Experimental results have proved the superiorities of the proposed method in people detection and counting.

Keywords: MCMC, LDA, Template Matching, People Counting.

1 Introduction

Counting people is an important and challenging problem in video surveillance. It can be used in various public places like shopping malls, public transport stations, theaters, department stores, or trade fairs for security issues, and resource allocation. The problem is challenging due to the occlusions between persons, variations between human shapes, lighting changes, camera motions, and various clothing styles in appearance. During the past few decades, extensive studies [1]-[6] have been conducted on people counting directly from videos. For example, Rabaud and Belongie [1] used a KLT tracker to track corner features and obtained their trajectories from which different objects were clustered and counted using local rigidity constraints and a simple object model. Antonini and Thiran [2] developed a visual surveillance system that models and counts humans using a trajectory clustering technique. Wu and Nevatia [6] used a set of "edgelet" features as weak classifiers to train a strong

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 761–771, 2012.

classifier to detect pedestrians without background subtraction. Leibe *et al.* [3] used an implicit shape model to generate top-down pedestrian models from a set of scale-invariant points feature for pedestrian detection. Lin *et al.* [4] decomposed a human model to different body parts and then detected occluded pedestrians from videos using a hierarchical part-template matching technique. Dong *et al.* [5] mapped the global shapes of humans to various configurations with different Fourier descriptors and then solved their parameters by a dynamic programming technique for crowd segmentation. Eshel and Mose [8] used a set of cameras with overlapping fields of views to detect heads and then tracked and counted pedestrians in a dense crowd using a planar homograph transformation. One key problem in the above methods is that the feature contrast between the objects and their backgrounds should be larger. In addition, most of approaches require good head position initializations for parameter searching and then counting people. This paper will propose a novel LDA-based approach to tackle the above problems and then detect and count occluded pedestrians from videos in real time.

Fig. 1 shows the flowchart of this system. To deal with the occlusion problem, we use a Bayesian inference model to formulate the people problem. Then, the MCMC scheme is used to capture the dynamics of the pedestrian model and thus effectively and efficiently derives its parameters. In real conditions, the accuracy of the MCMC scheme in people counting strongly depends on the results of head detection. Thus, an intelligent head detector is developed for identifying all possible head candidates from videos. However, due to lighting changes or background colors, the features of an object are not always clear to its background. The unclear feature often leads to lots of miss in object detection. To tackle this problem, a novel LDA approach is used to enhance the boundaries between objects and features. The usage of LDS in object detection is very different its common applications in object recognition. Then, a head-shoulder detector is proposed for detecting all possible candidates from the results of LAD. Due to LDA, lots of misses in object detection can be avoided. To improve the efficiency of model matching, the MCMC scheme is adopted for quickly filtering out impossible candidates and determining the optimal model parameters. Another critical problem is the construction of model space. Since different occluded pedestrians will form different contours, it is not easy to decide the size and space of pedestrian model. To tackle this problem, different templates will be dynamically generated when a new contour is searched. Then only a few of templates should be generated for pedestrian detection. The performance of the proposed method has been rigorously tested on various videos with qualitative and quantitative results corroborating its superiority in people counting.

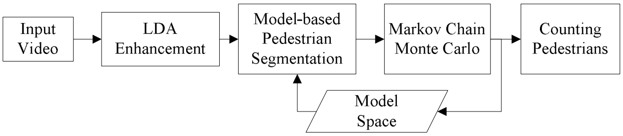


Fig. 1. Flowchart of people counting by MCMC

The remainder of the paper is organized as follows. In the next section, the problem of people counting was defined. Then, Section 3 discusses the details of head-shoulder detector. Section 4 describes the LDA scheme to enhance the discriminant capability from the object (head-shoulder) region to its background region. Section 5 reports a variety of experimental results and finally a conclusion will be presented in Section 6.

2 Pedestrian Detection under Occlusions

To count occluded pedestrians from videos, we use a Bayesian inference model to formulate this problem and then estimate its parameters through the MCMC scheme.

2.1 Bayesian Estimation

Given a region R , the problem of people counting can be formulated as a maximum a posterior estimation as follow:

$$n^* = \arg\max_n P\left(\Theta_n \middle| R\right), \tag{1}$$

where n the number of persons in R and Θ_n is the solution space of template sets of n persons. Θ_n can be further represented as

$$\Theta_n = \left\{\theta_1^n, \theta_2^n, ..., \theta_k^n, ..., \theta_{K_n}^n\right\}. \tag{2}$$

θ_k^n denotes the kth set of models for representing the appearances of n persons. Let M denote the set of used models. Then, θ_k^n is represented as

$$\theta_k^n = \left\{m_1, m_2, ..., m_i, ..., m_n\right\},$$

where $\theta_j \in M$. K_n is the number of elements in Θ_n and satisfies

$$K_n \leq \left(L \times I_h \times I_w \times H\right)^n, \tag{3}$$

where I_w and I_h denote the width and height of R , The parameters of m_i include a template φ_i at the position $\left(x_i, y_i\right)$ with the height h_i. L represents the number of models used for representing a pedestrian, and H is the maximum height of pedestrian. According to Bayesian rules, the posterior probability $P\left(\Theta_n \middle| R\right)$ can be decomposed into a prior distribution $P\left(\Theta_n\right)$ and a likelihood $P\left(R \middle| \Theta_n\right)$ as follows:

$$P\left(\Theta_n \middle| R\right) \propto P\left(R \middle| \Theta_n\right) P\left(\Theta_n\right). \tag{4}$$

$P\left(\Theta_n\right)$ is independent to time and can be represented as the form

$$P(\Theta_n) = P(n)\prod_{i=1}^{n} P_{template}(\varphi_i) P_{pos}(x_i, y_i) P_{height}(h_i), \tag{5}$$

where $P(n)$ is the prior on the number of persons in R and modelled as

$$P(n) = \exp(-\gamma n), \tag{6}$$

where γ is a predefined parameter (closer to zero) to control the growing rate of *n*. $P_{template}(\varphi_i)$ is the prior probability of the template φ_i and defined as

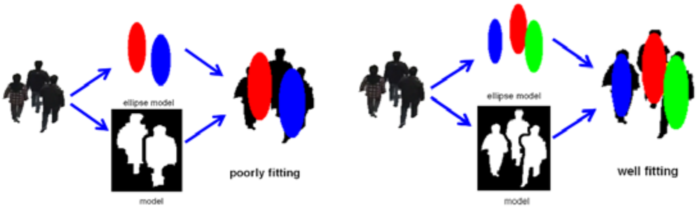
$$P_{template}(\varphi_i) = [1 + N_{used}(\varphi_i)]\left[\sum_{i=1}^{|\Psi|} N_{used}(\varphi_i)\right]^{-1}, \tag{7}$$

where $|\Psi|$ is the number of templates φ_i collected in Ψ and $N_{used}(\varphi_i)$ is the times of template φ_i when it appears in the training videos. $P_{pos}(x_i, y_i)$ denotes the prior probability of the template model m_i appearing at the position (x_i, y_i). Let h_R be the number of heads detected from R. Then, a set of mixture Gaussian function is used to model $P_{\text{mod}}(x_i, y_i)$, *i.e.*,

$$P_{\text{mod}}(x_i, y_i) = \sum_{k=1}^{h_R} \pi_k N_k^{head}(x_i, y_i), \tag{8}$$

where π_k is the coefficient for weighting the *k*th Gaussian function $N_k^{head}(x_i, y_i)$ and $\sum_{k=1}^{h_R} \pi_k = 1$. The form of $N_k^{head}(x_i, y_i)$ will be discussed in Section 3. $P_{height}(h_i)$ is the prior probability of model height defined as:

$$P_{height}(h_i) = \exp(-\frac{(h_i - \mu_H)^2}{\sigma_H^2}). \tag{9}$$



(a) (b)

Fig. 2. Different cases of model fitting. (a) Poorly fitting. (b) Well fitting.

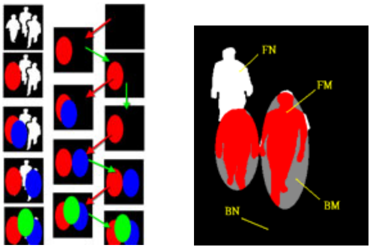
Usually, if two objects are overlapped together, their overlapping area should include lots of edges. Like Fig. 2, (a) shows the case of poorly fitting result and (b) is the result of well fitting. For a model θ_k^n in Θ_n, we use $O(\theta_k^n)$ to denote the overlapping area between templates in θ_k^n. Then, the edge likelihood $P_{edge}\left(R|\Theta_n\right)$ can be represented as

$$P_{edge}\left(R|\Theta_n\right) = \max_{\theta_k^n \in \Theta_n} \frac{\sum_{p \in O(\theta_k^n)} Edge(p)}{|O(\theta_k^n)|},$$

where $Edge(p)$ is 1 if p is an edge pixel and zero, otherwise.

2.2 Estimation through Markov Chain Monte Carlo

To quickly find the solution n from Eq.(1), an intelligent head-shoulder detector will be proposed. Then, the MCMC technique is used to sample the posterior probability space for parameter estimation. Fig. 3 (a) shows the learning progress of MCMC. The left-most column represents the result of foreground extraction and the right-most column shows the best hypothesis generation. Then, predict the next state by estimating the best hypothesis from the pervious state. The state transition is guided by a proposal density function. The transition state contains three actions: *Human hypothesis addition*, *hypothesis removal*, *Model update/switch*.



(a) (b)

Fig. 3. The MCMC estimation process. (a) MCMC process. (b) Different sub-regions are extracted for defining the proposal density function.

To build the proposal density function, this paper uses a background subtraction technique to extract foreground objects from videos. Based on the subtraction result, the input region R can be converted to a binary map. Like Fig. 3(b), after overlapping a template on R, R can be divided to four kinds of sub-regions, i.e., BN (background pixels), BM (background pixels but contained in the template), FN (foreground pixels but not contained in the template) , and FM (foreground pixels contained in the template). Let $\mathbb{C} = BM \bigcup FN \bigcup FM$. Then, three rates (α, β, γ) are obtained as follows:

$$\alpha = |FN||\mathbb{C}|^{-1}, \beta = |BM||\mathbb{C}|^{-1}, \text{and } \gamma = |FM||\mathbb{C}|^{-1}.$$

In addition, let $\xi = \max\left(\alpha, \beta, \gamma\right)$, the proposal density function for determining the next state transitions is given as follows:

$$P(Action_t \mid \alpha,\beta,\gamma) = \begin{cases} addition & if\ \xi = \alpha, \\ removal & if\ \xi = \beta, \\ exchange\ or\ move & if\ \xi = \gamma. \end{cases}$$

Details of each action are described as follows.

(a) Template Hypothesis Addition

According to the head point detected previously, randomly select a reference point for placement of a new model hypothesis and estimate the posterior probability.

1) Create the Head Candidate GMM map H(u) from the result of head-shoulder detection;
2) Randomly select a point “u” from H(u);
3) Randomly dice a number “x”, where 0 <= x <= 1;
4) If x<H(u), we accept it; otherwise, repeat step (2).

(b) Human Hypothesis Removal

Randomly select an existing hypothesis model from the best hypothesis space, and then remove it according to the following step:

1) Randomly select a template hypothesis “z”;
2) Calculate the overlapping rate $P(B \mid z)$ between the background and z;
3) Randomly dice a number “x” 0 <= x <= 1
4) If x> $P(B \mid z)$, remove it; otherwise, repeat step (1).

(c) Exchange Template hypothesis

Randomly select an existing hypothesis model from the best hypothesis space and replace it with a hypothesis model of high probability according to the following step:

1) Randomly select a template hypotheses “p” with the center C_p;
2) Randomly select second template hypotheses “q” with the center C_q;
3) Randomly dice a number “x”, where 0 <= x <= 1;
4) If x< $e^{-\|C_p - C_q\|}$, exchange their positions, otherwise, repeat step (2).

(d) Move Template Hypothesis

Randomly select a template hypothesis and movie it’s position according the steps:

1) Randomly select a template hypothesis “z” with the center C_z.
2) Extract the true positive region TPR_z from z.
3) Calculate the center $C_{tpr,z}$ of TPR_z
4) Move it’s center C_z to the candidate position $C_{tpr,z}$.

3 Head-Shoulder Hypothesis Generation

To count persons from a region R, a novel head-shoulder detector is proposed for detecting all possible heads from videos for guiding the MCMC estimation process.

3.1 Head Detection from Multiple Templates

This paper uses several templates $\hbar_i$ to represent a head-shoulder hypothesis. Let $\hbar$ denote the set of head-should templates, i.e., $\hbar = \{\hbar_i\}$. Like Fig. 4, for each template $\hbar_i$, we hierarchically decompose it to different parts, i.e., $\hbar_{ij}$. Then, given a template $\hbar_i$, the probability $P(R\,|\,\hbar_i)$ can be obtained by combining all its components, i.e.,

$$P(R\,|\,\hbar_i) = \sum_{\hbar_{ij}\in\hbar_i} \alpha_j P(R\,|\,\hbar_{ij}), \tag{10}$$

where α_i is a weight for combining head components $\hbar_{ij}$ together. The probability $P(R\,|\,\hbar)$ to detect whether there is on head candidate in R is estimated as

$$P(R\,|\,\hbar) = \max_{\hbar_i\in\hbar} P(R\,|\,\hbar_i).$$

To build $P(R\,|\,\hbar_{ij})$, three features are used to describe $\hbar_{ij}$; that is, Ω -type contour, symmetry feature, and centroid context.

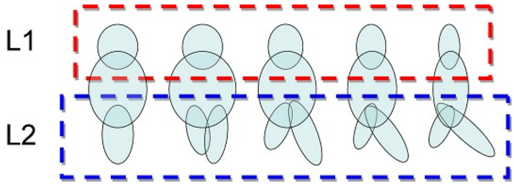


Fig. 4. Five templates $\hbar_i$ used to describe a pedestrian

3.2 Head-Shoulder Ω model

Actually, the shape of a head looks like a symbol Ω. Thus, like Fig. 5, we can use an Ω model to represent a head. Given a head shoulder template $\hbar_i$, its contour positions and directions can be extracted along the Ω model and represented as $P_{\hbar_i} = \{\vec{p}_{\hbar_i}(1),...,\vec{p}_{\hbar_i}(k),...,\vec{p}_{\hbar_i}(n)\}$ and $V_{\hbar_i} = \{\vec{v}_{\hbar_i}(1),...,\vec{v}_{\hbar_i}(k)),...,\vec{v}_{\hbar_i}(n)\}$. In addition, the contour positions and directions in R is represented as P_R and V_R. Then, the distance between P_R and $P_{\hbar_i}$ is represented:

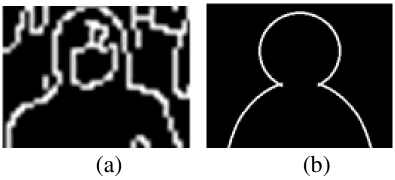

(a) (b)

Fig. 5. Head-shoulder model for head detection. (a) Edge map of a head. (b) Ω model to represent a head.

$$d(P_{\hbar_i}, P_R) = \frac{1}{n}\sum_{k=1}^{n} | \vec{p}_R(k) - \vec{p}_{\hbar_i}(k) |^2 .$$

In addition, the distance between V_R and $V_{\hbar_i}$ is defined as

$$d(V_{\hbar_i}, V_R) = 1 - \frac{1}{n}\sum_{k=1}^{n} | \vec{v}_R(k) \bullet \vec{v}_{\hbar_i}(k) | .$$

Then, the probability of a detection window R containing $\hbar_{ij}$ can be described by

$$P_\Omega(\hbar_{ij} \,|\, R) = \exp[-\frac{d(V_{\hbar_i}, V_R)}{\sigma_V^2} - \frac{d(P_{\hbar_i}, P_R)}{\sigma_P^2}].$$

3.3 Head Symmetry

According to the contour of head-shoulders, the symmetry property exists between the left and right sides on a head-shoulder template part $\hbar_{ij}$. Let E_{R_l} and E_{R_r} denote the edge maps of the left and right sides of R . In addition, $\sim E_R$ denotes the mirror of E_R according to the central vertical line in R . The symmetry used to detect the head-shoulder candidates is defined as:

$$P_{sym}(R \,|\, \hbar_{ij}) = \exp(-\frac{1}{2}[Dt_{\hbar_{ij}}(E_{R_l}, \sim E_{R_r}) + Dt_{\hbar_{ij}}(\sim E_{R_l}, E_{R_r})]),$$

where $Dt_{\hbar_{ij}}(A, B)$ is the distance between two edge maps A and B based on their distance transforms within the part $\hbar_{ij}$.

3.4 Histogram of Gradients on Polar Coordinate

This paper also uses the histogram of gradients on a polar coordinate for describing the contour of objects. Then, two histograms $h_R(k)$ and $h_{\hbar_{ij}}(k)$ is obtained from R and $\hbar_{ij}$, respectively. Thus, the similarity between them can be estimated by

$$P_{Grad}(R \,|\, \hbar_{ij}) = \sum_{k=1}^{K_{bin}} \min\left\{h_R(k) \;,\; h_{\hbar_{ij}}(k)\right\} \left[\sum_{k=1}^{K_{bin}} \max\left\{h_R(k) \;,\; h_{\hbar_{ij}}(k)\right\}\right]^{-1} .$$

To verify a pedestrian candidate more accurately, its outer shape should play a more important role than its inner shapes. Thus, for each pixel p_i , a weight w_i is included for weighting its importance, where w_i increases according to the distance between p_i and the central of R. Assume that r_i is the distance between p_i and the central of R, and the circumcircle of R has the radius z. Then, w_i is defined by

$$w_i = \begin{cases} \exp(-|r_i - z|^2), \text{ if } r_i \le \text{z}, \\ 0, \qquad \text{elsewise.} \end{cases}$$

Then, Eq.(10) can rewritten as:

$$P(R|\hbar_i) = \sum_{\hbar_{ij}\in \hbar_i} \alpha_j P_\Omega(\hbar_{ij}|R) P_{sym}(R|\hbar_{ij}) P_{Grad}(R|\hbar_{ij}). \tag{11}$$

4 LDA-Based Object Detection

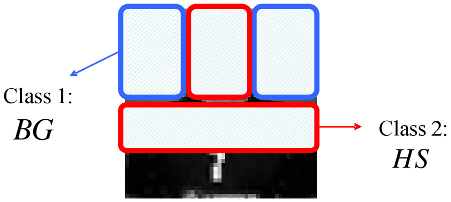


Fig. 6. 2-Class features extracted from a head-shoulder candidate

To detect and count people from videos, most of methods depend very strongly on the characteristics of the edge. However, if the color contrast is very low or not clear, the task of people counting will tend to fail. To tackle this problem, this section takes advantages of Linear Discriminant Analysis (LDA) to enhance the discriminant capability from the object (head-shoulder) region to its background region. Like Fig. 6, given a training head region, two regions are manually extracted. Here, BG and HS mean the background and the head-shoulder regions, respectively. Their feature means μ_{HS} and μ_{BG} can be estimated as follows:

$$\mu_{HS} = \frac{1}{|HS|}\sum_{p\in HS} Head(p) \text{ and } \mu_{BG} = \frac{1}{|BG|}\sum_{p\in BG} Head(p). \tag{12}$$

The within-class scatter matrix S_W is the sum of the inner-class variance of the pixels in HS and the background BG, *i.e.*,

$$S_W = S_{HS} + S_{BG}, \tag{13}$$

where $S_{HS} = \sum_{p\in HS} (head_p - \mu_{HS})(head_p - \mu_{HS})^{\mathrm{T}}$ and $S_{BG} = \sum_{p\in BG} (head_p - \mu_{BG})(head_p - -\mu_{BG})^{\mathrm{T}}$. In addition, the between-class scatter matrix S_B is the between-class variance of the mean between the object and the background, which is calculated as

$$S_B = (\mu_{HS} - \mu_{BG})(\mu_{HS} - \mu_{BG})^{\mathrm{T}}. \tag{14}$$

To enhance the foreground feature from the background, we can find the optimal projective basis W by maximizing the term $J(W)$.

$$J(W) = \frac{\left|W^{\mathrm{T}} S_B W\right|}{\left|W^{\mathrm{T}} S_W W\right|}. \tag{15}$$

After calculations, W can be found by the form:

$$W = S_W^{-1}(\mu_{HS} - \mu_{BG}). \tag{16}$$

For each scanning window, we can apply LDA first for enhancing the head-shoulder feature from the background.

5 Experimental Results

This paper collected a set of 10GB (412,298 frames) videos for the testing data, with 89 video segments of different lengths at different environments. The video sources including CDVP, OTCBVS, BEHAVE, CAVIAR and PETS. The performance of our system is about 11 fps. Fig. 7 shows the results of head-shoulder detection. Fig. 8 shows the results using full bodies. Fig. 9 shows the detection results of different pedestrians under different environments. Table 1 shows the accuracy analysis of people counting under different databases. Experimental results have shown that our counting method is superior in terms of accuracy, robustness, and stability.

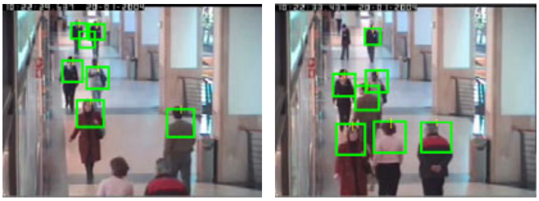


Fig. 7. Results of Head-Shoulder detection

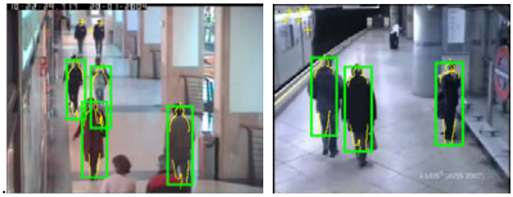


Fig. 8. Results of Pedestrian detection. Yellow-contour model is the best hypothesis for the hierarchy models.

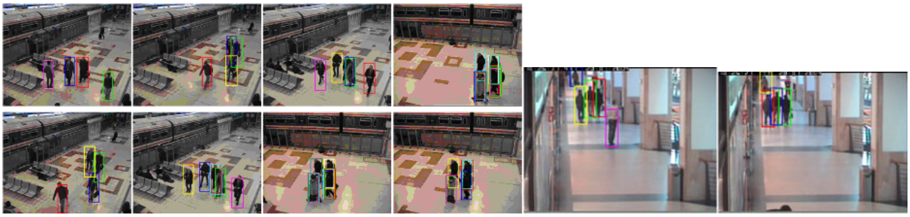


Fig. 9. Detection results of different pedestrians under different environments

Table 1. Accuracy analysis Of PEOPLE Counting

Video	frames	Accuracy	False alarm	miss	RMSE
OTCBVS	5193	96.73%	1.23%	0.0328	0.1910
BEHAVE	73566	84.80%	8.87%	15.19%	1.1011
CAVIAR I	15775	88.20%	6.79%	11.79%	0.4825
CAVIAR II	20525	84.82%	10.74%	14.64%	0.5781

Acknowledgments. This work was supported in part by National Science Council of Taiwan under Grant NSC97-2221-E-155-060, Taiwan, and the Ministry of Economic Affairs under contract no. 98-EC-17-A-02-S1-032.

References

1. Rabaud, V., Belongie, S.: Counting Crowded Moving Objects. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 705–711 (2006)
2. Antonini, G., Thiran, J.: Counting Pedestrians in Video Sequences Using Trajectory Clustering. IEEE Transactions On Circuits And Systems For Video Technology 16, 1008–1020 (2006)
3. Leibe, B., Seemann, E., Schiele, B.: Pedestrian Detection in Crowded Scenes. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 878–885 (2005)
4. Lin, Z., Davis, L.S., Doermann, D., DeMenthon, D.: Hierarchical Part-Template Matching for Human Detection and Segmentation. In: International Conference on Computer Vision, pp. 1–8 (2007)
5. Dong, L., Parameswaran, V., Ramesh, V., Zoghlami, I.: Fast Crowd Segmentation Using Shape Indexing. In: International Conference on Computer Vision, pp. 1–8 (2007)
6. Wu, B., Nevatia, R., Li, Y.: Segmentation of Multiple, Partially Occluded Objects by Grouping, Merging. Assigning Part Detection Responses. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)
7. Kim, Z.: Real Time Object Tracking based on Dynamic Feature Grouping with Background Subtraction. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)
8. Eshel, R., Moses, Y.: Homography Based Multiple Camera Detection and Tracking of People in a Dense Crowd. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)

Visual-Based Spatiotemporal Analysis for Nighttime Vehicle Braking Event Detection

Duan-Yu Chen and Chia-Hsun Chen

Department of Electrical Engineering, Yuan Ze University,
32003 Chung-Li, Taiwan
dychen@saturn.yzu.edu.tw, s970509@mail.yzu.edu.tw

Abstract. In this paper, we propose a novel visual-based approach that can detect brake lights at night by analyzing the tail lights based on the thee-dimensional Nakagami imaging which can provide robust information of brake lights. Instead of using the knowledge of the heuristic features, such as symmetry and position of rear facing vehicle, size and so forth, we focus on extracting the invariant features based on modeling the scattering of brake lights and therefore can conduct the detection process in a part-based manner. Experiment from extensive dataset shows that our proposed system can effectively detect vehicle braking under different lighting and traffic conditions, and thus prove its feasibility in real-world environments.

Keywords: Brake Light Detection, Spatiotemporal Analysis.

1 Introduction

Advanced safety vehicle is a critical issue in recent years for automobiles, especially when the number of vehicles is growing rapidly worldwide. The cost down of general cameras makes it feasible to have an intelligent system of visual-based event detection in front for forward collision avoidance and mitigation. When driving at nighttime, vehicles in front are generally visible by their tail lights. The brake lights are particularly important due to their consequent events that drivers need to focus on.

In the related works, researchers focus on detecting vehicles [1][8-9][19], traffic lights [2][5-6], tail lights of vehicles in front [3][7][10] and collision warning at night [4]. To detect vehicles at nighttime, Chen et al. [1] perform bright object segmentation and verify the segmented regions if they are the targets by conducting the process of spatial clustering using the features of shape, texture, and the relative positions, i.e., the symmetric property. The color features are useful for detecting most tail lights and then vehicles could be identified by some rule-based approach.

Park and Jeong [2] use color features to extract candidate regions and then select candidate pixels that may come from signal light from grabbed image based on statistical features. Remaining regions are examined to see if each region were of a shape of circle since there are many light sources that have color like signal light. In [3], O'Malley *et al.* proposed a novel image processing system to detect and track vehicle

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 772–781, 2012.

rear-lamp pairs in forward-facing color video. First, they suggested a camera-configuration process that optimizes the appearance of rear lamps for segmentation. Rear-facing lamps are segmented from low-exposure forward-facing color video using a red-color threshold. Their color threshold is directly derived from automotive regulations and adapted for real-world conditions in the hue–saturation–value (HSV) color space. Then, lamps are paired using color cross-correlation symmetry analysis and tracked using Kalman filtering. Due to the use of the symmetry characteristic of tail lights, the detection of tail lights is difficult to be conducted in a part-based manner.

In [4], Thammakaroon and Tangamchit developed an approach for predictive brake warning which is based on the distance estimation from the analysis of tail lights in front. In order to localize the regions of interest, the red color of the rings is detected based on the RGB color model. They calculate the brake lights distribution from 50 collected samples to determine the threshold for each color channel. Consequently, three candidate Region of Interest (ROI) from RGB channels are verified by computing the intersection. The intersected ROI(s) is regarded as the tail lights and their size is used to determine the distance between the vehicle(s) in front and the camera. In [5], Yelal *et al.* proposed an approach for the signal lights detection. They used the luminous values of the glowing points combined with color values present in the image frame as the features in real time. Lab color model is used to extract the luminous values in the image frame, and contour tracking is conducted for shape detection of the signal lights. The image frames extracted from the real time video are subjected to region segmentation to extract the ROIs. The ROI is extracted by analyzing the position of the signal lights in an image frame based on road tracking. This road tracking technique uses the edges of a road as reference lines. Based on these reference lines the image frame is segmented into different regions.

In [6], Gong *et al.* proposed an approach for the recognition and tracking of traffic lights. First, the candidate region of the traffic light is extracted using the threshold segmentation method and the morphological operation. Then, the recognition algorithm of the traffic light based on machine learning is employed. To avoid false negatives and tracking loss, the target tracking algorithm CAMSHIFT (Continuously Adaptive Mean Shift), which uses the color histogram as the target model, is adopted. In [7], Schamm *et al.* proposed an approach that can recognized vehicles in front of the own car by detection of their front or rear lights using a perspective blob filter and subsequently searching for corresponding light pairs. For preceding vehicles, the activity of the third break light is estimated, to distinguish the maneuver state of the vehicle. Their approach uses a flexible 2d Gaussian filter, which can detect circular light blobs even under difficult conditions. A rule-based clustering approach is used to find light pairs. The light pair association is combined with a symmetrical analysis using the Sum of Squared Differences to confirm valid associations. Using a color based comparison, the system is able to distinguish between the front and rear position lights. For preceding vehicles, the activity of the third break light is estimated, so the maneuver state of the vehicle can be detected.

In [8], Sun *et al.* proposed an approach of rear-view vehicle detection using feature extraction and classification. Their vehicle detection algorithm consists of two main steps a multiscale driven hypothesis generation step and an appearance-based hypothesis verification step. During the hypothesis generation step, image locations where vehicles might be present are extracted. This step uses multiscale techniques not only to speed up detection, but also to improve system robustness. The appearance-based hypothesis verification step verifies the hypotheses using Gabor features and SVMs.

In [9], Kim *et al.* detected and tracked vehicles regardless of the light and road conditions at any distance using vision and sonar sensors. First, they used a simple method that can determine the light condition by observing several images and this light condition is used by selecting one of several detection methods. The method in the day time image can extract the shadow region represented by the boundary between a vehicle and the road and further verify by using other vehicle features, such as symmetry rate, vertical edge, and lane information. The vehicle tracking method in the day time uses on-line template matching using the mean image created by several consecutive detection results. The method in the night time extracts bright regions caused by the headlights, taillights, brake lights, etc. and these candidates are verified by observing several consecutive frames.

In [10], Wang *et al.* proposed an approach that can perform lane detection and vehicle recognition at nighttime. In lane detection, three features including lane markers, brightness, slenderness and proximity are applied to detect the positions of lane markers in the image. On the other hand, vehicle recognition is achieved by using an evident feature which is extracted through four steps: taillight standing-out process, adaptive threshold, centroid detection, and taillight pairing algorithm.

However, to the best of our knowledge, few approaches focus on detecting brake lights of vehicles. To detect the brake lights robustly, instead of employing the knowledge of the heuristic features, such as symmetry and position of rear facing vehicle, size and so forth, in this work, we focus on finding the invariant features from the regions of brake lights and aim to conduct the detection process in a part-based manner. Considering the environment-independent features, the significant characteristic of the scattering in the region of brake lights is our concern and is employed for the analysis of tail lights and the consequent detection of brake lights.

The remainder of this paper is organized as follows. Section 2 details the proposed approach of brake light detection based on 3D Nakagami imaging. Section 3 shows the experiment results and discussions, and Section 4 concludes this work.

2 Brake Lights Detection

In this section, we describe the approach of brake light detection, in which we aim to detect brake lights of vehicles in front without the need of the complete taillights in shape.

2.1 Preprocessing - Contrast Enhancement

The color intensity image is defined as

$$C_i = \frac{Max(R,G,B)}{255}, \tag{1}$$

where R, G and B are the three color channels in RGB color space respectively. The color intensity image C_i can then be obtained using Eq.(1). In addition to employing the property of the large contrast between the regions of tail lights and non-tail lights, the scattering property of brake lights is investigated. In the real-world environment, scattering is a general physical process [11] where some forms of radiation, such as light, sound, or moving particles, are forced to deviate from a straight trajectory by one or more localized non-uniformities in the medium through which they pass. Therefore, we focus on discovering the invariant features from the regions of brake lights based on the scattering property and thus aim to conduct the detection process in a part-based manner.

Before conducting the process of Nakagami imaging, to reduce the noise generated from non-tail lights, a simple step function $T(u)$ is first applied to the color intensity image C_i and is defined by

$$T(u) = \begin{cases} 1 & , if\ u \geq \theta_u \\ 0 & , otherwise \end{cases} . \tag{2}$$

Using Eq.(2), the color intensity image C_i is filtered by

$$U_T = C_i \times T(C_i) , \tag{3}$$

From the color intensity distribution shown in Fig.1, it can be observed that we can approximately separate the tail lights from non-tail lights. However, the distributions of brake lights and tail lights are difficulty to discriminate since they are of high similarity. Therefore, to overcome this problem, we propose an approach that can discriminate brake lights from tail lights based on 3D Nakagami image.

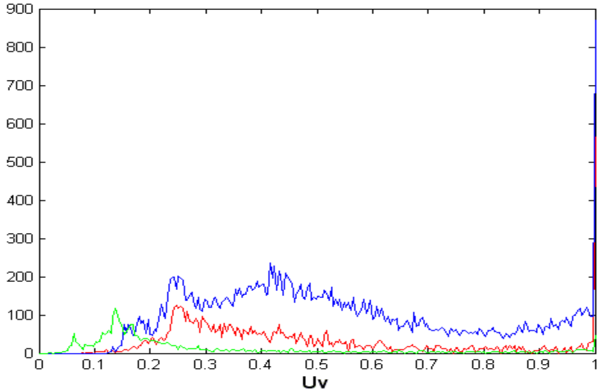


Fig. 1. Demonstration of the color intensity histogram of tail light (red), brake light (blue) and non-tail light (green)

2.2 Modeling Tail Lights by Nakagami Distribution

The Nakagami statistical model initially used for the analysis of the returned radar echoes and may be applicable to ultrasound by using two associated parameters, namely m-parameter and the scaling parameter [12-13]. The Nakagami statistical model is also widely used in wireless communications for modeling the fading channels [14]. The probability density function of the Nakagami distribution is defined as

$$f(r)=\frac{2m^{m}r^{2m-1}}{\Gamma(m)\Omega^{m}}\exp\left(-\frac{m}{\Omega}r^{2}\right) \quad ,r\geq 0, \tag{4}$$

where $\Gamma(\cdot)$ is the gamma function. The symbol r means possible values of the random variable R in the region of interest. The scaling parameter Ω and the Nakagami parameter m associated with the Nakagami distribution can be obtained respectively from Eq.(5) and Eq.(6).

$$\Omega=E(R^{2}), \tag{5}$$

$$m=\frac{\Omega^{2}}{E[(R^{2}-\Omega)^{2}]}, \tag{6}$$

where $E(\cdot)$ denotes the statistical mean. The Nakagami parameter m is a shape parameter determined by the probability distribution function of the random variable R. Therefore, to detect one or more brake lights present in the image, the 3D Nakagami image should be computed. The 3D Nakagami image is obtained based on the 3D Nakagami parameter m_{w}, which can be obtained by Eq.(6). However, a sliding cube should be defined first for estimating the local 3D Nakagami parameter, m_{w}, which is assigned as the new voxel located in the center of the cube. The variable R is the voxel set of the region of interest selected by the sliding cube. Based on sliding the cube with the 3D data, each local Nakagami parameter, m_{w}, in its regions of interest is computed and thus the 3D Nakagami image can be obtained.

The size of cube is inversely proportional to the resolution of the 3D Nakagami image. Although, the smaller cube cause a better resolution, this also means that we need more time and less 3D data lead to an unstable estimation of the Nakagami parameter m_{w}. Therefore, to find the stable parameter m_{w} and an acceptable 3D Nakagami resolution, it is necessary to determine the optimal size of the sliding cube. To resolve the optimal size of the cube, we first calculate Nakagami parameter m and average Nakagami parameter $\overline{m}$ using the sliding cube to process the 3D data, in which $\overline{m}$ is an indicator representing the global scattered statistics. Each value m_{w} in the 3D volume and the average value $\overline{m}_{w}$ are subsequently estimated using sliding cubes with increasing sizes. When the cube becomes large enough to satisfy the stable estimation of m_{w}, the local mean should approach the global mean. Namely, an appropriate cube size is determined once $\overline{m}_{w}\approx m_{w}$.

Due to the scattering from brake lights, the red lights appear wider and brighter than non-brake lights. It can be observed that the area covering the part of tail light and the scattering regions can be emphasized by 3D Nakagami imaging. It can be observed that based on 3D Nakagami imaging, the area of the brake light including the scattering region can be discriminated from the area of non-brake light.

To model the brake lights based on 3D Nakagami imaging, we observe that the variable R with the optimal sliding cube in the 3D Nakagami image possesses the Nakagami parameter of the voxels with larger value,which is higher in the area of brake lights than that in the area of non-brake lights.

2.3 Adaptive Decision Making for Brake Light Detection

To determine the threshold T_m adaptively, a dataset generated randomly is employed to simulate what the brake lights behave. To discriminate brake lights from non-brake lights based on the scattering property, we can model the scattering intensity I_s by

$$I_s = \frac{1}{2} I_0 (1 + \cos^2 \Theta) , \tag{7}$$

where I_0 is the intensity of the incident light and Θ is the scattering angle. According to the scattering property, the scattering intensity of tail lights would be generally larger than that of non-tail lights since the angle between the camera and the tail light is smaller than that between the camera and the non-tail lights. It means that higher scattering intensity would result in larger scattering region of a light source.

Furthermore, the varied distance between the tail lights in front and the camera is also critical since the signal of tail lights would attenuate with the vehicle going further. Therefore, the threshold T_m used to detect the brake lights should be adaptive to the distance and the size of cube. The distance between the tail lights and the camera can be approximately estimated based on the computation of the vanishing point by lane line detection [15].

Finally to determine the threshold T_m adaptively according to the evaluated distance, a set of tail lights is collected in distinct distances between the vehicles and the camera and used to approximate the signal variations using curve fitting [17-18]. Sometimes, the chosen curve passes through the data points, but on other points, the curve closes to them rather than passing through them. In most cases, we use the least square curve fitting to make the square error of the data minimum points. We can thus obtain an adaptive threshold T_m according to the distance D_{dis} by

$$T_m = \begin{cases} 1000 & , if\ D_{dis} \geq H_d \\ a{D_{dis}}^2 + bD_{dis} + c & , otherwise \\ 2 & , if D_{dis} < L_d \end{cases} , \tag{8}$$

where the weighting set $\{a, b, c\}$ is evaluated in the training dataset, and D_{dis} is the distance between the candidate regions to the horizontal line. H_d and L_d is the upper and lower bound of the distance, respectively.

3 Experimental Results

The extensive test dataset is captured using a CCD camera under different road environments, and traffic conditions. The resolution is 80?60?10. For qualitative evaluations, some detected results are shown in Fig.2. The first column shows the original video frame, the second column denotes the Nakagami image, and the third column introduces the criterion used to make decision of braking or non-braking. In this test sequence, the vehicles are further from the camera. From Fig.2, we can see that the value of Nakagami parameter m_w of the tail lights of non-braking and the street lights are relatively smaller than the value of Nakagami parameter $m_w$of the braking light, and accordingly would not be regarded as brake lights.

In Fig.3, the red line is the threshold, T_m, and the red point where is the max Nakagami parameter m_w. From Fig.3(c), we can see that the value of m_w would be larger than the T_m when brake-lights appear in the video frame. Examples can be observed from Fig.3(c) to Fig.3(d).

The quantitative evaluation of the detection rate is shown in Table 1. The performance of braking event detection is quite promising since the overall detection rate is up to 85% in the real road environment.

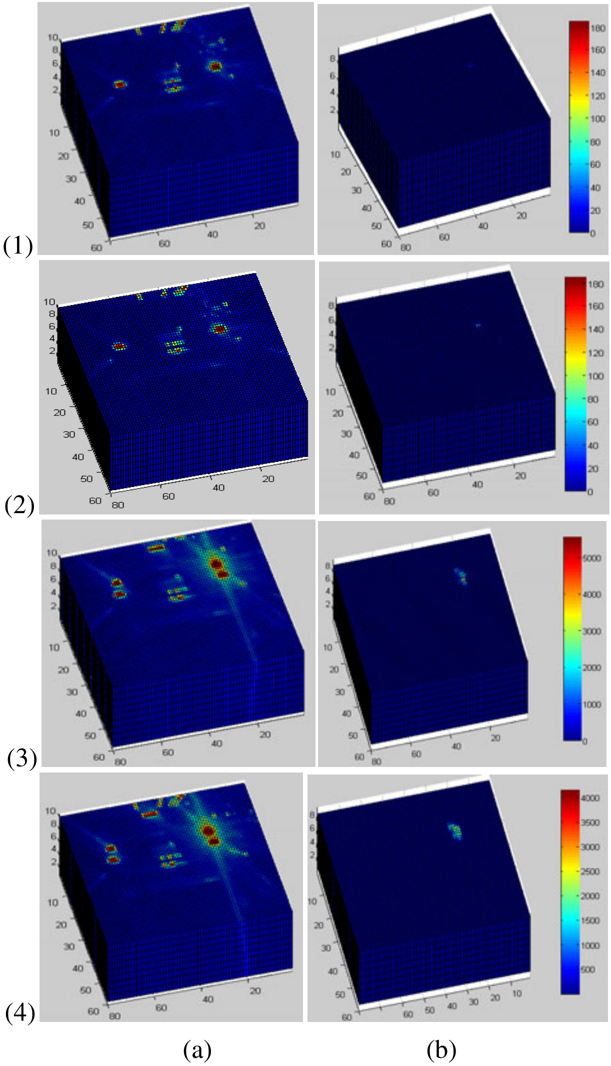
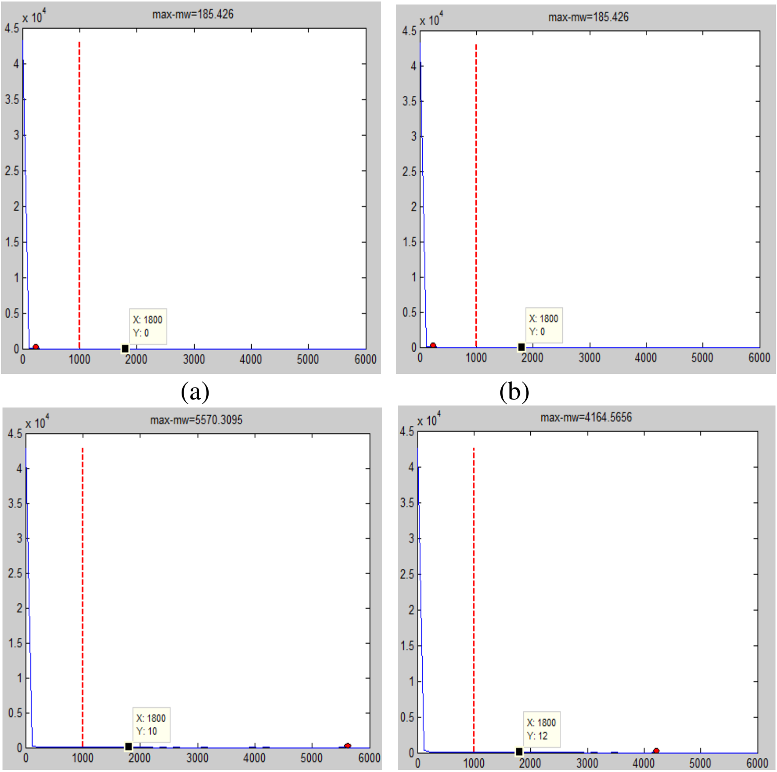


Fig. 2. (a) Frames shown from the status of non-braking to braking; (b) 3D Nakagami image(m_w)



(a) (b)

(c) (d)

Fig. 3. (a) to (d) is the histograms of Fig.2(1) to Fig.2(4)

Table 1. The detection rate evaluated using the dataset defined by four categories

Video Classification		Ground Truth	Detected	Detection Rate	Avg.
Case	Dist.				
1	Closer	32	30	93.75%	85%
2	Med.	32	27	84.375%	
3	Far	32	25	78.125%	

4 Conclusions

In this paper, we have proposed a novel visual-based approach that can detect brake lights at night by analyzing the tail lights based on the Nakagami-m distribution. Instead of using the knowledge of the heuristic features, such as symmetry and position of rear facing vehicle, size and so forth, we focus on finding the invariant features modeling the brake lights scattering by 3D Nakagami imaging and therefore can conduct the detection process in a part-based manner. Experiment from extensive dataset has shown that our proposed system can effectively detect vehicle braking under different lighting and traffic conditions. The detection rate has achieved about 85% and thus proves its feasibility in real-world environments.

References

1. Chen, Y.L., Chen, Y.H., Chen, C.J., Wu, B.F.: Nighttime vehicle detection for driver assistance and autonomous vehicles. In: Proc. IEEE ICPR, pp. 687–690 (2006)
2. Park, J.H., Jeong, C.S.: Real-time Signal Light Detection. In: Proc. International Conf. on Future Generation Communication and Networking Symposia (2008)
3. O'Malley, R., Jones, E., Glavin, M.: Rear-Lamp Vehicle Detection and Tracking in Low Exposure Color Video for Night Conditions. IEEE Trans. on Intelligent Transportation Systems 11(2) (June 2010)
4. Thammakaroon, P., Tangamchit, P.: Predictive Brake Warning at Night using Taillight Characteristic. In: Proc. IEEE International Symposium on Industrial Electronics (2009)
5. Yelal, M.R., Sasi, S., Shaffer, G.R., Kumar, A.K.: Color-based Signal light Tracking in Real-time Video. In: Proc. IEEE International Conference on Video and Signal Based Surveillance (2006)
6. Gong, J.W., Jiang, Y.H., Xiong, G.M., Guan, C.H., Tao, G., Chen, H.Y.: The Recognition and Tracking of Traffic Lights Based on Color Segmentation and CAMSHIFT for Intelligent Vehicles. In: Proc. IEEE Intelligent Vehicles Symposium (2010)
7. Schamm, T., von Carlowitz, C., Zollner, J.M.: On-Road Vehicle Detection During Dusk and at Night. In: Proc. IEEE Intelligent Vehicles Symposium (2010)
8. Sun, Z., Bebis, G., Miller, R.: Monocular Precrash Vehicle Detection: Features and Classifiers. Proc. IEEE Transactions on Image Processing (2006)
9. Kim, S.Y., Oh, S.Y., Kang, J.K., Ryu, Y.W., Kim, K.S., Park, S.C., Park, K.H.: Front and Rear Vehicle Detection and Tracking in the Day and Night Times Using Vision and Sonar Sensor Fusion. In: Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (2005)
10. Wang, C.C., Huang, S.S., Fu, L.C.: Driver Assistance System for Lane Detection and Vehicle Recognition with Night Vision. In: Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (2005)
11. Stover, J.C.: Optical Scattering: Measurement and Analysis. SPIE Optical Engineering Press (1995) ISBN 0-8194-1934-6
12. Nakagami, M.: The M-Distribution–A General Formula of Intensity Distribution in Rapid Fading. In: Hoffman, W.C. (ed.) Statistical Methods on Radio Wave Propagation. Pergamon Press, New York (1960)
13. Shankar, P.M.: A General Statistical Model for Ultrasonic Backscattering from Tissues. IEEE Trans. Ultrason. Ferroelec. Freq. Contr. 47, 727–736 (2000)
14. Charash, U.: Reception through Nakagami fading multipath channels with random delays. Proc. IEEE Transactions on Communications 27, 657–670 (1979)
15. Tseng, S.P., Liao, Y.S., Yeh, C.H., Huang, L.K.: A DSP-based Lane Recognition Method for the Lane Departure Warning System of Smart Vehicles. In: Proc. International Conference on NSC, pp. 823–828 (2009)

16. John, B.M., Donald, N.: Application of the Hough Transform to Lane detection and Following on High Speed Roads. signal & system Group, Department of Computer Science, National University of Ireland, pp. 1–9 (1999)
17. Xu, G.S.: Sub-pixel Edge Detection Based on Curve Fitting. In: Proc. IEEE International Conference on Information and Computing Science (2009)
18. Shen, W., Wu, L., Tu, L.: A curve fitting based image segmentation method. In: Proc. International Conference on Computer Science and Information Technology (2009)
19. Takahashi, H., Ukishima, D., Kawamoto, K., Hirota, K.: A Study on Predicting Hazard Factors for Safe Driving. IEEE Trans. on Industrial Electronics 54(2), 781–789 (2007)
20. Mori, H., Charkari, M., Matsushita, T.: On-line vehicle and pedestrian detections based on sign pattern. IEEE Tran. on Industrial Electronics 41(4) (1994)

Fusing Template and Point Information to Track Planes with Large Interframe Displacement

Naeem Akhter

Technical University Vienna, Austria
noomi702@yahoo.com

Abstract. This paper presents a hybrid approach by fusing template and keypoint based tracking to track pose of planar textured targets with large interframe displacement. The fusion is made such that it adds to accuracy and convergence of template based tracking without involving feature selection, introducing pose drift strategy, and incorporating sophisticated prediction or motion model. The approach is not only robust against illumination changes and partial occlusion, but also free from offline pose learning and prior knowledge about background which makes it flexible to adapt change in scene.

Keywords: Monocular, Pose tracking, Textured, Planar, Interframe displacement.

1 Introduction

Pose tracking is a fundamental task in several robotic applications for example self localization and object grasping. It has been of general interest in computer vision literature, particularly in scenes with cluttered background, noise, partial occlusion, or changing illumination. Accordingly, there is a large number of related publications. While substantial advances have been made, there is still a need for systems which can tolerate rapid translations, rotations and accelerations in unconstrained environments. To cope with this, a hybrid approach is proposed that integrates two approaches, namely template based and keypoint based tracking.

On a broad level, approaches of pose tracking for textured targets can be divided into two groups: pose tracking by detection and pose tracking by modeling. The first type of approaches [10], [1], [2], [11], [5] assume there exist a unique casual-effect relationship between pose and certain properties of the target [9]. A 2D-3D mapping is constructed using training data consisting of several views of the target. The constructed mapping is then used to find pose of a given 2D target image, which makes them rigid to learned scenarios. Scenes in which targets are easy to detect are assumed [10]. Although suitable for large interframe displacement as strong prior on the pose is not required, they are less accurate and more computationally intensive than the second type of approaches [25].

The second type of approaches pre-assume a 3D model of target but do not require prior training to learn pose. They require a strong prior on the pose

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 782–792, 2012.

to iteratively evolve to actual pose. Typically, they recover pose by first establishing 2-3D feature correspondence and then solving for the pose using a pose estimation technique [9]. Based on the type of feature, they are further divided into template based and keypoint based approaches. Template based approaches [14], [15], [16] estimate pose of a reference template by minimizing an error measure based on image brightness [13]. In general, they work under diffused lighting, no occlusion, and small interframe displacement [12], [25]. Keypoint based approaches [12], [17], [18], [4] exploit local appearance of targets. They work opposite to template based approaches but relatively computationally expensive [25]. A common problem with the second type of approaches is pose drift due to error accumulation over long sequences [25]. In general, pose drift is delayed by approximating target pose by modeling or predicting its motion, or selecting features that are least vulnerable to add to the drift.

Approaches [3], [12], [6], [7] also exist that combine more than one type of approaches with intention to increase accuracy and/or achieve robustness. There is none that simultaneously addresses large interframe displacement and flexibility to adapt change in scene. Moreover, rather than fusing they work either by feeding output of one approach to second approach or by switching between the two approaches. The present approach intrinsically assimilate template based and keypoint based tracking due to their complementary role in achieving the goal. In contrast to estimate pose from pre-learned samples, deformation in the template is used. In place of intensity, point based error measure is defined to find the deformation. Tasks of detection and pose estimation are performed simultaneously without imposing constraints on background. The approach intrinsically delays pose drift.

Rest of the paper is divided into following sections. Section 2 describes fusing between the two approaches. Evaluation of the resultant approach is presented in Section 3. In the end Section 4 concludes the work.

2 Fusing Point and Template Information

To fuse point information into template, template based tracking introduced by Mei et al. [14] is chosen. Their tracking is based on efficient second order minimization (ESM) algorithm, which achieves second order convergence at computational cost and consequently speed of first order methods. The algorithm performs better in terms of accuracy and convergence than the alternatives [23], [15], [16]. Pseudocode of the algorithm with point information incorporated is given in figure 1.

Let I_1 be a reference image of a monocular sequence I_k, $k = 1...K$, such that a region I_{ref} (reference patch) of this contains projection of the planar target. Given an approximate transformation $\tilde{T}$ consisting of rigid motion (rotation $\tilde{R}$, translation $\tilde{t}$) in terms of camera motion, features F_{ref} extracted from I_{ref}, and a set of thresholds, the algorithm returns the actual T. Theoretically, it is equivalent to map I_{ref} to desired region defined by T in the current image I_k that minimizes sum of square distance (SSD) over all feature points.

```
Input: $I_1$, $I_{ref}$, $I_k$, $F_{ref}$, $\tilde{T}$, thresholds(maxIter,num,err)
Output: T

Iter = 0

While(iter < maxIter)

    Compute $\tilde{R}$, $\tilde{t}$
    H = $\tilde{R} + \tilde{t}n_d'$
    $I_{cur}$ = definePatch($I_k$, $I_{ref}$, H)
    $F_{cur}$ = extractFeatures($I_{cur}$)
    matches = matchFeatures($F_{ref}$, $F_{cur}$)
    removeOutliers(matches)

    x = $-2J^+D(\mathbf{0})$

    if  $\|x\|$ < err
        T = $\tilde{T}$
        break
    else
        $\tilde{T}$ = T(x)$\tilde{T}$
    end

end
```

Fig. 1. Pseudocode of the ESM algorithm fused with point information

Algorithm starts by computing transformation of the target in image plane using homography H associated to $\tilde{T}$. Such that

$$\tilde{T} = \begin{bmatrix} \tilde{R} & \tilde{t} \\ \mathbf{0} & 1 \end{bmatrix} \tag{1}$$

$$H = (\tilde{R} + \tilde{t}n_d^{'}) \tag{2}$$

$$p = \pi(w(H(\tilde{T})))\pi^{-1}(p^*) \tag{3}$$

where n_d is a vector defined as $\frac{n}{d}$ consisting of normal n and distance d of the target plane from camera. w is a warping function that defines a coordinate transformation between points on a unit plane (normalized plane). π is a projection function that defines projection of a point on the unit plane to image plane. Practically, this is to find the new position p in the current image of a pixel p^* in the reference image. With this a patch I_{cur} (current patch) in the current image is defined. This leads to four benefits. One region to search the target in image is confined. Second explicit detection or segmentation of the target is avoided. Third likelihood of correspondence with background is reduced. Fourth background is intrinsically ignored which in turn makes background dynamics irrelevant.

In the next step features F_{cur} are extracted from the defined patch. The Scale Invariant Feature Transform (SIFT) is an approach for extracting local features that are reasonably invariant to scaling, translation, rotation, illumination changes, image noise, affine distortion, occlusion, and viewpoint change [24].

Further motivation comes from its use in real-time tracking on mobile phones [8]. Therefore, this work uses SIFT. Outcome of this step is a set of interest points each comprised of a feature point and its descriptor. A feature point corresponds to 2D location $[u\ v]'$ and is described using its local neighborhood.

Extracted features are then matched with the F_{ref} using K-d tree. False correspondence is avoided by first removing points with multiple correspondence. Then further removing whose Euclidian distance and slope exceeds a specific range. Based on the number of features, empirically determined two strategies are employed. If the number exceeds 40, Gaussian distribution is assumed and the range is defined by equation 4. Otherwise, it is defined by equation 5.

$$\mathit{Mean}\ \{\mathit{slope,\ distance}\} \pm 1.5 \times \mathit{its\ Standard\ deviation} \tag{4}$$

$$\mathit{Median}\ \{\mathit{slope,\ distance}\} \pm 0.66 \times \mathit{Median}\ \{\mathit{slope,\ distance}\} \tag{5}$$

Theoretically, four non-collinear points are sufficient [19]. However, the four-points assumption is viable if projection is linear, calibration is perfect, image is noise free, feature points are located accurately, and there is no false correspondence. Which is hard to attain. Therefore, several approaches [20], [21], [22] including this work, use as many points as are available. Once outliers are removed, cost of matching is then computed between the corresponding points. Let the corresponding points are $\{l_{i,j}\}$ and $\{m_{i,j}\}$ in the reference and current patches respectively. Let D_q be the distance between q^{th} pair of corresponding points, the cost is defined as:

$$\forall i \in 1, 2, ..., q \quad D_i = l_i - m_i \tag{6}$$

If the SSD value of vector D approaches to zero, the estimated pose becomes equal to the actual pose. Tracking jumps to the next image. Otherwise, we need to update $\tilde{T}$. Let the update is denoted by $T(x)$. Where x is a parameter vector that consists of coefficients of base elements: three for translation B_1-B_3 and three for rotation B_4-B_6 such that

$$T(x) = exp(\sum_{i=1}^{6} x_i B_i) \tag{7}$$

$$B_1 = \begin{bmatrix} 0\,0\,0\,1 \\ 0\,0\,0\,0 \\ 0\,0\,0\,0 \\ 0\,0\,0\,0 \end{bmatrix} B_2 = \begin{bmatrix} 0\,0\,0\,0 \\ 0\,0\,0\,1 \\ 0\,0\,0\,0 \\ 0\,0\,0\,0 \end{bmatrix} B_3 = \begin{bmatrix} 0\,0\,0\,0 \\ 0\,0\,0\,0 \\ 0\,0\,0\,1 \\ 0\,0\,0\,0 \end{bmatrix}$$

$$B_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} B_5 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} B_6 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

More precisely, the problem of pose estimation is to minimize the cost of matching which in terms of the parameter vector can be described as

$$\forall i \in 1, 2, ..., q \ \ D_i(x) = \pi(w(H(T(x)\tilde{T})))\pi^{-1}(l_i) - m_i \tag{8}$$

Minimizing this expression is a nonlinear optimization task. Let cost function $D(x)$ be the vector $[D_1(x)\ D_2(x)\ D_3(x)\ ...\ D_q(x)]'$ that corresponds to the distance over all points at the given parameter vector x. By the second order approximation of $D(x)$ about $x = \mathbf{0}$ using Taylor series and simplification [14]

$$D(x) \approx D(\mathbf{0}) + \frac{1}{2} J x \tag{9}$$

$$J = J_\pi J_w J_H J_T \tag{10}$$

where J is jacobian of the D with respect to the x. In contrast to the original Jacobian which is composed of jacobians of each of image J_I, image projection function J_π, image warping function J_w, homography J_H, and transformation J_T. In this work, it is composed of the other fours except J_I. This is to reduce non-linearity in the cost function. In the former case there are two factors that introduce non-linearity in the cost function. First corresponds to non-linear projection and second corresponds to intensity information. In fact pixel values are essentially un-related to pixel coordinates [15], therefore, J_I is ignored. This allows to use fewer detail, regions in spite of the complete reference patch. Moreover, impact of non-linearity should be reduced as the cost function is better defined. Outcome of testing with simulated sequence confirms this. Expression of each of the jacobian for each feature point l_i normalized to the unit plane is

$$J_\pi = \nabla_P \pi(P)|_{P=l_i} \tag{11}$$

$$J_w = \nabla_H (w(H))(P)|_{H=H(0)=I} \tag{12}$$

$$J_H = \nabla_T H(\tilde{T})^{-1} H(T\tilde{T})|_{T=T(0)=I} \tag{13}$$

$$J_T = \nabla_x T(x)|_{x=0} \tag{14}$$

Solution to the problem lies in finding a parameter vector x_0 such that $D(x_0) = \mathbf{0}$. This is obtained by iteratively solving the cost function such that for a vector $x = x_0$ we have

$$\nabla D(x)|_{x=x_0} = \mathbf{0} \tag{15}$$

At each iteration an updated x is calculated as follows

$$x = -2J^+ D(\mathbf{0}) \tag{16}$$

where $D(\mathbf{0})$ is the cost at $x = \mathbf{0}$, and J^+ means pseudo-inverse of J. Once convergence ($\|x\| < err$) is achieved in the current image, we obtain the optimal transformation T_k between the reference I_1 and the current image I_k. The algorithms finishes with this image and restarts with the next image I_{k+1}. Let $T_k(x_0)$ be the relative transformation between the last two consecutive frames I_{k-1} and I_k. Pose estimation starts in the next image with the following approximation

$$\tilde{T}(k+1) = T_k(x_0) T_k \tag{17}$$

Tracking continues till the last image I_K is reached and we find a total transformation T_K without pose drift.

3 Results and Discussion

Evaluation of the approach is made using both the simulated and real sequences. In the first case, it is made with reference to the Mei et al. using the same simulated sequence on which the referenced approach was tested. Figure 2 shows three images of the simulated sequence.

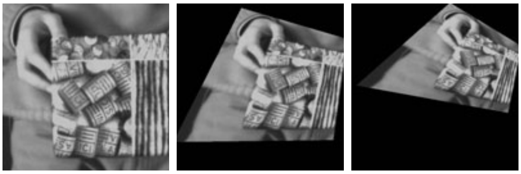


Fig. 2. Images 1, 50, and 100 respectively in the simulated sequence

Figure 3 shows outcome in terms of absolute translational error, absolute rotational error, and number of iterations elapsed to converge by increasing interframe displacement. The interframe displacement is increased by skipping multiple images at regular intervals from the original sequence. Started by skipping alternate images and ending with two images in the sequence. One can see by fusing point information into the pure template based tracking both the errors remain more than half below. The errors oscillate in the beginning for the reason of small baseline effect while stabilizes later. In the case of number of iterations, although difference between the two is small in the beginning but immediately that is after skipping just two images raises dramatically. The proposed approach showed consistent behavior. The most considerable fact is that the referenced approach fails tracking beyond 10 number of images skipped. This is due to its reliance on strong prior on the pose.

In the second case, the approach is tested with real sequences. These sequences consist of flight of ten cubical objects thrown horizontally across the principal axis of camera. For each object 50 sequences are collected. They are thrown at a distance of $1.6_{\pm 0.45}$ m from camera with their largest plane exposed to the camera. Figure 4 shows the planes. Their sizes and number of features extracted from each plane at this distance are given in Table 1. Horizontal field of view at this distance is 1.2 m. Before leaving field of view, they lie at 1.44±0.45 and 0.07±0.21 m from camera along Z and Y axis respectively. Another calibrated camera is used to find the distances and normal to the plane using stereo vision. The range of estimated rotation along each of the X, Y, and Z axis is -37.93 to 35.08, -30.50 to 50.90, and -15.50 to 44.68 degrees respectively.

Evaluation is made using the methodology introduced in [26]. Corners of the plane are set as reference points. Their image coordinates are used as a ground truth. The ground truth is generated manually. Accuracy in each image is then determined with the help of root mean square (RMS) distance between the estimated position of the reference point p and its corresponding ground truth p^*. Such that

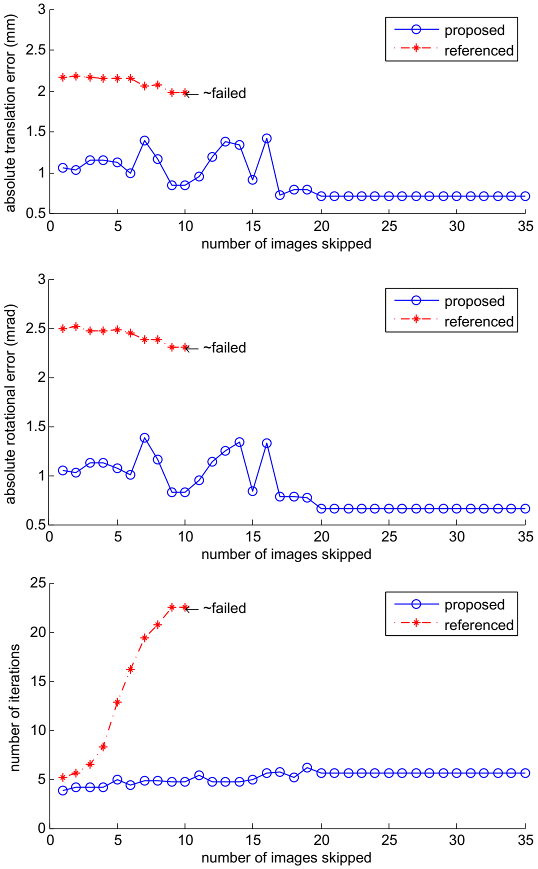


Fig. 3. Comparison based on interframe displacement

Table 1. Sizes and feature amount of the planes

Plane	**Number of features**	**Size** (mm^2)
Daisy	155	300 × 160
Garment	177	
Donuts	99	285 × 120
Monster	130	
Rice	185	250 × 160
Chicken	114	
China	188	200 × 175
Biscuit	107	
Juice	69	240 × 115
Bravo	102	

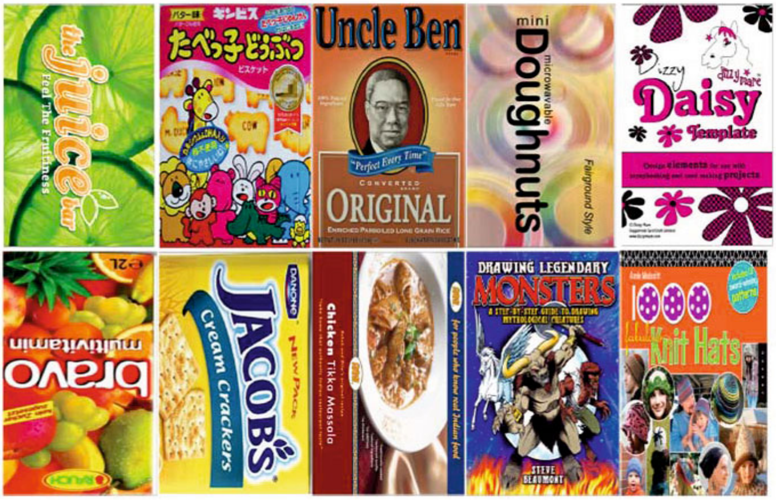


Fig. 4. Planes of the objects and their assigned names. Top to bottom followed by left to right: (a) Juice, (b) China, (c) Rice, (d) Donuts, (e) Daisy, (f) Bravo, (g) Biscuit, (h) Chicken, (i) Monster, and (j) Garment.

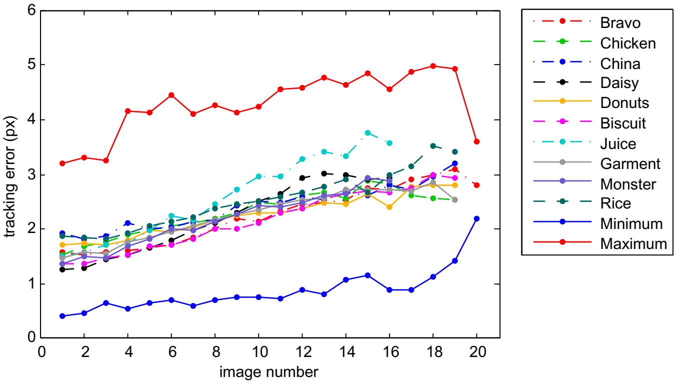


Fig. 5. Error in tracking real sequences

$$tracking\ error = \sqrt{\frac{1}{4}\sum_{i=1}^{4}\|p_i - p_i^*\|^2} \tag{18}$$

Figure 5 shows tracking error on average and extreme bases. For each plane the average is taken per image over all the 50 throws. The uppermost line shows the maximum error, whereas, the lowermost line shows the minimum error. One can see the approach performs equally well in all the cases except Juice. This is due its much lower amount of texture (number of features) relative to the rest. A common trend among all the planes is that the error increases with the increase

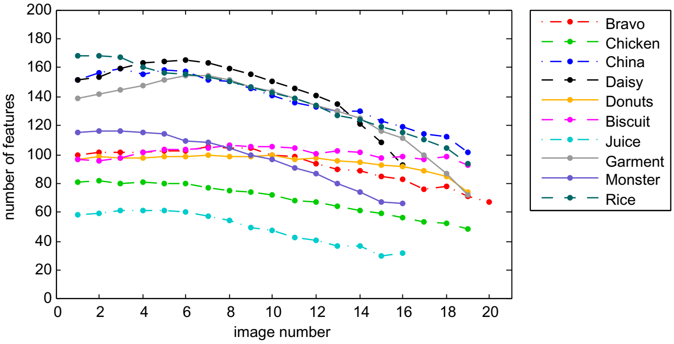


Fig. 6. Feature decay

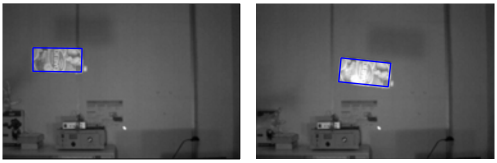


Fig. 7. Two images of a sequence in which appearance of the plane changes due to non-diffused lighting

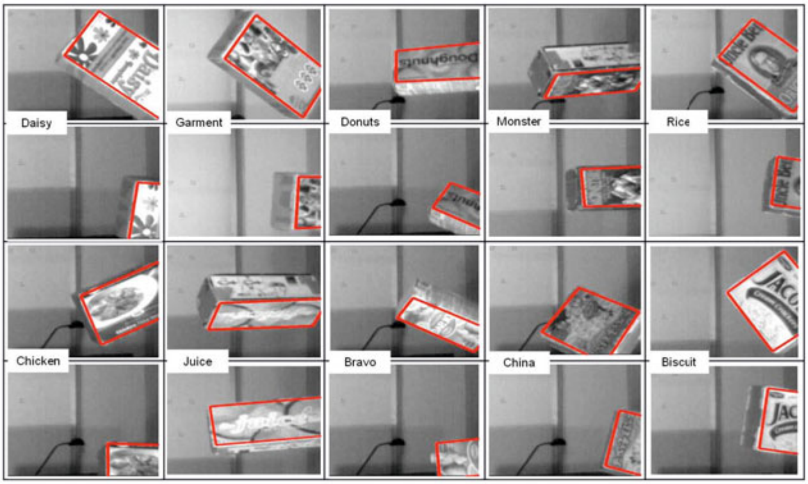


Fig. 8. Two instances of tracking each plane under extreme occlusion

in image number. This is partially due to error accumulation and partially due to loss in features. The loss is due to throwing objects in front of the camera. So in the subsequent frames planes move farther and farther from the camera, which results in losing finer texture. Figure 6 shows decrease in feature amount on average bases with the increase in image number. The average is taken per image over all the 50 throws. The interframe displacement was large enough that in no case the referenced approach is able to track the plane.

Sequences are acquired without diffused lighting. Lights are used only to enhance brightness, hence to reduce exposure time of the camera. So one can expect change in illumination with the change in plane orientation. Figure 7 confirms this. Having success with this shows robustness of the approach against illumination changes. To further show robustness of the approach against partial occlusion, Figure 8 present two instances of tracking under extreme occlusion for each plane before it leaves field of view.

4 Conclusion

A hybrid approach by fusing point and template based tracking to track planar-textured targets with large interframe displacement is introduced. The approach is flexible to adapt change in scene. Its evaluation is made using both the simulated and real sequences. In the first case, the approach performs better in terms of accuracy, convergence, and interframe displacement. In the second case, a consistent behavior is seen with the change in target. Robustness of the approach against partial occlusion and illumination changes is also shown. One may argue the approach is computationally expensive in terms of feature employed. To compensate this a part of image is exploited. Moreover, faster convergence further weakens the argument, particularly, when the interframe displacement is large. In future, its implementation on real time will be analyzed.

References

1. Masson, L., Dhome, M., Jurie, F.: Robust Real Time Tracking of 3D Objects. In: International Conference on Pattern Recognition, UK, pp. 252–255 (2004)
2. Bjorkman, M., Kragic, D.: Combination of Foveal and Peripheral Vision for Object Recognition and Pose Estimation. In: IEEE International Conference on Robotics and Automation, USA, pp. 5135–5140 (2004)
3. Choi, C., Christensen, H.I.: Real-time 3D model-based tracking using edge and keypoint features for robotic manipulation. In: IEEE International Conference on Robotics and Automation, USA, pp. 4048–4055 (2010)
4. Collet, A., Berenson, D., Srinivasa, S.S., Ferguson, D.: Object recognition and full pose registration from a single image for robotic manipulation. In: IEEE International Conference on Robotics and Automation, Japan, pp. 3534–3541 (2009)
5. Lepetit, V., Pilet, J., Fua, P.: Point Matching as a Classification Problem for Fast and Robust Object Pose Estimation. In: International Conference on Computer Vision and Pattern Recognition, USA, pp. 244–250 (2004)
6. Rosten, E., Drummond, T.: Fusing points and lines for high performance tracking. In: IEEE International Conference on Computer Vision, China, pp. 1508–1515 (2005)
7. Vacchetti, L., Lepetit, V., Fua, P.: Combining Edge and Texture Information for Real-Time Accurate 3D Camera Tracking. In: IEEE and ACM International Symposium on Mixed and Augmented Reality, USA, pp. 48–57 (2004)

8. Wagner, D., Reitmayr, G., Mulloni, A., Drummond, T., Schmalstieg, D.: Pose tracking from natural features on mobile phones. In: IEEE and ACM International Symposium on Mixed and Augmented Reality, UK, pp. 125–134 (2008)
9. Ji, Q.: 3D Face pose estimation and tracking from a monocular camera. Image and Vision Computing 24, 499–511 (2002)
10. Azad, P.: State of the Art in Object Recognition and Pose Estimation. Cognitive Systems Monographs: Visual Perception for Manipulation and Imitation in Humanoid Robots 04, 7–47 (2009)
11. Ekvall, S., Kragic, D., Hoffmann, F.: Object recognition and pose estimation using color cooccurrence histograms and geometric modeling. Image and Vision Computing 23, 943–955 (2005)
12. Ladikos, A., Benhimane, S., Navab, N.: High Performance Model-Based Object Detection and Tracking. Theory and Applications: Computer Vision and Computer Graphics 21, 191–204 (2009)
13. Benhimane, S., Malis, E.: Homography-based 2D Visual Tracking and Servoing. International Journal of Robotics Research 26, 661–676 (2007)
14. Mei, C., Benhimane, S., Malis, E., Rives, P.: Efficient Homography-Based Tracking and 3-D Reconstruction for Single-Viewpoint Sensors. IEEE Transactions on Robotics 24, 1352–1364 (2008)
15. Baker, S., Matthews, I.: Lucas-Kanade 20 Years On: A Unifying Framework. International Journal of Computer Vision 56, 221–255 (2004)
16. Jurie, F., Dhome, M.: Hyperplane Approximation for Template Matching. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 996–1000 (2002)
17. Vacchetti, L., Lepetit, V., Fua, P.: Stable Real-Time 3D Tracking Using Online and Offline Information. IEEE Transactions on Pattern Analysis and Machine Intelligence 26, 1385–1391 (2004)
18. Johnson, J.E., Hebert, M.: Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence 21, 433–449 (1999)
19. Schweighofer, G., Pinz, A.: Robust Pose Estimation from a Planar Target. IEEE Transactions on Pattern Analysis and Machine Intelligence 28, 2024–2030 (2006)
20. Arun, K.S., Huang, T.S., Blostein, S.D.: Least squares fitting of two 3D point sets. IEEE Transactions on Pattern Analysis and Machine Intelligence 09, 698–700 (1987)
21. Bolle, R.M., Cooper, D.B.: On optimally combining pieces of information with application to estimating 3D complex-object position from range data. IEEE Transactions on Pattern Analysis and Machine Intelligence 08, 619–638 (1986)
22. Haralick, R.M., Joo, H., Lee, C.N., Zhuang, X., Vaidya, V.G., Kim, M.B.: Pose estimation from corresponding point data. IEEE Transactions on Systems, Man, and Cybernetics 19, 1426–1445 (1989)
23. Hager, D.G., Belhumeur, P.N.: Efficient Region Tracking With Parametric Models of Geometry and Illumination. IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 1025–1039 (1998)
24. Sangle, P., Kutty, K., Patil, A.: A Method for Generation of Panoramic View based on Images Acquired by a Moving Camera. IJCA Special Issue on Artificial Intelligence Techniques - Novel Approaches and Practical Applicationse 03, 11–14 (2011)
25. Lepetit, V., Fua, P.: Monocular model-based 3D tracking of rigid objects. Foundations and Trends in Computer Graphics and Vision 01, 1–89 (2005)
26. Lieberknecht, S., Benhimane, S., Meier, P., Navab, N.: Benchmarking template-based tracking algorithms. Virtual Reality 15, 99–108 (2010)

Author Index