Kyoung Mu Lee
Yasuyuki Matsushita
James M. Rehg
Zhanyi Hu (Eds.)

LNCS 7727

Computer Vision – ACCV 2012

11th Asian Conference on Computer Vision
Daejeon, Korea, November 2012
Revised Selected Papers, Part IV

Springer

Lecture Notes in Computer Science 7727

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Kyoung Mu Lee Yasuyuki Matsushita
James M. Rehg Zhanyi Hu (Eds.)

Computer Vision – ACCV 2012

11th Asian Conference on Computer Vision
Daejeon, Korea, November 5-9, 2012
Revised Selected Papers, Part IV

Springer

Volume Editors

Kyoung Mu Lee
Seoul National University
Department of Electrical and Computer Engineering
1 Gwanak-ro, Gwanak-gu, 151-744 Seoul, Korea
E-mail: kyoungmu@snu.ac.kr

Yasuyuki Matsushita
Microsoft Research Asia
No. 5, Danling st., Haidian District, 100080 Beijing, P.R. China
E-mail: yasumat@microsoft.com

James M. Rehg
Georgia Institute of Technology
School of Interactive Computing
801 Atlantic Drive, CCB 315, Atlanta, GA 30332, USA
E-mail: rehg@gatech.edu

Zhanyi Hu
Chinese Academy of Sciences
Institute of Automation
National Laboratory of Pattern Recognition
Zhong Quan Cun East Road 95, Haidian District, 100190 Beijing, P.R. China
E-mail: huzy@nlpr.ia.ac.cn

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-37446-3 e-ISBN 978-3-642-37447-0
DOI 10.1007/978-3-642-37447-0
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2013934230

CR Subject Classification (1998): I.4.1-10, I.5.1-4, I.2.10, I.2.6, I.3.5, H.3.4, H.2.8, F.2.2

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

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 11th Asian Conference on Computer Vision (ACCV 2012) took place in South Korea in the city of Daejeon, a well-known center of research and high-tech industry. Following the tradition of previous meetings, ACCV 2012 had a number of events co-located with the main conference, including nine workshops, two tutorial sessions, 12 on-site demos featuring a wide range of advanced vision technology, and a special competition on RGB-D camera applications. In addition, there were three keynote speakers: Tomaso Poggio (Invariant Recognition in the Visual Cortex), Du Sik Park (The Color and Image Processing Technology for CE Device: Current and Future), and Andrew Fitzgibbon (3D Vision in a Changing World).

The ACCV Steering Committee, consisting of Katsushi Ikeuchi, Yasushi Yagi, and Tieniu Tan, provided guidance throughout the organizational process and we are grateful for their support. We were fortunate to be able to work closely with the General Chairs, In So Kweon, Chilwoo Lee, and Akihiro Sugimoto, who arranged the financing and logistics. Thanks to their efforts we were able to secure the Daejeon Convention Center as an excellent venue for our meeting. Special thanks go to our Publication Chairs In Kyu Park and Tae-Wuk Bae, for handling the daunting task of assembling the conference proceedings and meeting the publication deadlines.

Additional support for ACCV 2012 was provided by our 13 sponsors, who contributed at four levels: Platinum (Daejeon Metropolitan City, Daejeon International Marketing Enterprise, Daejeon Convention Center, Korea Tourism Organization, DigiCar Center, Mobile Device Interface Research Center, and Seoul National University), Gold (Samsung AIT and Puloon Technology), Silver (Mando Corporation, Qualcomm, and 4D View Solutions), and Bronze (NVIDIA Corporation).

In order to support an on-line review process, we utilized Microsoft's CMT system, with special thanks to Yasuyuki Matsushita for managing the CMT process. Continuing the trend of increasing submissions to ACCV, we received 869 submissions by the deadline of July 1, 2012. This represents an 18% increase in submissions over 2010. We received submissions from 43 countries, with Asia (63%), Europe (23%), and North America (12%) making up the bulk of the submissions by region. Submitted papers that did not conform to the submission criteria regarding author anonymity, formatting, and length, were desk rejected and removed from consideration.

The four Program Co-chairs assembled a group of 33 leading vision researchers to serve as Area Chairs (ACs) and conduct the review process. These Chairs managed a group of 479 reviewers, who provided expert assessment of the submitted papers. Each paper received a minimum of three reviews, as well as a consolidation report from the responsible AC, which detailed the outcome of

the decision process. Review decisions were finalized at the AC meeting, which was held at Seoul National University during September 17–18, 2012. Special thanks to Kyoung Mu Lee for handling the arrangements for this meeting. ACs were organized into triples, so that papers with varying review scores could be discussed by multiple ACs. The triples in turn were organized into four panels, which finalized all of the paper decisions. The AC panels were instructed to use their best judgement in determining which papers to accept. While review scores were an input to the decision process, these scores alone did not determine the outcome. The Program Chairs strictly followed the recommendations of the panels with regard to acceptance. We asked for clarification where it was needed, and requested detailed and clear consolidation reports. Each consolidation report was checked by at least one Program Chair.

We wish to acknowledge the invaluable help of a number of people in making this conference possible. The logistical talents of the Organizing Committee made it possible to conduct a well-run meeting with a diverse set of activities. We extend our thanks to everyone who was involved in the submission and review process: the ACs, reviewers, and authors. Without your dedication and hard work there would be no meeting. We look forward to the continuing evolution of ACCV as one of the top conferences in the field.

November 2012

Kyoung Mu Lee
Yasuyuki Matsushita
James M. Rehg
Zhanyi Hu

Organization

Steering Committee

Katsushi Ikeuchi	The University of Tokyo, Japan
Yasushi Yagi	Osaka University, Japan
Tieniu Tan	The National Laboratory of Pattern Recognition, China

General Chairs

In So Kweon	KAIST, Korea
Chilwoo Lee	Chonnam National University, Korea
Akihiro Sugimoto	National Institute of Informatics, Japan

Program Chairs

Kyoung Mu Lee	Seoul National University, Korea
Yasuyuki Matsushita	Microsoft Research Asia, China
Jim Rehg	Georgia Institute of Technology, USA
Zhanyi Hu	Chinese Academy of Science, China

Workshop Chairs

Jongil Park	Hanyang University, Korea
Junmo Kim	KAIST, Korea
Hideo Saito	Keio University, Japan
Yanxi Liu	The Pennsylvania State University, USA
Ming-Hsuan Yang	University of California at Merced, USA

Finance Chair

Kiryong Kwon	Bukyung National University, Korea

Publication Chairs

In Kyu Park	Inha University, Korea
Tae-Wuk Bae	Stanford University, USA

Publicity Chairs

Chang-Su Kim	Korea University, Korea
Burkhard Wunsche	University of Auckland, New Zealand
Takeshi Oishi	The University of Tokyo, Japan
Robert Fisher	University of Edinburgh, UK

Web Chair

Kanghyun Jo	University of Ulsan, Korea

Demo Chairs

Il dong Yun	Hankuk University of Foreign Studies, Korea
Yongduek Seo	Sogang University, Korea
Hajime Nagahara	Kyushu University, Japan
Tat Jen Cham	Nanyang Technological University, Singapore

Tutorial Chairs

Chang Dong Yoo	KAIST, Korea
Yoshinori Kuno	Saitama University, Japan
Michael S. Brown	National University of Singapore, Singapore

Local Chairs

Kuk-Jin Yoon	GIST, Korea
Jongwoo Lim	Hanyang University, Korea
Ju Yong Chang	ETRI, Korea

Special Session Chair

Yu-Wing Tai	KAIST, Korea

Industrial Chair

Chang Yeong Kim	Samsung Advanced Institute of Technology, Korea

Area Chairs

Serge Belongie	University of California, San Diego, USA
Michael Brown	National University of Singapore, Singapore
Nam Ik Cho	Seoul National University, Korea
Robert Collins	The Pennsylvania State University, USA

Larry Davis	University of Maryland, USA
Kristen Grauman	University of Texas at Austin, USA
Abhinav Gupta	Carnegie Mellon University, USA
Bohyung Han	POSTECH, Korea
Richard Hartley	Australian National University, Australia
Jiaya Jia	Chinese University of Hong Kong, Hong Kong
Neel Joshi	Microsoft Research, USA
Koichi Kise	Osaka Prefecture University, Japan
Nikos Komodakis	University of Crete, Greece
Sang Wook Lee	Sogang University, Korea
Ales Leonardis	University of Ljubljana, Slovenia
Vincent Lepetit	EPFL, Switzerland
Yasuhiro Mukaigawa	Osaka University, Japan
Nikos Paragios	Ecole Centrale de Paris, France
Shmuel Peleg	The Hebrew University of Jerusalem, Israel
Hideo Saito	Keio University, Japan
Imari Sato	National Institute of Informatics, Japan
Shin'ichi Satoh	National Institute of Informatics, Japan
Shiguang Shan	Chinese Academy of Sciences, China
Jianbo Shi	University of Pennsylvania, USA
Cristian Sminchisescu	Universität Bonn, Germany
Chi-Keung Tang	HKUST, Hong Kong
Marshall Tappen	University of Central Florida, USA
Fernando de la Torre	Carnegie Mellon University, USA
Kenneth K.-Y.Wong	Hong Kong University, Hong Kong
Jianxin Wu	Nanyang Technological University, Singapore
Shuicheng Yan	National University of Singapore, Singapore
Ming-Hsuan Yang	University of California, Merced, USA
Ruigang Yang	University of Kentucky, USA

Program Committee Members

Austin D. Abrams	Atsuhiko Banno
Catherine Achard	Yufang Bao
Emre Akbas	Adrian Barbu
Karteek Alahari	Nick Barnes
Mitsuru Ambai	John Barron
Bjoern Andres	Abdessamad Ben Hamza
Gaston R. Araguas	Chiraz BenAbdelkader
Nafiz Arica	Moshe Ben-Ezra
Yasuo Ariki	AndrewTeoh Beng-Jin
Chetan Arora	Achraf Ben-Hamadou
Abdullah Arslan	Benjamin Berkels
Xiang Bai	Horst Bischof
Vineeth Balasubramanian	Prabir Biswas

Soma Biswas
Matthew Blaschko
Konstantinos Blekas
Adrian Bors
Michael Boshra
Nizar Bouguila
Edmond Boyer
Steve Branson
Michael M. Bronstein
Andres Bruhn
Asad A. Butt
Ricardo S. Cabral
David W. Cai
Jinhai Cai
Francesco Camastra
Xiaochun Cao
Xun Cao
Barbara Caputo
Joao Carreira
Yaron Caspi
Umberto Castellani
Turgay Celik
Kap Luk Chan
Kwok-Ping Chan
Sharat Chandran
Hong Chang
Vincent Charvillat
Rama Chellappa
Bing-Yu Chen
Chu-Song Chen
Haifeng Chen
Hwann-Tzong Chen
Jie Chen
Jiun-Hung Chen
Ling Chen
Qiang Chen
Terrence Chen
Tsuhan Chen
Xiangyu Chen
Xiaowu Chen
Hong Cheng
MingMing Cheng
Shyi-Chyi Cheng
Yuan Cheng
Liang-Tien Chia
Shao-Yi Chien
Tat-Jun Chin
Minsu Cho
Wen-Sheng Chu
Yung-Yu Chuang
Albert CS Chung
Pan Chunhong
Arridhana Ciptadi
Javier Civera
Carlo Colombo
Jason Corso
Marco Cristani
Beleznai Csaba
Jinshi Cui
Jeremiah D. Deng
Qieyun Dai
Kostas Daniilidis
Petros Daras
Francois de Sorbier
Fatih Demirci
Joachim Denzler
Anthony Dick
Santosh Divvala
Csaba Domokos
Qiulei Dong
Test Dong
Michael Donoser
Gianfranco Doretto
Bruce Draper
Fuqing Duan
Zoran Duric
Ulrich Eckhardt
Michael Eckmann
Wolfgang Einhauser
Hazim Ekenel
Francisco Escolano
Jialue Fan
Wen-Pinn Fang
Micha Feigin
Jianjiang Feng
Jiashi Feng
Francesc J. Ferri
Pierre Fite Georgel
Katerina Fragkiadaki
Juan Francisco Giro Martín

Chi-Wing Fu
Chiou-Shann Fuh
Hironobu Fujiyoshi
Giorgio Fumera
Ryo Furukawa
Juergen Gall
Li Gang
Jun Hong Gao
Yongsheng Gao
Weina Ge
Andreas Geiger
Arkadiusz Gertych
Bernard Ghanem
Guy Godin
Roland Goecke
Bastian Goldluecke
Yunchao Gong
Bogdan T. Goras
Stephen Gould
Hayit Greenspan
Irene Gu
Josechu Guerrero
Richard Guest
Guodong Guo
Yanwen Guo
Yaniv Gur
Vu Hai
Lin Hai-Ting
Kiana Hajebi
Peter Hall
Onur Hamsici
Hu Han
Mei Han
Tony Han
Allan Hanbury
Zhou Hao
Kenji Hara
Tatsuya Harada
Osman Hassab Elgawi
Jean-Bernard Hayet
Junfeng He
Ran He
Joon Hee Han
Shinsaku Hiura
Jeffrey Ho
Yo-Sung Ho
Christopher Hollitt
Hyunki Hong
Ki Sang Hong
Kazuhiro Hotta
Seiji Hotta
Edward Hsiao
Winston Hsu
Gang Hua
Chunsheng Hua
Chun-Rong Huang
Dong Huang
Fay Huang
Jonathan Huang
Kaiqi Huang
Peter Huang
Xinyu Huang
Benoit Huet
Yi-Ping Hung
Mohamed Hussein
Cong Phuoc Huynh
Sung Ju Hwang
Naoyuki Ichimura
Ichiro Ide
Yoshihisa Ijiri
Sei Ikeda
Nazli Ikizler-Cinbis
Atsushi Imiya
Kohei Inoue
Catalin Ionescu
Rui Ishiyama
Yoshio Iwai
Nathan Jacobs
Arpit Jain
Yangqing Jia
Yunde Jia
Shuqiang Jiang
Xiaoyi Jiang
Yu-Gang Jiang
Nianjuan Jiang
Yushi Jing
Kang-Hyun Jo
Matjaz Jogan
Manjunath V. Joshi
Frederic Jurie

Shingo Kagami
Zdenek Kalal
Amit Kale
George Kamberov
Kenichi Kanatani
Atul Kanaujia
Henry Kang
Sing Bing Kang
Mohan Kankanhalli
Abou-Moustafa Karim
Zoltan Kato
Harish Katti
Rei Kawakami
Hiroshi Kawasaki
Mark Keck
Sang Keun Lee
Saad-Masood Khan
Aditya Khosla
Hansung Kim
Kyungnam Kim
Sungwoong Kim
TaeHoon Kim
Tae-Kyun Kim
Benjamin Kimia
Ron Kimmel
Yasuyo Kita
Itaru Kitahara
Kris Kitani
Reinhard Klette
Georges Koepfler
Mario Koeppen
Kevin Koeser
Effrosyni Kokiopoulou
Iasonas Kokkinos
Alexander Kolesnikov
Sotiris B. Kotsiantis
Junghyun Kown
Norbert Kruger
Arjan Kuijper
Kashino Kunio
Yoshinori Kuno
Cheng-Hao Kuo
Suha Kwak
Bogdan Kwolek
Junseok Kwon

Ľubor Ladický
Alexander Ladikos
Shang-Hong Lai
Antony Lam
Zhiqiang Lao
Longin Jan Latecki
Francois Lauze
Duy-Dinh Le
Chan-Su Lee
Guee Sang Lee
Jae-Ho Lee
Seungyong Lee
Taehee Lee
Christian Leistner
Bocchi Leonardo
Marius Leordeanu
Matt Leotta
Wee-Kheng Leow
Bruno Lepri
Frederic Lerasle
Thomas Leung
Annan Li
Fuxin Li
Hongdong Li
Jia Li
Li-Jia Li
Rui Li
Yongmin Li
Yufeng Li
Chia-Kai Liang
Shu Liao
T. Warren Liao
Wen-Nung Lie
Jenn-Jier J. Lien
Jongwoo Lim
Joo-Hwee Lim
Joseph J. Lim
Ser-Nam Lim
Hai Ting Lin
Huei-Yung Lin
Weiyao Lin
Wen-Chieh(Steve) Lin
Zhouchen Lin
Haibin Ling
Baoyuan Liu

Cheng-Lin Liu
Hairong Liu
Jingchen Liu
Ligang Liu
Miaomiao Liu
Qingzhong Liu
Si Liu
Tianming Liu
Tyng-Luh Liu
Xiaobai Liu
Xiaoming Liu
Marco Loog
Huchuan Lu
Juwei Lu
Le Lu
Tong Lu
Ludovic Macaire
Anant Madabhushi
Subhransu Maji
Atsuto Maki
Yasushi Makihara
Koji Makita
Yoshitsugu Manabe
Rok Mandeljc
Al Mansur
Gian-Luca Marcialis
Tim Marks
Stephen Marsland
Jean Martinet
Aleix Martinez
Syed Zain Masood
Takeshi Masuda
Thomas Mauthner
Stephen J. Maybank
Kenton McHenry
Stephen McNeill
Gerard Medioni
Ramin Mehran
Domingo Mery
David Michael
Gregor Miller
Washington Mio
Ikuhisa Mitsugami
Anurag Mittal
Daisuke Miyazaki
Yoshihiko Mochizuki
Pascal Monasse
Vlad I. Morariu
Greg Mori
Bryan Morse
Yadong Mu
Jayanta Mukhopadhyay
Henning Mller
Hajime Nagahara
Shin-ichi Nakajima
Atsushi Nakazawa
Woonhyun Nam
Loris Nanni
Ram Nevatia
Shawn Newsam
Tian-Tsong Ng
Jifeng Ning
Masashi Nishiyama
Mark Nixon
Shohei Nobuhara
Vincent Nozick
Tom O'Donnell
Chi-Min Oh
Takeshi Oishi
Takahiro Okabe
Takayuki Okatani
Gustavo Olague
Maks Ovsjanikov
Yuji Oyamada
Paul Sakrapee Paisitkriangkrai
Kalman Palagyi
Gang Pan
Hailang Pan
Sharath Pankanti
In Kyu Park
Jong-Il Park
Ioannis Patras
Vladimir Pavlovic
Helio Pedrini
Pieter Peers
Yigang Peng
David W. Penman
Amitha Perera
Alessandro Perina
Janez Pers

Wong Ya Ping
Robert Pless
Thomas Pock
Dipti Prasad Mukherjee
Andrea Prati
Yael Pritch
Oriol Pujol Pujol
Amal Punchihewa
Zhen Qian
Xueyin Qin
Bogdan Raducanu
Luis Rafael Canali
Visvanathan Ramesh
Ananth Ranganathan
Nalini Ratha
Nilanjan Ray
EdelGarcia Reyes
Christian Riess
Tammy Riklin Raviv
Tron Roberto
Antonio Robles-Kelly
Mikel Rodriguez
Bodo Rosenhahn
Guy Rosman
Arun Ross
Peter Roth
Amit Roy Chowdhury
Xiang Ruan
Raif Rustamov
Fereshteh Sadeghi
Satoshi Saga
Ryusuke Sagawa
Fumihiko Sakaue
Mathieu Salzmann
Jorge A. Sanchez
Nong Sang
Angel Sappa
Michel Sarkis
Jun Sato
Tomokazu Sato
Walter Scheirer
Bernt Schiele
Frank Schmidt
Dirk Schnieders
William Schwartz
Stan Sclaroff
McCloskey Scott
Shuji Senda
Vinay Sharma
Chunhua Shen
Li Shen
Shuhan Shen
Qinfeng J. Shi
Hakjoon Shim
Nobutaka Shimada
Ikuko Shimizu
Ilan Shimshoni
Koichi Shinoda
Takaaki Shiratori
Abhinav Shrivastava
Leonid Sigal
Terence Sim
Sudipta Sinha
Danijel Skocaj
Eric Sommerlade
Jeany Son
Andy Song
Li Song
Zheng Song
Aristeidis Sotiras
Richard Souvenir
Jacopo Staiano
Chris Stauffer
Gideon Stein
Evgeny Strekalovskiy
Yu Su
Ramanathan Subramanian
Yusuke Sugano
Yasushi Sumi
Fengmei Sun
Jian Sun
Ju Sun
Min Sun
Weidong Sun
Xiaolu Sun
Yajie Sun
Jinli Suo
Rahul Swaminathan
Yu-Wing Tai
Taketomi Takafumi

Jun Takamatsu
Hugues Talbot
Toru Tamaki
Robby Tan
Tieniu Tan
Xiaoyang Tan
Masayuki Tanaka
Jinhui Tang
Jinshan Tang
Ming Tang
Rinichiro Taniguchi
João Manuel R. S. Tavares
Mutsuhiro Terauchi
Taipeng Tian
Joseph Tighe
Yu Ting
Reichl Tobias
Eno Toeppe
Matt Toews
Shoji Tominaga
Akihiko Torii
Bill Triggs
Werner Trobin
Ngo Thanh Trung
Yanghai Tsin
Pavan Turaga
Matt Turek
Matthew Turk
Seiichi Uchida
Hideaki Uchiyama
Toshio Ueshiba
Norimichi Ukita
Roberto Valenti
Michel F. Valstar
Pascal Vasseur
Changhu Wang
Chen Wang
Cheng Wang
Hanzi Wang
Hongcheng Wang
Liang Wang
Lu Wang
Min Wang
Ruiping Wang
Shiaokai Wang
Song Wang
Xianwang Wang
Xiaogang Wang
Yang Wang
Yu-Chiang Frank Wang
Yunhong Wang
Chaohui Wang
Li-Yi Wei
Yichen Wei
Chee Sun Won
Young W. Woo
John Wright
Tai Pang Wu
Xiaomeng Wu
Yi Wu
Peter Wurtz
Jianxiong Xiao
Jing Xiao
Yang Xiao
Xuehan Xiong
Changsheng Xu
Dong Xu
Li Xu
Ning Xu
Yong Xu
Jianru Xue
Yasushi Yagi
Osamu Yamaguchi
Pingkun Yan
Keiji Yanai
Fei Yang
Hao Yang
Herbert Yang
Jie Yang
Meng Yang
Ming Yang
Peng Yang
Yongliang Yang
Bangpeng Yao
Jong Chul Ye
Sai Kit Yeung
Alper Yilmaz
Zhaozheng Yin
Xianghua Ying
Kuk-Jin Yoon

Lap Fai Yu
Tianli Yu
Baozong Yuan
Junsong Yuan
Lu Yuan
Xenophon Zabulis
John Zelek
Gang Zeng
Zheng-Jun Zha

Cha Zhang
Changshui Zhang
Guofeng Zhang
Hong Hui Zhang
Hongbin Zhang
Hui Zhang
Lei Zhang
Li Zhang
Liqing Zhang
Xiaoqin Zhang
Yu Zhang
Xiao-Wei Zhao
Lu Zheng
Weishi Zheng
Wenming Zheng
Zhonglong Zheng
Baojiang Zhong
Feng Zhou
Zhi-Hua Zhou
Cai-Zhi Zhu
Feng Zhu
Jiejie Zhu
Zhigang Zhu
Ning Zhu
Danping Zou

External Reviewers

Farnaz Abtahi
Yasuhiro Akagi
Rushil Anirudh
Hirooki Aoki
Indriyati Atmosukarto
Qinxun Bai
Somdutta Banerjee
Yosuke Bando
Loris Bazzani
J. Bermudez
Fatih Cakir
Kevin Cannons
Che-Han Chang
Ding-Jie Chen
Hsin-Yi Chen
James Chen
Xida Chen
Hong Cheng
Shinko Cheng
Hung-Kuo Chu
Ahmed Sheikh Deeb
Idit Diamant
Liana Diesendruck
Xiaoyu Ding
Yuanyuan Ding
Carl Doersch
Keisuke Doman
Ralf Dragon
Marco Fornoni
David Fouhey
Nathan Frey
Hua Gao
Jizhou Gao
Yuli Gao
Haokun Geng
Fabian Gigengack
Arjan Gijsberts
Hitoshi Habe
Ralf Haeusler
Hossein Hajimirsadeghi
Patrick Harding
Kun He
Ariane Herbulot
Simon Hermann
Jacob Hinkle
Shang-Hong Lai

Tzu-Wei Huang
Tomoya Ishikawa
Hiroyuki Iwama
Yoshihiro Kanamori
Swarna Kamlam
Phil Kang
Wai L. Khoo
Kazuaki Kondo
Hiroshi Koyasu
Ilja Kuzborskij
Po-Lun Lai
Tian Lan
Ken-Yi Lee
Tung-Ying Lee
Daniel Leung
Yi Li
Yang Liu
Shugao Ma
Rouzbeh Maani
Rok Mandeljc
Samuele Martelli
Lucas Marti
Alhayat Ali Mekonnen
Chhaya Methani
Ikuhisa Mitsugami
Oliver Mller
T. Nathan Mundhenk
Daigo Muramatsu
Amit Padhy
Samunda Parera
Liliana Lo Presti
Ajita Rattani
Mahdi Rezaei
Samuel Rivera
Mike Roberts
Guy Rosman
Mohammad Rouhani
Muhammad Rushdi
Christian Schmaltz
Nataliya Shapovalova
Bin Shen
Farzad Siyahjani
Marcos Slomp
Tomokazu Takahashi
Danhang Tang
Hao Tang
Junli Tao
Tatiana Tommasi
Arash Vahdat
Jinjun Wang
Jun Wang
Junqiu Wang
Qing Wang
Tsaipei Wang
Yu-Shuen Wang
ZhengXiang Wang
Donglai Wei
Jie Wei
Chenyu Wu
Herb Yang
Yi Yang
Thibault Yohan
Jianming Zhang
Tianzhu Zhang
Wei Zhang
Ji Zhao
Bineng Zhong
Shengqi Zhu
Gali Zimmerman

ACCV 2012 Best Paper Award Committee

Sing Bing Kang	MicroSoft Research, USA
Ian Reid	University of Oxford, UK
Long Quan	HKUST, Hong Kong

ACCV 2012 Best Paper (The Saburo Tsuji Award)

Detecting Partially Occluded Objects with an Implicit Shape Model Random Field
Paul Wohlhart, Michael Donoser, Peter Roth, and Horst Bischof

ACCV 2012 Best Student Paper (The Sang Uk Lee Award)

Discriminative Dictionary Learning with Pairwise Constraints
Huimin Guo, Zhuolin Jiang, and Larry Davis

ACCV 2012 Best Application Paper (The Songde Ma Award)

Large-Scale Bundle Adjustment by Parameter Vector Partition
Shanmin Pang, Jianrue Xue, Le Wang, and Nanning Zheng

ACCV 2012 Best Paper Honorable Mention

Rapid Uncertainty Computation with Gaussian Processes and Histogram Intersection Kernels
Alexander Freytag, Erik Rodner, Paul Bodesheim, and Joachim Denzler

ACCV 2012 Best Student Paper Honorable Mention

Robust Visual Tracking Using Dynamic Classifier Selection with Sparse Representation of Label Noise
Yuefeng Chen and Qing Wang

ACCV 2012 Best Application Paper Honorable Mention

Efficient Learning of Linear Predictors Using Dimensionality Reduction
Stefan Holzer, Slobodan Ilic, David Tan, and Nassir Navab

ACCV 2012 Best Reviewers

Mitsuru Ambai
Steve Branson
Joao Carreira
Wen-sheng Chu
Hu Han
Gang Hua
Ichiro Ide
Yu-Gang Jiang
Mohan Kankanhalli
Junseok Kwon
Longin Jan Latecki
Marius Leordeanu
Fuxin Li
Jongwoo Lim
Cheng-Lin Liu
Steve Maybank
Paul Sakrapee Paisitkriangkrai
Arun Ross
Walter Scheirer
Yu-Wing Tai
Toru Tamaki
Bill Triggs
Liang Wang
Ruiping Wang
Jianxiong Xiao
Li Xu
Bangpeng Yao
Sai-Kit Yeung
Guofeng Zhang
Lei Zhang

ACCV 2012 Sponsors

Platinum	Daejeon Metropolitan City Daejeon International Marketing Enterprise Daejeon Convention Center Korea Tourism Organization DigiCar Center, KAIST Mobile Device Interface Research Center, Chonnam National University Seoul National University
Gold	Samsung Advanced Institute of Technology Puloon Technology
Silver	Mando Corporation Qualcomm 4D View Solutions
Bronze	NVIDIA Corporation

Table of Contents – Part IV

Oral Session 8: Shape Reconstruction and Optimization

Poster Session 8: Shape from X and Photometry

Oral Session 9: Applications of Computer Vision

Poster Session 9: Low-level Vision and Applications of Computer Vision

Self-calibration and Motion Recovery from Silhouettes with Two Mirrors

Hui Zhang[1,2], Ling Shao[3], and Kwan-Yee Kenneth Wong[4]

[1] Dept. of Computer Science, United International College,
28, Jinfeng Road, Tangjiawan, Zhuhai, Guangdong, China
[2] Shenzhen Key Lab of Intelligent Media and Speech,
PKU-HKUST Shenzhen Hong Kong Institution, Shenzhen, China
[3] Dept. of Electronic and Electrical Engineering,
The University of Sheffield, United Kingdom
[4] Dept. of Computer Science, The University of Hong Kong,
Pokfulam Road, Hong Kong

Abstract. This paper addresses the problem of self-calibration and motion recovery from a single snapshot obtained under a setting of two mirrors. The mirrors are able to show five views of an object in one image. In this paper, the epipoles of the real and virtual cameras are firstly estimated from the intersection of the bitangent lines between corresponding images, from which we can easily derive the horizon of the camera plane. The imaged circular points and the angle between the mirrors can then be obtained from equal angles between the bitangent lines, by planar rectification. The silhouettes produced by reflections can be treated as a special circular motion sequence. With this observation, technique developed for calibrating a circular motion sequence can be exploited to simplify the calibration of a single-view two-mirror system. Different from the state-of-the-art approaches, only one snapshot is required in this work for self-calibrating a natural camera and recovering the poses of the two mirrors. This is more flexible than previous approaches which require at least two images. When more than a single image is available, each image can be calibrated independently and the problem of varying focal length does not complicate the calibration problem. After the calibration, the visual hull of the objects can be obtained from the silhouettes. Experimental results show the feasibility and the preciseness of the proposed approach.

1 Introduction

Mirrors have been used for generating multiple views of an object, from which the visual hull can be obtained to recover the object shape and it has may applications [18] [14] [19]. The object and its reflections generally provide symmetric relationships for recovering parameters of the camera and the mirror [24] [5] (or a pair of mirrors [4]). In [7], Gluckman and Nayar discussed the geometry and calibration of a two-mirror system using point correspondences. Hu et. al. [10] later presented an approach for obtaining the camera calibration from the

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 1–12, 2013.

constraints imposed by both the silhouette outlines and point correspondences. Fujiyama et. al. [5] clearly presented the geometry of multiple view using one mirror. Forbes et. al. [3] introduced an approach based on silhouettes alone. However, they assumed an orthographic projection model and required a dense search in the parameter space to determine the initial estimates. Later in [2] they improved their method by providing closed form solutions for the initial parameter estimates using a perspective camera model. However, at least two snapshots were required for acquiring the calibration and estimating the motion. Besides, their method still required the assumption of an orthographic projection in the process of motion recovery. In another recent work, Huang [11] proved that the focal length can be recovered from a single snapshot of the setting, but it was based on the assumption that the principal point lied on the image center.

By exploring the geometry of two mirrors, this paper relates a two-mirror setting to a circular motion. Many studies have been conducted in circular motion [17], [1], [12], [16], [9]. Traditional method obtained the rotation angles by careful calibration [17], i.e., the camera internal parameters, rotation angles, camera orientations, etc are all accurately known. In [1], Fitzgibbon et al. developed a method to handle the case of uncalibrated camera with unknown rotation angles based on a projective reconstruction. Their method is based on the projective geometry of single axis motion, and it involves the computation of both fundamental matrices and trifocal tensors from point correspondences. Jiang et al. [12] further extended this approach by making use of the conic trajectories of the rotating point features, and developed an algorithm that requires neither the computation of fundamental matrices nor trifocal tensors. An alternative approach is to exploit the silhouettes of the object. Mendonça et al. [16] proposed to recover the structure and motion in several steps, each of which only involves a low dimensional optimization. However, the camera intrinsics are still required in the procedure for recovering the rotation angles and the subsequent Euclidean reconstruction. Zhang et al. [20] introduced an approach for uncalibrated silhouettes based on a new formulation of the circular point, and they further extended their method by making use of the 1D camera geometry [21].

Inspired by [16] and [20], it is derived in this paper that circular motions of a pair of symmetric objects can be obtained from the relationships between the image of the object and its reflections in two mirrors. The silhouettes produced by reflections can be treated as a special circular motion sequence. With this observation, technique developed for calibrating a circular motion sequence can be exploited to simplify the calibration of a single-view two-mirror system. Different from the state-of-the-art techniques [2] which assume orthogonal projection for recovering the motion, this work is totally based on perspective projection and hence it is applicable for any real scenes. More importantly, only one snapshot is required in this work for calibrating a natural camera (with three unknowns) and recovering the motion. This is more flexible than the previous approaches which require at least two images and the problem of varying focal length in multiple views will not complicate the calibration problem. Experimental results show the feasibility and the preciseness of the proposed approach.

The remainder of the paper is organized as follows. Section 2 gives the fundamental theories of the two-mirror setup. It also presents the relationship between the two-mirror setting and the circular motion. Section 3 describes self-calibration of the camera, with the recovery of image invariants, i.e., the circular points, the imaged rotation axis, the vanishing point of the x-axis of the real camera and the mirror angles, etc. Section 4 introduces implementation details of the proposed technique. Section 5 presents the experimental results, followed by discussions and conclusions in Section 6.

2 Two-Mirror Setup and Circular Motion

2.1 Two-Mirror Setup

In this section, we introduce the two-mirror setup in a 3D space. The reflections shown by mirrors are used to derive vanishing points for parallel tangent lines and these vanishing points all lie on the vanishing line $\mathbf{l}_h$ of a plane in which the real and virtual cameras lie.

Let us first consider a camera C capturing an object O and its reflection O_1 in a mirror M (see Fig.1(a)). Note that there would be a virtual camera C_1 which is the reflection of C in the mirror M. Consider two planes $\Pi_\top$ and $\Pi_\bot$ passing through the two cameras C, C_1 and tangent to both O and O_1 externally. As both sides of the mirror are symmetric, the tangent points on O and O_1, i.e., $\mathbf{X}$, $\mathbf{X}_1$ and $\mathbf{Y}$, $\mathbf{Y}_1$, provide two point correspondences with respect to the mirror. The joint lines $\mathbf{XX_1}$, $\mathbf{YY}_1$ and the line joining the camera centers CC_1 are parallel to each other and perpendicular to the mirror plane. Let the images of $\mathbf{XX_1}$ and $\mathbf{YY}_1$ in the real camera C be $\mathbf{l}_\top$, $\mathbf{l}_\bot$, respectively, which are the bitangents to the silhouettes of O and O_1. Their intersection point $\mathbf{v}_1$ indicates the vanishing point of the perpendicular direction of the mirror plane.

Now let us consider the two-mirror setup (see Fig.1(b)) capturing five objects. The camera C observes the real object O and also its four mirror reflections O_1, O_2, O_{12} and O_{21}. The virtual object O_1 is the reflection of O in the mirror M_1; O_2 is the reflection of O in the mirror M_2; O_{12} is the reflection of O_1 in the mirror M_2; and O_{21} is the reflection of O_2 in the mirror M_1. Note there are two virtual mirrors M_{v1}, M_{v2} which reflect O_1 to O_{21}, O_2 to O_{12}, respectively. There are also several virtual cameras which are the reflections of the real camera C, i.e., the virtual cameras C_1, C_2 (the reflection of C in the mirror M_1, M_2, respectively), the virtual camera C_{21} (the reflection of C_1 in the mirror M_{v1} and also the reflection of C_2 in M_1), the virtual camera C_{12} (the reflection of C_2 in the mirror M_{v2} and also the reflection of C_1 in M_2). Note all the cameras lie on a common plane Π and the bitangents $\mathbf{XX_1}$, $\mathbf{YY}_1$ in Fig.1(a) are parallel to Π, which implies the five (real and virtual) objects lie on a plane parallel to Π. Besides, note that the mirrors M_1, M_2, M_{v1}, M_{v2} intersect along a common line $\mathbf{L}_s$ which is perpendicular to Π.

Let the images of O, O_1, O_2, O_{21}, O_{12} be o, o_1, o_2, o_{21}, o_{12}, respectively, and the vanishing line of Π be $\mathbf{l}_h$ (see Fig.1(c)). From the mirror reflections, it can

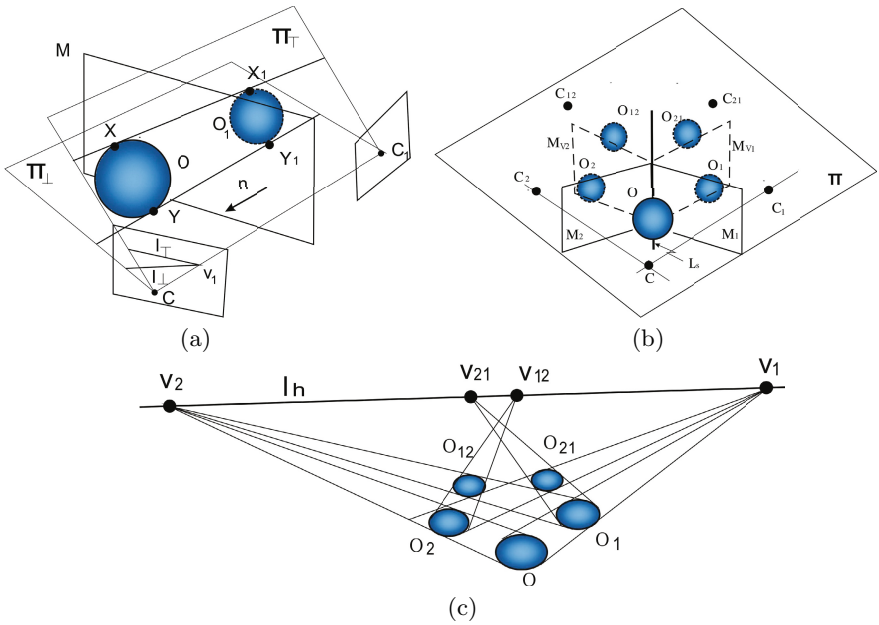


Fig. 1. Geometry of the mirror(s). (a) One mirror setup. (b) Two mirror setup. (c) The image of the two-mirror setup.

be seen that the outer bitangents of o, o_1 and that of o_2, o_{21} intersect with the horizon $\mathbf{l}_h$ at the vanishing point $\mathbf{v}_1$; the outer bitangents of o, o_2 and that of o_1, o_{12} intersect with $\mathbf{l}_h$ at $\mathbf{v}_2$; the outer bitangents of o_1, o_{21} intersect with $\mathbf{l}_h$ at $\mathbf{v}_{21}$; the outer bitangents of o_2, o_{12} intersect with $\mathbf{l}_h$ at $\mathbf{v}_{12}$. Hence the horizon $\mathbf{l}_h$ can be recovered as a line passing through all the vanishing points $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_{21}$ and $\mathbf{v}_{12}$.

2.2 Relating the Two Mirror Setting to the Circular Motion

We have observed that the silhouettes produced by reflections can be treated as a special circular motion sequences. In this section, we will illustrate this in detail. Consider the top view of Fig.1(b). The real and virtual cameras C, C_1, C_2, C_{21}, C_{12} are all on the plane Π. Let the real camera C lie on the negative Z-axis of the world coordinate system and the mirror intersection line $\mathbf{L}_s$ coincides with the Y-axis (see Fig.2(a)).The projection matrix of C is

$$\mathbf{P}_C = \mathbf{KR}[\mathbf{I}| - \mathbf{T}], \tag{1}$$

where $\mathbf{K}$ is the camera intrinsic matrix, $\mathbf{R}$ is the camera initial orientation and $\mathbf{T} = [0\ 0\ -1]^T$ is the camera center. Let the angle between the mirror M_1 and the negative Z-axis be σ, and the angle between the mirror M_2 and the negative Z-axis be φ. Then the angle between M_1 and M_2 is $\theta = \sigma + \varphi$. From the mirror

reflections, it can be seen that $|OC| = |OC_1| = |OC_2| = |OC_{21}| = |OC_{21}|$, where $|AB|$ indicates the length of AB. Hence the camera centers C, C_1, C_2, C_{21}, C_{12} lie on a circle (see Fig.2(a)). Besides, note that the angle between M_1 and OC_1 is σ, and the angle between M_2 and OC_2 is φ. Similarly, the other angles between virtual cameras and mirrors can also be easily derived as shown in Fig.2(a).

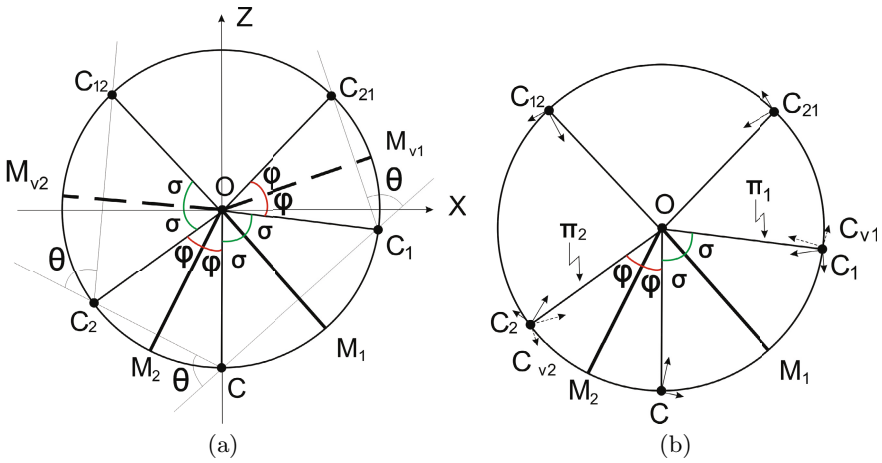


Fig. 2. Top view of the mirror setup. (a) The camera centers lie on a circle. (b) The cameras perform circular motion.

Imagine that there is a plane mirror Π_1 which passes through $\mathbf{L}_s$ and C_1. Let C_{v1} be the reflection of C_1 according to Π_1. The camera projection matrices for C_1 and C_{v1} can be represented by

$$\begin{aligned} \mathbf{P_{C_1}} &= \mathbf{KR}[\mathbf{R_Y}(\sigma) \sum \mathbf{R_Y}(-\sigma)| - \mathbf{T}], \\ \mathbf{P_{C_{v1}}} &= \mathbf{KR}[\mathbf{R_Y}(\mathbf{2}\sigma)| - \mathbf{T}], \end{aligned} \tag{2}$$

where $\sum = diag([-111])$, $\mathbf{R}_Y(\sigma)$ indicate rotation around Y-axis by an angle σ. Similarly, let C_{v2} be the reflection of C_2 according to Π_2, where Π_2 is a virtual plane mirror passing through $\mathbf{L}_s$ and C_2. We can easily derive the projection matrices for C_2 and C_{v2} in a similar way.

Now it can be easily observed that C_{21} is obtained by rotating C counter-clockwise about the point O with an angle $2(\varphi + \sigma)$, i.e., twice of the angle θ between the mirror M_1 and M_2. Similarly, C_{12} is obtained by rotating C clockwise about the point O with 2θ. C_2 is obtained by rotating C_1 clockwise about the point O with 2θ. Therefore, it can be observed that C, C_{v1}, C_{21}, C_{12}, C_{v2} are the cameras performing a circular motion and the rotation axis is the Y-axis. Besides, it can also be derived that the angles have the following constraints

$$\angle CC_1C_{21} = \angle C_1CC_2 = \angle CC_2C_{12} = \pi - \theta. \tag{3}$$

Under circular motion, the fundamental matrix relating any two views can be explicitly parameterized in terms of the image invariants, and is given by [1][15]

$$\mathbf{F}(\psi) = [\mathbf{v}_x]_\times + \kappa \tan\frac{\psi}{2}(\mathbf{l}_s\mathbf{l}_h^T + \mathbf{l}_h\mathbf{l}_s^T), \tag{4}$$

where ψ is the rotation angle between the two views. $\mathbf{l}_s$ is the imaged rotation axis and $\mathbf{v}_x$ is the vanishing point of X-axis. κ is an unknown but fixed scalar used to account for the different scales used in the homogeneous representations of the two terms in the summation [22].

3 Self-calibration of Two-Mirror Setting

In this section, a novel approach for self-calibrating the two-mirror setup will be introduced. The imaged circular points of the horizontal camera plane are firstly derived by metric rectification of the horizontal plane. The angle between the mirrors can thus be easily obtained. From the metric rectification, the imaged rotation axis can be derived. The vanishing point of the X-axis can thus be obtained by a cross ratio relationship. These image invariants could be used for a camera self-calibration.

3.1 Recovery of the Circular Point and the Mirror Angle

First, from the horizon $\mathbf{l}_h = [l_1\ l_2\ l_3]^T$ estimated in Section 2, the image in Fig.1(b) can be rectified to an affine plane using a 'pure projective' transformation [13], given by

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{bmatrix}. \tag{5}$$

Let the circular points be $[\alpha \mp j\beta,\ 1,\ 0]^T$ on the affine plane, the plane can be further transformed to a metric plane using an affine transformation [13] given by

$$\mathbf{A} = \begin{bmatrix} \frac{1}{\beta} & -\frac{\alpha}{\beta} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{6}$$

Note that in equation (3) the angle $\angle CC_1C_{21}$ formed by the line $\mathbf{l}_{CC_1} = [l_{a1}\ l_{a2}\ l_{a3}]^T$ and $\mathbf{l}_{C_1C_{21}} = [l_{b1}\ l_{b2}\ l_{b3}]^T$, and the angle $\angle C_1CC_2$ formed by $\mathbf{l}_{C_1C} = [l_{p1}\ l_{p2}\ l_{p3}]^T$ and $\mathbf{l}_{CC_2} = [l_{q1}\ l_{q2}\ l_{q3}]^T$ are equal unknown angles on the world plane. Hence the 2D point (α, β) can be shown lying on the circle with center on the point $(c_\alpha, c_\beta) = (\frac{aq-bp}{a-b-p+q}, 0)$ and squared radius $r^2 = (\frac{aq-bp}{a-b-p+q})^2 + \frac{(a-b)(ab-pq)}{a-b-p+q} - ab$, where $a = -\frac{l_{a2}}{l_{a1}}$, $b = -\frac{l_{b2}}{l_{b1}}$, $p = -\frac{l_{p2}}{l_{p1}}$ and $q = -\frac{l_{q2}}{l_{q1}}$ indicate the directions of each line. Similarly, by making use an additional unknown equal angle $\angle CC_2C_{12}$ in equation (3), (α, β) can be determined easily. Hence the pair of circular points in the original image can be recovered, by $\mathbf{i}, \mathbf{j} = [(\alpha \pm j\beta)l_3, l_3, -\alpha l_1 - l_2 \mp j\beta l_1]^T$.

From $\mathbf{i}, \mathbf{j}$, the angle between the two mirrors can thus be directly obtained by using the Laguerre's formula

$$\theta = \frac{1}{2j} \log\{\mathbf{v_1}, \mathbf{v_2}; \mathbf{i}, \mathbf{j}\} \tag{7}$$

where $\{\mathbf{v_1}, \mathbf{v_2}; \mathbf{i}, \mathbf{j}\}$ denotes a cross-ratio, $j^2 = -1$.

3.2 Recovery of the Imaged Rotation Axis

By making use of the projection and affine transformations $\mathbf{P}$ and $\mathbf{A}$ (see Section 3.1), the imaged circular points $\mathbf{i}, \mathbf{j}$ are expected to be rectified to their genuine position $\mathbf{I}, \mathbf{J} = [1 \pm j \ 0]^T$. However, we still need a rotation $\mathbf{R}$ to transform the imaged circular points to their genuine position. Hence by the same transformations, the imaged rotation axis $\mathbf{l}_s$ can be rectified to a plane $[1\ 0\ 0]^T$ passing through the camera center and the rotation axis. Thus $\mathbf{l}_s$ can be initialized as

$$\mathbf{l}_s = (\mathbf{RAP})^T \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}. \tag{8}$$

The vanishing point $\mathbf{v}_z$ of the Z-axis can be obtained as the intersection between $\mathbf{l}_s$ and $\mathbf{l}_h$. The vanishing point $\mathbf{v}_x$ can also be easily recovered from the cross ratio

$$\{\mathbf{i}, \mathbf{j}; \mathbf{v}_x, \mathbf{v}_z\} = -1. \tag{9}$$

The angle σ between the mirror M_1 and the negative Z-axis (see Fig.2(a)) can be obtained by $\sigma = \pi/2 - \log\{\mathbf{v}_1, \mathbf{v}_z; \mathbf{i}, \mathbf{j}\}/(2j)$ and the angle φ between the mirror M_2 and the negative Z-axis can be obtained by $\varphi = \pi/2 - \log\{\mathbf{v}_2, \mathbf{v}_z; \mathbf{i}, \mathbf{j}\}/(2j)$, where $j^2 = -1$.

Note the pair of circular points $\mathbf{i}$, $\mathbf{j}$ of the circular plane are given by [22]

$$\mathbf{i}, \mathbf{j} \sim \mathbf{v}_x \mp j\, \kappa \mathbf{l}_s \times \mathbf{l}_h, \tag{10}$$

where κ is the same scalar in equation (4). As $\mathbf{i}$, $\mathbf{j}$, $\mathbf{v}_x$, $\mathbf{l}_s$, $\mathbf{l}_h$ are known variables, κ can be easily obtained. Hence the epipoles $\mathbf{e}_i$ $(i = 1, 2)$ between a pair of the images of the circular motion can be obtained from [15]

$$\mathbf{e}_i \sim \mathbf{v}_x - (-1)^i \kappa \tan\frac{\psi}{2} \mathbf{l}_s \times \mathbf{l}_h. \tag{11}$$

And the refinement of the imaged rotation axis $\mathbf{l}_s$ can be carried out as a two dimensional optimization problem by minimizing the distance between the transformed epipolar tangents $\mathbf{l}'_i$ and the silhouette in the second image (see Fig.3). The transformation is defined by a *harmonic homology* [8][15] $\mathbf{W}^{-T}$, which is given by $\mathbf{W} = \mathbf{I} - 2\frac{\mathbf{v}_x \mathbf{l}_s^T}{\mathbf{v}_x^T \mathbf{l}_s}$.

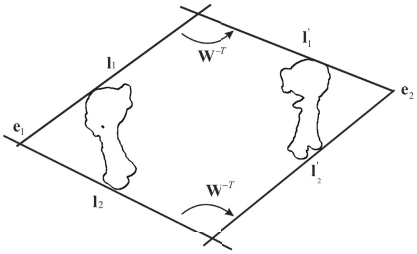


Fig. 3. The overlapping of two silhouettes and their epipolar tangents under the circular motion. $\mathbf{l}_1$,$\mathbf{l}_1'$,$\mathbf{l}_2$ and $\mathbf{l}_2'$ are the outer epipolar tangent lines.

3.3 Self-calibration and Motion Recovery

The obtained imaged circular points can be used to find the camera intrinsics since they lie on the image of the absolute conic (IAC) ω. Besides, the imaged rotation axis $\mathbf{l}_s$ and the vanishing point $\mathbf{v}_x$ define a pole-polar relationship w.r.t. ω [8]. ω can then be estimated from the following constraints:

$$\begin{cases} \mathbf{i}^T\omega\mathbf{i} = 0 \quad \text{and} \quad \mathbf{j}^T\omega\mathbf{j} = 0, \\ \mathbf{l}_s = \omega\mathbf{v}_x \ . \end{cases} \tag{12}$$

Since these provide three independent constraints, given only a single image, a natural camera with zero skew and unit aspect ratio can be calibrated by Cholesky decomposition [6] of ω. For multiple images captured with varying focal length, each image can be calibrated independently. Hence the problem of varying focal length does not complicate the calibration problem.

4 Implementation

Here we introduce using one snapshot to calibrate the camera and recover the motion. Cubic B-spline snakes are used to extract silhouettes from the images with sub-pixel localization accuracy. The horizon $\mathbf{l}_h$ is initially obtained by robustly fitting a line to the vanishing points constructed from the outer tangents to the object silhouettes in the image. $\mathbf{l}_h$ and the vanishing points are then refined by minimizing the distance between the tangent lines and the corresponding silhouettes.

The image can then be transformed to an affine plane by equation (5). Then the imaged circular points $\mathbf{i}$, $\mathbf{j}$ can be obtained by making use equal unknown angles in the world plane (see Section 3.1 in detail). The imaged rotation axis $\mathbf{l}_s$ is then initialized as the rectified YZ-plane by equation (8) and the vanishing point $\mathbf{v}_x$ of X-axis can be recovered by (9). $\mathbf{l}_s$ and $\mathbf{v}_x$ can be refined by the finding of a line tangent to one silhouette which is transformed by the harmonic homology $\mathbf{W}^{-T}$ to a line tangent to another silhouette under the circular motion [16]. From the estimated $\mathbf{l}_h$, $\mathbf{i}$, $\mathbf{j}$, $\mathbf{l}_s$, $\mathbf{v}_x$, the fixed scalar κ and the rotation angles

can be easily derived (see Section 3.2 for detail). Besides, a natural camera can be calibrated with the recovered $\mathbf{l}_s$, $\mathbf{v}_x$ and $\mathbf{i}$, $\mathbf{j}$, by equation (12). The camera extrinsic parameters can then be estimated by aligning the images of the horizon and the rotation axis through rectifying each image independently by a homography induced by a rotation about the camera center such that $\mathbf{L_s}$ coincides with the Y-axis of the world coordinate and the Z-axis of the camera world coordinate coincides with the Z-axis of the world coordinate.

Besides, if multiple snapshots were taken, we need to specify the five silhouettes from different views in a common reference frame to refine the estimation. This can be achieved by firstly aligning the world coordinate recovered with different snapshots and rectifying the five-view silhouette sets with the camera matrices so that the cameras all point towards the rotation axis $\mathbf{L_s}$. The silhouette sets are then scaled and translated along the rotation axis so that the outer epipolar tangents coincide with the projected tangents from silhouettes in the other silhouette set.

Finally, a bundle-adjustment using Levenberg Marquardt minimization is applied to refine all the parameters. The intrinsics and the angle θ between mirrors M_1 and M_2 are then estimated with the optimized entities, followed by a constructing the visual hull from silhouettes.

5 Experiments and Results

Real experiments were carried out to test the feasibility of the approach. The first experiment consisted views of a girl (see the first column Fig.4). The image had a resolution of 1296×861. Provided with only one single snapshot, the camera was self-calibrated under the assumption of a natural camera (zero-skew and unit aspect ratio). Column 2-4 of Table 1(a) compare the estimated camera matrix and the recovered mirror angle with that of the ground-truth (obtained with a planar calibration pattern [23]) and the approach introduced in [2]. It can be seen that the recovered angle θ between the mirrors has a high resolution. The focal length f and the u_0 coordinate of the principal point were both precisely estimated while v_0 was not. This is due to the error in the estimated $\mathbf{v}_x$. Column 2-4 of Table 1(b) show the experimental results with two snapshots. It can be seen the calibration results is better with more snapshots involved in estimation. From the recovered motion, Fig.4(c) shows the 3D model reconstructed with only one single snapshot and Fig.4(d) shows that with two snapshots. The model becomes more accurate may due to the reason that more snapshots may provide more accuracy in the camera calibration and the visual hull.

The second experiment consisted views of a monster (see the second colum of Fig.4). The image had a resolution of 1296×861. With only one single snapshot, column 6-9 of Table 1(a) compare the estimated camera matrix and the mirror angle with that of the ground-truth (obtained with a planar calibration pattern [23]) and the approach introduced in [2]. Column 6-9 of Table 1(b) show the result with two snapshots. From the estimated motion, Fig.4(d) shows the 3D model reconstructed with only one single snapshot and Fig.4(f) shows that with two snapshots.

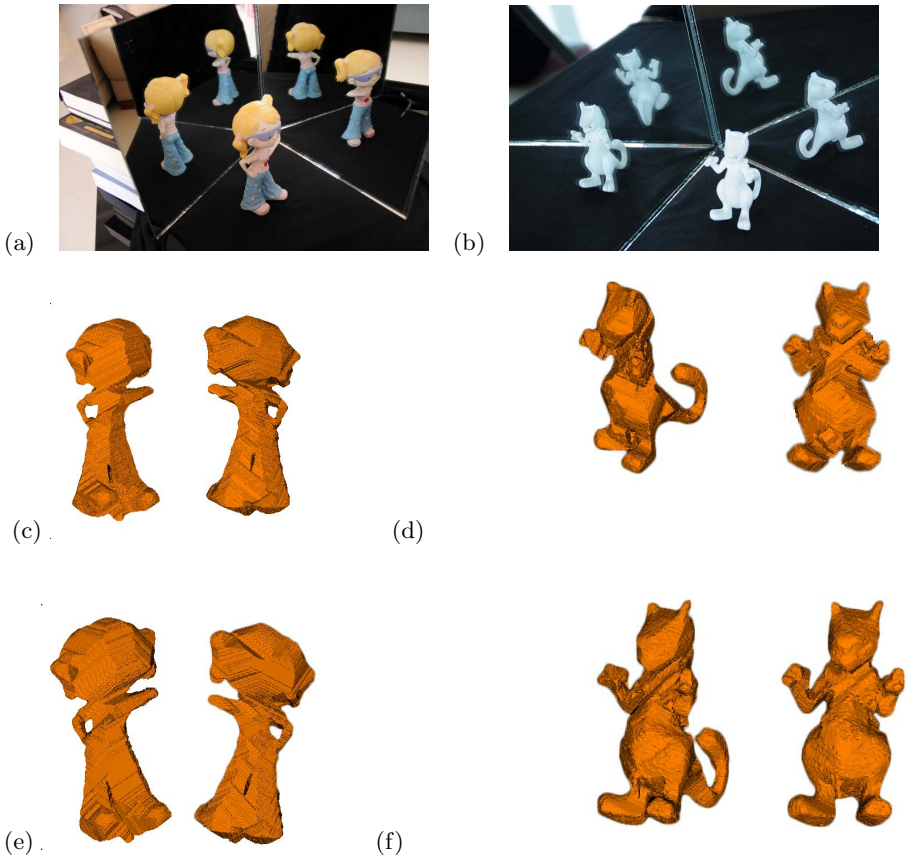


Fig. 4. Real experiment. The 1st column is for the experiments of the girl and the 2nd column is for that of the little monster. (a)&(b) An image of the two mirror setting. (c)&(d) The reconstructed 3D models from a snapshot. (e)&(f) The reconstructed 3D models from 2 snapshots.

6 Conclusions

In this paper, we have presented a practical and efficient approach for self-calibrating a camera from only a single snapshot obtained under a setting of two-mirror. We relate it with a circular motion and use image rectification to find the initial estimation of the imaged rotation axis. Different from the state-of-the-art approaches, only one snapshot is required in this work for calibrating a natural camera and recovering the motion. This is more flexible than the previous approaches which require at least two images. Hence the problem of varying focal length in multiple images does not complicate the calibration problem. After calibration, a visual hull of the object can be obtained from the silhouettes. Experiments have produced convincing 3D models, demonstrating the practicality of our algorithm.

Table 1. Comparative results of the intrinsic and the angle between the mirrors. Column 2-5 show experiments with images of a girl. Column 6-9 show experiments with images of a monster.(a) From a single snapshot. (b) From two views of the two-mirror settings.

		Girl				Monster			
	-	f	u_0	v_0	θ	f	u_0	v_0	θ
(a)	Ground-truth	1178.4	663.78	440.74	74.3°	2971.4	623.89	415.31	74.3°
	Method in [2]	1224.5	648	430.5	-	2950.6	648	430.5	-
	Proposed method	1196.7	633.43	413.2	74.18°	2958.8	606.05	365.82	74.43°
	Percentage error to GT	1.55%	1.73%	2.34%	0.16%	0.42%	0.60%	1.67%	0.17%
		Girl				Monster			
	-	f	u_0	v_0	θ	f	u_0	v_0	θ
(b)	Ground-truth	1178.4	663.78	440.74	74.3°	2971.4	623.89	415.31	74.3°
	Method in [2]	1173.2	646.9	432.71	-	2959.2	664.32	403.28	-
	Proposed method	1172.7	648.97	426.45	74.35°	2976.9	659.46	375.4	74.21°
	Percentage error to GT	0.48%	1.26%	1.21%	0.07%	0.18%	1.20%	1.34%	0.12%

Acknowledge. The work is supported by the National Natural Science Foundation of China (Project no. 61005038) and an internal funding from United International College.

References

1. Fitzgibbon, A.W., Cross, G., Zisserman, A.: Automatic 3D Model Construction for Turn-Table Sequences. In: Koch, R., Van Gool, L. (eds.) SMILE 1998. LNCS, vol. 1506, pp. 155–170. Springer, Heidelberg (1998)
2. Forbes, K., Nicolls, F., de Jager, G., Voigt, A.: Shape-from-silhouette with two mirrors and an uncalibrated camera. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 165–178. Springer, Heidelberg (2006)
3. Forbes, K., Voigt, A., Bodika, N.: Visual hulls from single uncalibrated snapshots using two plannar mirrors. In: Proc. 15th South African Workshop on Pattern Recognition (2004)
4. Francois, A.R.J., Medioni, G.G., Waupotitsch, R.: Reconstructing mirror symmetric scenes from a single view using 2-view stereo geometry. In: International Conference on Pattern Recognition, vol. 4, pp. 12–16 (2002)
5. Fujiyama, S., Sakaue, F., Sato, J.: Multiple view geometries for mirrors and cameras. In: International Conference on Pattern Recognition, pp. 45–48 (August 2010)
6. Gentle, J.E.: Numerical Linear Algebra for Applications in Statistics. Springer (1998)
7. Gluckman, J.M., Nayar, S.K.: Planar catadioptric stereo: Geometry and calibration. In: Proc. Conf. Computer Vision and Pattern Recognition.
8. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge (2000)
9. Hernndez, C., Schmitt, F., Cipolla, R.: Silhouette coherence for camera calibration under circular motion. IEEE Trans. on Pattern Analysis and Machine Intelligence 29(2), 343–349 (2007)

10. Hu, B., Brown, C.M., Nelson, R.C.: Multiple-view 3-d reconstruction using a mirror. Technical Report TR863 (May 2005)
11. Huang, P.H., Lai, S.H.: Contour-based structure from reflection. In: Proc. Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 379–386 (June 2006)
12. Jiang, G., Tsui, H.T., Quan, L., Zisserman, A.: Geometry of single axis motions using conic fitting. IEEE Trans. on Pattern Analysis and Machine Intelligence 25(10), 1343–1348 (2003)
13. Liebowitz, D., Zisserman, A.: Metric rectification from perspective images of planes. In: Proc. Conf. Computer Vision and Pattern Recognition, pp. 482–488 (1998)
14. Mariottini, G.L., Scheggi, S., Morbidi, F., Prattichizzo, D.: Catadioptric stereo with planar mirrors: multiple-view geometry and camera localization. In: Visual Servoing via Advanced Numerical Methods, pp. 3–21 (2010)
15. Mendonça, P.R.S., Wong, K.-Y.K., Cipolla, R.: Camera Pose Estimation and Reconstruction from Image Profiles under Circular Motion. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1843, pp. 864–877. Springer, Heidelberg (2000)
16. Mendonça, P.R.S., Wong, K.-Y.K., Cipolla, R.: Epipolar geometry from profiles under circular motion. IEEE Trans. on Pattern Analysis and Machine Intelligence 23(6), 604–616 (2001)
17. Niem, W.: Robust and fast modelling of 3d natural objects from multiple views. In: SPIE Proceedings - Image and Video Processing II, vol. 2182, pp. 388–397 (1994)
18. Rurainsky, J., Eisert, P.: Mirror-based multi-view analysis of facial motions. In: Prof. of International Conference on Image Processing, vol. 3, pp. 73–76 (2007)
19. Smith, B.M., Stork, D.G., Zhang, L.: Three-dimensional reconstruction from multiple reflected views within a realist painting: An application to scott frasers three way vanitas. In: The 21st Annual IST/SPIE Symposium on Electronic Imaging, vol. 7239 (January 2009)
20. Zhang, H., Wong, K.-Y.K.: Self-calibration of turntable sequences from silhouettes. IEEE Trans. on Pattern Analysis and Machine Intelligence 31(1), 5–14 (2009)
21. Zhang, H., Wong, K.-Y.K., Zhang, G., Liang, C., Zhang, G.: 1d camera geometry and its application to the self-calibration of circular motion sequences. IEEE Trans. on Pattern Analysis and Machine Intelligence 30(12), 2243–2248 (2008)
22. Zhang, H., Zhang, G., Wong, K.-Y.K.: Auto-calibration and motion recovery from silhouettes for turntable sequences. In: Proc. British Machine Vision Conference, Oxford, UK, vol. I, pp. 79–88 (September 2005)
23. Zhang, Z.Y.: A flexible new technique for camera calibration. IEEE Trans. on Pattern Analysis and Machine Intelligence 22(11), 1330–1334 (2000)
24. Zhang, Z.Y., Tsui, H.T.: 3d reconstruction from a single view of an object and its image in a plane mirror. In: International Conference on Pattern Recognition, vol. II, pp. 1174–1176 (1998)

Stereo Reconstruction and Contrast Restoration in Daytime Fog

Laurent Caraffa and Jean-Philippe Tarel*

Université Paris-Est, LEPSiS, IFSTTAR,
58 Boulevard Lefèbvre, F-75015 Paris, France

Abstract. Stereo reconstruction serves many outdoor applications, and thus sometimes faces foggy weather. The quality of the reconstruction by state of the art algorithms is then degraded as contrast is reduced with the distance because of scattering. However, as shown by defogging algorithms from a single image, fog provides an extra depth cue in the gray level of far away objects. Our idea is thus to take advantage of both stereo and atmospheric veil depth cues to achieve better stereo reconstructions in foggy weather. To our knowledge, this subject has never been investigated earlier by the computer vision community. We thus propose a Markov Random Field model of the stereo reconstruction and defogging problem which can be optimized iteratively using the α-expansion algorithm. Outputs are a dense disparity map and an image where contrast is restored. The proposed model is evaluated on synthetic images. This evaluation shows that the proposed method achieves very good results on both stereo reconstruction and defogging compared to standard stereo reconstruction and single image defogging.

1 Introduction

The first dense stereo reconstruction algorithms were proposed forty years ago. There is now more than one hundred algorithms listed on the Middlebury evaluation site. Nevertheless, several new algorithms or improvements are proposed each year. The reason for this constant interest is the high usefulness of the 3D reconstruction which serves in many applications such as: driver assistance, automatic driving, environment simulators, augmented reality, data compression, 3D TV. While the Middlebury database contains only indoor scenes of good quality, outdoor applications are confronted with more difficult weather conditions such as fog, rain and snow. These weather conditions reduce the quality of the stereo pairs and introduce artifacts. Reconstruction results are thus usually degraded.

The principle of stereo reconstruction is to find, for every pixel in the left image, the pixel in the right image which minimizes a matching cost along the epipolar line. Depending on the scene, the matching cost can be ambiguous or

* Thanks to the ANR (French National Research Agency) for funding, within the ICADAC project (6866C0210).

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 13–25, 2013.

wrongly minimal. A prior on the disparity map is thus added, for instance to enforce that close pixels have similar disparity. As a consequence, the stereo reconstruction is set as the minimization of an energy which derives from a Markov Random Field (MRF) model, see for instance [1,2]. Thanks to recent advances in numerical analysis, the optimization of this energy can be performed quickly without being trapped by most of the local minima.

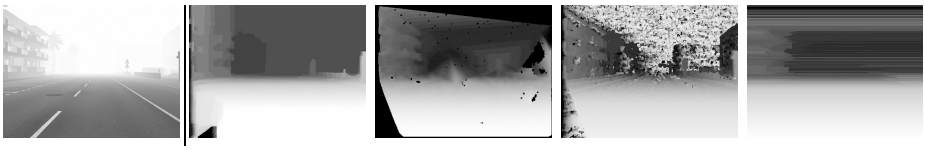


Fig. 1. From left to right: original left image of the stereo pair, disparity maps obtained using α-expansion on MRF [1], Libelas [3], correlation windows and dynamic programing on each line

We observed that stereo reconstructions are degraded in the presence of fog. As an illustration, in Fig. 1, we show disparity maps obtained on a foggy stereo image by four stereo reconstruction algorithms: α-expansion on MRF [1], Libelas [3], correlation windows and dynamic programing on each line. Results are not satisfactory; in the best case, they are correct only up to a critical distance. Indeed, in a foggy scene, the more distant an object, the whiter its color. As a consequence, contrast is a decreasing function of distance, which makes matching all the more difficult to perform. If stereo disparity is important for 3D reconstruction, in foggy scenes, the gray-level of distant objects is also a depth cue. This depth cue is used in contrast restoration algorithms but had not been used in 3D reconstruction yet. The defogging problem can also be set as a MRF problem, see [4]. The atmospheric veil depth cue is particularly interesting since it is complementary to the stereo depth cue: the former is reliable only for remote objects, while the latter is reliable only for near by objects. Our idea is thus to combine a MRF model of both stereo reconstruction and defogging problems into a unified MRF model to take advantage of both depth cues. As far as we know, there is no algorithm dedicated to dense stereo reconstruction in foggy weather conditions.

The article is structured as follows. In Sec. 2, we state the problem, and explain how fog affects the scene image. The classic dense stereo reconstruction and image defogging problems are derived from a general formulation. In Sec. 3, our model of the stereo reconstruction and defogging problem is proposed. At last, Sec. 4 is dedicated to an evaluation on synthetic images and tests on camera images.

2 Problem Statement

The inputs are the left and right images of a stereo pair $\{I_L, I_R\}$. These images are observed after perturbation by atmospheric scattering and camera optics.

The images without all these perturbations are denoted I_{0L} and I_{0R}, respectively, and are of course unknown. Also unknown is the depth map represented by its disparity map D. Our goal being to fuse depth cues from the stereo and from the atmospheric veil to achieve better reconstruction, it seems natural to search for a Bayesian formulation of the problem so that prior knowledge can be included to remove possible ambiguities. The two unknowns that we want to estimate are the disparity map D and the clean left image I_{0L}. The right one I_{0R} is not an unknown since, not considering occluded objects, it is a function of D and I_{0L}.

The maximum a posteriori principle tells us to maximize the following posterior probability, which can be rewritten using Bayes' rule as:

$$p(D, I_{0L}|I_L, I_R) \propto p(I_L, I_R|D, I_{0L})\ P(D, I_{0L}) \tag{1}$$

where $p(I_L, I_R|D, I_{0L})$ is the data likelihood and $P(D, I_{0L})$ is the prior on the unknowns (D, I_{0L}). Instead of posterior probability maximization, in practice, it is its log which is minimized, leading to the following formulation in terms of energy, or log-likelihood:

$$E(D, I_{0L}|I_L, I_R) = \underbrace{E(I_L, I_R|D, I_{0L})}_{E_{data}} + \underbrace{E(D, I_{0L})}_{E_{prior}} \tag{2}$$

The term E_{data} is also known as the data cost or fidelity term, and E_{prior} as the prior or regularization term.

2.1 Dense Stereo Reconstruction without Fog

Without fog, I_L and I_R are only affected by the noise of the sensor which is generally low. Following [2], the Bayesian formulation of the dense stereo reconstruction is approximated by assuming that I_L is without noise. In (2), the unknown variable I_{0L} can be thus substituted by I_L leading to the approximate but simpler energy minimization:

$$E(D|I_L, I_R) = \underbrace{E(I_R|D, I_L)}_{E_{data\_stereo}} + \underbrace{E(D|I_L)}_{E_{prior\_stereo}} \tag{3}$$

Data term: E_{data_stereo} is the error in intensity between a pixel in the left image and a pixel in the right image given a disparity. It is usually chosen as:

$$E_{data\_stereo} = \sum_{(i,j)\in X} \rho_S\left(\frac{|I_L(i,j) - I_R(i - D(i,j), j)|}{\sigma_S}\right) \tag{4}$$

where X is the set of image pixels, ρ_S is a function related to the distribution of the intensity noise with scale σ_S. This intensity noise takes into account the camera noise, but also the occlusion, and it can be one of the functions used in robust estimation to remove outliers.

Prior term: This term enforces the smoothness of the disparity map. Because of constant intensity objects, the data term can be rather ambiguous. It is thus necessary to introduce a prior on the disparity map to interpolate the ambiguous areas correctly. The smoothness prior tells that two close pixels have a greater chance to be the projection of a same object with the same depth than remote pixels. This assumption is not always true due to gaps in depth for example. As a consequence, a robust function ρ_D should be used in this term. The classical prior term is:

$$E_{prior\_stereo} = \lambda_D \sum_{(i,j)\in X} \sum_{(k,l)\in N} W_D(\nabla I_L(i,j))\ \rho_D(|D(i,j) - D(i+k, j+l)|) \quad (5)$$

where λ_D is a factor weighting the strength of the prior on D, N is the set of relative positions of pixel neighbors (4, 8 connectivity or other), and W_D is a monotonically decreasing function of image intensity gradients. The weight W_D is introduced to smooth low-gradient ambiguous areas more than gradient edges. Usually W_D is chosen as a decreasing exponential function of the image gradient: $W_D(\nabla I) = e^{-\frac{|\nabla I|}{\sigma_g}}$, where σ_g is a scale parameter. It is even better to use a function of the image Laplacian in order to avoid sensitivity to linear intensity variations: $W_D(\nabla I) = e^{-\frac{|\Delta I|}{\sigma_g}}$.

2.2 Effects of Fog

With a linear response camera, assuming an object of intrinsic intensity I_0, the apparent intensity I in presence of a fog with extinction coefficient β is modeled by Koschmieder law:

$$I = I_0 e^{-\beta p} + \underbrace{I_s(1 - e^{-\beta p})}_{V} \quad (6)$$

where p is the object depth, and I_s is the intensity of the sky. From (6), it can be seen that fog has two effects: first an exponential decay $e^{-\beta p}$ of the intrinsic luminance I_0, and second the addition of the atmospheric veil V which is an increasing function of the object distance p. The depth p can be rewritten as a function of the disparity $p = \frac{\delta}{D}$ where δ is related to the stereo calibration parameters. It is important, for the following, to notice that there is one situation where D can be obtained from a single image using V: when I_0 is close to zero, i.e when the object is dark. It is also important to notice that when the disparity D is zero, the intensity I_0 cannot be obtained. Moreover, I_0 being positive, the photometric constraint $V < I$ is deduced from (6).

For road images, several algorithms exist for detecting the fog and estimating the extinction coefficient β, see for instance [5]. The parameter β is thus assumed known in the following, as well as I_s.

2.3 Single Image Defogging Knowing the Depth

Before we describe our model for fused stereo reconstruction and defogging, we focus on the simpler problem of defogging from a single image I given the

disparity map D. Using the previous notations, only the left image is used in this section. We thus drop L in the indexes. The unknown I_0 is the image without fog and noise. Both I and D are assumed known. Even though D or the depth p is not accurately known, an approximate map is enough. The defogging problem knowing the disparity D can be set as a particular case of (1), i.e the maximization of the posterior probability:

$$p(I_0|D,I) \propto p(I|D,I_0)P(I_0|D)\ P(D) \tag{7}$$

or equivalently as the minimization of the energy:

$$E(I_0|D,I) = \underbrace{E(I|D,I_0)}_{E_{data\_fog}} + \underbrace{E(I_0|D)}_{E_{prior\_fog}} \tag{8}$$

Data term: The data term is the log-likelihood of the noise probability on the intensity, taking into account that I_0 is observed through the fog, see (6):

$$E_{data\_fog} = \sum_{(i,j)\in X} \rho_P\left(\frac{|I_0(i,j)e^{-\frac{\beta\delta}{D(i,j)}} + I_s(1-e^{-\frac{\beta\delta}{D(i,j)}}) - I(i,j)|}{\sigma_P}\right) \tag{9}$$

where ρ_P is a function related to the intensity noise due to the camera and σ_P is the scale of this noise. ρ_P and σ_P are thus directly related to the probability density function (pdf) of the camera noise and can be estimated off-line when calibrating the camera. It can be noticed for D close to zero that the data term does not constrain the distribution of I_0 which tends to the uniform pdf.

Prior term: We found that the following prior term produces nice restoration results:

$$E_{prior\_fog} = \lambda_{I_0} \sum_{(i,j)\in X} \sum_{(k,l)\in N} e^{-\frac{\beta\delta}{D(i,j)}}\ W_{I_0}(\nabla D(i,j))\ \rho_{I_0}(|I_0(i,j) - I_0(i+k,j+l)|) \tag{10}$$

where λ_{I_0} is a factor weighting the strength of the prior on I_0. Function W_{I_0} is the equivalent to W_D in the stereo, only now it is applied on the disparity map gradient rather than on image gradient. We use $W_{I_0}(\nabla D) = e^{-\frac{|\Delta D|}{\sigma'_g}}$, where σ'_g is a scale parameter. Function ρ_{I_0} is a robust function used for similar reasons as ρ_D. An extra weight $e^{-\frac{\beta\delta}{D(i,j)}}$ is introduced, and it is a key point, to take into account that in presence of fog, there is an exponential decay of contrast with respect to (w.r.t.) depth. This has the effect of giving less and less importance to the prior as depth increases. This is necessary to be consistent with the fact that the distribution of I_0 is less and less constrained by the data term for large distances. Without this extra factor, the intensity of close objects may wrongly diffuse on remote objects.

2.4 Optimization

While MRF formulations are successful to model image processing and computer vision problems, it is also necessary to have reliable optimization algorithms to minimize the derived energies. A large class of useful MRF energies is of the form:

$$f(Y) = \sum_{x \in X} \Phi_x(Y_x) + \sum_{x \in X, x' \in X} \Phi_{x,x'}(Y_x, Y_{x'}) \tag{11}$$

When the variable Y is binary, it has been shown long ago that for sub-modular functions Φ, the global minimum of the previous problem can be obtained in polynomial time. For non-binary variables, one of the most efficient technique to optimize (11) approximately is the α-expansion algorithm, which is based on the decomposition of the problem in successive binary problems. The global optimum of each binary sub-problem is obtained in polynomial time, when the prior term is sub-modular.

When the function Φ is not sub-modular, other heuristics such as $\alpha - \beta$ swap, Belief propagation, TRW, roof duality were proposed which produce interesting results.

3 Stereo Reconstruction and Defogging

The model we now propose for fused stereo reconstruction and defogging shares similarities with the single image defogging model presented in [4]. Indeed in [4], the model is set as a MRF model and both depth p and restored image I_{0L} are estimated successively. The main difference is that stereo is used in our approach, while the approach in [4] is monocular. In particular, this last approach cannot work with gray-level images, contrary to our stereo approach. Another difference is that, in [4], Koschmieder's law (6) is rewritten, after algebraic manipulations and use of the log function, in such a way that the depth and intensity appear as a linear combination of independent functions of each of these two variables. This rewriting allows a simpler optimization. However, the noise is non linearly transformed and this is not taken into account. The stereo approach we now present contains non-linear equations where the image noise is better handled.

3.1 MRF Model

Data term: In stereo with fog, the data term (9) applies on the left image. On the right, a similar term taking into account the disparity D is also introduced. This leads to the following log-likelihood of the stereo data in fog:

$$\begin{aligned} E_{data\_fog\_stereo} = & \sum_{(i,j) \in X} \rho_P\left(\frac{|I_{0L}(i,j)e^{\frac{-\beta\delta}{D(i,j)}} + I_s(1 - e^{\frac{-\beta\delta}{D(i,j)}}) - I_L(i,j)|}{\sigma_P}\right) \\ & + \rho_P\left(\frac{|I_{0L}(i,j)e^{\frac{-\beta\delta}{D(i,j)}} + I_s(1 - e^{\frac{-\beta\delta}{D(i,j)}}) - I_R(i - D(i,j),j)|}{\sigma_P}\right) \end{aligned} \tag{12}$$

Notice that when $\beta = 0$, i.e without fog, the first term in (12) enforces $I_{0L} = I_L$, and the second term is the stereo log-likelihood E_{data_stereo}. This shows that D can be estimated from both log-likelihoods. We thus propose to linearly combine the two log-likelihoods in the data term:

$$E_{data^*} = \alpha E_{data\_stereo} + (1-\alpha) E_{data\_fog\_stereo} \tag{13}$$

with $0 \leq \alpha \leq 1$. During the estimation of both I_{0L} and D, the value of I_{0L} can be temporarily far from the true value. The advantage of introducing E_{data_stereo} in the data term is that the minimization of E_{data_stereo} provides correct estimates of D at short distances even if I_{0L} is badly estimated.

Photometric constraint and assumption on white pixels: As introduced in Sec. 2.2, the photometric constraint on the atmospheric veil V must be verified both on the left and right images of the stereo pair. Due to noise, the photometric constraint is not very strict but it helps to reduce the search space of I_{0L}.

Due to fog, the contrast of remote objects is very low and stereo does not work. As remote objects are nearly white, we add a zero disparity assumption on those pixels with an intensity equal to I_s. This assumption is of course wrong for white objects. Taking into account the photometric constraint and the assumption on white pixels, the data term is:

$$E_{data} = \begin{cases} E_{data^*} & \text{if } V(i,j) \leq I_L(i,j) + 3\sigma_P \\ & \text{and } V(i,j) \leq I_R(i - D(i,j), j) + 3\sigma_P \\ & \text{and } I_L(i,j) \neq I_s \\ 0 & \text{if } I_L(i,j) = I_s \text{ and } D(i,j) = 0 \\ +\infty & \text{else.} \end{cases} \tag{14}$$

Prior term: In (1), the prior probability $P(D, I_{0L})$ is related to two variables: the disparity D and the intensity I_{0L}. Unfortunately, this kind of mixed prior term is actually difficult to optimize. To be consistent with previous stereo and defogging prior terms, (5) and (10) respectively, the two variables D and I_{0L} cannot be assumed independent of one another. We thus propose to write the prior probability as $P(D, I_{0L}) = P(D|\ddot{I}_{0L})P(I_{0L}|\ddot{D})$, where $\ddot{D}$ and $\ddot{I}_{0L}$ are fixed approximations of D and I_{0L}, given as priors. We thus propose the following prior term for the stereo reconstruction and defogging problem:

$$\begin{aligned} E_{prior} = & \sum_{(i,j)\in X} \sum_{(k,l)\in N} \lambda_{I_0}\, e^{-\frac{\beta\delta}{\ddot{D}(i,j)}}\, W_{I_0}(\nabla \ddot{D}(i,j))\, \rho_{I_0}(|I_{0L}(i,j) - I_{0L}(i+k, j+l)|) \\ & + \lambda_D\, W_D(\nabla \ddot{I}_{0L}(i,j))\, \rho_D(|D(i,j) - D(i+k, j+l)|) \end{aligned} \tag{15}$$

The fact that $\ddot{D}$ and $\ddot{I}_{0L}$ are approximated is not a problem since they appear only in the weights, such as W_{I_0} and W_D, which are very smooth functions. Indeed, the weight W_D is set, like in the stereo reconstruction case, to $W_D(\nabla I) = e^{-\frac{|\Delta I|}{\sigma_g}}$. The weight W_{I_0} is set, like in the defogging case, to $W_{I_0}(\nabla D) = e^{-\frac{|\Delta D|}{\sigma'_g}}$.

Initial $\ddot{D}$ and $\ddot{I}_{0L}$: Variables $\ddot{D}$ and $\ddot{I}_{0L}$ are the approximate disparity and approximate intensity in E_{prior}. The atmospheric veil can be approximately estimated on the left image using a single image defogging algorithm, see for instance [6,7]. Here, it is approximated by minimizing the following w.r.t. $\ddot{V}$:

$$\sum_{(i,j)\in X} |I_L(i,j) - \ddot{V}(i,j)| + \lambda \sum_{(k,l)\in N} |\ddot{V}(i,j) - \ddot{V}(i+k,j+l)| \tag{16}$$

using α-expansion. The small features in the image I_L are lost in $\ddot{V}$, but thanks to the L_1 robust terms, large objects with low contrast are kept. This atmospheric veil $\ddot{V}$ has the important property: it contains object edges. The weights W_D and $W_{I_{0L}}$ in E_{prior} are introduced to attenuate the regularization through these edges. By definition from (6), $V = 1 - e^{-\frac{\beta\delta}{D}}$ (assuming $I_s = 1$ without loss of generality). As a consequence, $\ddot{D}$ can be obtained from $\ddot{V}$. This implies that the factor $e^{-\frac{\beta\delta}{D}}$ in E_{prior} can be substituted by $1 - \ddot{V}$. Another consequence is that $\Delta\ddot{D}$ in W_{I_0} can be approximated by $\Delta\ddot{V}$. Rather than search for a close approximation of $\ddot{I}_{0L}$, we use $\frac{\Delta\ddot{V}}{1-\ddot{V}}$ as a good approximation of $\Delta\ddot{I}_{0L}$.

Complete model: In summary, the stereo reconstruction and defogging problem is set as the following minimization:

$$\min_{D, I_{0L}} E_{data} + E_{prior} \tag{17}$$

In practice, the functions ρ_D and ρ_{I_0} are chosen as the identity. The noise on the image being assumed Gaussian, ρ_P is the square function. For those pixels which verify the photometric constraint and which are not white, the energy which is minimized is, after introduction of $\ddot{V}$:

$$\begin{aligned}
E(D, I_{0L}) = \sum_{(i,j)\in X} \Bigg\{ & \frac{1-\alpha}{\sigma_P^2} \Big(|I_{0L}(i,j) e^{\frac{-\beta\delta}{D(i,j)}} + I_s(1 - e^{\frac{-\beta\delta}{D(i,j)}}) - I_L(i,j)|^2 \\
& + |I_{0L}(i,j) e^{\frac{-\beta\delta}{D(i,j)}} + I_s(1 - e^{\frac{-\beta\delta}{D(i,j)}}) - I_R(i - D(i,j), j)|^2 \Big) \\
& + \alpha\, \rho_S\Big(\frac{|I_L(i,j) - I_R(i - D(i,j), j)|}{\sigma_S}\Big) \\
& + \sum_{(k,l)\in N} \Bigg\{ (1-\alpha)\lambda_{I_0}(1 - \ddot{V}(i,j)) e^{-\frac{|\Delta\ddot{V}(i,j)|}{\sigma_g}} |I_{0L}(i,j) - I_{0L}(i+k, j+l)| \\
& + \lambda_D\, e^{-\frac{|\Delta\ddot{V}(i,j)|}{\sigma_g(1-\ddot{V}(i,j))}} |D(i,j) - D(i+k, j+l)| \Bigg\} \Bigg\}
\end{aligned} \tag{18}$$

As this energy is known up to a scale factor, (18) can be arbitrarily divided by $(1-\alpha)\lambda_{I_0}$. This is used in the next section to estimate σ_P from image residuals.

3.2 Optimization

In (18), D and I_{0L} appear in non-linear unary functions and independently in binary functions. It is thus possible to optimize (18) by means of a two-step alternate minimization: one step consists in minimizing w.r.t. I_{0L} and the other in minimizing w.r.t. D. The first step is defogging and the second step is stereo reconstruction. The energies in both steps being sub-modular, α-expansion is used for the minimization. With the alternate minimization, convergence towards a local minima is guaranteed. Before the first step, the disparity is initialized by stereo reconstruction assuming no fog, i.e by minimizing (18) with $\alpha = 1$.

As pointed in [4], the gradient distribution of a hazy image can be very different from that of a foggy image. This implies that after division of (18) by $(1-\alpha)\lambda_{I_0}$, the factor $\sigma_P\sqrt{\lambda_{I_0}}$ must be set differently from one image to another. When this factor is not correctly set, the chance to converge towards an interesting local minimum decreases. Hopefully, the first term of (18) being quadratic, the factor $\sigma_P\sqrt{\lambda_{I_0}}$ can be easily estimated by estimating the standard deviation of the left intensity residuals $I_{0L}(i,j)e^{\frac{-\beta\delta}{D(i,j)}} + I_s(1-e^{\frac{-\beta\delta}{D(i,j)}}) - I_L(i,j)$.

In summary, the optimization scheme is:

- Compute $\ddot{V}$ by minimization of (16), using α-expansion.
- Initialize D by minimizing (18) w.r.t D, with $\alpha = 1$, using α-expansion.
- Until convergence, iterate:
  1. Until convergence, iterate:
     (a) Minimization of (18) w.r.t I_{0L}, using α-expansion.
     (b) Minimization of (18) w.r.t D, using α-expansion.
  2. Update σ_P by computing the standard deviation of the left intensity residuals.
- $\sigma_P\sqrt{\lambda_{I_0}}$ is enforced to value 1 and a last optimization w.r.t. I_{0L} is performed to better emphasize the detailed texture.

4 Evaluation

4.1 Parameters Setting

The proposed MRF model is mainly parametrized by α which is the weight between the photometric log-likelihood $E_{photo\_fog\_stereo}$ of left and right images and the log-likelihood E_{photo_stereo} of the stereo. When α is close to zero, the obtained disparity map is smooth in homogeneous areas, but the disparity of close objects may be biased as well as the intensity I_{0L}. When α is close to one, the disparity obtained from the stereo log-likelihood is usually correct for close objects but the quality of the reconstruction decreases with the contrast and thus with the depth. Therefore, we recommend to set α close to 0.5 or a little higher.

Another important parameter is the initial value of $\sigma_P\sqrt{\lambda_{I_0}}$. The choice of this value can have an effect on the local minima selected at convergence.

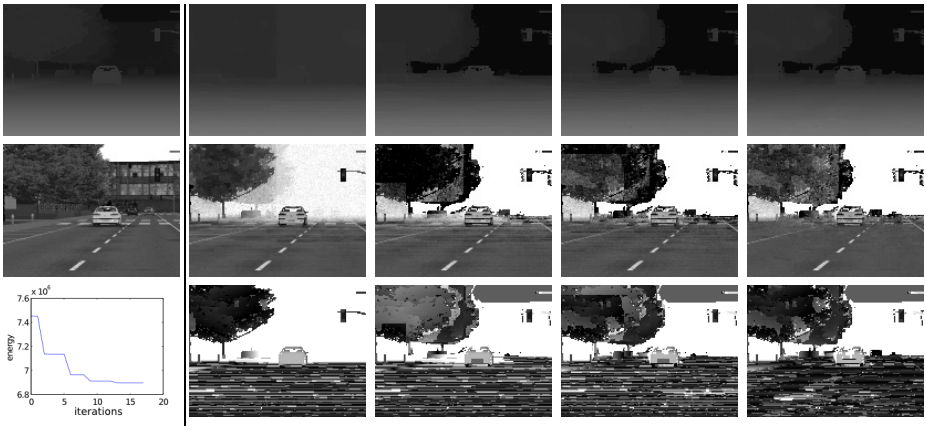


Fig. 2. First column, from top to down: the ground truth disparity map, the image without fog and the energy decrease with iterations. Second column: the disparity map, the restored image with $\sigma_P\sqrt{\lambda_{I_0}} = 1$ and the disparity error map obtained with stereo reconstruction without considering fog. Third column: results of the proposed method when $\sigma_P\sqrt{\lambda_{I_0}} = 20$ at the initialization. Fourth column: the first iteration. Last column: after convergence. For comparison purpose, restorations are processed using the last optimization step to emphasize details.

The bigger $\sigma_P\sqrt{\lambda_{I_0}}$ at the initialization, the smoother is the depth map after convergence. Fig. 2 shows several iterations of the algorithms with $\sigma_P\sqrt{\lambda_{I_0}} = 20$. We can notice that, after one iteration, the large scale of $\sigma_P\sqrt{\lambda_{I_0}}$ allows a better reconstruction and restoration around the closest vehicle. When the number of iteration increases, the scale $\sigma_P\sqrt{\lambda_{I_0}}$ becomes smaller, and the restoration is improved step by step for remote objects. Thus, the two far away vehicles appear. A too large scale can cause wrong stereo matching. However, when α is larger than 0.5, these wrong matches are unusual.

4.2 Synthetic Images

To evaluate the stereo reconstruction in foggy weather, we rely on synthetic images due to the difficulty to have the 3D model of a scene and images of this scene with and without fog. We generate synthetic stereo images using SiVIC$^{\text{TM}}$ software which allows to build physically-based road environments. Uniform fog is added knowing the depth map, see Fig. 3. To make the image more realistic and evaluate the ability of the algorithm to manage the noise, we also added a Gaussian noise on every pixels of left and right images, with standard deviation 1. This database is named FRIDA3 and is available online for comparative studies[1].

We compared the results of three methods: first, the stereo reconstruction based on the classic MRF model without fog; second, the first iteration of the

[1] `http://perso.lcpc.fr/tarel.jean-philippe/visibility/fogstereo.zip`

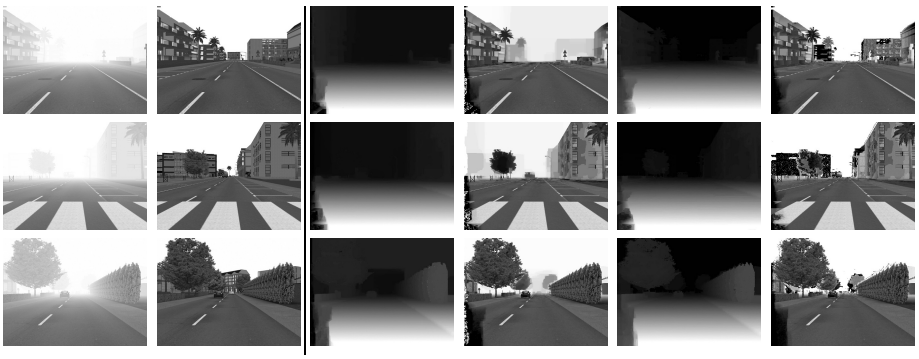


Fig. 3. Results on three images of the synthetic FRIDA3 stereo image database. First column: foggy left images. Second column: same scene without fog. Third and fourth columns: disparity maps obtained using stereo reconstruction without fog and restored images using these disparity maps. Fifth and sixth columns: disparity maps with the proposed method and associated restored images.

Table 1. Comparison of the percentage of correct disparities in average on 66 synthetic stereo pairs using the classic MRF approach without fog (STEREO, see Sec. 2.1), with the photometric constraint and assumption on white pixels added to stereo (STEREO+PC), at the first iteration (FIRST) and after converging (FINAL). Percentages are given for different values of the maximum error err on the disparity (in pixel).

Algorithm	$err < 1$	$err < 0.66$	$err < 0.33$
STEREO	0.776	0.722	0.514
STEREO+PC	0.811	0.764	0.548
FIRST	0.822	0.771	0.552
FINAL	0.828	0.780	0.573

proposed method; third, the proposed method after convergence (with initial $\sigma_P \sqrt{\lambda_{I_0}} = 20$ and $\alpha = 0.5$). Results are shown in Tab. 4.2, in average on 66 stereo pairs. This percentage takes into account only the pixels seen in both images with disparity larger than one, i.e not considering the sky. The stereo without fog (STEREO) achieves 72.2% of correct disparities in the whole image, for a maximum error of 0.66 pixels. When the photometric constraint due to fog veil is added (STEREO+PC), the percentage of correct disparities is improved to 76.4%. This step STEREO+PC corresponds to the initialization of the proposed method. The first iteration of the proposed method (FIRST) achieves 77.1%. After convergence (FINAL), this percentage is increased to 78.0%. From Tab. 4.2, it is clear that the proposed method outperforms the classic stereo reconstruction which does not take the presence of fog into account. In percentage, the improvement due to iterations may seem reduced on the whole image, but these iterations are important to improve correct disparities at long distances.

This fact is illustrated in Fig. 3 which displays obtained disparity maps and restored left images on three stereo pairs of the FRIDA3 database.

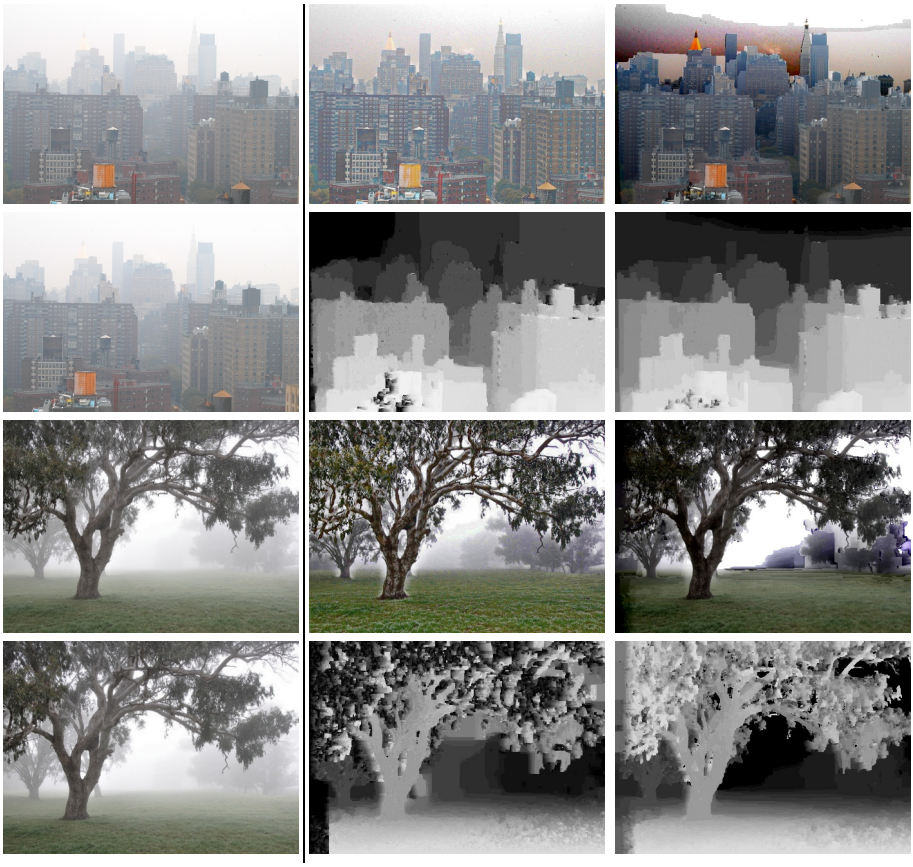


Fig. 4. First column: foggy stereo pair. Second column: Single image defogging with [6] and disparity map obtained by stereo reconstruction without fog. Third column: restored left image and disparity map obtained using the proposed method.

4.3 Camera Images

We compared the proposed method to the stereo reconstruction without fog described in Sec. 2.1 and image defogging described in [6]. β is manually selected. Results show that both the reconstruction and restoration are of better quality. In Fig. 4, results are compared on urban and country side stereo pairs. One may note that the obtained stereo reconstruction are dense at both short and long distances, contrary to stereo reconstruction without taking into account the fog. The stereo restoration obtained by the proposed method is of good quality compared to single image defogging results. At close distances, outliers

are avoided thanks to the photometric constraint and the true intensity of objects is kept. At a far distances, the contrast is greatly enhanced without amplifying the noise to much.

5 Conclusion

We proposed a MRF model to solve the stereo reconstruction and image defogging in daytime fog. It is an extension of two sub-models: the classical stereo reconstruction without fog and newly introduced image restoration when the depth is known. The proposed model includes the photometric constraint and priors on white pixels. It leads to the optimization of an energy which can be solved by an alternate scheme based on the application of successive α-expansion optimizations. The convergence towards a local minimum is thus guaranteed. Tests on both synthetic stereo pairs and camera stereo pairs show the relevance of the model. Thanks to the stereo depth clue, the disparity is correct at short distances, and thanks to the atmospheric veil depth cue, the disparity is drastically improved at long distances. The obtained restored results are better than the ones obtained without stereo thanks to the simultaneous estimation with the disparity map. Perspectives for future research are to take into account non constant sky, non Gaussian noise to improve scale estimation, to explicitly take into account occlusions in the formulation, to speed up the algorithm for real time applications and to extend the previous model to heterogeneous fog.

References

1. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, 1222–1239 (2001)
2. Woodford, O., Torr, P., Reid, I., Fitzgibbon, A.: Global stereo reconstruction under second-order smoothness priors. IEEE Transactions on Pattern Analysis Machine Intelligence 31, 2115–2128 (2009)
3. Geiger, A., Roser, M., Urtasun, R.: Efficient Large-Scale Stereo Matching. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 25–38. Springer, Heidelberg (2011)
4. Nishino, K., Kratz, L., Lombardi, S.: Bayesian defogging. International Journal of Computer Vision 98, 263–278 (2012)
5. Hautière, N., Tarel, J.-P., Lavenant, J., Aubert, D.: Automatic fog detection and estimation of visibility distance through use of an onboard camera. Machine Vision and Applications 17, 8–20 (2006)
6. Tarel, J.-P., Hautière, N.: Fast visibility restoration from a single color or gray level image. In: Proceedings of IEEE International Conference on Computer Vision (ICCV 2009), Kyoto, Japan, pp. 2201–2208 (2009)
7. Tarel, J.-P., Hautière, N., Caraffa, L., Cord, A., Halmaoui, H., Gruyer, D.: Vision enhancement in homogeneous and heterogeneous fog. IEEE Intelligent Transportation Systems Magazine 4, 6–20 (2012)

Large-Scale Bundle Adjustment by Parameter Vector Partition

Shanmin Pang, Jianrue Xue, Le Wang, and Nanning Zheng

Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University

Abstract. We propose an efficient parallel bundle adjustment (BA) algorithm to refine 3D reconstruction of the large-scale structure from motion (SfM) problem, which uses image collections from Internet. Different from the latest BA techniques that improve efficiency by optimizing the reprojection error function with Conjugate Gradient (CG) methods, we employ the parameter vector partition strategy. More specifically, we partition the whole BA parameter vector into a set of individual subvectors via normalized cut (Ncut). Correspondingly, the solution of the BA problem can be obtained by minimizing subproblems on these subvector spaces. Our approach is approximately parallel, and there is no need to solve the large-scale linear equation of the BA problem. Experiments carried out on a low-end computer with 4GB RAM demonstrate the efficiency and accuracy of the proposed algorithm.

1 Introduction

The large-scale structure from motion (SfM) technique [1], [2], [3], [4], [5] which uses image collections from Internet has become a popular topic in recent years, and attracted more and more attention to the bundle adjustment (BA) technique. BA, which aims to refine a visual reconstruction to produce jointly optimal 3D structure and camera parameter estimates [6], is used as the last step of each SfM algorithm. Even though rapid progress [7], [8], [9], [10], [11] has been made in this field, the efficiency of BA algorithms is still an open problem due to the very large number of parameters involved.

Much effort has been spent on traditional BA problems (i.e., source images of SfM come from video sequences and the size of BA is usually small). Shum et al. [12] introduce an efficient way to reduce the number of parameters by using two virtual key frames to represent a sequence, and it results in a significant speedup of the BA algorithm. However, the convergence of the proposed algorithm is still a pending issue. Instead of iteratively adjusting all the structure and motion parameters, Steedly and Essa [13] propose an incremental BA algorithm that only optimizes the parameters of change when adding a new frame. Though the algorithm converges, and is faster than the original BA, it cannot work in the case that the data are highly interdependent. The technique in [14] does not solve the normal equations directly, instead it permutes the Hessian matrix of the reprojection error function by spectral partitioning such that the large problem can be partitioned into several smaller and well-conditioned subproblems. Its

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 26–39, 2013.

limitation, however, is that at each iteration, a partition is needed, which might increase the complexity of the algorithm. The method in [10] executes BA in an out-of-core manner, which decouples the original problem into several submaps, so that the problem can be solved in parallel. However, an expensive merging step is needed to obtain the final complete solution.

Recently, several methods have been presented to address large-scale BA problems. In [15], an inexact Newton method which pairs with relatively simple preconditioners is employed to get an approximation solution of the normal equations at each iteration. The approach in [16] applies the Conjugate Gradient Least Square (CGLS) algorithm to BA, which avoids formulating the Hessian matrix of the reprojection error function, thus saving memory and computing time. Another work is proposed by Wu et al. [17], in which they address BA on a multicore computer to increase efficiency. Although these methods can solve large-scale optimization problems in theory, their computing cost of the whole optimization is still huge in practice. There is still a computing power gap between the computational requirement of BA algorithms and that can be provided by a normal computer. As claimed in [15], [17], the authors perform experiments on a workstation with dual Quad-core CPUS clocked at 2.27Ghz with 48GB RAM, and another same situation happens in [16]. The latest method [11] also cannot fit into a 8GB RAM memory when using the BAL datasets [15], which further demonstrates that the state-of-the-art BA algorithms still cannot run on a low-end computer due to the high consummation of memory.

In this paper, we propose a parameter vector partition bundle adjustment (VPBA) algorithm by exploiting sparsity of large-scale BA problems. Specifically, we first use the normalized cut (Ncut) algorithm [18] to partition the whole parameter vector into a set of individual sub-vectors, then iteratively solve BA subproblems on these sub-vector spaces to converge to the optimal solution of the original BA problem. Our work is similar to methods in [10] and [14]. However, The proposed VPBA algorithm is different from the method in [10] in that VPBA does not need any merging step, which is an important and necessary step in [10]. This feature makes VPBA more suitable for large-scale BA problems. Moreover, instead of partitioning the original BA problem into several subproblems at each iteration in [14], our algorithm partitions parameter vector only once, and is therefore more simple and efficient.

In summary, the proposed VPBA algorithm has the following characteristics: 1) It does not need to compute the Hessian matrix of the reprojection error function. More importantly, the approach avoids solving large-scale linear systems, which is often a heavy load for large-scale BA problems. Therefore, a significant amount of memory and computation time can be saved. 2) The partition strategy of VPBA makes the algorithm approximately parallel, and each BA subproblem can be easily accomplished on a low-end computer. 3) The experimental results (Section 4) show that the proposed algorithm is reliable, accurate and fast in practice, though we currently cannot provide a theoretical proof of the convergence of VPBA.

The rest of the paper is organized as follows: In Section 2, a brief introduction to the BA problem is provided. The proposed VPBA algorithm is presented in Section 3 and evaluated in Section 4. Finally, a conclusion is given in Section 5.

2 Revisit Bundle Adjustment

Assume that n 3D points are observed in m views, and let x_{ij} be the projected measurement of the ith 3D point on image j. BA minimizes the reprojection error with respect to all 3D points x_i $(i \in 1, \ldots, n)$ and camera parameters c_j $(j \in 1, \ldots, m)$, specifically

$$\min_{S} \|f(S)\|^2 = \min_{c_j, x_i} \sum_{i=1}^{n} \sum_{j=1}^{m} \|h(c_j, x_i) - x_{ij}\|^2, \tag{1}$$

where $\|f(S)\|^2$ is the sum of squares of reprojection error, and $h(c_j, x_i)$ is the predicted projection of 3D point i on image j. For simplicity, we let $S = (c_1, \ldots, c_m, x_1, \ldots, x_n)^T$ denote all unknown parameters.

The Gauss-Newton algorithm is a standard algorithm for Eq. (1). Usually, f is approximated by a small $\|\delta_S\|$, i.e,

$$f(S + \delta_S) \approx f(S) + J\delta_S, \tag{2}$$

where J is the Jacobian matrix of f. At each iteration, minimizing $\|f(S + \delta_S)\|$ leads to the following *normal equations*:

$$(J^T J)\delta_S = -J^T f, \tag{3}$$

where $J^T J$ is an approximation to the Hessian matrix of $\|f\|^2$. However, it is difficult to meet with the requirement of a suitable step control policy to guarantee convergence of the Gauss-Newton algorithm, especially when J is rank-deficient, or nearly so. The Levenberg-Marquardt (LM) algorithm avoids this by adding a damping term λI $(\lambda > 0)$ to $J^T J$, where λ is referred to as the damping term. This leads to solve the following damped system:

$$(J^T J + \lambda I)\delta_S = -J^T f. \tag{4}$$

LM is still inefficient in solving Eq. (4) when it is large-scale.

To reduce the size of the large linear system, one well known method, *Schur complement trick*, is widely adopted. Specifically, we can partition the Jacobian matrix into a camera part J_C and a point part J_P as $J = [J_C, J_P]$ by exploiting the structure of the BA parameter space. Thus $J^T J$ has the form:

$$\begin{bmatrix} J_C^T \\ J_P^T \end{bmatrix} \begin{bmatrix} J_C, J_P \end{bmatrix} = \begin{bmatrix} J_C^T J_C & J_C^T J_P \\ J_P^T J_C & J_P^T J_P \end{bmatrix} = \begin{bmatrix} U & W \\ W^T & V \end{bmatrix}, \tag{5}$$

where $U \in \mathbb{R}^{mc \times mc}$ is a block diagonal matrix with m blocks of size $c \times c$, and c is the number of the parameters of a single camera; $V \in \mathbb{R}^{np \times np}$ is a block diagonal

matrix with n blocks of size $p \times p$, and p is the number of the parameters of a single 3D point. Applying Gaussian elimination to Eq. (4) yields a simplified system

$$(U^* - WV^{*-1}W^T)\delta_{S_C} = -J_C^T f + WV^{*-1}J_P^T f, \tag{6}$$

where * denotes the augmentation of the diagonal elements of U and V. After we get δ_{S_C} with Eq. (6), we can then get δ_{S_P} by

$$V^*\delta_{S_P} = -J_P^T f - W^T\delta_{S_C}. \tag{7}$$

The *Schur complement trick* reduces the size of the linear system from $(mc + np) \times (mc+np)$ to $(mc) \times (mc)$. In practical applications, m is often much smaller than n, so huge amount of memory and computations can be saved.

In the case that there are several hundred cameras, Eq. (6) can be efficiently handled by many efficient strategies. One of most popular algorithms employing the *Schur complement trick* is sparse BA (SBA) [19]. SBA solves Eq. (6) via the Cholesky factorization method, and achieves a high performance. As reported in [15], SBA is successful for small problems. However, for large-scale problems ($m = 10^3 \sim 10^4$), SBA may still fail because the cost of cholesky factorization is prohibitively expensive. In order to solve this challenging problem, Conjugate Gradient (CG) methods [15], [16], [17] are used to solve Eq. (4) at the cost of obtaining an approximate solution of Eq. (4).

However, all these aforementioned BA algorithms still have to deal with huge matrix operations and the needs of solving large-scale linear systems, which are not trivial.

3 The Vector Partition BA Algorithm

In this section, we present the proposed VPBA algorithm. The latest BA algorithms proposed in [15], [16] and [17] try to improve the efficiency of solving Eq. (4). Being distinct from these approaches, the VPBA contrives to partition the whole parameter vector into a set of individual sub-vectors, and decomposes the original optimization problem Eq. (1) into a set of individual subproblems. After partitioning, each subproblem can be solved by the LM algorithm on a low-end computer. The final solution of Eq. (1) is a straightforward combination of solutions of these subproblems.

3.1 Exploiting Sparsity

BA becomes a large-scale optimization problem when the size of parameter vector S is large. Fortunately, the large-scale BA problem has useful properties of structure and sparseness. This motivates us to design an efficient BA algorithm by exploiting the structure and sparseness of the large-scale problem.

By investigating the reprojection error function Eq. (1), we find that each individual component only depends on two composite parameters c_j and x_i.

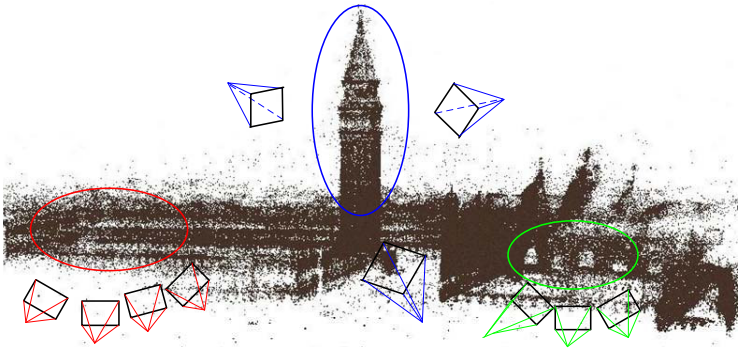


Fig. 1. Illustration of sparsity of BA. 3D points in blue ellipse are mainly reconstructed by cameras marked in blue, and these 3D points have fewer projections on images marked in red and green. Cameras marked in red and green photograph different parts of the scene, and thus no 3D points to connect images in red and green together. **Remarks**: 1) The Venice model with 1778 cameras is optimized by our VPBA algorithm, and this initial 3D model is released by Agarwal et al [15]. 2) Only schematic positions of a few cameras are illustrated.

This means that the reprojection error function is a *partial separable* function [20]. This structural property forms a solid basis for our parameter vector partition, and inspires us to consider the sparseness of the camera parameter space and 3D point parameter space separately.

Firstly, we consider the sparseness of the camera parameter space. Each camera only photographs a very small portion of landmarks due to its limited view scope. For example, for the scene containing 4.5 millon points and 13682 cameras in the BAL datasets [15], a camera covers at most 20,000 3D points, and most of cameras can only cover several hundred or thousand points. This means that if we fix j in Eq. (1), except for a few cameras, components of Eq. (1) related to camera c_j account for a very small part of the overall parameter vector.

Secondly, the sparseness also exists in the 3D points. For a specific scene, only extremely few points are simultaneously visible in the hundreds of cameras, and most of points are only observed by dozens of cameras. In other words, when we fix i in Eq. (1), the number of components of Eq. (1) depended on point x_i is relatively small, too.

For a single image, only a small fraction of the image collection can be matched with a large number of feature points. With the sparseness of the camera parameter space, we can conclude that the number of 3D points reconstructed by these matched images is small compared with the size of the whole 3D point set. Meanwhile, we can infer these reconstructed 3D points are mainly visible in these images, and have fewer projections on other images by the sparseness of 3D point parameter space. Fig. 1 illustrates this observation.

Additionally, the sparseness of the parameter space gives rise to another apparent fact: some cameras photograph different parts of a large scene, and they share no common content. This leads to a situation that we cannot reconstruct any 3D points from these images. In other words, there are no 3D points to connect these images together. This is also illustrated in Fig. 1.

Based on the aforementioned structural properties, we obtain a parameter vector partition strategy as follows: 1) partition the camera parameter vector into individual camera groups, 2) partition the 3D point parameter vector into point groups according to the partitioned camera groups. Through this partition, a large-scale BA problem can be decomposed into subproblems accordingly. This partition strategy makes the VPBA algorithm avoid huge matrix operations, such as computing and inverting of the full Hessian matrix, and thus results in speedup, and memory saving.

3.2 Partition Parameter Vector

In this section, we present the parameter vector partition in detail. To do this, let us first define a partition of vector $S \in \mathbb{R}^n$ as follows:

Definiton 1 (Partition) *For the parameter vector S composed by variables a_1, $a_2, \cdots, a_m$, where $a_i \in \mathbb{R}^{i_k}$ and $\sum_{i=1}^{m} i_k = n$. namely, $S = (a_1^T, a_2^T, \cdots, a_m^T)^T$. if S is partitioned into $S = (S_1, S_2, \cdots, S_q)^T$, where $S_j = (a_{j_1}^T, a_{j_2}^T, \cdots, a_{j_{l_j}}^T)$ and $j_t (t = 1, 2, \cdots, l_j) \in \{1, 2, \cdots, m\}$, then we say these sub-vectors S_i form a partition $\{S_1, \cdots, S_q\}$ of the original vector S. That is, $\bigcup_{j=1}^{q} S_j = S$ and $S_i \cap S_j = \emptyset$, if $i \neq j$. Moreover, $\bar{S}_j$, which is the complement vector of S_j, satisfies $\bar{S}_j \cup S_j = S$ and $\bar{S}_j \cap S_j = \emptyset$, $\forall j \in \{1, \cdots, q\}$.*

According to this definition, we can decompose any minimization problem $\min_{S \in \mathbb{R}^n} g(S)$ as

$$g(S) = g(S_j, \bar{S}_j) + g(\bar{S}_j), \tag{8}$$

where $g(\bar{S}_j)$ only depends on the parameters in $\bar{S}_j$.[1] After getting $g(\bar{S}_j)$, $g(S_j, \bar{S}_j)$ can be computed by $g(S_j, \bar{S}_j) = g(S) - g(\bar{S}_j)$.

Now, we state our method to solve the large-scale minimization problem $\min_{S \in \mathbb{R}^n} g(S)$: first, partition the vector $S \in \mathbb{R}^n$ into sub-vectors $\{S_1, \cdots, S_q\}$, and then decompose the original problem into subproblems defined on these sub-vector spaces according to Eq. (8), and finally, solve the original problem by iteratively minimizing these q subproblems (we describe details in Subsection 3.3).

Thus, the first step of VPBA becomes clear: to partition the whole parameter vector S into a partition P. Considering the convergence speed and the size of subproblems, the partition P should meet two basic requirements: 1) the size of sub-vectors should be small enough so as to be solved with a normal computer; 2) the number of coupled parameters should be as small as possible.

[1] We give a simple example to make Eq.(8) more readable: suppose $g(S) = (x_1 - x_2)^2 + (x_1 - x_3)^2 + (x_2 - x_3)^2$, and $S = (x_1, x_2, x_3)^T$; if we let $S_1 = x_1$ and $\bar{S}_1 = (x_2, x_3)^T$, then $g(S_1, \bar{S}_1) = (x_1 - x_2)^2 + (x_1 - x_3)^2$ and $g(\bar{S}_1) = (x_2 - x_3)^2$.

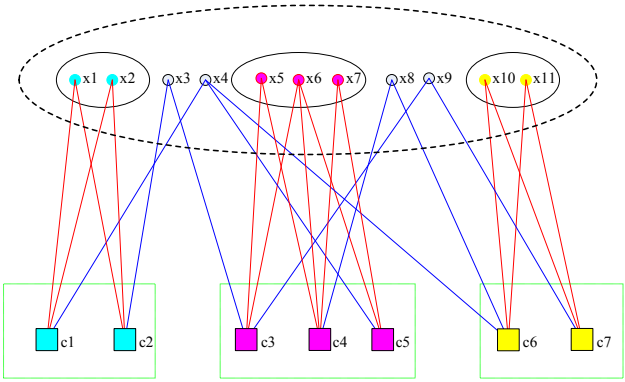


Fig. 2. Schematic illustration of decomposing all camera and 3D point parameters. This example assumes that 7 cameras observe 11 points. According to image similarities, we first use Ncut to partition 7 cameras into 3 groups: C_1, C_2, C_3, where $C_1 = \{c_1, c_2\}$, $C_2 = \{c_3, c_4, c_5\}$, $C_3 = \{c_6, c_7\}$. Correspondingly, points are partitioned into 4 groups: $X_1 = \{x_1, x_2\}$, $X_2 = \{x_5, x_6, x_7\}$, $X_3 = \{x_{10}, x_{11}\}$, $X_4 = \{x_3, x_4, x_8, x_9\}$ (see text in detail). Thus, $S_i = \{C_i, X_i\}(i = 1, 2, 3)$ and $S_4 = X_4$ constitute a partition.

In order to trade-off between these two requirements, we first divide the camera parameters by Ncut. Other grouping algorithms, such as K-means and Mean shift, may also be easily adopted in our framework. Specifically, we build an undirected weighted graph for the image collection, where each node denotes a single image, and each edge denotes the connection of each pair of images. The weighted matrix W is built in this way: w_{ij} stands for the number of 3D points that image i and image j share. This implies that if image i and image j are very similar, w_{ij} is large, and vice versa. Once W is constructed, the Ncut algorithm can partition the full image set into groups. In order to satisfy requirement (1), we adopt two-way Ncut repeatedly until every camera group contains a small number of cameras.

Next, we partition 3D point parameters into groups. Given the K camera groups (we denote them as $C_1, \cdots, C_K$) that we have already obtained by Ncut, images within a same group are with strong similarities, and jointly represent a segment of the scene. Different groups share few or no connection, and represent different segments of the scene. With this observation, we can divide the 3D point sets into two classes: intra-points, and inter-points. Intra-points are those observed by cameras within a same group C_l $(1 \leq l \leq K)$. Inter-points are points which do not meet with this condition (i.e., they are observed by cameras from at least two groups). In this way, all these 3D points are divided into $K + 1$ groups: $X_1, \cdots, X_{K+1}$, where X_l $(1 \leq l \leq K)$ is made up by intra-points, and X_{K+1} is made up by inter-points, respectively.

Finally, we do a merging step to partition the whole parameter vector into $K + 1$ sub-vectors: parameters of both the camera group C_l and 3D point group

X_l are merged to span a parameter sub-vector S_l ($1 \leq l \leq K$). Thus, according to Definition 1, $P = \{S_1, \cdots, S_{K+1}\}$ constitutes a partition, where $S_{K+1} = X_{K+1}$. The complete process of parameter vector partition is illustrated in Fig. 2.

Corresponding to partition P, the original BA problem defined on vector space S is decomposed into $K+1$ subproblems defined on vector spaces $S_1, .., S_{K+1}$. These subproblems are interacted with each other only by coupled parameters. As we will see in Subsection 3.3, all inter-points (i.e., S_{K+1}) and cameras related to them make up coupled parameters. Obviously, coupled parameters are not too many based on the features of Ncut, and this can be further verified with BAL datasets [15] in Subsection 4.1.

Now, the parameter vector partition algorithm can be summarized as follows:

1. Represent the full image set as a graph and set up the weighted matrix W, then use Ncut to partition all cameras into K groups: $C_1, \cdots, C_K$.
2. Use camera groups to cut 3D points and get $K+1$ groups of points: $X_1, \cdots, X_{K+1}$, where points in X_l ($1 \leq l \leq K$) are only observed by cameras in C_l. Points which do not meet this condition form X_{K+1}.
3. Let $S_l = \{C_l, X_l\}$ ($1 \leq l \leq K$) and $S_{K+1} = X_{K+1}$, then $\{S_1, \cdots, S_{K+1}\}$ is a partition of the parameter vector.

It should be noted that, in order to meet requirement 1), Ncut sometimes produces a few groups with too small size. However, this is not a problem, since we can simply merge these small groups into a group. This step is necessary, because it can reduce the number of coupled parameters, and also can keep a balance between the largest groups and the smallest ones, which is important to the parallelization of the VPBA algorithm (see Subsection 3.4).

3.3 Iterate to Convergence

Given a partition, the corresponding minimization functions have defined expressions. Let us denote f_l as a minimization function corresponding to vector space S_l. According to Eq. (8), f_l ($1 \leq l \leq K$) is only dependent on set C_l, X_l, and $X_{K+1,l}$, where $X_{K+1,l}$ is a a subset of X_{K+1}, and points in it have image projections on cameras in C_l. Clearly,

$$f_l = \sum_{x_i \in X_l} \sum_{c_j \in C_l} \|h(c_j, x_i) - x_{ij}\|^2 + \sum_{x_i \in X_{K+1,l}} \sum_{c_j \in C_l} \|h(c_j, x_i) - x_{ij}\|^2. \quad (9)$$

We fix points in set $X_{K+1,l}$ (i.e., $X_{K+1,l}$ is the constant parameter) and optimize every element of S_l when minimizing Eq. (9). This demonstrates that f_l has nothing to do with another minimization function f_q ($l \neq q$), so that they can be minimized in parallel.

Note that compared with the original BA problem in Eq. (1), Eq. (9) only adds some fixed points. This indicates Eq. (9) and Eq. (1) have nearly the same

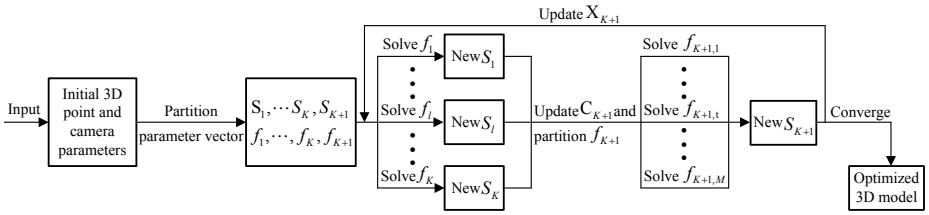


Fig. 3. The framework of the VPBA algorithm. Note that $f_1, \cdots, f_K$ are independent, so we solve them in parallel. After updating S_{K+1}, we handle f_{K+1} on M processors since it is a separable function. Moreover, the number of parameters of f_{K+1} (see Table 1) is not much, so solving it takes a little time at each iteration.

structure, but Eq. (9) has much fewer parameters to be optimized, so we can use previous BA methods with a slight modification. In the experiment, we choose the SBA algorithm to solve them, since as reported in [15], SBA has the best performance in solving these small BA problems.

Similarly, according to Eq. (8), f_{K+1} depends on inter-points (i.e., X_{K+1}) and cameras (we call C_{K+1}) which observe inter-points. Its parameters and constant parameters are X_{K+1} and C_{K+1}, respectively. Likewise, we state f_{K+1} as follows:

$$f_{K+1} = \sum_{x_i \in X_{K+1}} \sum_{c_j \in C_{K+1}} \|h(c_j, x_i) - x_{ij}\|^2. \tag{10}$$

When minimizing Eq. (10), we need to fix camera parameters and optimize inter-points. Given the camera parameters, each point can be optimized independently (i.e., f_{K+1} is *a separable function*). This means solving f_{K+1} is much easier than f_l, and its solution can be obtained only by solving $|X_{K+1}|$ linear systems with size $2|C_{K+1}| \times 3$ [12].

Finally, we can solve the original BA problem by iteratively solving subproblems as follows:

1. Fix $X_{K+1,l}$ and solve one step of f_l $(1 \leq l \leq K)$ simultaneously using any monotonically descent and convergent algorithm (for example, SBA [19]), get new $S_1, \cdots, S_K$.
2. Use the result of step 1 to update C_{K+1} into f_{K+1}, then fix C_{K+1} and solve one step of f_{K+1}, get new X_{K+1} and in turn update f_l.
3. Repeat steps 1 and 2 until convergence.

The algorithm alternates coupled parameters X_{K+1} and C_{K+1} between steps 1 and 2, and updates uncoupled parameters $X_1, \cdots, X_K$ only at step 1. Since algorithms employed for solving subproblems are monotonically descent, hence the total reprojection error is decreased at each iteration. Furthermore, VPBA can be implemented almost in parallel, and experimental results show its efficiency. Fig. 3 illustrates the entire implementation process of the VPBA algorithm.

3.4 Parallel Measures

A difficult issue in the implementation of VPBA is that, though we start $f_1, \cdots, f_K$ at the same time, they may end at different time due to their different size. In order to shorten waiting time, we reduce the exchanging frequency of coupled parameters by two measures. First, at step 1 of the VPBA algorithm, rather than just running one step of f_l, we iterate it several times, and different subproblems have different iterations (i.e., for larger subproblems, we set fewer iteration numbers, and vice versa.). This idea ensures the cost time of these subproblems is roughly the same.

Second, given the updated camera parameters C_{K+1}, we minimize f_{K+1} with respect to X_{K+1} at step 2 of the VPBA algorithm. f_{K+1} is separable and has a relatively small number of parameters (see Table 1) to be optimized. We can handle f_{K+1} on multiple processors, thus it only takes a little time in each loop. These two simple measures make the VPBA algorithm approximately parallel.

4 Experiments and Results

In this section, we evaluate the VPBA algorithm using BAL datasets released by Agarwal et al. [15]. The BAL datasets contain five categories of datasets: Dubrovnik, Final, Ladybug, Trafalgar Square and Venice. Each of them has dozens of 3D models that are reconstructed with different numbers of cameras. We choose 24 large models to evaluate the performance of the VPBA algorithm for large-scale BA problems.

The models initially contain a relatively large number of outliers. Similar to methods in [1], [16], [17], we remove outliers as follows: 1) remove 3D points that are in the back of (or close to) camera planes; 2) reject points with a large reprojection error; 3) filter out cameras whose calibration information is obvious wrong, such as focal length is negative.

4.1 Comparison with SBA

In this section, we design experiments to demonstrate whether the parameter vector partition strategy of VPBA is effective. For this purpose, we implement both VPBA and SBA using a low-end computer with 4GB RAM. Since the SBA algorithm does not use our parameter vector partition strategy, thus we can compare the performance of VPBA with SBA using the same datasets.

In the implementation of the VPBA algorithm, we stop partitioning the parameter vector until the largest camera group has fewer than 1000 cameras for each BA problem. Table 1 lists the maximum number of cameras including all the subproblems f_l $(1 \leq l \leq K)$. Inter-points and cameras related to them constitute all coupled parameters, and points usually account for most of these coupled parameters. We also list the number of inter-points for each BA problem in Table 1. It clearly shows that inter-points account for a very small fraction $(0.02 \sim 0.10)$ of all 3D points in the Venice and Ladybug datasets. It should be

noted, for the Final dataset, the fraction of inter-points becomes a little larger, and is about twenty percent of all the 3D points. We think this is probably caused by two factors: 1) the connectivity graph of cameras is more complex than that in the Venice and Ladybug datasets, and grouping these cameras leads to more coupled parameters ; 2) the size of these models is much larger than that in the Venice and Ladybug datasets. To make it feasible running on a PC, we need more groups of cameras which inevitably gives rise to more coupled parameters.

For each initial model in the Ladybug and Venice datasets, VPBA and SBA are stopped by the same criteria: maximum iterations (50) or the relative

Table 1. Comparison results of VPBA and SBA. The first column corresponds to the name and index in the original datasets: "L" for "Ladybug", "V" for "Venice" and "F" for "Final". m and n denote the number of cameras and 3D points of the original problem, respectively. K is the number of camera groups and m_s stands for the maximum number of cameras including all the subproblems f_l ($1 \leq l \leq K$). n_u is the amount of coupled 3D points. We evaluate our algorithm in terms of time (in minutes) and the final mean squared reprojection error (in pixels). The last column denotes the speed up ratio of VPBA over SBA. '–' means SBA cannot fit into memory on our platform (4GB RAM).

							VPBA		SBA		
name	m	m_s	K	n	n_u	n_u/n	error	time	error	time	r_t
L-17	969	392	3	121,633	3,856	0.03	0.86	16	0.92	157	9.8
L-19	1064	419	3	121,633	4,681	0.04	0.83	25	0.85	318	12.7
L-22	1,197	499	3	121,633	2,896	0.02	0.78	35	0.81	450	12.9
L-24	1,266	503	3	127,787	2,953	0.02	0.72	35	0.76	557	15.9
L-25	1,340	501	3	129,306	3,033	0.02	1.01	38	0.98	724	19.1
L-26	1,469	520	3	140,029	4,388	0.03	0.83	38	0.81	853	22.5
L-27	1,586	545	4	145,006	4,512	0.03	0.70	43	0.72	1061	24.7
L-29	1,690	570	4	149,121	4,593	0.03	0.72	47	–	–	–
L-31	1,712	573	4	149,707	4,598	0.03	0.77	54	–	–	–
V-04	423	271	2	272,523	16,352	0.06	2.51	25	2.47	63	2.5
V-05	740	412	2	475,217	17,234	0.04	2.02	62	1.99	198	3.2
V-09	1,179	424	3	699,114	23,752	0.03	1.97	66	1.95	573	8.7
V-11	1,281	552	3	743,047	20,405	0.03	1.81	90	1.81	1121	12.5
V-12	1,343	514	3	766,029	26,482	0.03	1.74	68	1.75	1279	18.8
V-16	1,483	430	4	796,053	26,310	0.03	1.97	62	–	–	–
V-18	1,537	444	4	802,756	28,153	0.04	1.98	67	–	–	–
V-26	1,689	439	4	840,442	72,779	0.09	1.94	53	–	–	–
V-29	1,770	486	4	849,761	86,463	0.10	1.92	54	–	–	–
F-03	869	494	2	418,517	96,572	0.23	1.59	73	1.56	231	3.2
F-04	961	546	2	161,069	30,909	0.19	1.68	48	1.70	270	5.6
F-05	1,936	599	4	561,238	150,127	0.27	1.97	119	–	–	–
F-06	3,017	875	6	252,466	41,268	0.16	1.81	293	–	–	–
F-07	4,557	922	6	1,280,289	260,998	0.20	1.53	364	–	–	–
F-08	13,608	923	**17**	3,773,337	684,261	0.18	1.78	378	–	–	–

reduction in the magnitude of the reprojection error (10^{-12}). VPBA works well on these two datasets: it's much faster than SBA (3 times $\backsim$ 24 times speedup), and can reach the comparable reprojection error (Table 1).

For models F-03 $\sim$ F-08, we run the VPBA algorithm for a maximum 100 iterations (50 iterations for SBA), and can reach the comparable reprojection error with SBA. The slow convergence on these models is probably caused by more coupled parameters than those in the Venice and Ladybug datasets. However, note that BA problems has cubic complexity, it's worth increasing the number of iterations because the size of each subproblem is much smaller than the original problem. More importantly, even with 100 iterations, our algorithm is much faster than SBA, and it can solve large-scale BA problems on a low-end computer.

4.2 Comparison with the State-of-the-Art BA Algorithms

In this section, we compare VPBA with the state-of-the-art algorithms presented in [15] in terms of speed. It's difficult for us to compare our VPBA algorithm with them directly, as they need to perform on a workstation with large memory and powerful CPU. However, we can compare VPBA with the latest algorithms indirectly. Specifically, Agarwal et al. [15] propose four new algorithms: explicit-jacobi, normal-jacobi, implicit-ssor, implicit-jacobi, and they also report the

Table 2. Compare our algorithm with four latest algorithms proposed in [15] in terms of speed. The first column denotes the test datasets. From the second column to the last column, each one shows the speed up ratio over SBA. The comparison result of VPBA with SBA is obtained in a same computing platform. The other four algorithms are compared with SBA with another computing platform, and their results are reported in [15]. Since authors of [15] do not publish their source code, so we use SBA as a basis to perform the comparison. This indirect way shows that the VPBA algorithm is much faster than SBA, hence outperforms these four latest algorithms.

name	VPBA	explicit-jacobi	implicit-jacobi	implicit-ssor	normal-jacobi
L-17	**9.8**	1.6	5.8	0.5	0.9
L-19	**12.7**	0.7	6.1	0.2	0.3
L-22	**12.9**	5.1	10.4	3.4	4.1
L-24	**16.9**	3.1	10.0	1.0	1.5
L-25	**19.1**	1.5	11.2	0.7	0.7
L-26	**22.5**	4.3	11.6	0.7	0.7
L-27	**24.7**	4.9	7.8	2.3	2.1
V-04	**2.5**	1.1	0.6	0.7	0.1
V-05	**3.2**	0.9	2.2	0.5	0.9
V-09	**8.7**	1.0	0.6	0.1	0.3
V-11	**12.5**	0.8	1.1	1.1	0.5
V-12	**18.8**	0.8	1.1	1.1	0.5
F-03	**3.2**	1.4	0.5	1.4	2
F-04	5.6	2.6	**11.9**	6.9	1.4

comparison results of these four algorithms with SBA. These comparison results can enable us to compare VPBA with these four algorithms indirectly. Table 2 lists speed up ratios of these four algorithms over SBA on platform reported in [15]. It clearly shows that explicit-jacobi, normal-jacobi, and implicit-ssor have no significant advantage than SBA. However, VPBA is much faster than SBA, so we can conclude that the VPBA algorithm is faster than these three algorithms. In addition, VPBA can compare with implicit-jacobi, which is the best of the four algorithms in [15]. All of these indicate that our VPBA algorithm is fast.

5 Conclusions

We have presented a new VPBA algorithm to large-scale BA problems that avoids huge matrix operations by decomposing the original optimization problem into subproblems. We first partition the large-scale parameter vector into a set of sub-vectors according to the features of BA problems, then define subproblems on these sub-vectors, and finally solve them iteratively. The structure of subproblems is similar with the original problem, but have much fewer number of cameras and points than the original problem, so each of them can be more efficiently solved. A key contribution of our work is that we can accomplish large-scale BA problems on a low-end computer. We demonstrate the performance of the VPBA algorithm in our experiments, and the results are promising, though we currently can not provide theoretical proof of its convergence.

Our future work includes three aspects. First, the VPBA algorithm has not reached parallel completely, so how to parallelize the algorithm is a task to investigate. Second, like other BA algorithms, the proposed algorithm converges fast during the first few steps, but slows down after dozens of steps. We will further study how to make the algorithm perform faster. Third, we will prove the convergence of the VPBA algorithm and apply this result to other large-scale optimization problems in computer vision.

References

1. Snavely, N., Seitz, S., Szeliski, R.: Modeling the world from internet photo collections. International Journal of Computer Vision 80, 189–210 (2008)
2. Agarwal, S., Snavely, N., Simon, I., Seitz, S., Szeliski, R.: Building rome in a day. In: IEEE 12th International Conference on Computer Vision, pp. 72–79 (2009)
3. Snavely, N., Seitz, S., Szeliski, R.: Skeletal graphs for efficient structure from motion. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
4. Farenzena, M., Fusiello, A., Gherardi, R.: Structure-and-motion pipeline on a hierarchical cluster tree. In: IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops), pp. 1489–1496 (2009)
5. Crandall, D., Owens, A., Snavely, N., Huttenlocher, D.: Discrete-continuous optimization for large-scale structure from motion. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3001–3008 (2011)

6. Triggs, B., McLauchlan, P., Hartley, R., Fitzgibbon, A.: Bundle adjustment–a modern synthesis. In: Vision Algorithms: Theory and practice, pp. 153–177 (2000)
7. Jeong, Y., Nister, D., Steedly, D., Szeliski, R., Kweon, I.: Pushing the envelope of modern methods for bundle adjustment. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1474–1481 (2010)
8. Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., Sayd, P.: Generic and real-time structure from motion using local bundle adjustment. Image and Vision Computing 27, 1178–1193 (2009)
9. Lourakis, M., Argyros, A.: Is levenberg-marquardt the most efficient optimization algorithm for implementing bundle adjustment? In: IEEE 10th International Conference on Computer Vision, pp. 1526–1531 (2005)
10. Ni, K., Steedly, D., Dellaert, F.: Out-of-core bundle adjustment for large-scale 3d reconstruction. In: IEEE 11th International Conference on Computer Vision, pp. 1–8 (2007)
11. Jian, Y., Balcan, D., Dellaert, F.: Generalized subgraph preconditioners for large-scale bundle adjustment. In: IEEE 13th International Conference on Computer Vision, pp. 1–8 (2011)
12. Shum, H., Ke, Q., Zhang, Z.: Efficient bundle adjustment with virtual key frames: A hierarchical approach to multi-frame structure from motion. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. II:538–II:543 (1999)
13. Steedly, D., Essa, I.: Propagation of innovative information in non-linear least-squares structure from motion. In: IEEE 8th International Conference on Computer Vision, pp. 223–229 (2001)
14. Steedly, D., Essa, I., Dellaert, F.: Spectral partitioning for structure from motion. In: IEEE 9th International Conference on Computer Vision, pp. 996–103 (2003)
15. Agarwal, S., Snavely, N., Seitz, S., Szeliski, R.: Bundle adjustment in the large. In: European Conference on Computer Vision, pp. 29–42 (2010)
16. Byröd, M., Åström, K.: Conjugate gradient bundle adjustment. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 114–127. Springer, Heidelberg (2010)
17. Wu, C., Agarwal, S., Curless, B., Seitz, S.: Multicore bundle adjustment. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3057–3064 (2011)
18. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 888–905 (2000)
19. Lourakis, M., Argyros, A.: Sba: A software package for generic sparse bundle adjustment. ACM Transactions on Mathematical Software (TOMS) 36, 1–30 (2009)
20. Nocedal, J., Wright, S.: Numerical optimization. Springer (2006)

Learning Feature Subspaces for Appearance-Based Bundle Adjustment

Chia-Ming Cheng[1] and Hwann-Tzong Chen[2]

[1] MediaTek Inc., Taiwan
[2] National Tsing Hua University, Taiwan

Abstract. We present an improved bundle adjustment method based on the online learned appearance subspaces of 3D points. Our method incorporates the additional information from the learned appearance models into bundle adjustment. Through the online learning of the appearance models, we are able to include more plausible observations of 2D features across diverse viewpoints. Bundle adjustment can benefit from such an increase in the number of observations. Our formulation uses the appearance information to impose additional constraints on the optimization. The detailed experiments with ground-truth data show that the proposed method is able to enhance the reliability of 2D correspondences, and more important, can improve the accuracy of camera motion estimation and the overall quality of 3D reconstruction.

1 Introduction

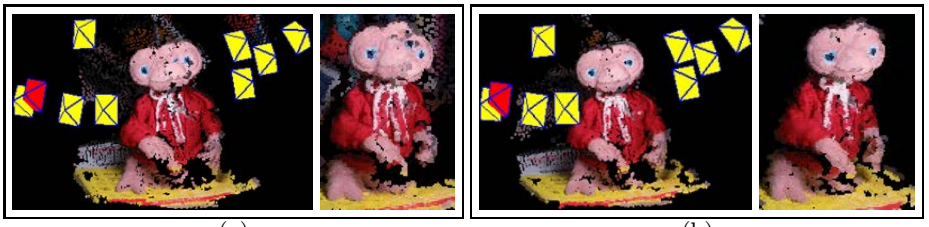
Recent structure from motion (SfM) systems such as [1,3,6,8,14] usually build on two key techniques: one is a distinctive-feature detector for image matching, e.g. [10,17], and the other is an optimization process based on bundle adjustment [15]. SIFT [10] is arguably the most popular feature-extraction method for image matching. It has been successfully used in 3D modeling systems [13,14] to extract local features for finding 2D correspondences to the same 3D point. The optimization process in an SfM system is usually based on bundle adjustment. For example, the handy SfM system *Bundler* [13,14] uses a modified version of sparse bundle adjustment package [9] to solve the joint optimization of camera parameters and 3D point positions. More efficient algorithms on solving bundle adjustment have also been continually developed [2,4]. The coupling of feature matching and bundle adjustment enables modern SfM systems like Bundler to model large-scale 3D structures from unordered image collections.

The sparse bundle adjustment used in Bundler requires good feature-matching results to provide reliable initial correspondences. However, local features across wide-baseline views and varied lighting conditions are not easy to be matched due to the nontrivial transformation of the feature's appearance. Havlena et al. [6] use a model-growing scheme to connect images and create new 3D points for the 3D model. More correspondences can thus be included in bundle adjustment. Our approach shares a similar notion of adding new views as [6], but we explore the use of online learning mechanisms in SfM. We seek to improve the matching

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 40–53, 2013.

quality by incorporating the online learned appearance models of 3D points into bundle adjustment. Various learning-based feature descriptors have been devised to improve image matching, e.g. [17]. Our goal is different in that we attempt to build feature representations for structure-from-motion rather than for general-purpose image matching. We incrementally update the appearance models of 3D points after each iteration of bundle adjustment, and use the appearance models to formulate a more robust bundle adjustment process.

Based on the online learning scheme for the appearance models of 3D points, we present the *appearance-based bundle adjustment* to solve the SfM problem. A feature subspace is associated with each 3D point as the appearance model, and the subspace is incrementally updated when new observations are available after each iteration of bundle adjustment. Local features in a new view are directly compared with the appearance model of each 3D point to find correspondences. Through the online learning of the appearance models, we are able to include more plausible observations of 2D features across diverse viewpoints. The experiments show that our approach is effective in improving both the visibility rates and the track lengths of correctly matched features. The appearance-based bundle adjustment is preferable to the point-based bundle adjustment in terms of the formulation of optimization problems. Relying on merely the positions of 2D points to evaluate the reprojection error might either lead to wrong estimations or make lots of points be removed as outliers. Our formulation can use the appearance information to avoid being trapped in poor local minima. Fig. 1 shows an example of using the appearance-based bundle adjustment to obtain a more consistent structure. In the experiments shown in Section 4, we use ground-truth data to show that our approach can enhance the reliability of the reconstructed 3D points, and as a result, can improve the accuracy of camera motion estimation and the overall quality of 3D reconstruction.



(a) (b)

Fig. 1. (a) The PMVS [5] reconstruction based on the result generated by a standard SfM pipeline with sparse bundle adjustment. Although the sparse bundle adjustment yields a small reprojection error, the inconsistency in the reconstructed structure is noticeable at the middle part, corresponding to the boundaries between the two clusters of views. (b) The PMVS output of our approach. Combining the geometry and the appearance helps to resolve the problem caused by insufficient matchings between the two clusters of views.

2 Learning the Subspace Representations of Local Features

In SfM, bundle adjustment is performed according to the initial pose estimation and the correspondences found by image matching. During bundle adjustment, dubious correspondences might be excluded from the optimization as outliers. A camera view that does not contain enough inlier corresponding points might thus be removed and does not contribute to the reconstruction. When more views are added into bundle adjustment, the increasing amount of information may help to identify correct matchings. Our approach to adding new views is to take account of the new information derived from the results of previous iterations of bundle adjustment. We explore the new view to find feature points that can actually fit the scene structure. To enable such an adaptive mechanism for finding 2D correspondences, we propose to learn the subspace representations for image features. The proposed subspace representations can be plugged in the appearance-based bundle adjustment optimization, which will be described in the next section.

The subspace representations are expected to model the variations of local features exhibited in former observations. We start by using SIFT to detect keypoints and extract local features. Instead of modeling 2D features image by image, we build a feature subspace associated with each 3D point. The detected local features in a new view are compared with the existing subspaces to find correspondences. The subspace representations are equipped with an incremental update scheme, such that, after bundle adjustment, local features can be used to update the subspaces.

We choose to use the $\mathcal{L}_\infty$ subspace described in [7] as the appearance model. The $\mathcal{L}_\infty$ subspace is originally presented for visual tracking. It has been shown that the $\mathcal{L}_\infty$ subspace outperforms the $\mathcal{L}_2$ (PCA-like) subspace in tracking objects under lighting changes and geometric transformations. The computation is also easier for $\mathcal{L}_\infty$ subspace since, unlike $\mathcal{L}_2$ subspace, no eigen-decomposition is involved.

Consider a set of SIFT feature vectors $\{v_1, \ldots, v_k\}$ associated with a 3D point. Our goal is to learn a subspace L that minimizes an error function given by

$$\text{Error}^\infty\left(L, \{v_1, \ldots, v_k\}\right) = \max_{t \in \{1,\ldots,k\}} d(L, v_t)\,, \tag{1}$$

where the function $d(\cdot, \cdot)$ measures the distance from a vector to a subspace in a least-squares sense. A subspace spanned by the entire observations of SIFT feature vectors $\{v_1, \ldots, v_k\}$ should minimize the above error function. We can find one of the subspaces that approximate to the span of $\{v_1, \ldots, v_k\}$ by applying the Gram-Schmidt process to $\{v_1, \ldots, v_k\}$, and an orthonormal basis can be obtained to represent the subspace.

The dimension of $\mathcal{L}_\infty$ subspace spanned by $\{v_1, \ldots, v_k\}$ will grow as the number k of data increases. To enable the subspace to be updated under a bounded dimension, we use a local-means method similar to the ones proposed in [12]. We keep at most s local means $\{z_1, \ldots, z_s\}$ to form the subspace. For each

3D point we learn its $\mathcal{L}_\infty$ subspace using the local means $\{z_1, \ldots, z_s\}$ rather than $\{v_1, \ldots, v_k\}$. The Gram-Schmidt process is applied to the local means $\{z_1, \ldots, z_s\}$ and yields an orthonormal basis Q for the $\mathcal{L}_\infty$ subspace. The local means are incrementally updated through the observations of $\{v_1, \ldots, v_k\}$.

In our SfM method, the orthogonal bases $\{Q_j\}$ of the learned $\mathcal{L}_\infty$ subspaces are used as the appearance models for 3D points $\{X_j\}$. Each 3D point X_j has an associated orthonormal basis Q_j. Given a detected 2D point in a new view i for camera C_i, we may find its most possible corresponding 3D point by projecting its SIFT feature vector onto the appearance subspace of each 3D point. We search for the subspace spanned by basis Q_{j^*} that has the minimum squared Euclidean distance from the SIFT feature vector to its orthogonal projection on the subspace. That 2D point is thus denoted as a 2D correspondence u_{ij^*} of the 3D point X_{j^*} in view i.

The SIFT feature vector of the 2D point is then used to update the corresponding basis Q_{j^*}. We add the SIFT vector into the closest local mean to update the set of local means. If the maximum number s of local means is not achieved and the distance from the SIFT vector to the closest mean is larger than a threshold, we create a new mean and add it into the set of local means. The updated set of local means is then used to generate a new orthonormal basis of the subspace by applying the Gram-Schmidt process. The Gram-Schmidt process is efficient. In our case we choose $s = 10$ and find that the overhead of recomputing Gram-Schmidt is negligible.

3 Appearance-Based Bundle Adjustment

Bundle adjustment is formulated as a process of simultaneously refining 'the sparse 3D points of the scene structure' and 'the parameters of cameras capturing the images'. The underlying optimization problem often involves minimizing the reprojection error of 3D points according to their 2D correspondences across images. Assume that we have m cameras $\mathbf{C} = (C_1, \ldots, C_m)$ observing n points $\mathbf{X} = (X_1, \ldots, X_n)$ in 3D space. An observation of 2D point is denoted by u_{ij}, which is derived from the observation model $f(C_i, X_j)$ that yields the 2D image coordinates of the 3D point X_j projected into the view of camera C_i plus some unknown noise. The visibility of point X_j in the view of camera C_i is indicated by an index set $\mathcal{I}$, such that $(i, j) \in \mathcal{I}$ if and only if point X_j is observed in the ith image.

We present an appearance-based formulation of bundle adjustment in which the learned appearance subspaces of 3D points can be used to provide additional evidence for the measurement of the reprojection error. Instead of estimating the parameters $\{\mathbf{C}, \mathbf{X}\}$ through minimizing the reprojection error of 3D points, we incorporate the appearance into the optimization problem defined by

$$\{\mathbf{C}^*, \mathbf{X}^*\} = \arg\min_{\mathbf{C},\mathbf{X}} \sum_{(i,j)\in\mathcal{I}} \phi_{ij} \left\| f(C_i, X_j) - u_{ij} \right\|^2, \tag{2}$$

where we multiply the reprojection error $\|f(C_i, X_j) - u_{ij}\|^2$ by an appearance weight ϕ_{ij}. For a camera C_i that has been considered in previous bundle adjustment iterations, the appearance weight ϕ_{ij} is defined by

$$\phi_{ij} = \exp\left\{-\frac{d(Q_j, v_{ij})^2}{2\sigma_a^2}\right\}, \tag{3}$$

where v_{ij} is the SIFT feature vector for the unknown 2D correspondence u_{ij} of X_j in view i, and $d(Q_j, v_{ij})$ is the distance from v_{ij} to its matched appearance subspace spanned by basis Q_j.

On the other hand, for a new camera view i', we select the 2D correspondence $\hat{u}_{i'j}$ whose feature $\hat{v}_{i'j}$ best fits the subspace Q_j, that is, yields the smallest value $d(Q_j, \hat{v}_{i'j})$ among the candidates within a radius r from the initial reprojection coordinates $f(\bar{C}_{i'}, \bar{X}_j)$ before the current iteration of bundle adjustment, where $\bar{C}_{i'}$ and $\bar{X}_j$ are previous estimations. The new view is then associated with an appearance weight

$$\phi_{i'j} = \exp\left\{-\frac{d(Q_j, \hat{v}_{i'j})^2}{2\sigma_a^2} - \frac{\|\hat{u}_{i'j} - f(\bar{C}_{i'}, \bar{X}_j)\|^2}{2\sigma_s^2}\right\}, \tag{4}$$

where we lessen the weight according to how far $\hat{u}_{i'j}$ diverges from the initial reprojection coordinates. We set $\sigma_s = 0.4r$ as a spatial scale factor based on the radius r. In our experiments we set $r = 5.0$, $\sigma_s = 2.0$ and $\sigma_a = 0.6$. Note that we use the factor of re-projection error in (4) because we would like to introduce a soft decision boundary for the inclusion of $\hat{u}_{i'j}$. If we use only the factor of $d(Q_j, \hat{v}_{i'j})$ in (4), we actually adopt a hard boundary to decide whether we should include $\hat{u}_{i'j}$. Such a hard decision boundary would be more sensitive to the parameter setting for the search radius r.

The optimization can be expressed in matrix form:

$$\{\mathbf{C}^*, \mathbf{X}^*\} = \arg\min_{\mathbf{C},\mathbf{X}} \left\| \boldsymbol{\Phi} \left(f(\mathbf{C}, \mathbf{X}) - \hat{\mathbf{U}} \right) \right\|^2, \tag{5}$$

where $\|\cdot\|$ is the Frobenius norm, $\boldsymbol{\Phi}$ contains the appearance weights in the corresponding matrix elements, and $\hat{\mathbf{U}}$ consists of the 2D correspondences. Let $\mathbf{J} = [\partial f/\partial \mathbf{C}\ \partial f/\partial \mathbf{X}]^\mathsf{T}$. By the first order Taylor approximation we may write the solution as

$$\begin{bmatrix} \Delta\mathbf{C} \\ \Delta\mathbf{X} \end{bmatrix} = (\mathbf{J}^\mathsf{T}\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{\Phi}\mathbf{J})^{-1}\mathbf{J}^\mathsf{T}\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{\Phi}\left(\hat{\mathbf{U}} - f(\bar{\mathbf{C}}, \bar{\mathbf{X}})\right). \tag{6}$$

3.1 Comparison with the Original Bundle Adjustment [15]

Although we formulate the optimization in a form of weighted least squares as in [15] so that stable numerical solutions can be more easily obtained, the notion of our formulation is quite different from [15], where the weight matrix is just an inverse covariance matrix modeling the uncertainty. Our formulation includes

the additional information provided by the learned appearance models, and we perform the optimization and learning in an EM-like manner that is embedded in the iterations of bundle adjustment. At each iteration of bundle adjustment we search among the candidate appearance models to associate individual 2D points in the new view with the 3D points. After an iteration of bundle adjustment, we can update the appearance models using the current results of 2D to 3D correspondences.

Our appearance-based formulation is also different from the intensity-based model which solves for the transformations between image patches, as is mentioned in [15]. For the problem of SfM, the transformations between image patches on surfaces are not fully dependent on the parameters of the camera poses and the scene structures of interest. To include extra parameters of patch transformations might burden the optimization rather than alleviate the adjustment computation. Our formulation does not include the extra parameters but make use of the appearance information to avoid infeasible solutions found by point-based bundle adjustment.

4 Experiments

In the first part of the experiments, we evaluate the performance of learning the subspace representations for local features. We show that the proposed learning method can be applied to large datasets and can achieve very good precision-recall rates, significantly better than the baseline strategy of descriptor averaging. Our learning method performs comparably well as the direct matching strategy (nearest-neighbor criterion), in which all descriptors are kept for matching without any learning. However, our learning method is much more efficient than the direct matching, especially for large datasets.

In the second part of the experiments, we evaluate the structure-from-motion results using the appearance-based bundle adjustment. We use three datasets that provide calibrated cameras and ground-truth correspondences for evaluation. Our method shows the advantages of increasing the track length and the number of observations per view. More important, the accuracy of camera motion estimation and 3D reconstruction is also improved, in comparison with the point-based sparse bundle adjustment.

4.1 Evaluation of Learning Subspace Representations

We use the datasets provided by Winder and Brown in [17] to evaluate the effectiveness of learning the subspace representations. The image data are taken from *photo tourism* [13] reconstructions of Trevi Fountain, Notre Dame, and Half Dome. Each dataset consists of $100,000$ grayscale patches, which are obtained by projecting 3D points from *photo tourism* reconstructions back into the original images. Due to the mechanism of deciding the scales and orientations of the 2D projected points, many of the correspondences identified in the datasets may not have been matched using SIFT descriptors. The patches might also have some local occlusion due to parallax.

For each dataset, we select the 3D points that have at least twelve 2D correspondences (twelve corresponding patches), since we would like to see how effectively the subspace representations can perform for modeling longer tracks of matched 2D correspondences. As a result, the number of selected 3D points is 852, 515, and 1,071 for Trevi Fountain, Notre Dame, and Half Dome. Totally there are 15,267, 8,164, and 17,050 patches selected from the three datasets. The average number of patches of a selected 3D point for Trevi Fountain, Notre Dame, and Half Dome is 18, 16, and 16, respectively, and the histograms regarding the number of patches of selected 3D points are shown in Fig. 2. Some of the selected 3D points may have more than 30 corresponding patches. We separate the patches of each dataset into a training set and a test set, with a ratio of 4 : 1. The size of a patch is 64×64 pixels. We extract the SIFT descriptor from each patch for subspace learning.

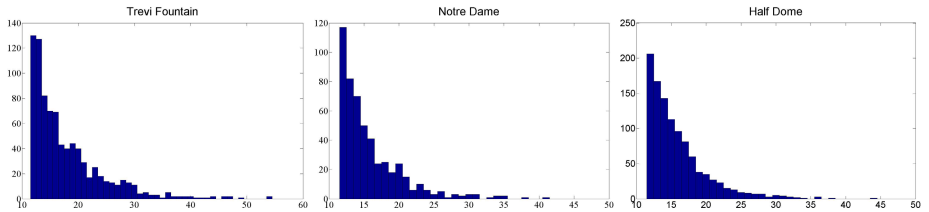


Fig. 2. The histogram of the number of patches corresponding to the selected 3D points (≥ 12 patches) for (a) Trevi Fountain, (b) Notre Dame, and (c) Half Dome datasets. Some of the selected 3D points may have more than 30 corresponding patches.

Precision-Recall. We apply the proposed learning method to each of the three training sets and build the feature subspaces for the corresponding 3D points. The maximum number s of local means is 10, as described in Section 2. For the test data, the correspondences to the 3D points are decided by finding the closest subspaces. We can verify the ground-truth correspondences to evaluate the quality of matching results. If we set a threshold for the distance between a test feature and its closest subspace, we may remove some incorrect correspondences. By modulating the threshold value, we can derive a precision-recall curve. Precision is the number of 'true positives' divided by the sum of 'true positives' and 'false positives'; recall is the number of 'true positives' divided by the sum of 'true positives' and 'false negatives'. If we set a larger threshold value, then the recall rate will be higher but the precision might decrease. The precision-recall curves for the three test sets are shown in Fig. 3. The subspace learning method is compared with two strategies: The first one is to average all the SIFT descriptors that belong to the same 3D point, and use the mean descriptor as the feature representation of the 3D point. To find the correspondence for a test descriptor, we measure the similarity between the test descriptor and each of the mean descriptors using the Euclidean distance. The second strategy is to keep all SIFT descriptors of the training data and use the nearest-neighbor criterion

to find 2D correspondence for the test descriptor, where the Euclidean distance is also used as the measurement for SIFT descriptors. As shown in Fig. 3, our subspace method can achieve comparable performances as the nearest-neighbor strategy. The averaging strategy does not perform very well because the mean descriptors might not be distinctive enough for large datasets.

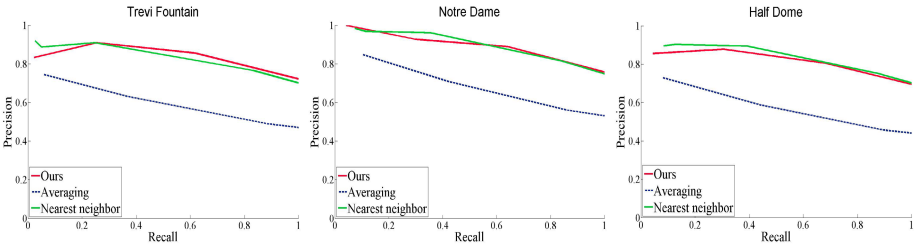


Fig. 3. The precision-recall curves for (a) Trevi Fountain, (b) Notre Dame, and (c) Half Dome datasets. Our subspace representations can achieve comparable performances as the nearest-neighbor strategy. The averaging strategy does not perform very well, probably because the mean descriptors are not distinctive enough for large datasets.

Timing. Learning the subspace representations using our method is very fast. For example, the subspaces for the 13,640 training descriptors of the Half Dome data can be learned in less than 2 seconds, in MATLAB on a PC with quad-core 2.8GHz CPU and 12GB memory. The training time for the averaging strategy is close to our method. The nearest-neighbor strategy does not require training, and only some overhead processing time is involved. Regarding the matching between the test data and the training data for finding correspondences, our method and the averaging strategy are faster. The nearest-neighbor strategy, as expected, is very slow. The timing results for matching are shown in Table 1.

Further Discussions. The evaluation shows that the learned appearance subspaces provide effective representations for finding correspondences to 3D points. By using the learned subspaces, we can have similar precision-recall rates without keeping all the descriptors of 2D features, and therefore greatly reduce the

Table 1. The timing results of feature matching using different strategies

	# of test patches	# of training patches	Timing for matching		
			Nearest neighbor	Averaging	Subspace
Trevi Fountain	3,053	12,214	729s	46s	51s
Notre Dame	1,632	6,532	252s	17s	18s
Half Dome	3,410	13,640	989s	65s	71s

time required for matching. Since we set the maximum number s of local means to be 10, the dimension of a learned subspace is at most 10. We find that the average dimension of the learned subspaces is 9, 8, and 8 for Trevi Fountain, Notre Dame, and Half Dome. The distributions of the subspace dimensions are shown in Fig. 4. We may choose a larger value of s to allow higher dimensional subspaces to be built, particularly when the dataset is very large, but the training and matching time might also increase. The trade-off of descriptiveness and efficiency would be dependent on the data. For a dataset with a scale about 1,000 3D points and 15,000 2D features, our current setting seems suitable.

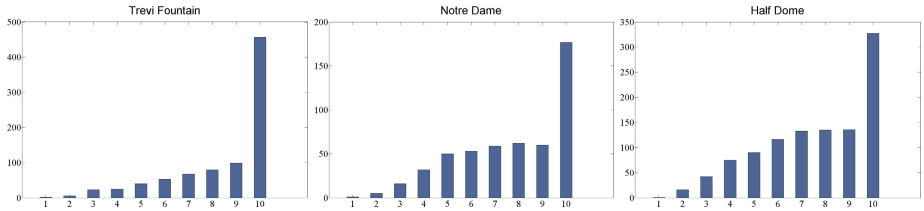
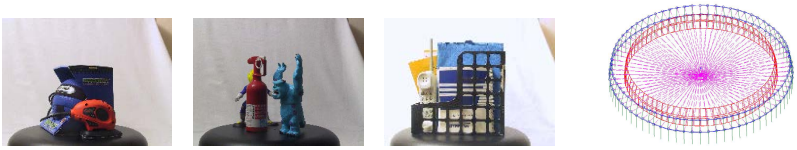


Fig. 4. The histogram of subspace dimensions for (a) Trevi Fountain, (b) Notre Dame, and (c) Half Dome datasets

4.2 Evaluation of Appearance-Based Bundle Adjustment Using Ground-Truth Data

We use the datasets created by Moreels and Perona [11] to evaluate the performance of the appearance-based bundle adjustment. The images in the datasets are captured by a calibrated stereo system with a turntable. The advantage of using these datasets is that we are able to verify the correctness of correspondences based on the ground-truth geometric constraints. We choose three of the datasets, BallSander, Standing, and StorageBin, as shown in Figs. 5a– 5c. The 'ground-truth' camera poses are shown in Fig. 5d. The world center is set at $(0, 0, 0)$, and the average distance between each camera and the world center is 1.0. The proposed appearance-based bundle adjustment is compared with the sparse bundle adjustment in respect of several evaluation metrics which we will describe later in this section. For fair comparison, the numbers of initial 2D features extracted by SIFT are the same for both methods.

Evaluation Metrics. We focus on the comparisons between the point-based sparse bundle adjustment [9] and our online-learned appearance-based bundle adjustment. The pipeline of incremental SfM is not taken into consideration for the evaluation. Several metrics are used to evaluate the performances: *i)* the visibility rate, *ii)* the outlier rate, *iii)* the false 3D-point rate, *iv)* the camera motion estimation error (the average rotation and translation errors), and *v)* the average 3D reconstruction error.



(a) (b) (c) (d)

Fig. 5. Three of the datasets created by Moreels and Perona [11]: (a) the BallSander dataset, (b) the Standing dataset, and (c) the StorageBin dataset. (d) The camera poses for those datasets are derived from the calibrated stereo system with a turntable. We set the world center at $(0, 0, 0)$, and the average distance between each camera and the world center is 1.0. The evaluations of the 3D errors are based on the scale after this normalization.

The visibility rate is computed by (# of observations)/(# of views $\times$ # of 3D points). By 'outlier' we mean that a 2D feature within a track does not satisfy the ground-truth geometry constraint. The outlier rate is defined by (# of outliers)/(# of observations). Furthermore, we can use the ground-truth geometry constraints to verify the correctness of a reconstructed 3D point. We compute the false 3D-point rate by (# of false 3D points)/(# of all reconstructed 3D points).

Incorrect matching results would induce outliers into the minimization of the reprojection error. Outliers might bias the solution due to overemphasizing the errors. Equipping the point-based bundle adjustment with an outlier-removal mechanism might increase the robustness, but would also make bundle adjustment prone to be trapped in trivial local minima. Ideally, the reprojection error should be minimized under the assumption that all 3D points can be observed in all views. A higher visibility rate and a lower outlier rate are preferable in a sense that they imply the ideal case of the original objective of bundle adjustment.

To further evaluate the quality of camera motion estimation and 3D reconstruction, we use the ground-truth camera poses and geometry constraints derived from the datasets of Moreels and Perona. As mentioned earlier, we measure the errors of camera motion estimation and 3D reconstruction based on a normalized scale: the average distance between each camera and the world center $(0, 0, 0)$ is 1.0. The quality of camera motion estimation is evaluated by the translation error and the rotation error of camera pose. We align all of the estimated camera poses to the normalized ground-truth coordinates shown in Fig. 5d. The translation error is computed as the distance between the estimated camera center and the ground-truth camera center. The rotation error is measured by the geometric mean of the Euler angles of $\mathbf{R}_{\text{est}}\mathbf{R}_{\text{gt}}^{\mathsf{T}}$, where $\mathbf{R}_{\text{est}}$ is an estimated rotation matrix and $\mathbf{R}_{\text{gt}}$ is the ground-truth rotation matrix. To compute the 3D reconstruction error, we exclude the false 3D points from the reconstructed 3D points. We then aligned the reconstructed 3D structure with the ground-truth structure by applying absolute pose estimation [16]. The average 3D reconstruction error is measured by the average distance from each aligned 3D point to its corresponding ground-truth 3D point.

Results. We summarize all of the evaluation results in Tables 2, 3, & 4. The results show that the appearance-based bundle adjustment achieves better performance than the point-based sparse bundle adjustment on all of the evaluation metrics. The average track length and the visibility rate of 2D features both significantly increase. The improved outlier rate means that the appearance-based bundle adjustment is capable of removing more incorrect correspondences. The appearance-based bundle adjustment can also achieve a very low false 3D-point rate, which means that its reconstruction of 3D points is quite reliable. Most important, the appearance-based bundle adjustment indeed improves the accuracy and quality of camera motion estimation and 3D structure reconstruction, as explicitly shown in the evaluation results.

Table 2. Evaluations with the BallSander dataset. We compare the proposed appearance-based bundle adjustment with the sparse bundle adjustment (SBA).

	SBA	Appearance-based
# of 3D points	943	494
average track length	4.11	9.87
visibility rate (%)	10.81	25.97
outlier rate (%)	1.29	0.72
false 3D-point rate (%)	1.70	0.20
average camera rotation error	2.061	1.793
average camera translation error	0.0073	0.0070
average 3D reconstruction error	0.0074	0.0059

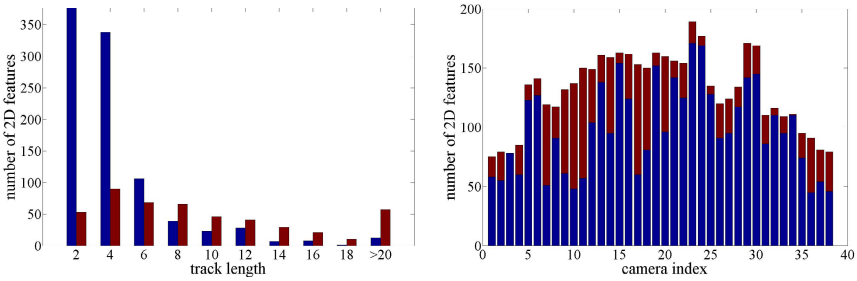
Table 3. Evaluations with the Standing dataset. We compare the proposed appearance-based bundle adjustment with the sparse bundle adjustment (SBA).

	SBA	Appearance-based
# of 3D points	1,226	621
average track length	4.86	12.50
visibility rate (%)	12.15	31.25
outlier rate (%)	1.16	0.98
false 3D-point rate (%)	1.47	0.00
average camera rotation error	1.603	1.402
average camera translation error	0.0065	0.0059
average 3D reconstruction error	0.0056	0.0055

Further Discussions. After learning the subspaces and applying the learned representations to the appearance-based bundle adjustment, we can find more 2D features that can be modeled by the learned subspaces. From the results shown in Figs. 6, 7, & 8, we observe that the online learned appearance representations can help to increase the track length as well as the number of registered 2D

Table 4. Evaluations with the StorageBin dataset. We compare the proposed appearance-based bundle adjustment with the sparse bundle adjustment (SBA).

	SBA	Appearance-based
# of 3D points	1,741	697
average track length	3.82	10.85
visibility rate (%)	8.88	25.22
outlier rate (%)	5.67	1.48
false 3D-point rate (%)	6.03	0.01
average camera rotation error	1.923	1.646
average camera translation error	0.0100	0.0076
average 3D reconstruction error	0.0108	0.0074



(a) (b)

Fig. 6. The BallSander dataset. (a) The distribution of the track length. (b) The number of registered 2D points in each view. Blue bars: before subspace learning. Red bars: after subspace learning.

features in each view. These newly-included 2D correspondences will contribute to solving the 3D points in later iterations. Overall, the integrated mechanism of subspace learning and appearance-based bundle adjustment provides a plausible way of computing structure and motion.

Although the reliability of the 3D points is enhanced, a limitation of our approach is that it would merge short tracks into longer ones, and as a result, the number of reconstructed 3D points might greatly decrease. The number of 3D points reconstructed by our approach is about half of the number of 3D points obtained by the point-based sparse bundle adjustment, as can be observed in Tables 2, 3, & 4. This is a trade-off between ensuring a more consistent structure and reconstructing as more 3D points as possible.

About the time complexity, the additional computational cost of the appearance based bundle adjustment is due to the computation of the appearance-weight matrix, of which the size is the number of views times the number of 3D points. We also need to compute the appearance weights and multiply the appearance-weight matrix by the Jacobian matrix, but the computation of Jacobian matrix is efficient owing to the the longer tracks and the reduced number of

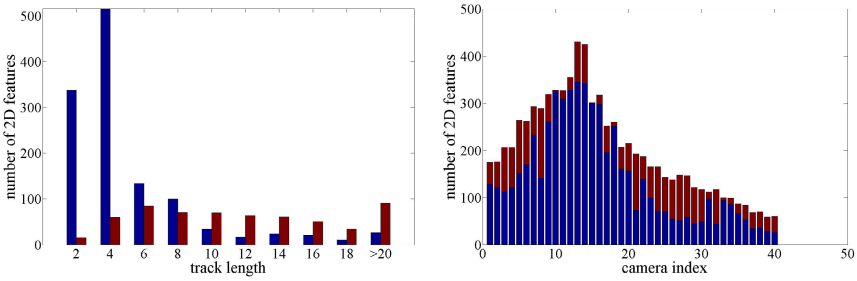


Fig. 7. The Standing dataset. (a) The distribution of the track length. (b) The number of registered 2D points in each view. Blue bars: before subspace learning. Red bars: after subspace learning.

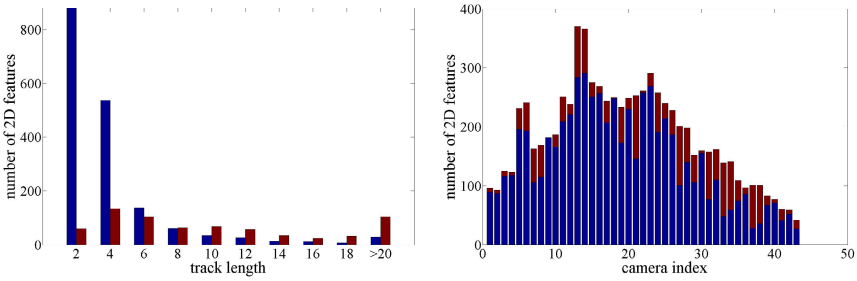


Fig. 8. The StorageBin dataset. (a) The distribution of the track length. (b) The number of registered 2D points in each view. Blue bars: before subspace learning. Red bars: after subspace learning.

redundant points. In practice the computation time of solving the appearance-based bundle adjustment is close to solving the sparse bundle adjustment if the optimization involves similar numbers of views and 3D points.

5 Conclusion

We have presented a new bundle adjustment method based on an online-learned appearance model associated with each 3D point. The proposed appearance-based bundle adjustment is able to include more 2D observations into the optimization. As shown in our experiments, the lengths of most tracks in conventional sparse bundle adjustment are usually quite small. The appearance-based bundle adjustment is able to achieve a significant increase in the number of long tracks and the number of correctly matched features. The visibility rates of 2D correspondences and the outlier rates are greatly improved by appearance-based bundle adjustment. Through the detailed evaluations on the ground-truth

datasets, we show that our method can improve the accuracy of camera motion estimation and the quality of 3D reconstruction, in comparison with the point-based sparse bundle adjustment.

References

1. Agarwal, S., Furukawa, Y., Snavely, N., Curless, B., Seitz, S.M., Szeliski, R.: Reconstructing rome. IEEE Computer 43(6), 40–47 (2010)
2. Agarwal, S., Snavely, N., Seitz, S.M., Szeliski, R.: Bundle Adjustment in the Large. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 29–42. Springer, Heidelberg (2010)
3. Agarwal, S., Snavely, N., Simon, I., Seitz, S.M., Szeliski, R.: Building rome in a day. In: ICCV (2009)
4. Byröd, M., Åström, K.: Conjugate Gradient Bundle Adjustment. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 114–127. Springer, Heidelberg (2010)
5. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multi-view stereopsis. In: CVPR (2007)
6. Havlena, M., Torii, A., Pajdla, T.: Efficient Structure from Motion by Graph Optimization. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 100–113. Springer, Heidelberg (2010)
7. Ho, J., Lee, K.-C., Yang, M.-H., Kriegman, D.J.: Visual tracking using learned linear subspaces. In: CVPR (1), pp. 782–789 (2004)
8. Li, X., Wu, C., Zach, C., Lazebnik, S., Frahm, J.-M.: Modeling and Recognition of Landmark Image Collections Using Iconic Scene Graphs. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 427–440. Springer, Heidelberg (2008)
9. Lourakis, M.I.A., Argyros, A.A.: Sba: A software package for generic sparse bundle adjustment. ACM Trans. Math. Softw. 36(1) (2009)
10. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
11. Moreels, P., Perona, P.: Evaluation of features detectors and descriptors based on 3d objects. International Journal of Computer Vision 73(3), 263–284 (2007)
12. Park, H., Jeon, M., Rosen, J.B.: Lower dimensional representation of text data based on centroids and least squares. BIT 43 (2003)
13. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. ACM Trans. Graph. 25(3), 835–846 (2006)
14. Snavely, N., Seitz, S.M., Szeliski, R.: Modeling the world from internet photo collections. International Journal of Computer Vision 80(2), 189–210 (2008)
15. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment - a modern synthesis. In: Workshop on Vision Algorithms, pp. 298–372 (1999)
16. Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns. IEEE Trans. Pattern Anal. Mach. Intell. 13(4), 376–380 (1991)
17. Winder, S.A.J., Brown, M.: Learning local image descriptors. In: CVPR (2007)

Toward Efficient Acquisition of BRDFs with Fewer Samples

Muhammad Asad Ali[1], Imari Sato[2], Takahiro Okabe[1], and Yoichi Sato[1]

[1] Institute of Industrial Science, The University of Tokyo, Japan
[2] National Institute of Informatics, Japan
{asad,takahiro,ysato}@iis.u-tokyo.ac.jp, imarik@nii.ac.jp

Abstract. In this paper we propose a novel method for measuring reflectance of isotropic materials efficiently by carefully choosing a set of sampling directions which yields less modeling error. The analysis is based on the empirical observation that most isotropic BRDFs can be approximated using 2D bivariate representation. Further a compact representation in the form of basis is computed for a large database of densely measured materials. Using these basis and an iterative optimization process, an appropriate set of sampling directions necessary for acquiring reflectance of new materials are selected. Finally, the measured data using selected sampling directions is projected onto the compact basis to obtain weighting factors for linearly representing new material as a combination of basis of several previously measured materials. This compact representation with an appropriate BRDF parameterization allows us to significantly reduce the time and effort required for making new reflectance measurements of any isotropic material. Experimental results obtained using few sampling directions on the MERL dataset show comparative performance to an exhaustively captured set of BRDFs.

1 Introduction

Materials can be classified based on their optical properties as they modulate light differently depending upon the nature of surface. These properties provide us with a variety of clues about how a particular material will appear under different illumination conditions. Physically as well as computationally the optical properties of materials are effectively represented using a Bidirectional Reflectance Distribution Function (BRDF)[1].

Typically BRDF helps us characterize scene radiance, more formally it is a function of four variables $f(\theta_i, \phi_i, \theta_o, \phi_o)$, where θ_i, ϕ_i are polar and azimuthal angles of the incident light direction and θ_o, ϕ_o of the reflected direction respectively. It tells us how bright a surface patch will appear when viewed from one direction while light falls from another. There are several advantages of measuring the optical properties of materials in the form of 4D BRDF as it can be used for photo realistic rendering, preservation of historical heritage, analysis of remote sensing data, movie production and in computer vision it is often used for material and object recognition tasks. Moreover measured BRDF data can be helpful for the development and validation of analytic BRDF models.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 54–67, 2013.

This work focuses on an important sub-class of BRDFs called isotropic BRDFs for which rotations about the surface normal does not need to be considered. This generalization reduces the BRDF from a function of four variables to three $f(\theta_i, \theta_o, \phi_i - \phi_o)$. Even with this generalization uniform sampling still requires a huge amount of measurements i.e. suppose with an angular spacing of ψ the number of measurements necessary would be approximately $\pi^3/(4\psi^3)$[2].

Many researchers have attempted to make the traditional measurement process more efficient by proposing solutions which attempt to measure many different samples at once by using mirrors [6][7] or use spherical samples of the materials [10] which requires the material to be homogeneous. However optical elements usually do not allow measuring reflectance at near grazing angles and can be a source of indirect illumination resulting in incorrect measurements [3].

To overcome some of these issues, we propose a reflectance measurement procedure that significantly reduces the number of necessary measurements by carefully selecting an optimized set of few sampling directions using compact basis in this paper. This is achieved by using the observation that most isotropic BRDFs can be approximately represented by 2D bivariate form and further the variations in the data can be minimized by representing it in the form of basis. This appropriate representation significantly reduces the number of unknowns in the linear system which directly influence the reduction in number of necessary measurements for acquiring BRDFs of isotropic materials. Obtained results using the proposed method demonstrate that by using such an approach a new material can be acquired using 100 or fewer measurements with a fair amount of accuracy.

The proposed method explicitly differs from [3] as it uses 2D bivariate approximation [5] for isotropic BRDFs and further compression using compact basis. Also few sampling directions are selected robustly using basis representation by performing iterative optimization in a dimensionally reduced space which is significantly fast compared to an exhaustive search over all samples. We are motivated to use bivariate approximation as it reduces the dimensions of isotropic BRDF from three to two due to generalization of bilateral symmetry and the use of basis enables us to compactly capture variations present in broad category of materials which directly contribute towards our goal of reducing sampling directions.

2 Related Work

Ward [6] did the pioneering work by introducing the use of digital cameras as part of measurement setup. The key optical instruments of his device were a half silvered hemisphere and a camera with a fish eye lens. In his arrangement the light source and the sample holder are movable over all the incident angles and allows the measurement of anisotropic reflectance for a material sample.

However the first large collection of sparsely sampled BRDFs of 61 materials originated as part of the CUReT project by the work of Dana et al.[7]. Their system was able to measure spatially varying BRDF's also referred as Bidirectional

Texture Functions (BTF). They simultaneously measure the BTF and BRDF of the material at 200 different combinations of viewing and illuminations directions. Later in [8] they introduced a improved version of the BRDF/BTF measurement device allowing simultaneous measurements of multiple viewing directions which used curved mirrors to eliminate the need of hemispherical positioning of camera and illumination device.

Marschner et al.[10] developed an improved BRDF measurement system using two cameras, a light source, test sample of known shape and assume known geometry. Matusik et al.[3] based their BRDF measurement setup on the work of [10] for measuring reflectance of about 100 different materials. Marschner et al.[10] were not able to take into account the local spectral characteristics of BRDFs resulting in dense uniform sampling of the acquisition hemisphere. This was one of the main issue addressed in the work of Matusik [3] to significantly reduce the time and measurements necessary for acquiring BRDFs. They also analyzed the local signal variations in the BRDFs using wavelets and showed that good reconstruction can be performed using 69000 measurements by using wavelet basis. Further they went on to show that it was possible to represent reflectance of an arbitrary material as a linear combination of reflectance of several other material samples using linear representation. They showed that 800 sampling directions are enough to represent new BRDFs using this framework.

Mukaigawa et al.[11][12] developed a high speed method for BRDF measurement using ellipsoidal mirrior and projector arrangement without a mechnical drive for changing incident angles. They can measure reflectance of a material in about 50 minutes. However the accuracy of the measured BRDFs was not evaluated and the use of fixed sampling interval without taking into account characteristics of BRDFs results in increased measurements. Similarly Gosh et al.[13] described a fast method for acquiring the BRDF directly into basis representation which results in capturing reflectance in 1-2 minutes. However obtained results show that there is still significant need for improvement specially in the direction of what kind of illumination basis functions can be ideal for the task.

Other existing methods like Lawrence et al.[14] focus on interactive editing of materials and introduce the use of inverse shade trees for representing arbitrary BRDFs non-parametrically using weighted sum of small number of materials. Similarly, Sato et al.[15] focus on modeling object appearance analytically and show that a set of suitable lighting directions for sampling images can be determined based on objects BRDF.

3 Proposed Technique

We propose the use of 2D bivariate approximation for representing isotropic materials based on the empirical observation that such materials are bilaterally symmetrical and further show little change when the light and view directions are swapped about the half vector thus transforming the dimensions of the isotropic BRDF from three to two. Besides the variations present in a large database of such materials are robustly captured using basis and are used for efficiently

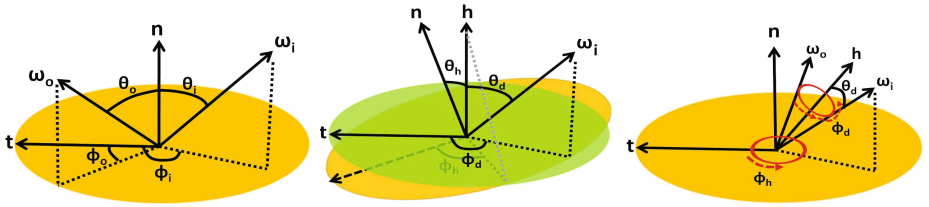


Fig. 1. Three parameterizations of the BRDF. (a) Original 4D BRDF. (b) Rusinkiewicz parameterization. (c) 2D Bivariate parameterization.

selecting an optimized set of few sampling directions in an iterative manner for acquiring new BRDFs of previously unknown materials. Captured materials using these 100 or fewer selected samples are then linearly represented using the compact basis for recovering complete BRDFs robustly.

3.1 Overview

In order to achieve the desired goal of reducing the number of necessary measurements, the BRDF is first transformed into an appropriate representation then all materials are arranged together in a matrix and dimensionality reduction is performed followed by the selection of suitable sampling directions using an iterative procedure. Each of these processes are explained in detail in a stepwise manner in the sections ahead followed by the detailed experimental evaluation of the proposed framework.

3.2 Data Representation

As this work deals with reducing the acquisition time of isotropic BRDFs of new materials so we opted to base our analysis on an already measured and well tested BRDF dataset of Mitsubishi Electric Research Lab (MERL)[4]. Acquired by Matusik [3][4] there are 100 materials in this dataset with BRDF measurements made for all three color channels i.e. Red, Green and Blue.

These measurements were made using Rusinkiewicz half vector parameterization [9] of the BRDF instead of the original 3D isotropic parameterization $f(\theta_i, \theta_o, \phi_i - \phi_o)$. They argued that the original representation requires dense angular sampling over the acquisition hemisphere to accurately measure the specular peaks otherwise resulting in poor highlight representation in the form of an ellipse depending upon the orientation of light source.

3.3 Rusinkiewicz BRDF Parameterization

Figure 1 shows the original as well as the half vector Rusinkiewicz parameterization [9] of BRDFs. In this parameterization four angles are used to describe the

BRDF namely: theta half (θ_h), theta difference (θ_d), phi difference (ϕ_d) where as phi half (ϕ_h) is not considered for isotropic BRDFs. The advantages gained by re-parameterizing the BRDF in this form are significant as the storage requirements are reduced allowing for fewer basis for robust representation besides important BRDF phenomenon such as specular and retro-reflective peaks are decoupled to be a function of one of the parameterized angles and only show weak dependence on a combination of axis.

In this new parameterization the range of θ_h, θ_d is $[0, \pi/2]$ and that of ϕ_d is $[0, \pi]$ due to reciprocity. The three angles $\theta_h, \theta_d, \phi_d$ are then further discretized to have 90, 90, 180 bins respectively. Thus for each color channel of a material sample we have a total of 90 x 90 x 180 = 1458000 BRDF measurements and for three color channels this amounts to a total of 1458000 x 3 = 4374000 BRDF measurements. Further the theta half angle θ_h is sampled more densely near the direction of specular reflection and the non-linear angle conversion can be approximated as $\theta_h = {\theta_{h\text{index}}}^2/(\pi/2)$, where $\theta_{h\text{index}}$ corresponds to the number of bins and varies from [0, $\pi/2$]. The mapping from angles to discretized bins remains linear for θ_d and ϕ_d.

3.4 Bivariate BRDF Representation

In order to further reduce the variations in isotropic BRDFs, its dimensions are constrained without surrendering the ability to represent important BRDF phenomenon. Such an approach called Bivariate representation was introduced by Romerio et al.[5]. It considers an additional projection of Rusinkiewicz BRDF representation that reduces the dimensions of an isotropic BRDF from three to two. This projection of isotropic BRDF on a 2D domain is acceptable as long as a BRDF shows little change for rotation of the input light(ψ_i) and output view(ψ_o) directions as a fixed pair about the half vector. For interpretation Figure 1(c) shows the Bivariate representation.

Practically the 2D Bivariate representation is a minimization of the original Rusinkiewicz representation with a summation defined over ϕ_d due to bilateral symmetry. The formula used for the computation of the bivariate BRDF parameterization is:

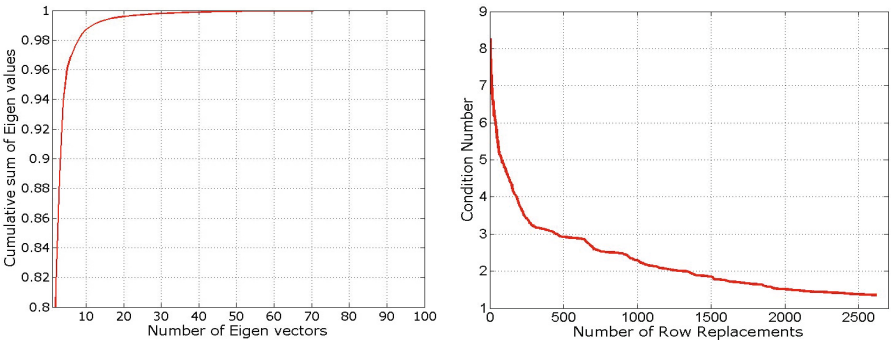
$$f(\theta_h, \theta_d) = \frac{1}{R} \sum_{\phi_d=0}^{\pi/2} f(\theta_h, \theta_d, \phi_d) \tag{1}$$

where $f(\theta_h, \theta_d, \phi_d)$ represents the 3D Rusinkeiwicz parameterized BRDF, R is the number of valid BRDF values in the interval $[0, \pi/2]$ for ϕ_d.

Dimensions of BRDF are significantly reduced in this representation as the two dimensions of BRDF now only comprise of θ_h and θ_d. Thus for each color channel of a material sample we have a total of 90 x 90 = 8100 measurements and for three color channels this amounts to a total of 8100 x 3 = 24300 measurements.

3.5 Data Organization

Next, we want to construct a matrix H of BRDF data for all materials. In order to prepare this data for processing later, it is necessary to arrange the BRDF



(a) (b)

Fig. 2. (a) Plot of cumulative sum of Eigen values for the 2D Bivariate data of matrix H. (b) Convergence plot demonstrating the decrease in condition number as row replacements are performed using the optimization process of section 3.8.

samples of all 100 materials in H such that the correspondence between the original acquisition angles in bivariate space is preserved. Thus the BRDF data of each material samples is arranged such that the red channel data goes first in column of matrix H followed by green channel data and then by blue channel data. Following this procedure BRDF data corresponding to all material samples is arranged in this matrix column-wise. The dimensions of matrix H after BRDF of all materials is arranged in it is M rows by N columns, where M is the number of sampling directions and N is the number of materials.

Normally BRDF values of specular and matt surfaces are scaled differently (high dynamic range). This results in large difference in magnitude of values among various types of materials. If these values are used with original scaling then future numerical analysis will associate more importance to noise in specular highlight as compared to non-specular components. To address this issue natural logarithm of all BRDF data values in matrix H is computed which significantly scales down the range of data for further analysis.

3.6 Dimensionality Reduction

The size of matrix H constructed above is still large and for acquiring BRDFs of new materials efficiently it would be beneficial if the dimensions of matrix H can be reduced by decorrelating various dependent components using multivariate analysis. To achieve this Principal Component Analysis (PCA) using covariance matrix of the form $H^T H$ is performed. After analyzing the reconstruction error using different number of basis vectors it is observed that PCA is able to capture the correlations among various BRDFs adequately. A plot of cumulative sum of Eigen values is shown in Figure 2(a) demonstrating significant reduction in dimensionality of the data.

3.7 Basis Projection Framework for Materials

Having performed dimensionality reduction we now set out to compute the projection of a newly acquired material sample on previously acquired BRDF data of several materials in matrix H. The objective here is to show that BRDF of a new material that is not part of H can be well represented by the linear combination of several other materials. What this means is that the BRDF of a new material is just a linear combination of BRDF of materials in H with a weighting factor only. These weighting factors are in fact the coefficients which need to be estimated as part of the projection. But before moving forward let us represent this in a form of linear equation:

$$H\boldsymbol{c} = \boldsymbol{b} \tag{2}$$

where H is the matrix of all BRDFs with dimensions M x N, $\boldsymbol{c}$ represents the coefficients vector which are to be estimated with dimension N and $\boldsymbol{b}$ corresponds to the BRDF of a new material which in this case must equal to M.

There are N material in H so at least N coefficients need to be estimated for each new material $\boldsymbol{b}$ by projecting it on H and then use the calculated coefficient $\boldsymbol{c}$ to reconstruct the new material sample as a linear combination of BRDFs of existing material using the linear generative model of equation (2).

However, having seen earlier in section 3.6 that fewer basis can capture majority of the variance in BRDFs of matrix H, so instead of using H for computing the linear projection, top K Eigen Basis V_K can be used to represent all the BRDFs. The linear equation with this change can be expressed as:

$$V_K\boldsymbol{c} + \boldsymbol{m} = \boldsymbol{b} \tag{3}$$

$$\boldsymbol{c} = (V_K^T V_K)^{-1} V_K^T (\boldsymbol{b} - \boldsymbol{m}) \tag{4}$$

where matrix V_K represents the top K Eigen basis of matrix H and $\boldsymbol{m}$ represents the mean vector of matrix H which must be subtracted from the new material measurements before computing its projection and then added back after the reconstruction step.

Moreover the above system of equations is highly over constrained. Suppose with $K = 35$, there are 35 unknown coefficients and the number of linear equations equals $M = 24300$ for each newly acquired BRDF of which majority are linearly dependent. This means that the number of necessary BRDF measurements can be significantly reduced for a material by selecting an appropriate sub-set from this large number of equations which can help us efficiently estimate the desired coefficients $\boldsymbol{c}$. If such a small subset of equations can be found which can represent a newly acquired BRDF as a weighted combination of BRDFs of several materials then any new material can be measured by using the combination of only a few light source and view directions corresponding to the selected set of equations (sampling directions / rows) in an efficient manner.

3.8 Selection of Suitable Directions for Acquisition

In order to estimate a subset of rows of Eigen Basis V_K, iterative optimization needs to be performed which attempts to reduce the condition number of the linear system of equations described above. The condition number is used here to find out how inaccurate the solution will be after an approximation using selected set of rows is obtained.

Normally for well conditioned matrices all the diagonal terms are of same order and for ordinary matrices the Eigen values will have the same order of magnitude as the diagonal terms of the original matrix. So the Eigen values will be close to diagonal terms for a diagonally dominated matrix. This means that the ratio of the highest to the smallest Eigen value should give a smaller number if the matrix is well-conditioned, since all the diagonal terms are of the same order. However if this ratio is large i.e. the order of difference among the diagonal terms is more, then the matrix is ill-conditioned. Now let us go into the details of this optimization process in a stepwise manner:

1. Select a subset of L rows from V_K randomly. Let us represent this row subset with matrix X.
2. Select one row from subset X and one row from outside of set X and swap them by inserting the row from outside into set X.
3. Then perform PCA on the covariance matrix $X^T X$ to obtain Eigen values.
4. Calculate the ratio between the highest and the lowest Eigen value (Max / Min) which approximates the condition number of the system.
5. If new condition number is less than the previous condition number then keep the newly inserted row in set X otherwise discard the new row and restore set X to its previous state.
6. This process is repeated iteratively from step 2 to step 5 until no more rows can be swapped for successive tests of all rows.
7. Repeat procedure from step 1 to step 6 several times and finally select the solution which has the lowest condition number among all obtained solutions.

The iterative procedure described above allows us to obtain an optimal solution over multiple runs and produces a stable set of rows in set X at the end of optimization which guarantees the system to be numerically well conditioned. Figure 2(b) plots the change in condition number with row replacements for a sample case. Sampling directions obtained in set X are based on the statistics of 2D BRDFs of different kinds of materials. These sampling directions are thus general and can be used for modeling various types of materials without the need for calculating them for each material.

The obtained set of equations can also be referred to as the most informative set and are selected irrespective of the red, green and blue channels. However selecting equations equal to the number of unknowns in our system may not generalize well over the set of known BRDFs so while performing row reduction we make sure to select the rows appropriately. Finally having selected a subset of equations the linear system can be updated to represent this fact as:

$$\boldsymbol{c} = (X_K^T X_K)^{-1} X_K^T (\boldsymbol{b_X} - \boldsymbol{m_X}) \tag{5}$$

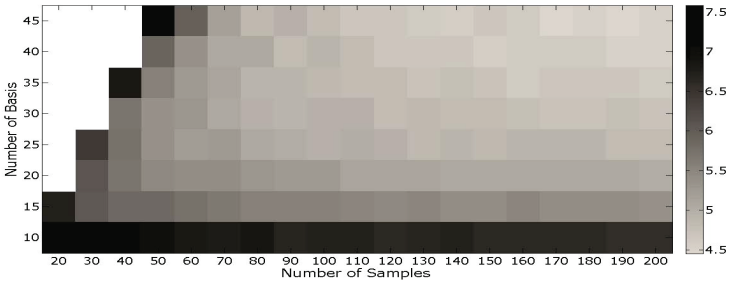


Fig. 3. Experimental results obtained using the proposed method with different combinations of basis and sampling directions. Sampling directions vary along the x-axis and basis vary along the y-axis. Gray color indicates average percentage error.

where X_K represents the Eigen Basis with selected set of rows, $\boldsymbol{b_X}$ is the acquired BRDF of a new material using selected sampling directions and $\boldsymbol{m_X}$ is the mean vector of H corresponding to selected directions.

4 Experimental Results

In order to test the effectiveness of the proposed method with few sampling directions several experiments are conducted using MERL dataset [4] besides the obtained results are compared with that of Matusik et al.[3] and a randomly selected set of samples. We compute the percentage error between the actual measured BRDF of a material and its approximation obtained using very few selected set of sampling directions suggested by our method for evaluation:

$$PercentageError = \left(\frac{\sqrt{\frac{1}{N}\sum_{\theta_h,\,\theta_d,\,\phi_d}\left(\frac{1}{C}\sum_{\mathrm{R,G,B}}(\rho_{\mathrm{org}}-\rho_{\mathrm{approx}})^2\right)}}{\sqrt{\frac{1}{N}\sum_{\theta_h,\,\theta_d,\,\phi_d}\left(\frac{1}{C}\sum_{\mathrm{R,G,B}}(\rho_{\mathrm{org}})^2\right)}} \right) * 100 \qquad (6)$$

where ρ_{org} represent the original measured BRDF in logarithm space and ρ_{approx} is its approximation using selected sampling directions, C is the number of color channels, N is the total number of sampling directions in 3D Rusinkiewicz parameterized data, while computing the error using 3D data with 2D bivariate approximation we evaluate approximated data against each ϕ_d value for a given pair of θ_h and θ_d.

First, to find out a suitable combination of basis and samples for representing arbitrary BRDFs, all possible combinations are densely evaluated. To perform such experiments the MERL [4] dataset is divided into two groups, a basis set and a test set of materials. The basis set is used for computing compact basis whereas the test set contains material from which selected set of samples will be taken as a representation of actual BRDF acquisition process using the sampling directions selection process described in section 3.8. Since there are 100 material in the dataset, we divide them into two sets as: 80 materials for calculating basis

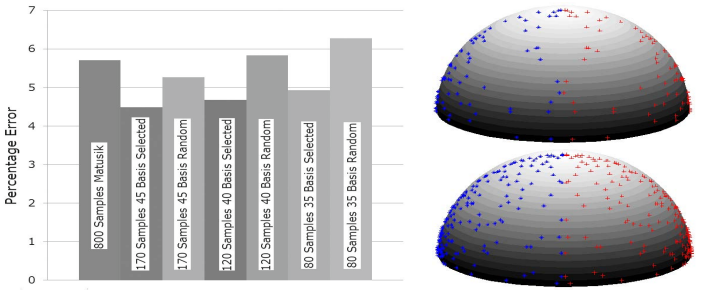


Fig. 4. (a) Comparison of the proposed method with Matusik et al.[3] and a randomly selected set of sampling directions. (b) Plot of selected sampling directions in 4D BRDF form for 80 Samples 35 Basis (top) and 170 Samples 45 Basis (bottom) visualized as pairs of light source (red+) and view (blue*) positions across the hemisphere.

and 20 materials for testing and 10 such configurations of 80-20 combinations of basis and test set are constructed randomly.

Detailed results obtained using the procedure described above are shown in Figure 3. In the figure it can be seen that as we increase the number of basis to 35 and onwards little improvement in reconstruction is observed by increasing the number of samples beyond a certain level. For 35 basis, only a 0.3% improvement occurs when number of samples are increased from 80 to 200. Similarly for 40 basis 0.2% improvement occurs as number of samples are increased from 120 to 200. Specially no improvement occurs at all in reconstruction error by increasing the number after 170 samples for the case of 45 basis. Based on these observation it seems that as few as 35 basis and 80 samples will be sufficient for capturing the variations presents in a large class of isotropic BRDFs quite effectively. However to generalize well we select three combinations of basis and samples for further analysis and comparisons i.e. 35 basis 80 samples, 40 basis 120 samples, 45 basis 170 samples. Figure 4(b) visualizes the selected sampling directions for two combinations. A value of $\phi_h = 0$ and $\phi_d = \pi/2$ is used for the mapping from 2D bivariate to 4D BRDF representation which allows us to compactly display the sampling directions in the form of pairs across the hemisphere.

Using these three combinations the proposed method is compared with the work of Matusik et al.[3] using 800 samples and a randomly selected set of samples. We use our own implementation of their work described in [3]. Figure 4(a) shows the comparison using averaged results for all methods. From this comparison it becomes quite evident that by using bivariate representation and basis approximation significant reduction in the number of necessary sampling directions is possible for a large variety of materials which show little change for rotations of the light and view direction about the half vector. Further these results show that the use of sophisticated sampling method described in section 3.8 allows considerable improvement over a randomly selected set with similar

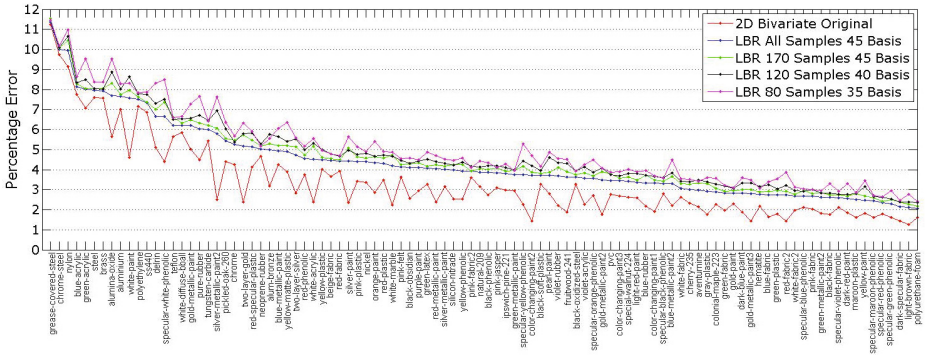


Fig. 5. Detailed Results of 100 materials from the MERL dataset. Comparison of three selected sampling directions (80,120,170) using the Linear Basis Representation (LBR) is shown with all samples and 2D Bivariate BRDF to demonstrate the effectiveness of the proposed method. (Image embedded at high resolution. Please zoom in.)

basis combination. We also explicitly compare results of four materials with the method of Matusik et al.[3]. It is important to mention here that their method uses 3D Rusinkiewicz parameterization [9] of the BRDFs while our proposed method uses 2D bivariate parameterization and further compression via PCA. Results format is: Material Name(Results of [3], Proposed method with 170 samples and 45 basis): Dark Red Paint(4.5%, 2.6%), Gold Paint (3.2%, 2.9%), Aluminum-Bronze(5.7%, 5.2%), Red Plastic (4.9%, 4.5%).

Detailed results of the proposed method on 100 materials from the MERL BRDF dataset [4] are also shown in Figure 5 using the selected combinations. The reconstruction achieved with fewer samples is also compared with the maximum achievable reconstruction using all samples and 45 basis combination to demonstrate how closely fewer samples compare to an exhaustively selected set of sampling directions. The results have been obtained by projecting a single material on basis computed from 99 materials from the dataset. Besides an explicit comparison of 2D bivariate representation of all materials with original 3D MERL data is also shown in Figure 5, with an average of 3.36% over the MERL database it can adequately capture the variations present in different materials.

In order to further demonstrate the effectiveness of reconstructing with fewer samples a visual comparison of reconstructed BRDFs is shown in Figure 6 for several materials. Renderings using original ground truth, randomly selected set of sampling directions and [3] are also included in comparison. Tone mapping algorithm of [16] is used for these renderings. This visual comparison highlights the fact that by using very few sampling directions it is possible to recover the original BRDF of an arbitrary material with a fair amount of accuracy.

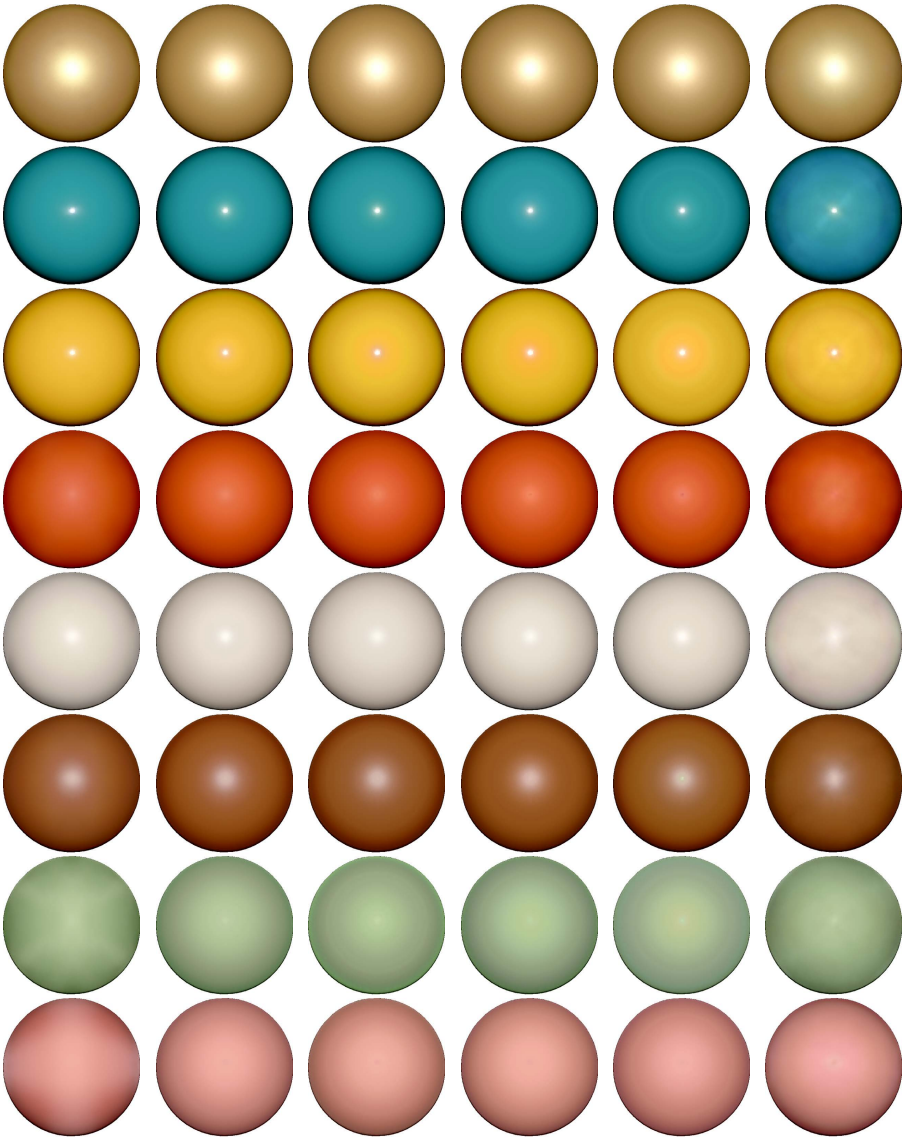


Fig. 6. Visual comparison using several materials between the original renderings and proposed method. (a) Original Measured Data. (b) Reconstruction with 170 Samples 45 Basis. (c) Reconstruction with 120 Samples 40 Basis. (d) Reconstruction with 80 Samples 35 Basis. (e) Reconstruction with Random 80 Samples 35 Basis. (f) Matusik [3] 800 Samples. Materials (along rows from top) are: gold paint, plastic, yellow phenolic, red fabric, rubber, maple, green latex, pink fabric.

5 Conclusion

In this paper we proposed a new method for acquiring BRDFs which significantly reduces the number of necessary measurements for isotropic materials. Our method achieves this by exploiting the inherent similarities present in materials using bivariate parameterization alongside a compact basis representation of a large database of materials. The detailed experimental results demonstrate the effectiveness of the proposed method with few measurements against an exhaustively captured set for a large set of materials from the MERL database. In future we plan to extend this framework to anisotropic and spatially varying BRDFs in an appropriate manner which can enable there acquisition efficiently.

Acknowledgment. The authors would like to acknowledge the financial support of Ministry of Education, Culture, Sports, Science and Technology (MEXT).

References

1. Nicodemus, F.E., Richmond, J.C., Hsia, J.J., Ginsberg, I.W., Limperis, T.: Geometric considerations and nomenclature for reflectance. National Bureau of Standards, Monographs. U.S. Department of Commerce (1977)
2. Koenderink, J.J., Doorn, A.J.V.: Phenomenological description of bidirectional surface reflection. Journal of Optical Society of America 15, 2903–2912 (1998)
3. Matusik, W., Pfister, H., Brand, M., McMillan, L.: Efficient Isotropic BRDF Measurement. In: Eurographics Workshop on Rendering (EGRW), pp. 241–247 (2003)
4. Matusik, W., Pfister, H., Brand, M., McMillan, L.: A Data Driven Reflectance Model. ACM Transactions on Graphics 22, 759–769 (2003)
5. Romeiro, F., Vasilyev, Y., Zickler, T.: Passive Reflectometry. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 859–872. Springer, Heidelberg (2008)
6. Ward, G.J.: Measuring and modeling anisotropic reflection. SIGGRAPH Computer Graphics 26, 265–272 (1992)
7. Dana, K.J., Ginneken, B.V., Nayar, S.K., Koenderink, J.J.: Reflectance and texture of real-world surfaces. ACM Transactions on Graphics 18, 1–34 (1999)
8. Dana, K.J.: BRDF/BTF measurement device. In: IEEE International Conference on Computer Vision (ICCV), pp. 460–466 (2001)
9. Rusinkiewicz, S.: A New Change of Variables for Efficient BRDF Representation. In: Eurographics Workshop on Rendering (EGRW), pp. 11–22 (1998)
10. Marschner, S.R., Westin, S.H., Lafortune, E.P.F., Torrance, K.E.: Image-Based Bidirectional Reflectance Distribution Function Measurement. Applied Optics 39, 2592–2600 (2000)
11. Mukaigawa, Y., Sumino, K., Yagi, Y.: Multiplexed Illumination for Measuring BRDF Using an Ellipsoidal Mirror and a Projector. In: Yagi, Y., Kang, S.B., Kweon, I.S., Zha, H. (eds.) ACCV 2007, Part II. LNCS, vol. 4844, pp. 246–257. Springer, Heidelberg (2007)
12. Mukaigawa, Y., Sumino, K., Yagi, Y.: Rapid BRDF measurement using an Ellipsoidal Mirror and a Projector. IPSJ Transactions on Computer Vision and Applications 1, 21–32 (2009)

13. Ghosh, A., Achutha, S., Heidrich, W., Toole, M.: BRDF acquisition with basis illumination. In: IEEE International Conference on Computer Vision (ICCV), pp. 1–8 (2007)
14. Lawrence, J., Artzi, A.B., DeCoro, C., Matusik, W., Pfister, H., Ramamoorthi, R., Rusinkiewicz, S.: Inverse Shade Trees for non-parametric material representation and editing. ACM SIGGRAPH, 735–745 (2006)
15. Sato, I., Okabe, T., Sato, Y.: Appearance sampling of real objects for variable illumination. International Journal of Computer Vision 75, 29–48 (2007)
16. Reinhard, E., Stark, M., Shirley, P., Ferwerda, J.: Photographic tone reproduction for digital images. ACM Transactions on Graphics 21, 267–276 (2002)

Shadow-Free TILT for Facade Rectification

Lumei Li[1,2,⋆], Hongping Yan[1], Lingfeng Wang[2], and Chunhong Pan[2]

[1] College of Information and Engineering, China University of Geosciences, Beijing
[2] NLPR, Institute of Automation, Chinese Academy of Sciences

Abstract. In this paper, we propose a shadow-free TILT method to rectify facade images corrupted by shadows. The proposed method is deduced from the original TILT, and improve it by introducing a multiplicative shadow factor. That is, in our method, the constraint is represented that the rectified image equals to the low-rank image multiplied by the shadow image, yet with the additive noise corruption. Moreover, the objective function is improved by incorporating the smooth shadow model. Experimental results on both synthetic and real images demonstrate that our method provides more accurate and stable rectification results as compared with the original TILT, especially when shadows are strong in the input images.

1 Introduction

Image-based architecture modeling is a famous application on both computer vision and computer graphics. Numerous methods have been proposed that can be mainly classified into two categories, i.e., the multi-image based methods [1,2,3] and the single-image based methods [4,5]. Generally, the multi-image methods model architectures by using potential information that is obtained from image matching. Their main limitation is the matching precision. As to the single-image methods, they utilize only one image as input, and are more convenient than the multi-image ones. However, the single-image methods often have the viewpoint problem. Hence, it always needs to rectify the input image before using it. In this work, we focus on the image rectification problem. The rectified image can be directly utilized for facades modeling (refer to [6,7]).

The facades of architectures often have notable geometric structures. Thus, traditional methods on image rectification rely on the local features, such as salient points and edges. The famous methods are based on vanishing points [8,9], which are obtained from a family of parallel lines or the geometric relationship between other vanishing points and the optical center. For example, in [8,10,11], vanishing points are obtained through the Cascaded Hough Transformation. Unfortunately, the calculation of vanishing points is very sensitive to the noise, since local features often fail to be detected. Hence, these methods often fail when the background is in a clutter or the facade is corrupted by occlusions.

⋆ This work was supported in part by the Fundamental Research Funds for the Central Universities, and the NSFC under Grant 61075016, 61005036.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 68–79, 2013.

Since the vanishing point based methods suffer from the noise sensitive problem, the texture based methods have been proposed. These methods combining the theoretical framework of low-rank and sparse representation [12,13,14], which do not detect local geometric features directly, but utilize them holistically. One popular approach is the transform invariant low-rank textures (TILT) method proposed in [15]. The TILT assumes that the rectified image is low-rank, since the facades of man-made architectures have meaningful structures, such as regular shapes, symmetric structure, and repeated patterns. Compared with vanishing point based methods, this method does not need to do some pre-processing, e.g., feature detection. Moreover, the iterative algorithm in the TILT is inherently robust to gross errors caused by partial occlusions or corruptions. However, there are some circumstances that the TILT can not handle well. One is the plane difficulty, that is, the deformed domain may be in the different planes. Even we need to connect the conjoint planes in holistic 3D reconstruction. In [16], it has solved the plane difficulty by identifying the intersection line via the low-rank method. The other problem in the TILT is the shadow difficulty, that is, the facade images may be corrupted by shadows, which are caused by neighboring high buildings or some self-protruding parts.

In this work, we mainly focus on the shadow difficulty of the original TILT. We first improve it by introducing a multiplicative shadow factor, and then propose a new shadow-free TILT model. In our shadow-free TILT model, the rectified image equals to the low-rank image multiplied by the shadow image, yet with the additive noise corruption. We proposed a new objective function which further consider the inside proprieties on these images, for example, the smoothness of shadow image and the sparseness of noise. Finally, our shadow-free TILT model is optimized based on the ALM iterative algorithm. Experimental results demonstrate that our method is better than the original TILT on many real facade images, especially when images are under shadows.

The remainder of this paper is organized as follows: Section 2 gives the motivation of improving the original TILT. Section 3 is the sketch of the TILT as well as gives the constraint and objective function of the original TILT according the low-rank textures. Section 4 proposes our shadow-free TILT model, and gives an efficient solution based on the ALM iteration algorithm. In Section 5, some experiments both on synthetic and real facade images are presented by comparing with TILT model. In Section 6, we give a conclusion of this work, and discuss the future work.

2 Motivation

The main problem addressed in this work is to rectify the viewpoint of a facade image. Generally, the facade has rich geometric structures, which are composed by all kinds of regular or symmetric texture. To rectify the viewpoint of a facade, the TILT utilizes regular and symmetric properties of texture by introducing a low-rank texture representation, that is, the rank of rectified facade image is lower as compared with the original input image.

However, in practice, the facade image is often corrupted by shadows, which may be caused by neighboring high buildings or some self-protruding parts. In such cases, the rank of rectified facade image may not be lower as compared with the original input image. Accordingly, the TILT may fail to rectify the viewpoint of a facade image with shadows. Fortunately, the shadow-free rectified facade image, in which shadows are removed, is also low-rank. Hence, to solve the shadow problem, we improve the original TILT by incorporating a shadow model. More precisely, the rectified facade image is decomposed into two parts, i.e., a shadow-free rectified facade image and the corresponding shadow image.

3 Overview of TILT

Constraint: Denote the input facade image as a matrix $I(\mathbf{x})$, $\mathbf{x} \in \Omega$, where Ω is the image domain belonging to $\mathbb{R}^2$. The TILT approach assumes that a transformed facade image $I \circ \tau$ is composed by two components, i.e., a low-rank texture I^o and a corruption E, namely, $I^o + E = I \circ \tau$. Here, $\tau : \mathbb{R}^2 \to \mathbb{R}^2$ is a transformation that belongs to a certain Lie group.

Objective function: Moreover, the original TILT assumes the error matrix E is sparse, since the corruptions are mainly caused by some weak noise or partial occlusions. Accordingly, combining with the low-rank constraint of texture I^o, the objective function can be formulated as $\text{rank}(I^o) + \gamma\|E\|_0$, where γ is a positive parameter that trades off the rank versus the sparsity of the error.

To sum up, the original TILT optimizes Eqn. (1) to obtain the low-rank texture I^o, given by

$$\min_{I^o,E,\tau} \left(\text{rank}(I^o) + \gamma\|E\|_0\right), \quad \text{s.t. } I^o + E = I \circ \tau\,. \tag{1}$$

From Eqn. (1), the original TILT is reasonable to recover low-rank images, since real facade images exist regular and near-regular patterns. Moreover, as reported in [15], it provides excellent experimental results on facade images, especially when images are taken under consistent illumination conditions. Unfortunately, if illumination condition varies in facade image domain, e.g., the image is corrupted by shadows, the original TILT often fails. In Section 4, we mainly focus on the problem brought by shadows, and propose a shadow-free TILT model.

4 The Shadow-Free TILT Model

In this section, the shadow-free TILT model is presented to recover the low-rank texture as well as the shadow from a corrupted facade image. In the following, we first reformulate the *constraint* and *objective function* by introducing shadow model, and summarize the shadow-free TILT model. Then, we present the solution and corresponding algorithm flowchart.

4.1 Constraint of Shadow-Free TILT

As described in Section 2, the rectified facade image $I \circ \tau$ is decomposed into a shadow-free low-rank image I^o and a corresponding shadow image S. Generally, the shadow S can be regarded as the multiplying bias on the original low-rank image I^o, namely, $S \odot I^o \approx I \circ \tau$. The operation $\odot$ is the Hadamard product (element-by-element product). Hence, by incorporating the additive sparse corruptions E, the constraint can be formulated as follows:

$$S \odot I^o + E = I \circ \tau \,. \tag{2}$$

The constraint presented in Eqn. (2) is non-linear because of the transformation τ [15]. A common approach to overcome this difficulty is to linearize the constraint by the first order Taylor expansion around the current estimated transformation τ. Hence, the linearized version of Eqn. (2) becomes

$$S \odot I^o + E = I \circ (\tau + \Delta\tau) \approx I \circ \tau + \nabla I \Delta\tau, \tag{3}$$

where ∇I is the Jacobian: derivatives of the input image w.r.t the transformation parameters. And also ∇I is a $h \times w \times p$ tensor, where h, w are the height and width of the input image I, and p is the number of the parameters of τ.

All values in shadow matrix S should be positive. Thereby, Eqn. (3) can be rewritten as follows (the *constraint*),

$$\begin{aligned} S \odot I^o + E &= I \circ \tau + \nabla I \Delta\tau \\ I^o + \frac{1}{S} \odot E &= \frac{1}{S} \odot (I \circ \tau + \nabla I \Delta\tau) \\ I^o + \widehat{E} &= \widehat{S} \odot (I \circ \tau + \nabla I \Delta\tau), \end{aligned} \tag{4}$$

where $\widehat{E} = \dfrac{1}{S} \odot E$ and $\widehat{S} = \dfrac{1}{S}$. Without confusion, $\widehat{S}$ is also named as shadows in the following.

4.2 Objective Function of Shadow-Free TILT

The main goal of this work is to recover the transformation τ from the constraint of Eqn. (4). Based on the observations on three images, i.e., the low-rank image I^o, the noise image E, and the shadow image S, the three corresponding objects are listed as follows:

1. *The rank of I^o should be low*: Similar with the original TILT [15], if without any corruptions and shadows, the rank of rectified image should be low. That is, the assumption on I^o is formulated as $\text{rank}(I^o)$.
2. *The transformed noise image $\widehat{E}$ should be sparse*: The corruption image E is assumed to be sparse. As described in above subsection, the shadow S can be regarded as a scaling factor. Thus, $\widehat{E} = \frac{1}{S} \odot E$ is also a sparse matrix. Thereby, the corruption E is formulated as $\|\widehat{E}\|_0$.

3. *The shadow image $\widehat{S}$ should be smooth*: Practically, the shadow is piece-wise smooth, which makes the shadow image S is smooth in entire image domain. Moreover, the values in shadow image should be larger than zero. Thus, it is reasonable to assume that the shadow image $\widehat{S}$ is also smooth. Here, we use the Frobenius norm of the gradient of shadow image $\widehat{S}$ to define its smoothness, namely, $\|\nabla\widehat{S}\|_F^2$.

Combining the above objects, we obtain the following objective function:

$$f(I^o, \widehat{E}, \widehat{S}) = \text{rank}(I^o) + \gamma\|\widehat{E}\|_0 + \beta\|\nabla\widehat{S}\|_F^2, \tag{5}$$

where γ and β are weighting parameters.

Theoretically, the rank of I^o equals its number of positive singular values, that is,

$$\text{rank}(I^o) = \|\Lambda\|_0, \tag{6}$$

where Λ is the singular value matrix of I^o. However, 0-norm used in Eqns. (5) and (6) is difficult to optimize for its non-convexity. Fortunately, breakthroughs have been made in sparse representation [14]. We use 1-norm to relax 0-norm:

$$\|\Lambda\|_0 \to \|\Lambda\|_1, \qquad \|\widehat{E}\|_0 \to \|\widehat{E}\|_1 \tag{7}$$

where $\|\cdot\|_1$ represents the sum of the absolute values. Mathematically, $\|\Lambda\|_1 = \|I^o\|_*$, where $\|\cdot\|_*$ is the nuclear norm. To sum up, the objective function Eqn. (5) can be relaxed as follows:

$$f(I^o, \widehat{E}, \widehat{S}) = \|I^o\|_* + \gamma\|\widehat{E}\|_1 + \beta\|\nabla\widehat{S}\|_F^2. \tag{8}$$

The Shadow-free TILT Model: As presented in above two subsections, Eqns. (4) and (8) are proposed to describe the constraint and objective function, respectively.The shadow-free TILT model is summarized as follows:

$$\min_{I^o,\widehat{E},\widehat{S},\Delta\tau} \left(\|I^o\|_* + \gamma\|\widehat{E}\|_1 + \beta\|\nabla\widehat{S}\|_F^2\right) \quad \text{s.t. } I^o + \widehat{E} = \widehat{S} \odot (I \circ \tau + \nabla I \Delta\tau), \tag{9}$$

We obtain our model by incorporating a shadow model into the original TILT. Meanwhile, we convert our model to a convex optimization with a linear constraint. In the following, we give the optimization algorithm of our method.

4.3 Algorithm Based on Augmented Lagrangian Multiplier Method

To optimize our shadow-free TILT model, firstly given $I^o, \widehat{E}, \widehat{S}, \tau$, we solve the optimization problem to get $\Delta\tau$. Then, we update the transformation by $\tau = \tau + \Delta\tau$, and re-substitute τ into the problem. After several times of iteration, this optimization problem converges to a local minima of the original non-linear problem. This process is listed in Algorithm 1 (please refer to the *OUTER LOOP*).

The core part of above process is the updating of $\Delta\tau$. Motivated by the previous works [17,18,15] about sparse and low-rank problems, we adopt the Augmented Lagrangian Multiplier (ALM) iteration method to solve it. The ALM method converts a constrained optimization problem into an unconstrained problem by introducing the Lagrangian Multiplier and a penalty term. Thus, the optimization problem of Eqn. (9) can be reformulated as follows:

$$\mathcal{L}_\mu(I^o, \widehat{E}, \widehat{S}, \Delta\tau, Y) = \min_{I^o, \widehat{E}, \widehat{S}, \Delta\tau} \left(\|I^o\|_* + \gamma\|\widehat{E}\|_1 + \beta\|\nabla\widehat{S}\|_F^2 + \langle Y, R\rangle + \frac{\mu}{2}\|R\|_F^2 \right), \quad (10)$$

where Y is a Lagrange multiplier matrix of appropriate dimensions, parameter $\mu > 0$ is a penalty coefficient to weight the influence caused by infeasible solutions, and matrix $R = R(I^0, \widehat{E}, \widehat{S}, \Delta\tau)$ satisfies

$$R(I^0, \widehat{E}, \widehat{S}, \Delta\tau) = \widehat{S} \odot (I \circ \tau + \nabla I \Delta\tau) - I^o - \widehat{E}.$$

Combining the basic idea of ALM iteration, the problem presented in Eqn. (10) can be solved as following two steps:

$$\left(I^o_{k+1}, \widehat{E}_{k+1}, \widehat{S}_{k+1}, \Delta\tau_{k+1}\right) = \arg\min \mathcal{L}_{\mu_k} \left(I^o_k, \widehat{E}_k, \widehat{S}_k, \Delta\tau_k, Y_k\right), \quad (11)$$

$$Y_{k+1} = Y_k + \mu_k \left(\widehat{S}_k \odot (I \circ \tau + \nabla I \Delta\tau_k) - I^o_k - \widehat{E}_k\right). \quad (12)$$

Here, the parameter μ is updated as $\mu_{k+1} = \rho\mu_k$, where $\rho > 1$, $\mu_0 > 0$. However, it is difficult to minimize I^o_{k+1}, $\widehat{E}_{k+1}$, $\widehat{S}_{k+1}$, and $\Delta\tau_{k+1}$ simultaneously. Thus, we adopt an *alternating direction method* to obtain the objectives. For convenience, we first introduce the soft-thresholding (shrinkage) operator $H_\varepsilon[\cdot]$:

$$H_\varepsilon[x] = \text{sign}(x) \cdot (|x| - \varepsilon), \quad (13)$$

where ε is the soft-threshold. According to the well-known shrinkage analysis proposed in [19,20], the optimal solutions of I^o_{k+1}, $\widehat{E}_{k+1}$, $\widehat{S}_{k+1}$, and $\Delta\tau_{k+1}$ can be expressed as follows [1]:

$$\begin{aligned}
I^o_{k+1} &\leftarrow U_k H_{\mu_k^{-1}}[\Sigma_k] V_k^T \\
\widehat{E}_{k+1} &\leftarrow H_{\lambda\mu_k^{-1}}\left[\widehat{S}_k \odot M_k - I^o_{k+1} + \frac{Y_k}{\mu_k}\right] \\
\widehat{S}_{k+1} &\leftarrow \frac{\widehat{S}_k - \xi\left(2\beta\nabla^2\widehat{S}_k + Y_k \odot M_k - \mu_k M_k \odot \left(\widehat{E}_{k+1} + I^o_{k+1}\right)\right)}{\mathbb{I} + \xi\mu_k M_k \odot M_k} \\
\Delta\tau_{k+1} &\leftarrow \nabla I^\dagger \left(-I \circ \tau + \frac{1}{\widehat{S}_k} \odot \left(\widehat{E}_{k+1} + I^o_{k+1} - \frac{Y_k}{\mu_k}\right)\right)
\end{aligned} \quad (14)$$

where $M_k = I \circ \tau + \nabla I \Delta\tau_k$, $U_k\Sigma_k V_k^T$ is the SVD of $\left(\widehat{S}_k \odot M_k - \widehat{E}_k + \frac{Y_k}{\mu_k}\right)$, and $\mathbb{I}$ is an all-ones matrix with the same size of input image, ∇^2 is the Laplacian operator, and the $\nabla I^\dagger$ is the Moore-Penrose pseudo-inverse of ∇I.

[1] The update of $\widehat{S}_{k+1}$ is described in the supplementary.

As shown in the third row of Eqn. (14), the semi-implicit method is utilized to update the shadow $\widehat{S}$, and the parameter ξ is the iteration stepsize. Compared with the gradient descent method, the semi-implicit is less sensitive to the iteration stepsize. In practical, the shadow image obtained by Eqn. (14) exists noise. To overcome this limitation, we apply a bilateral-like filtering method, i.e., guide-filter filter [21], to decrease the noise while preserving edges. The shadow-free TILT model is summarized in Algorithm 1.

Algorithm 1. Shadow-free TILT Algorithm

Input: The initial rectangle, transformation τ, and shadow image S (S is a matrix with all values are one). Parameters: $k = 0$, $Y_0 = 0$, $E_0 = 0$, $\Delta\tau_0 = 0$, $\mu_0 = 1.25$, $\rho = 1.25$, $\xi = 10^2$, and $\beta = 5 * 10^{-3}$.

Output: The optimized I^o, $\widehat{E}$, $\widehat{S}$, and $\Delta\tau$.

OUTER LOOP:

while *not converge* **do**

  Calculating the normalization of current image, that is, $I \circ \tau = \frac{I \circ \tau}{\|I \circ \tau\|_F}$;

  Calculating the normalization of Jacobian ∇I w.r.t parameters of deformation τ, namely $\nabla I = \frac{\partial}{\partial \zeta}\left(\frac{vec(I \circ \zeta)}{\|vec(I \circ \zeta)\|_F}\right)|_{\zeta=\tau}$;

  INNER LOOP:

  while *not converge* **do**

    $(U_k, \Sigma_k, V_k^T) = \text{SVD}\left(\widehat{S}_k \odot (I \circ \tau + \nabla I \Delta\tau_k) - \widehat{E}_k + \frac{Y_k}{\mu_k}\right)$;

    $I^o_{k+1} = U_k H_{\mu_k^{-1}}[\Sigma_k] V_k^T$;

    $\widehat{E}_{k+1} = H_{\lambda\mu_k^{-1}}\left[\widehat{S}_k \odot (I \circ \tau + \nabla I \Delta\tau_k) - I^o_{k+1} + \frac{Y_k}{\mu_k}\right]$;

    $\widehat{S}_{k+1} = \frac{\widehat{S}_k - \xi\left(2\beta\nabla^2\widehat{S}_k + Y_k \odot (I\circ\tau + \nabla I\Delta\tau_k) - \mu_k (I\circ\tau + \nabla I\Delta\tau_k) \odot \left(\widehat{E}_{k+1} + I^o_{k+1}\right)\right)}{\mathbb{I} + \xi\mu_k (I\circ\tau + \nabla I\Delta\tau_k) \odot (I\circ\tau + \nabla I\Delta\tau_k)}$;

    $\Delta\tau_{k+1} = \nabla I^{\dagger}\left(-I \circ \tau + \frac{1}{\widehat{S}_k} \odot \left(\widehat{E}_{k+1} + I^o_{k+1} - \frac{Y_k}{\mu_k}\right)\right)$;

    $Y_{k+1} = Y_k + \mu_k\left(\widehat{S}_k \odot (I \circ \tau + \nabla I \Delta\tau_k) - I^o_k - \widehat{E}_k\right)$;

    $\mu_{k+1} = \rho\mu_k$;

  end

  Updating transformation: $\tau = \tau + \Delta\tau_{k+1}$;

end

5 Experimental Results

In this section, we present the experiments of our method by comparing with the original TILT. First, we give both visual and numeric results on a synthetic data in Subsection 5.1. In Subsection 5.2, we evaluate our method on real facade images from three aspects. The parameters related to shadow are set as follows: the shadow update stepsize $\xi = 10^2$ and the shadow weight $\beta = 5 * 10^{-3}$. Other parameters are the same with the original TILT (see Algorithm 1).

5.1 Synthetic Data

In this experiment, we use a synthetic checkerboard image with different shadow strength to evaluate the tolerance for the shadow of our method, compared with the original TILT. The test image is synthesized by the following equation:

$$I_m(\mathbf{x}) = \begin{cases} J_m(\mathbf{x})\dfrac{1}{1+m} & \mathbf{x} \in \text{shadow region} \\ J_m(\mathbf{x}) & \text{otherwise} \end{cases}, \tag{15}$$

where $J_m(\mathbf{x})$ is the image without shadows (see Fig. 1(a)); $I_m(\mathbf{x})$ is the image with shadow (see Fig. 1(b)); and $m \geq 0$ is the shadow strength value. Then, the image is deformed by projective transformation and added by Gaussian noise (see Fig. 1(c)). Thus, the Fig. 1(c) is regarded as the original input image.

Table 1. Comparison of our method with the original TILT on different shadow strengths (number of success)

Shadow Strength	0.0	0.1	0.3	0.6	1.0	2.0
TILT	10	9	9	7	6	0
Shadow-free TILT	10	10	10	10	10	3

We evaluate different strength of shadows by varying m from 0.0 to 2, and the comparison results are listed in Table 1. For a fixed shadow strength, we perform 10 experiments with different interactive regions, and the success numbers are shown in Table 1. For fair comparison, the interactive regions in the original TILT and our Shadow-free TILT model are the same. In this table, the success number of original TILT decreases gradually when the shadow strength is increasing. Surprisingly, our method is stable when shadow strength is not larger than 1.0. When the shadow strength $m = 2.0$, the success number of our method become lower. Fortunately, our method has 3 success times, while the results of original TILT model are all failed. Fig. 1 gives a visual comparison when $m = 1.0$. This experiment indicates that adding a multiplicative factor to weaken the influence of the shadow is meaningful, especially when shadow strength is large.

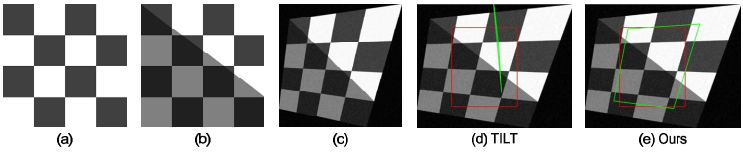


Fig. 1. The sub-figure (a) is a synthetic image, (b) is the image with shadows, (c) is the deformed image by projective transformation. The sub-figures (d) and (e) are the respective results of original TILT and ours. The red window denotes the input and the green denotes the output. In the following figures, we all use this discretion.

5.2 Experiments on Real Facade Images

A Detailed Comparison: This experiment presents a detailed comparison on a image with strong shadows, and the result is illustrated in Fig. 2. As shown in this figure, our result is almost correct, while the original TILT fails along the horizontal direction. The main reason is that the strong shadow not only introduces undesired information, but also causes the texture on the wall partitioned into two regions, i.e., one is under the sunshine and the other is in the shadow. However, by introducing shadow factor, our model can weaken the disturbance of shadow (please refer to the shadow-free image Fig. 2(e)).

Moreover, as shown in Fig. 2(e), the shadow region becomes shallow, however, it is not removed wholly. Hence, the rank of shadow-free image (Fig. 2(e)) may be not lower than the original image with shadow (Fig. 2(d)). Surprisingly, our method can still work. The main reason is that we use nuclear norm to relax the rank . When shadows are partially removed, the singular values will become smaller, even though the number of positive singular values may not become lower. Hence, the sum of singular values will become lower, correspondingly.

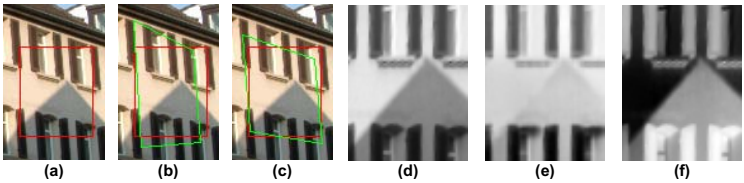


Fig. 2. A detailed experiment about our model. The sub-figure (a) is the input image, (b) is the output of the original TILT, (c) is the output of ours, (d) is gray image after rectification, (e) is the rectified image with shadow removed, and (f) is the corresponding shadow image.

More Comparisons of Our Method with The Original TILT: In this experiment, we present a number of comparisons, and the results are shown in Figs. 3 and 4. The tested images are all corrupted by shadows, and some of them also have other problems, e.g., the occlusion.

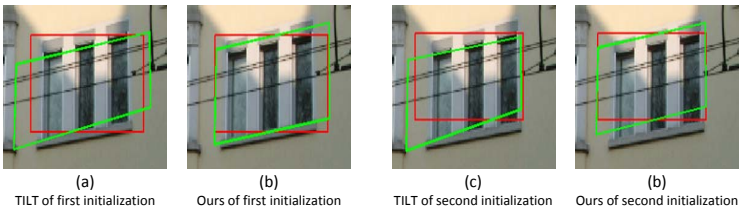


Fig. 3. Compared to the original TILT with two similar initializations. Sub-figures (a,c) are the original TILT results, and (b,d) are ours.

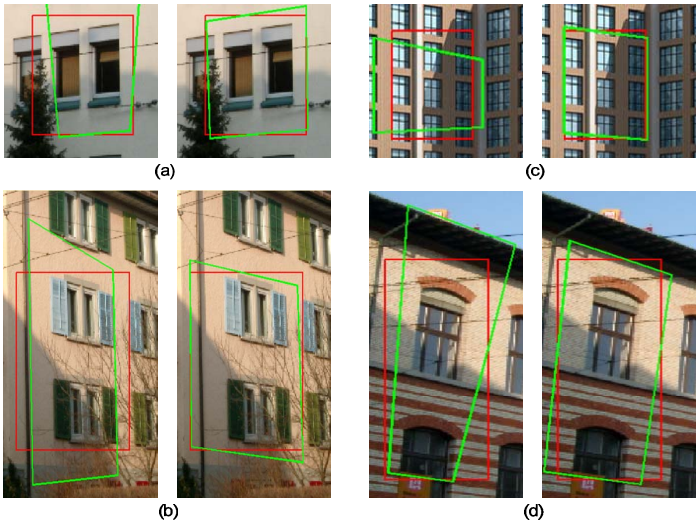


Fig. 4. Comparisons of the original TILT and our method in different difficulties, such as sunshine and occlusions. For each pair of images, the left one is the TILT result, and the right one is our result.

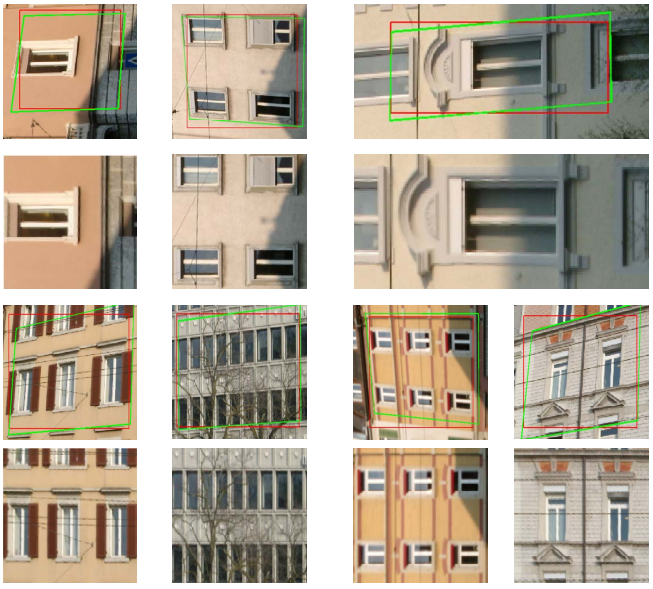


Fig. 5. Rectification results in different circumstances of our method. The first and third rows give the input region (in red) and the output region (in green), and the second and fourth rows are the corresponding rectified images.

Fig. 3 gives a comparison result with two different initializations. As shown in this figure, when shifting the initial region, our model (see Fig. 3 (b,d)) can get stable and correct results. However, the results obtained from the original TILT (see Fig. 3 (a,c)) have errors along the horizontal direction. This experiment shows that our method is less sensitive to the initialization to some extent.

In Fig. 4, we present more comparisons with other difficulties. For example, in Fig. 4(a,b), the input images are corrupted by large occlusions. As shown in this figure, our model still can handle those conditions. The main reason is that the sparseness assumption on the noise (or the occlusion) is also well introduced by our shadow-free TILT model. In Fig. 4(c,d), the sunshine is strong. The comparison results show that our model can handle this difficulty well.

Experiments on ZübuD Database: We use the ZübuD database [22] to evaluate our approach. ZübuD contains 1005 images of 201 buildings, which are taken from different illumination conditions and viewpoints. Fig. 5 presents the seven results with following challenges: different illumination conditions, viewpoints, occlusions, and building types. The experiment results illustrate that our method can also tackle these difficulties.

6 Conclusion and Future Work

In this paper, we propose a new shadow-free TILT model to rectify the deformed images with shadows. The main contribution is that we introduce a multiplicative and smooth shadow factor to improve the original TILT model on real facade image rectification problem. Algorithmically, we convert the shadow image into its reciprocal, which makes the optimization algorithm more convenient and reasonable. Experiment results show that comparing with the original TILT, our shadow-free TILT model can handle facade images with notable shadows better.

The proposed shadow-free TILT model is still rudimentary in handling shadow problem, especially when shadow is very strong. For example, in the synthetic experiment, when shadow factor reaches 2, our method may still fail. In the future, we will add shadow estimation module to improve our shadow-free TILT model. The estimated shadow not only can be used as initialization, but also can be regarded as a constraint in the iteration. Moreover, we can also add the geometric information, such as lines and points, to enhance our model.

References

1. Dick, A.R., Torr, P.H.S., Cipolla, R.: Modelling and interpretation of architecture from several images. Inter. J. of Computer Vision 60, 111–134 (2004)
2. Xiao, J., Fang, T., Tan, P., Zhao, P., Ofek, E., Quan, L.: Image-based facade modeling. ACM Transaction on Graphics 27, 161:1–161:10 (2008)
3. Xiao, J., Fang, T., Zhao, P., Lhuillier, M., Quan, L.: Image-based street-side city modeling. ACM Transaction on Graphics 28, 114:1–114:12 (2009)
4. Hong, W.: On symmetry and multiple-view geometry: Structure, pose, and calibration from a single image. Inter. J. of Computer Vision 60 (2004)

5. Müller, P., Wonka, P., Haegler, S., Ulmer, A., Van Gool, L.: Procedural modeling of buildings. ACM Transaction on Graphics 25, 614–623 (2006)
6. Zhao, P., Quan, L.: Translation symmetry detection in a fronto-parallel view. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1009–1016 (2011)
7. Teboul, O., Simon, L., Koutsourakis, P., Paragios, N.: Segmentation of building facades using procedural shape priors. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3105–3112 (2010)
8. Wu, C., Frahm, J.-M., Pollefeys, M.: Detecting Large Repetitive Structures with Salient Boundaries. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 142–155. Springer, Heidelberg (2010)
9. Micusk, B., Wildenauer, H., Kosecka, J.: Detection and matching of rectilinear structures. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–7 (2008)
10. Tuytelaars, T., Proesmans, M., Gool, L.J.V.: The cascaded hough transform as support for grouping and finding vanishing points and lines. In: Proceedings of the International Workshop on Algebraic Frames for the Perception-Action Cycle, pp. 278–289 (1997)
11. Tuytelaars, T., Gool, L.V., Proesmans, M., Moons, T., Mi, E.: The cascaded hough transform as an aid in aerial image interpretation. In: International Conference on Computer Vision, pp. 67–72 (1998)
12. Peng, Y., Ganesh, A., Wright, J., Xu, W., Ma, Y.: Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 763–770 (2010)
13. Wagner, A., Wright, J., Ganesh, A., Zhou, Z., Mobahi, H., Ma, Y.: Towards a practical face recognition system: Robust alignment and illumination via sparse representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 34, 372–386 (2012)
14. Candès, E.J., Li, X., Ma, Y., Wright, J.: Robust principal component analysis? Clinical Orthopaedics and Related Research abs/0912.3599, 1–39 (2009)
15. Zhang, Z., Liang, X., Ganesh, A., Ma, Y.: TILT: Transform Invariant Low-Rank Textures. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part III. LNCS, vol. 6494, pp. 314–328. Springer, Heidelberg (2011)
16. Mobahi, H., Zhou, Z., Yang, A.Y., Ma, Y.: Holistic 3d reconstruction of urban structures from low-rank textures. In: International Conference on Computer Vision Workshops, pp. 593–600 (2011)
17. Lin, Z., Ganesh, A., Wright, J., Wu, L., Chen, M., Ma, Y.: Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. In: Intl. Workshop on Comp. Adv. in Multi-Sensor Adapt. Processing, Aruba, Dutch Antilles, pp. 213–216 (2009)
18. Lin, Z., Chen, M., Wu, L.: The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. Analysis math.OC, 2209–2215 (2010)
19. Cai, J.F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. on Optimization 20, 1956–1982 (2010)
20. Hale, E.T., Yin, W., Zhang, Y.: Fixed-point continuation for 1-minimization: Methodology and convergence. SIAM J. on Optimization 19, 1107–1130 (2008)
21. He, K., Sun, J., Tang, X.: Guided Image Filtering. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 1–14. Springer, Heidelberg (2010)
22. Shao, H., Svoboda, T., Gool, L.V.: ZuBuD — Zürich buildings database for image based recognition. Technical Report 260, Computer Vision Laboratory, Swiss Federal Institute of Technology (2003)

Reconstructing Shape from Dictionaries of Shading Primitives

Alexandros Panagopoulos[1], Sunil Hadap[2], and Dimitris Samaras[1]

[1] Computer Science Dept., Stony Brook University, USA
[2] Adobe Systems Inc., USA

Abstract. Although a lot of research has been performed in the field of reconstructing 3D shape from the shading in an image, only a small portion of this work has examined the association of local shading patterns over image patches with the underlying 3D geometry. Such approaches are a promising way to tackle the ambiguities inherent in the shape-from-shading (SfS) problem, but issues such as their sensitivity to non-lambertian reflectance or photometric calibration have reduced their real-world applicability. In this paper we show how the information in local shading patterns can be utilized in a practical approach applicable to real-world images, obtaining results that improve the state of the art in the SfS problem. Our approach is based on learning a set of geometric primitives, and the distribution of local shading patterns that each such primitive may produce under different reflectance parameters. The resulting dictionary of primitives is used to produce a set of hypotheses about 3D shape; these hypotheses are combined in a Markov Random Field (MRF) model to determine the final 3D shape.

1 Introduction

Shape recovery is a classic problem in computer vision and a large body of prior work exists on the subject, including a variety of shape-from-X techniques. Shape-from-shading is the instance of the shape recovery problem where shape is inferred by the variations of shading in the image. The goal of this paper is to infer the 3D scene structure, in the form of a normal map, from a single 2D image using the information contained in shading. Although shading is a very important cue for human perception of shape and depth, shape-from-shading is a challenging and generally ill-posed problem in computer vision.

A vast amount of prior work exists in the field of shape from shading. Early work can be found in [1]. A variety of shape-from-shading algorithms are surveyed in [2], and more recently in [3], including approaches based on energy minimization and partial differential equations [4]. A variety of smoothness and curvature constraints in energy minimization is examined in [5] to improve the recovered needle maps. Energy minimization approaches suffer from deep local minima, as discussed in [6], which proposes a stochastic optimization approach to avoid them. Heavy shadows further complicate the SfS problem. In [7], shading is incorporated in the form of additional constraints to a deformable model,

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 80–94, 2013.

in order to estimate shape under varying reflectances and extended to the case of unknown illumination. An MRF formulation of the shape from shading problem is presented in [8], including integrability constraints. While SfS is an ill-posed problem in the case of orthographic projection under a distant light source, [9] shows that assuming a more realistic perspective projection and a point light source, SfS becomes well-posed.

In our approach we are interested in extracting and utilizing information in larger image regions (*image patches*) consisting of multiple pixels. Our motivation comes from the intuition that ambiguities inherent in the problem when looking at individual pixels are reduced when examining larger neighborhoods. A data-driven approach could capture the correlations between local image appearance and geometry, allowing us to perform shape reconstruction based on a relatively small set of hypotheses about local 3D structure that have been learned by observing real data, thus making the problem easier.

Some prior work [10,11] has examined shading and geometry in small image regions. [12] examines shading primitives capturing the shading patterns in folds and grooves of surfaces, including interreflections. A graphical model framework for incorporating patch-based priors in various computer vision problems is presented in [13]. Their results in the SfS problem are however limited to a small subset of synthetic images. Geometric primitives are also utilized in [14], to capture object-specific priors for reconstruction of known object classes, such as faces. In [15] a set of shading primitives is used to capture the folds in cloth, and the surface in between folds is interpolated through a two-level MRF model in order to reconstruct the 3D shape of cloth. Recently, [16] used learned shading primitives to deform the initially known 3D surface of a locally textured object. One of the few patch-based approaches for the general shape-from-shading problem is proposed in [17]. Their method uses a dictionary of spherical primitives and a variational approach to reconstruct the 3D shape of Lambertian objects.

In this work, we use a learned dictionary of geometric primitives to capture the relationship between the appearance and geometry of image patches. Each entry in the dictionary captures the geometry of a small rectangular region (*patch*) and a distribution of the possible image intensities associated with this geometry, as observed in a training set containing images of known geometry. We choose to describe the 3D geometry by a normal map. We assume that the scene is illuminated by a single distant point light. We do not assume a specific type of surface reflectance. In our initial approach to the problem, we assume that the object surface has uniform albedo, so that an image containing only shading variations is available. Shading variations in case of variable albedo could be extracted through other methods [18]. Furthermore, we do not model the effects of cast shadows and interreflections. However, since our method relies more on the higher-frequency components of local appearance, interreflections, which change relatively smoothly over the surface, will have limited influence on our method.

To reconstruct the shape of a new image, we first divide the image into patches. For each image patch, we search the dictionary for patches that have similar

appearance to the observed one. Patch appearance is described on a wavelet basis. We define the distance of the image patch to a dictionary patch as the Mahalanobis distance between the observed appearance and the distribution of appearances that can be produced by the dictionary patch. That distribution corresponds to different parameter choices in the Ward reflectance model [19]. Searching the dictionary for matches to an observed image patch produces a set of hypotheses about the local geometry. Despite the fact that there are infinite possible geometric explanations for the appearance of a given patch, our experiments show that certain explanations are much more probable, making our approach effective. The problem of inferring the shape of the objects in the scene becomes that of properly selecting the normal vectors given the set of local hypotheses obtained by the dictionary.

We combine the local hypotheses into the final 3D shape through a Markov Random Field (MRF) model. The MRF model contains one node per image pixel, with pairwise interactions between them, and the node labels indicate the normal vector at each corresponding pixel. The main contributions of this work are the following:

1. We propose a new metric to capture the similarity between local shading patterns and learned patches using a wavelet decomposition and the Mahalanobis distance. As a result, our method can reconstruct the shape of surfaces that significantly deviate from the lambertian model, and handle images that are not photometrically calibrated. These are both significant restrictions of previously proposed approaches.
2. We describe an algorithm that effectively combines information across multiple scales and combines the local geometric hypotheses to reconstruct the final normal map through an MRF model. Our method achieves state-of-the-art results in real images.
3. We show how a patch-based SfS approach can be used to refine and fill-in gaps in the geometry obtained with 3D sensors such as the Microsoft Kinect.

We present results on synthetic and real data. In both cases, our algorithm is able to recover both the general object shape and finer geometric details. In our experiments, dictionaries are learned on synthetic data, but we are able to use them to reliably reconstruct the shape of real photographs. Comparisons with other approaches [17,20,21,9] on real data show the advantages of our approach.

In the following sections we describe how image patches can be represented and how a dictionary of patches can be learned from a set of training images and their corresponding geometry (Sec.2), and how we can reconstruct the normal map from a test image, using the trained dictionary and formulating the problem as inference on a Markov Random Field (MRF) model (Sec.3). In Sec.4 we present results on synthetic datasets and real images with our method. Sec.5 concludes the paper.

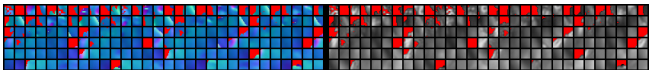


Fig. 1. The data stored in a learned dictionary. Left: the normal map of sample dictionary patches; Right: the mean appearance of each dictionary patch as reconstructed from the mean of appearance wavelet coefficients. Red indicates background pixels.

2 Patch Dictionary

We first construct a dictionary of local geometric primitives (*patches*) from a set of training images with known geometry. Each patch in the dictionary is a small normal map of size $n \times n$, representing the local 3D geometry. Along with the geometry for each patch, we store the distribution of pixel intensities (local appearances) that can be produced by that geometry under different reflectance models, given a light source direction. We refer to each of the learned geometric primitives in the dictionary as a *dictionary patch*. By *patch appearance* we refer to the $n \times n$ grid of pixel intensities describing the appearance of an image patch or dictionary patch. By *patch geometry* we refer to the $n \times n$ grid of normal vectors representing the patch geometry.

2.1 Patch Representation

We reduce the dimensionality of the normal map representation by applying PCA to a subset of patches from the training set and keeping the M_G first eigenvectors. Patch normal maps are therefore projected on the PCA basis and represented by the M_G resulting coefficients. We choose to represent the patch appearance using a Haar wavelet basis [22]. We use Haar wavelets of order 2, using the non-standard construction, resulting in a basis of size $M_A = 16$ for appearance patches.

The distribution of appearances that can be produced by the geometry of a dictionary patch is represented by the mean and variance of the coefficients of the patch appearance. Furthermore, each dictionary patch contains a mask that indicates which pixels belong to the foreground and which (if any) to the background. Therefore, a dictionary patch $\mathcal{D}_i$ is represented by a quadruplet $\{\mathbf{G}_i, \mathbf{M}_i, \mu_i^A, \sigma_i^A\}$, where $\mathbf{G}$ are the PCA coefficients describing the patch normal map, $\mathbf{M}_i$ is the patch foreground/background mask (an $n \times n$ grid of binary values), and μ_i^A and σ_i^A are the means and variances of the coefficients of the appearances that can be produced by the patch geometry.

An example set of patch appearances and geometries from a learned dictionary is shown in Fig.1.

2.2 Dictionary Construction

Let $\mathcal{T} = \left\{\left(T_k^G, T_k^M, \mathbf{t}_k^L\right)\right\}$ be the training set, where each training instance k consists of a normal map T_k^G, a foreground/background mask T_k^M and a light source direction $\mathbf{t}_k^L$. We assume that each training instance is illuminated by a

single distant light source. In order to obtain a good dictionary $\mathcal{D}$ from training set $\mathcal{T}$, we aim to learn a set of geometric primitives that could adequately describe the objects in the training set. Our approach is to: **1)** First examine only the geometry of the training set, learning a set of dictionary patches that correspond to distinct local geometric structures in our training set. **2)** As a second step, we examine the local appearance produced by each of the learned dictionary patches under different reflectances, and store statistics to describe the distribution of these appearances.

To learn the dictionary patch geometry, we first divide the geometry T_k^G of each training instance k into a set $\mathcal{P}$ of overlapping patches P_i of size $n \times n$. We then project the normal map P_k^G of each patch P_i onto the PCA basis, so that P_k^G is represented by a set of coefficients α_k^G. To decide if we should add this patch to the dictionary $\mathcal{D}$, we compute the distance between P_k and each dictionary patch $\mathcal{D}_i$ as:

$$\langle P_k, \mathcal{D}_i \rangle = \sum_{m=1}^{M_G} \left(\alpha_k^G(m) - \alpha_i^G(m)\right)^2 + w_M \sum_{p=0}^{n^2} \left[P_k^M(p), \mathcal{D}_i^M(p)\right], \qquad (1)$$

where the first term is the euclidian distance of the PCA coefficients representing the geometry and the second term the difference of the foreground/background masks, weighed by a weight w_M that determines how strictly we want the foreground/background mask to match between the two patches (a large value of $w_M = 100$ was used in our experiments).

If the distance to the closest patch already in the dictionary is above a threshold θ_D, then a new dictionary patch is added to the dictionary, with the geometry and mask of patch P_k. Therefore, after all patches in the training set have been examined, a (potentially large) dictionary $\mathcal{D}$ has been constructed, containing a variety of distinct local geometric structures.

The second step is to learn the distribution of appearances that can be produced by the geometry of each dictionary patch. In order to do that, we render the normal map of each dictionary patch $\mathcal{D}_i$ using the Ward [19] reflectance model and a set $\mathcal{R}$ of different reflectance parameters, which corresponds to surfaces of varying specularity, varying diffuse intensity and varying anisotropic specular properties. We project the image intensities produced by each reflectance parameter selection onto the wavelet basis, and we store the mean μ_i^A and variance σ_i^A for each appearance coefficient across all reflectance parameters.

Dictionary Light Source Direction. We train the dictionary of patches using a single, known light source direction. This known light source direction is used to associate each local geometric primitive in the dictionary with a range of appearances under different reflectance parameters, removing the dependence of local appearance on light direction.

When reconstructing a test image, the light source direction used to train the dictionary has to be the same as the one that corresponds to the test image. Therefore, we re-compute the distribution of appearances for each dictionary

patch as a first step every time we are provided with a new image to reconstruct and the corresponding light source direction. Generating the distribution of appearances for a dictionary of 30000 patches, such as the one used in our experiments, takes 1-3 minutes. This time is significantly less than the time needed to reconstruct the image from the dictionary, making this solution feasible.

This way, the dictionary does not have to capture the ambiguities caused by varying light source directions, which would lead to both an extremely large dictionary and a very difficult reconstruction problem.

3 Shape Reconstruction

In this section we describe how we reconstruct the geometry when provided with a new image $\mathbf{I}$ and a learned dictionary $\mathcal{D}$. We first divide the input image into a set of overlapping patches. We then find the dictionary patches in $\mathcal{D}$ that are closest in appearance to the patches extracted from the test image $\mathbf{I}$. Finally, we reconstruct the 3D shape from the results of the dictionary look-up using a Markov Random Field (MRF) model.

We divide the image $\mathbf{I}$ into a set of overlapping patches. We define an image patch P_j for each image pixel j, so that P_j is centered at pixel j and has size $n \times n$. This way, we extract all possible image patches from the input image $\mathbf{I}$. For each image patch, we search the dictionary for dictionary patches of similar appearance. We retrieve the k_D dictionary patches that are closest in terms of appearance to image patch P_j (we define the metric to compare patch appearances in the next section, Sec.3.1). Because we defined image patches centered at each pixel, a given pixel i is covered by up to n^2 overlapping image patches. As a result, there are up to $k_D n^2$ dictionary matches that include pixel i, with each dictionary match defining a normal vector for pixel i. Each of these results is considered a hypothesis about the vector at pixel i.

Because of the dependency of patches on scale, we repeat this search for a set of different scales $\mathcal{S}$. We use re-scaled versions of the original image, at scales both coarser and finer. We examine every patch at the coarsest scale. At finer scales, we only examine those image patches that have image variance above a given threshold (0.001 in our experiments). Moving to finer scales, the patches get smaller relative to the image. As a result, the average image variance per patch reduces, so that only finer details are examined at finer scales (see Fig.2). The dictionary matches of size $n \times n$ at each scale are then re-scaled to the scale of the original image. As a result, the final set of dictionary matches contains patches of varying sizes, corresponding to the different image scales used for the search.

The above procedure generates up to $|\mathcal{S}|k_D n^2$ normal vector hypotheses for each image pixel i. From this large set of hypotheses, we keep only the k normal vectors that correspond to the k dictionary patches with the lowest matching cost that contain this image pixel. These candidate normal vectors will be subsequently used in the MRF optimization described in section 3.2 to obtain the final normal map.

3.1 Dictionary Search

To determine how well a dictionary patch (consisting of a normal map patch and a set of appearance statistics) matches an image patch (consisting of a patch of image intensities) we use the Mahalanobis distance.

Let P_j be an image patch consisting of appearance P_j^A (a $n \times n$ patch of per-pixel intensities) and a foreground/background mask P_j^M. Projecting the foreground pixels of appearance P_j^A onto the appearance wavelet basis, we obtain a set of coefficients α_j^A that describe the image patch appearance. We compute the distance between the appearance of P_j and that of a dictionary patch $\mathcal{D}_i$ by the Mahalanobis distance:

$$D_A(\mathcal{D}_i, P_j) = \sqrt{\sum_{m=1}^{M_A} \frac{\left(\alpha_j^A(m) - \mu_i^A(m)\right)^2}{\left(\sigma_i^A(m)\right)^2}}, \tag{2}$$

where μ_i^A and σ_i^A are the mean and variance of the appearance coefficients of the appearances produced by dictionary patch $\mathcal{D}_i$ under different reflectances, as computed during training [1].

To compute the quality of the match between dictionary patch $\mathcal{D}_i$ and image patch P_j, we also compute the similarity of the foreground/background masks of the two patches:

$$D_M(\mathcal{D}_i, P_j) = \frac{1}{n^2} \sum_{x=1}^{n} \sum_{y=1}^{n} \left[\mathcal{D}_i^M(x,y) = P_j^M(x,y)\right], \tag{3}$$

where $\left[\mathcal{D}_i^M(x,y) = P_j^M(x,y)\right] = 1$ if both masks agree for pixel (x,y) and 0 otherwise.

Finally, we can take into account the similarity of dictionary patch $\mathcal{D}_i$ to a rough 3D shape prior. This term allows us to utilize the normal map estimate from the previous scale while searching for dictionary matches at the next scale, when examining multiple scales. Similarly, this term can allow the incorporation of rough geometry knowledge. Such an example is the refinement of 3D shape captured by a commercial 3D camera, such as a Kinect sensor. The geometry prior cost is defined as:

$$D_G(\mathcal{D}_i, P_j) = \sum_{m=1}^{M} \left(\alpha_i^G(m) - \alpha_j^G(m)\right)^2, \tag{4}$$

where $\alpha_i^G(m)$ is the m-th coefficient of the geometry of dictionary patch $\mathcal{D}_j$, $\alpha_j^G(m)$ is the m-th coefficient of the *coarse* geometry of the test patch j. Assuming that the geometry prior is coarse, only the first M geometry coefficients are taken into account, corresponding to the low-frequency components of the geometry prior. In our experiments, $M = 3$.

The final cost of using dictionary patch $\mathcal{D}_i$ to explain image patch P_j is then:

$$cost(\mathcal{D}_i, P_j) = D_A(\mathcal{D}_i, P_j) + w_M D_M(\mathcal{D}_i, P_j) + w_G D_G(\mathcal{D}_i, P_j), \tag{5}$$

where w_M and w_G are weight that control the relative strength of match and geometry prior matching ($(w_M, w_G) = (1000, 1)$ in our experiments).

[1] We have assumed that covariances between appearance coefficients are 0, which lead to no significant deterioration in results, but significantly faster training and testing.

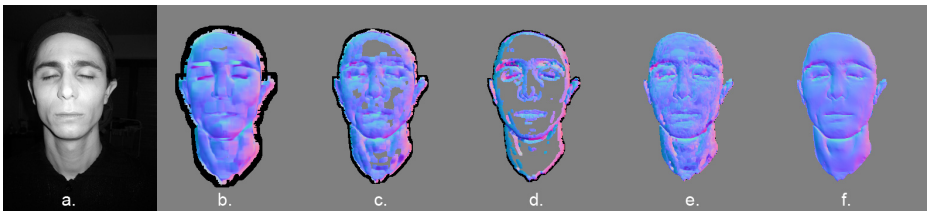


Fig. 2. Combining matches over different scales to produce an initial guess about the normal map. a) original image; b-d) the normal maps produced by averaging dictionary matches at 3 different scales; e) the combination of all scales to produce an initial guess about the normal map; f) the final result from our method.

3.2 Combination of Dictionary Matches

Having obtained a set of dictionary matches, we then produce an initial guess for the normal map. For each pixel i, we have recovered a potentially large set of normal vectors $\{\mathbf{n}_k^i\}$, across different scales. We compute the mean $\overline{\mathbf{n}}_i$ of all normals at pixel i. Then, we recompute the mean normals iteratively. At each iteration, we take the weighted mean of normals $\{\mathbf{n}_k^i\}$ at pixel i, where each normal is weighed by $1/||\mathbf{n}_k^i - \overline{\mathbf{n}}_i||_2$. This allows us to reduce the effect of outliers to the initial estimate [17]. The results we obtain at each scale and their combination to produce the initial guess are shown in Fig.2.

We refine this initial guess to produce the final normal map by modeling the problem as an MRF model. Through the MRF optimization, we estimate a normal map for the image that is both close to the discovered dictionary matches and that satisfies anisotropic smoothness constraints.

Our MRF model can be represented by a 4-connected 2D lattice, where each node corresponds to an image pixel. Each random variable x_i at pixel i indicates a normal vector $\mathbf{n}_i$. Therefore, the labels x_i take values from a continuous domain. The energy of the MRF model is:

$$E(\mathbf{x}) = \sum_{i \in \mathcal{I}} \phi_i(x_i) + w_2 \sum_{i,j \in \mathcal{N}} \psi_{ij}(x_i, x_j), \tag{6}$$

where $\mathcal{I}$ is the set of image pixels, $\mathcal{N}$ is the set of neighboring pixels in the 4-connected grid, $\phi_i(x_i)$ is the singleton potential that associates the labels x_i with the geometry hypotheses recovered from the dictionary $\mathcal{D}$ and $\psi_{ij}(x_i, x_j)$ is the pairwise potential associating neighboring pixels i and j. The weight w_2 was set to 0.1 in our experiments.

The form of the *singleton potential* is:

$$\phi_i(x_i) = w_i^I \sum_{j=1}^{D_i} \arccos\left(\mathbf{n}(x_i) \cdot \mathbf{n}(\mathcal{D}_j)\right) cost(\mathcal{D}_j), \tag{7}$$

where $\mathbf{n}(x_i)$ is the normal vector at pixel i as indicated by label x_i, D_i is the number of dictionary matches that contain pixel i, $\mathbf{n}(\mathcal{D}_j)$ is the normal vector at pixel i as predicted by match $\mathcal{D}_j$, and $cost(\mathcal{D}_j)$ is the cost associated with match

$\mathcal{D}_j$. Furthermore, w_i^I is a weight that corresponds to *how reliable we expect the dictionary matches at pixel i to be.*

We express w_i^I based on two observations: dictionary matches are more reliable when there is enough local image variability (flat image regions are the least informative), and dictionary matches are not reliable when the matches in different scales differ significantly from each other. Therefore, we define w_i^I as:

$$w_i^I = \frac{\sigma_i}{1 + q(i)}, \tag{8}$$

where σ_i is the local image variance at pixel i, which is computed as the variance of the image pixel intensities in a 6×6 patch centered at pixel i. The term $q(i)$ represents how much the recovered dictionary patches differ at pixel i, and is defined as:

$$q(i) = \frac{1}{\pi} \sum_{s=0}^{|\mathcal{S}|} \sum_{j} \arccos\left(\mathbf{n}(\mathcal{D}_j^s) \cdot \overline{\mathbf{n}}_i\right), \tag{9}$$

where $\mathcal{S}$ is the set of different scales we are examining, $\mathcal{D}_j^s$ indicates the j-th recovered dictionary patch for pixel i using scale s, and $\overline{\mathbf{n}}_i$ is the normal vector at pixel i obtained by averaging the normals at pixel i from all recovered dictionary matches at all scales.

The *pairwise potentials* $\psi_{ij}(x_i, x_j)$ enforce smoothness between the normals of neighboring pixels i and j:

$$\psi_{ij}(x_i, x_j) = w_{ij} \arccos\left(\mathbf{n}(x_i) \cdot \mathbf{n}(x_j)\right), \tag{10}$$

where w_{ij} is a weight computed as a function of the image gradient between pixels i and j:

$$w_{ij} = \max\left\{0, 1 - w_\nabla \nabla I_{ij}\right\}, \tag{11}$$

and w_∇ determines how sensitive the smoothing term is to image gradients (we set $w_\nabla = 4$ in our experiments).

We infer the final shape by minimizing the MRF energy over the labels $\mathbf{x}$.We chose to use the QPBO [23,24] and fusion-move [25] algorithms to perform inference. The QPBO algorithm is used to solve a binary MRF labeling problem between the current set of node labels $\hat{\mathbf{x}}$ and a set of proposed labels $\mathbf{x}'$. The solution is initialized to our initial guess about the normal map, produced by keeping the average normal of the finest scale available for each pixel. We perform a predefined number of iterations, and at each iteration we generate the set of proposed normals (indicated by labels $\mathbf{x}'$) by adding a small random offset to each normal vector in the current solution $\hat{\mathbf{x}}$.

4 Experimental Evaluation

We evaluated our method on both real (Fig.5) and synthetic (Fig.3) data. For evaluation on synthetic data, we used a set of 3D models rendered assuming Lambertian reflectance. The set consisted of 6 models of real objects captured with a 3D scanner [26,27] and rendered from 142 different viewpoints and a set of 2.5D range images of 11 different objects [28], captured from 66 different

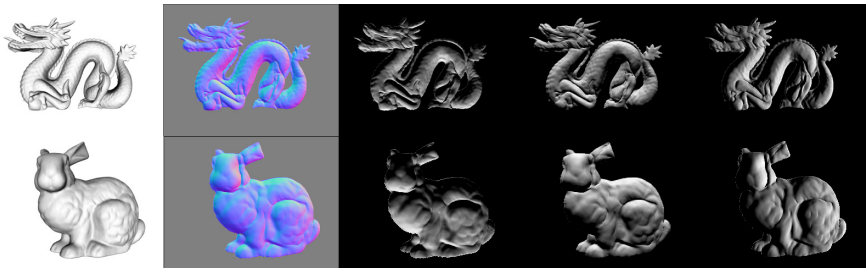


Fig. 3. Reconstruction of normal maps of synthetic images. The images are generated by rendering depth maps of objects collected by 3D scanning [26,27]. We show the reconstructed normal maps and renderings of the reconstructed shape under different illuminations.

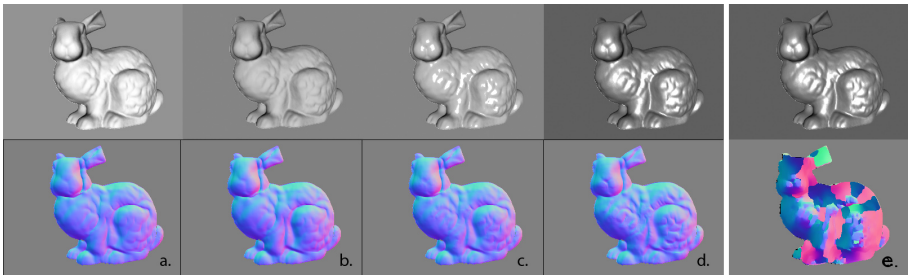


Fig. 4. Effect of non-lambertian reflectance: a-d) reconstruction using the Mahalanobis distance metric, e) reconstruction using Euclidian distance. a) Lambertian reflectance; b) Lambertian reflectance, under-exposed image; c,d,e) Specular reflectance using the Ward model. Our approach achieves results that are robust to reflectance and photometric calibration, while it is impossible to reconstruct a specular surface using just the Euclidian distance. Notice also that the surface in (d) is more specular than the most specular reflectance parameters used while training, showing the ability of our approach to generalize over reflectance parameters.

viewpoints. We used a subset of the viewpoints available, resulting in a set of 150 images. We used leave-one-out cross-validation to evaluate our algorithm: we reconstructed the shape from an image of model i using a dictionary trained on all models other than i (excluding multiple views of the same object as well). We used 4 scales (1/4, 1, 2 and 4 times the size of the original image) to recover matching patches from the dictionary. The smaller scale better captures the overall shape of the object, while finer scales can better capture detail. A total of 5000 iterations was performed during MRF inference. The running time of our algorithm was 20-40 minutes per image, depending on image size and the size of the dictionary (running time measured on an Intel Core i5 machine). Training for a dataset of 150 images takes slightly over an 1hr. We integrated the normal maps estimated by our method using the M-estimator [29], in order to produce the final 3D surfaces (Fig 6).

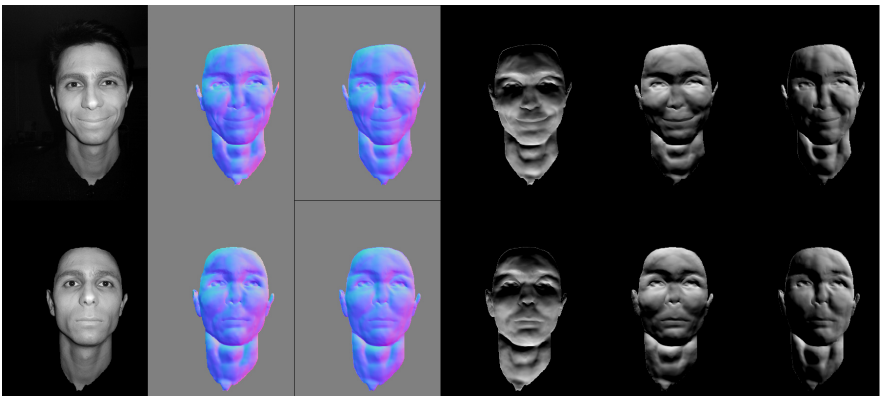


Fig. 5. Reconstruction from a real photograph. From left to right, original image (from [9]); the normal map estimated with our method; the normal map after integrating our estimate using the M-estimator [29]; 3 rendered images with the normal map we estimated and different light directions.

For our experiments, we used a dictionary of 30000 patches of size 12×12 pixels. We used a Haar wavelet basis of size 16 and the first 90 PCA eigenvectors for the patch normal maps. We observed that dictionaries of at least 10000 patches were necessary in order to get satisfactory reconstructions, while having more than 30000 patches (for the selected patch size) was usually only marginally beneficial to our results. Furthermore, it was apparent from our experiments that the patch size needs to be at least 8×8 pixels in order to properly capture local shape. We can demonstrate this through a custom dictionary containing only patches of spherical surfaces. Reconstructing an image from that dictionary is significantly more accurate with patch sizes over 8×8 pixels, which would imply that relatively large patch sizes are required to reliably capture the local curvature of surfaces, since this custom dictionary ignores finer details. Furthermore, in these experiments, using a 16×16 patch size on an image that has been rescaled to be 4 times larger than the original (without adding any detail/information) is significantly more accurate than using 4×4 patches on the original image.

In our experiments, our method significantly outperforms previous shape-from-shading approaches (Fig.7,8). It is able to reliably capture the general orientation of surfaces and is able to reconstruct much more local detail than other approaches [20,21,9]. This can be attributed to the fact that most shape-from-shading approaches rely on some kind of smoothness constraint, whereas in our case such constraints are replaced by the learned primitives. Smoothness needs to be enforced much more weakly during our MRF inference, allowing the solution to retain a lot of local detail. In our experiments with real data, our method also outperforms the shape-from-shading approach of [9] that applies to specific cases of the problem that can be well-posed. The ability of our method to

Fig. 6. Examples of 3D surfaces reconstructed from the normal maps estimated with our method, using the M-estimator [29]

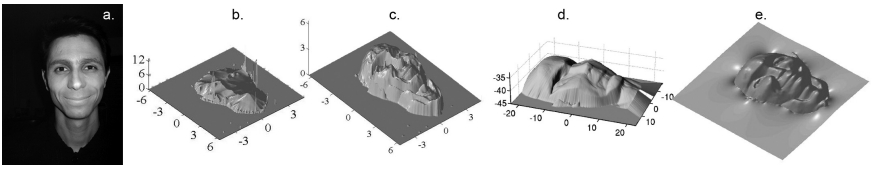


Fig. 7. Comparison of our method with other approaches: a) original image; Surface estimates by: b) [20]; c) [21]; d) [9]; e) our approach. Our approach captures both the overall shape of the object as well as the details better, resulting in a 3D face with clearly discernible features and a closer resemblance to the original face

handle surfaces that are not lambertian is one extra reason for the improved performance on real images. The use of the Mahalanobis distance further allows us to cope with images that are not photometrically calibrated (e.g. underexposed images), which can be challenging when matching the local patch appearance, since in the set of reflectances used to build the distributions of appearances in the dictionary we have also included surfaces with lower uniform albedo.

One weakness of our method is that the quality of the results diminishes in the case of objects with large flat surfaces, indicating that flat patches are significantly more ambiguous than patches that contain even slight shading variations.

Refining Coarse Geometry. We can also use our approach to refine a coarse normal map. We obtain the initial geometry using a Microsoft Kinect (a consumer device that includes a 3D scanner and a camera). The collected data are an image and a depth map. The depth values in the depth map are reliable but of low resolution. Therefore, computing the normal vectors from the depth map leads to unsatisfactory results, even when smoothing is used on the depth values, as shown in Fig.9. Furthermore, the collected depth map contains a lot of holes, especially around the occlusion borders of objects. We can use our approach to refine such results, by including the geometry information captured in Eq.5.

Fig.9 shows the results for an example scene captured using a Kinect. Our method is able to complete the holes in the collected depth map, and to obtain a convincing normal map. We show the normal maps we obtain from the Kinect depth data using various levels of smoothing on the depth values for comparison.

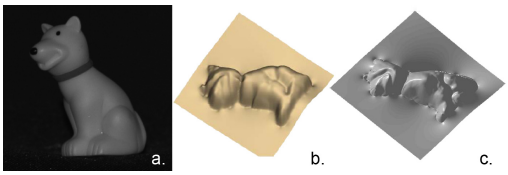


Fig. 8. Comparison of our method with [17] on a real image (from [17]): a) original image; Surface estimates: b) Result as shown in [17]; c) by our approach. Our method is able to recover more detail and a more accurate overal shape

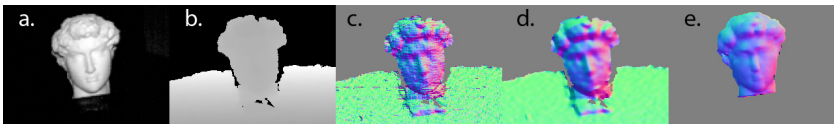


Fig. 9. Refinement of geometry captured with a Kinect: a) the image captured by the Kinect; b) the depth map captured by the Kinect; c) normals computed by the depth map; d) normals computed by the depth map after gaussian smoothing of depth values; e) normals computed by refining the smoothed normal map (d) using our method. We have correctly completed all the object edges, as well as increased the detail in the object while removing noise

5 Conclusions

In this paper we presented a data-driven approach to the problem of shape-from-shading from a single image. We described how we can build a dictionary that captures the correlations between different structures in local shading and geometry. We propose a way to recover hypotheses about the local 3D geometry from the local appearance in a way that is robust to non-lambertian reflectance and photometric calibration. We recover the final 3D shape by combining these hypotheses in an MRF model. The advantages the proposed data-driven approach are that it removes a lot of typical considerations in SfS algorithms, such as boundary conditions or the choice of camera model, and enables us to explicitly deal with surfaces that deviate from the lambertian reflectance model. The results with this approach outperform previous shape-from-shading approaches, even when such approaches make significantly more assumptions than ours. The versatility of such an approach also allows us to use it in order to refine coarse geometric data captured from other sources. Future work will incorporation of priors about albedo in our dictionary representation.

Acknowledgements. This work was supported by grants NSF CNS-0627645, IIS-0916286, IIS-1111047, Adobe Systems Inc. and DIGITEO-Subsample.

References

1. Brooks, M.J.: Shape from shading. MIT Press, Cambridge (1989)
2. Zhang, R., Tsai, P.S., Cryer, J.E., Shah, M.: Shape from shading: A survey. IEEE TPAMI 21, 690–706 (1999)
3. Durou, J.D., Falcone, M., Sagona, M.: Numerical methods for shape-from-shading: A new survey with benchmarks. CVIU 109, 22–43 (2008)
4. Prados, E., Faugeras, O.: A generic and provably convergent shape-from-shading method for orthographic and pinhole cameras. Int. J. Computer Vision 65, 97–125 (2005)
5. Worthington, P.L., Hancock, E.R.: New constraints on data-closeness and needle map consistency for shape-from-shading. IEEE TPAMI 21, 1250–1267 (1999)
6. Crouzil, A., Descombes, X., Durou, J.D.: A multiresolution approach for shape from shading coupling deterministic and stochastic optimization. PAMI 25 (2003)
7. Samaras, D., Metaxas, D.: Incorporating illumination constraints in deformable models for shape from shading and light direction estimation. PAMI 25, 247–264 (2003)
8. Potetz, B.: Efficient belief propagation for vision using linear constraint nodes. In: CVPR 2007. IEEE Computer Society, Minneapolis (2007)
9. Prados, E., Faugeras, O.: Shape from shading: a well-posed problem? In: CVPR, vol. II, pp. 870–877. IEEE (2005)
10. Potetz, B., Lee, T.S.: Statistical correlations between two-dimensional images and three-dimensional structures in natural scenes. Journal of the Optical Society of America A 20, 1292–1303 (2003)
11. Potetz, B., Lee, T.S.: Scaling laws in natural scenes and the inference of 3D shape. In: NIPS 18, pp. 1089–1096. MIT Press, Cambridge (2006)
12. Haddon, J., Forsyth, D.: Shading primitives: Finding folds and shallow grooves. In: ICCV, pp. 236–241 (1998)
13. Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning Low-Level Vision. International Journal of Computer Vision 40, 25–47 (2000)
14. Hassner, T., Basri, R.: Example based 3d reconstruction from single 2d images. In: Beyond Patches Workshop at IEEE Conference on Computer Vision and Pattern Recognition, p. 15. IEEE Computer Society (2006)
15. Han, F., Zhu, S.C.: A two-level generative model for cloth representation and shape from shading. IEEE TPAMI 29, 1230–1243 (2007)
16. Varol, A., Shaji, A., Salzmann, M., Fua, P.: Monocular 3d reconstruction of locally textured surfaces. PAMI (2011)
17. Huang, X., Gao, J., Wang, L., Yang, R.: Examplar-based shape from shading. In: Proceedings of the Sixth International Conference on 3-D Digital Imaging and Modeling, pp. 349–356. IEEE Computer Society, Washington, DC (2007)
18. Tappen, M.F., Freeman, W.T., Adelson, E.H.: Recovering intrinsic images from a single image. PAMI 27, 1459–1472 (2005)
19. Ward, G.J.: Measuring and modeling anisotropic reflection. SIGGRAPH Comput. Graph. 26, 265–272 (1992)
20. Falcone, M., Sagona, M.: An Algorithm for the Global Solution of the Shape-from-Shading Model. In: Del Bimbo, A. (ed.) ICIAP 1997. LNCS, vol. 1310, pp. 596–603. Springer, Heidelberg (1997)
21. Tsai, P., Shah, M.: Shape from shading using linear approximation. IVC (12) 487–498

22. Haar, A.: Zur theorie der orthogonalen funktionensysteme. Mathematische Annalen 69, 331–371 (1910)
23. Hammer, P.L., Hansen, P., Simeone, B.: Roof duality, complementation and persistency in quadratic 0-1 optimization. Mathematical Programming 28, 121–155 (1984)
24. Kolmogorov, V., Rother, C.: Minimizing nonsubmodular functions with graph cuts-a review. PAMI 29, 1274–1279 (2007)
25. Lempitsky, V., Rother, C., Blake, A.: Logcut - efficient graph cut optimization for markov random fields. In: ICCV (2007)
26. Turk, G., Levoy, M.: Zippered polygon meshes from range images. In: Turk, G., Levoy, M. (eds.) SIGGRAPH 1994, pp. 311–318. ACM, New York (1994)
27. Curless, B., Levoy, M.: A volumetric method for building complex models from range images. In: SIGGRAPH 1996, pp. 303–312. ACM, New York (1996)
28. Hetzel, G., Leibe, B., Levi, P., Schiele, B.: 3d object recognition from range images using local feature histograms. In: CVPR, pp. 394–399 (2001)
29. Agrawal, A., Raskar, R., Chellappa, R.: What Is the Range of Surface Reconstructions from a Gradient Field? In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 578–591. Springer, Heidelberg (2006)

Iterative Feedback Estimation of Depth and Radiance from Defocused Images

Xing Lin, Jinli Suo, Xun Cao, and Qionghai Dai

Dept. of Automation, Tsinghua University*
lin-x10@mails.tsinghua.edu.cn, jlsuo.lhi@gmail.com,
xuncao@gmail.com, qionghaidai@tsinghua.edu.cn

Abstract. This paper presents a novel iterative feedback framework for simultaneous estimation of depth map and All-In-Focus (AIF) image, which benefits each other in each stage to obtain final convergence: For the recovery of AIF image, sparse prior of natural image is incorporated to ensure high quality defocus removal even under inaccurate depth estimation. In depth estimation step, we feed back the constraints from the high quality AIF image and adopt a numerical solution which is robust to the inaccuracy of AIF recovery to further raise the performance of DFD algorithm. Compared with traditional DFD methods, another advantage offered by this iterative framework is that by introducing AIF, which follows the prior knowledge of natural images to regularize the depth map estimation, DFD is much more robust to camera parameter changes. In addition, the proposed approach is a general framework that can incorporate depth estimation and AIF image recovery algorithms. The experimental results on both synthetic and real images demonstrate the effectiveness of the proposed method, especially on the challenging data sets containing large textureless regions and within a large range of camera parameters.

1 Introduction

Recovering scene depth has been a hot topic in computer vision and has broad applications, researchers explored lots of efforts and made large progress in the past decades. Among the large number of depth estimation approaches (e.g. multi-view stereo, structure from motion, depth from shading), Depth-From-Defocus (DFD) and Depth-From-Focus (DFF) are insensitive to occlusion and registration errors[1]. However, there are two main disadvantages, the accuracy of depth estimation deteriorates in the textureless regions, and the images captured in *DFD* or *DFF* suffer from defocus blur, from which we cannot obtain high quality all-focused image of the scene. In this paper, we focus on recovering high quality scene depth and all-focused image simultaneously under a feedback scheme, which converges by estimating two items iteratively and each item benefits from the other one within an iterative step.

* This work was supported by the National Basic Research Project No. 2010CB731800 and the Project of NSFC No. 61035002 & 61170194.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 95–109, 2013.

1.1 Previous Works

Generally, depth estimation approaches using the defocus cues fall into two streams: *DFF* needs to capture a series of defocused images by changing the camera focal setting gradually, and label the local depth to be the focal setting with the highest focal level. Therefore, the key task is to define the proper focal criterion. Various methods of *DFF* are proposed, such as [2],[3],[4], etc. The main shortcoming of *DFF* approaches is that a large number of images are required to ensure that their whole depth of field covers the whole depth range, and the image number increases with the estimation accuracy. DFD measures the amount of blur to infer the final depth using one or multiple defocused images at different focus levels [5][6]. Accuracy of depth estimation from a single image such as [7][8] is limited due to the intrinsic ambiguity in depth estimation process, thus lots of DFD algorithms adopt two or more images. DFD algorithm dates back to Pentland [9] and a large number of variants are proposed in the latter years.

Most DFD approaches model image defocus as a convolution process, the models can be further divided into spatial [10][11] and frequency domain representation [9][12][13][14][15]. According to the formulation of defocus cues, these approaches recover scene depth by integrating cues from local regions, while [5][16][17][18][19] optimize a function defined over the global image. Compared to the local algorithms, the global ones can obtain higher accuracy but are more computationally intensive. The algorithm by Favaro et al.[20] combines benefits from global blur model and local regularization to recover depth maps containing thin structures. From the computational perspective, the depth inference can be performed either in a deterministic manner or under a statistical framework [5][16][17], which adopt Markov Random Field (MRF) to model both scene structure and appearance.

Different from the aforementioned methods, some other researchers[17][21][22] model defocusing as a diffusion process and represent it mathematically using the heat equation.

In spite that a large number of DFD algorithms are proposed and can even deal with some complex cases[1][6], estimation on scene without rich texture is still quite challenging. Another limitation of the previous work is that most approaches focus on estimating scene depth but ignore the advantage of the AIF image, which is of great importance in real world applications. Recently, some researchers try to obtain both depth and AIF images, such as [16], [7], [18], etc. However, either prior of AIF image and depth map are not properly defined[16][18], or two items are computed separately instead of optimizing in an unified way and thus both accuracy and robust are limited.

1.2 Our Approach

This paper utilizes two defocused images focused at different depth planes to estimate depth map as well as AIF image under an iterative feedback framework. This framework incorporates the natural image sparse prior into deblur process to recover AIF image under a coarse initial depth map, and adds constraints

from AIF into the depth estimation algorithm to raise the accuracy of depth map. The experimental results also show that performance of depth estimation is more robust to camera parameter changes than traditional non-iterative DFD algorithms[18][20][21]. The framework of our approach is illustrated in Figure 1, including three stages: depth initialization, AIF recovery and Depth refinement.

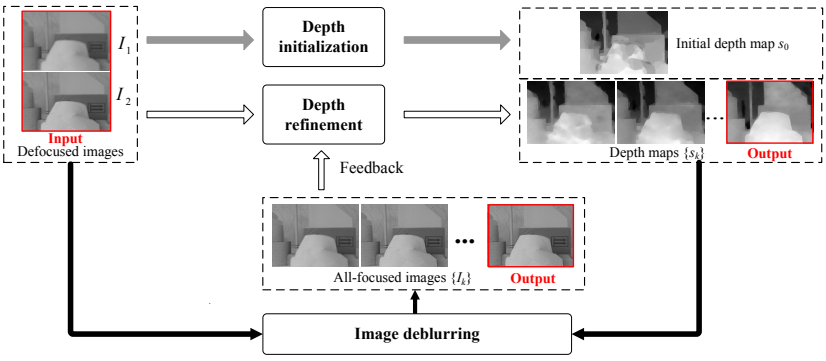


Fig. 1. Diagram of the proposed iterative feedback framework. Here gray arrows, outlined arrows and black arrows denote depth initialization, AIF recovery and depth refinement respectively.

Depth Initialization. For initial depth estimation, we apply relative blur based DFD algorithm similar to [20] by eliminating radiance from the convolution image blur model and leaving depth as the only unknown, then estimate initial scene depth with Total Variation (TV) regularization introduced to favor piecewise smooth depth estimation. From the result of running example, we can see that accuracy deteriorates in the textureless regions, as shown in Figure 1.

AIF Recovery. The spatially variant blur kernels for two defocused images can be coarsely calculated from the initial depth map, and we employ spatially variant non-blind deblurring algorithm to obtain the AIF image. Recently, many deblurring algorithms from multiple images have been proposed [23][24][25], but considering the promising experiment results obtained in [7] by incorporating $||\cdot||_{0.8}$ norm for natural image sparse prior modeling, we extend the spatially invariant single image deblurring method in [7] to be applicable for multiple input images and spatially variant blur. Since most of the depth estimation errors appear in the textureless areas, and we have data constraints from two defocused images along with the natural image sparse prior, we are able to achieve promising deblurring result even the initial depth is not perfectly accurate (see Figure 1).

Depth Refinement. After obtaining AIF, a data term defined between AIF image and the input defocused images is fed back to refine the depth estimation step. Benefiting from newly added data term constraint and a numerical solution robust to outliers in AIF, more accurate depth estimation can be obtained for textureless regions, as shown in Figure 1.

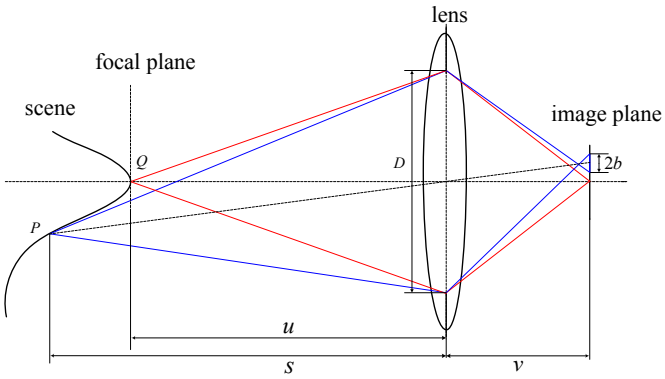


Fig. 2. Blur geometry for a thin lens, with the scene point Q focused and point P having a blur radius b in the image plane

In the rest of this paper, Section 2 firstly explains the imaging model, notations and adopted depth initialization method, Section 3 focuses on the iterative optimization of AIF image and depth map. The numerical solution for depth estimation and experiment validations are given in Section 4 and Section 5 respectively. Section 6 concludes this paper with some discussions.

2 Relative Blur Based Depth Initialization

In this section, we first introduce the adopted imaging model and notations, and then present the relative blur based depth estimation algorithm. Let I denote the AIF image, then its blurry version $I_b(\mathbf{y})$ focusing at a certain depth plane can be represented as:

$$I_b(\mathbf{y}) = \int_{\mathbf{x}\in\mathcal{N}_\mathbf{y}} h_\sigma(\mathbf{y},\mathbf{x}) I(\mathbf{x}) d\mathbf{x}. \tag{1}$$

Here $\mathbf{x}$ and $\mathbf{y}$ represent 2D pixel coordinates, $\mathcal{N}_\mathbf{y}$ is the pixels in I with their blur spots contributing to $\mathbf{y}$ in $I_b(\mathbf{y})$, the blur kernel $h_\sigma(\mathbf{y},\mathbf{x})$ can be approximated by following Gaussian convolution model

$$h_\sigma(\mathbf{y},\mathbf{x}) = \frac{1}{2\pi\sigma^2(\mathbf{y})} \exp\left(-\frac{||\mathbf{y}-\mathbf{x}||^2}{2\sigma^2(\mathbf{y})}\right), \tag{2}$$

in which $\sigma(\mathbf{y})$ is the amount of depth-related blurring at pixel $\mathbf{y}$

$$\sigma(\mathbf{y}) = \kappa b(\mathbf{y}), \tag{3}$$

with κ being the calibration parameter converting world coordinate to image plane and b being the blur radius at pixel $\mathbf{y}$.

According to geometrical optics in [6], the blur radius can be calculated by following equation:

$$b(\mathbf{y}) = \frac{Dv}{2}|\frac{1}{F} - \frac{1}{v} - \frac{1}{s(\mathbf{y})}|, \tag{4}$$

and the meaning of each notation is illustrated in Figure 2: D denotes the aperture diameter, s denotes the depth map, v and u denote focus setting and the distance from focal plane to lens, and the focal length is F.

Given two defocused images I_1, I_2 respectively captured at focal setting v_1 and v_2 but keeping the other camera parameters consistent, we can register them in a simply way[6]. From the convolution model in Eq. (1), the relative blur based convolution model between two defocused images is as follows:

$$I_2(\mathbf{y}) = \int \frac{1}{2\pi\Delta\sigma^2} e^{-\frac{||\mathbf{y}-\mathbf{x}||^2}{2\Delta\sigma^2}} I_1(\mathbf{x})d\mathbf{x}, \tag{5}$$

where $\Delta\sigma^2(\mathbf{y}) = \sigma_2^2 - \sigma_1^2$ is the square of relative depth-related blurring. In the case that $\sigma_1^2(\mathbf{y}) > \sigma_2^2(\mathbf{y})$, above equation turns into

$$I_1(\mathbf{y}) = \int h_{\sigma_1}(\mathbf{y},\mathbf{x})I(\mathbf{x})d\mathbf{x} \approx \hat{I}_{2,\Delta\sigma}(\mathbf{y}) = \int h_{\Delta\sigma}(\mathbf{y},\mathbf{x})I_2(\mathbf{x})d\mathbf{x}, \tag{6}$$

while when $\sigma_1^2(\mathbf{y}) < \sigma_2^2(\mathbf{y})$ we have

$$I_2(\mathbf{y}) = \int h_{\sigma_2}(\mathbf{y},\mathbf{x})I(\mathbf{x})d\mathbf{x} \approx \hat{I}_{1,\Delta\sigma}(\mathbf{y}) = \int h_{\Delta\sigma}(\mathbf{y},\mathbf{x})I_1(\mathbf{x})d\mathbf{x}. \tag{7}$$

Similar to [20], the depth initialization from I_1 and I_2 can be formulated as an optimization problem:

$$\hat{s} = \arg\min_s(\alpha E_d(s) + E_m(s)), \tag{8}$$

where $E_d(s)$ and $E_m(s)$ are respectively data term and regularization term, with α being a weighting factor.

Specifically, the data term can be written as:

$$E_d(s) = \int H(\Delta\sigma(\mathbf{y}))||I_1(\mathbf{y}) - \hat{I}_{2,\Delta\sigma}(\mathbf{y})||_2^2 d\mathbf{y} + \int (1 - H(\Delta\sigma(\mathbf{y})))||I_2(\mathbf{y}) - \hat{I}_{1,\Delta\sigma}(\mathbf{y})||_2^2 d\mathbf{y}, \tag{9}$$

with $H(\cdot)$ being the step function, and regularization term $E_m(s)$ is defined as the isotropic total variation to favor piecewise smooth scene depth

$$E_m(s) = \int ||\nabla s(\mathbf{y})||_2 d\mathbf{y}. \tag{10}$$

The depth initialization algorithm defines relative blur based data terms to separate shape from radiance, thus avoids radiance regularization and reduces computational complexity. The total variation regularization term used in the algorithm can preserve the depth edge better and avoid the over-smooth effect. However, the algorithm does not perform well in the case of textureless scene and is sensitive to camera parameter changes.

3 Iterative Optimization of Depth and AIF Image

After the depth initialization in the last section, the framework iterates between image deblurring and depth estimation until both AIF image and estimated depth map become stable.

3.1 Deblurring from Multiple Defocused Images

According to Eqn. (2)-(4) and initial depth estimation, we can calculate two spatially varying blur kernel maps h_{σ_1},h_{σ_2} corresponding to defocused images I_1,I_2 respectively. Then the optimization of AIF image can be formulated as a spatially variant deblurring:

$$\hat{I} = \arg\min_{I}(E_d(I) + \gamma E_m(I)), \tag{11}$$

where γ is the coefficient that balances data term $E_d(I)$ and regularization term $E_m(I)$. The data term is defined as

$$E_d(I) = || \int h_{\sigma_1}(\mathbf{y}, \mathbf{x}) I(\mathbf{x}) d\mathbf{x} - I_1(\mathbf{y})||_2^2 + || \int h_{\sigma_2}(\mathbf{y}, \mathbf{x}) I(\mathbf{x}) d\mathbf{x} - I_2(\mathbf{y})||_2^2. \tag{12}$$

and for regularization we incorporate the natural image sparse prior:

$$E_m(I) = \int ||\nabla I(\mathbf{y})||_{0.8} d\mathbf{y}. \tag{13}$$

To solve the optimization problem in Eq. (11) numerically, we extend the Iterative Re-weighted Least Squares process (IRLS) used in [7] in two facets: dealing with spatially variant convolution process and incorporating information from multiple input images.

With the constraints from multiple defocused images, relationship between blur kernels and the sparse prior or AIF image, we can get promising deblurring result even depth is not perfectly accurate. Another reason for robustness of AIF recovery to depth errors is that, most depth estimation inaccuracies occur in textureless areas, where the deblur result is quite stable.

3.2 Depth Refinement

After recovering AIF image, an energy term defined over which is fed back to depth estimation steps for refinement in next iteration:

$$E_f(s) = || \int h_{\sigma_1}(\mathbf{y}, \mathbf{x}) I(\mathbf{x}) d\mathbf{x} - I_1(\mathbf{y})||_2^2 + || \int h_{\sigma_2}(\mathbf{y}, \mathbf{x}) I(\mathbf{x}) d\mathbf{x} - I_2(\mathbf{y})||_2^2. \tag{14}$$

We add the feedback energy term to Eq. (8) and get

$$\hat{s} = \arg\min_{s} \alpha(E_d(s) + E_f(s)) + E_m(s). \tag{15}$$

The numerical solution of depth refinement defined in Eq. (15) is similar to the depth initialization algorithm in Section 2. Although using the similar numerical solution, depth refinement step introduces a feedback energy describing

constraints from recovered high quality AIF, depth tends to be improved compared to initialization results, especially in the textureless regions.

In all, with AIF image acting as a bridge between depth and defocused images, the proposed framework targets to refine depth and AIF iteratively. In each step, we adopt an estimation algorithm for one target robust to estimation errors of the other, and thus two modules of the iterative framework can benefit each other. Therefore, the proposed iterative refinement goes in the right direction and obtains promising results for both depth and AIF estimation.

4 Numerical Solutions

The optimization with TV regularization defined in Eq. (8) and Eq. (15) can be carried out in several ways. Considering both numerical efficiency and stability (e.g., robustness to outliers, insensitivity to parameters and initialization value), we modify the Alternating Direction Method (ADM) proposed in [26] to be applicable for our task.

Taking depth initialization as an example, the augmented Lagrangian function of Eq. (8) is as follows:

$$L_A(\mathbf{w}, s, \lambda) = \alpha E_d(s) + \sum\nolimits_{\mathbf{x}\in\Lambda} (||\mathbf{w}_\mathbf{x}||_2 - \lambda'_\mathbf{x}(\mathbf{w}_\mathbf{x} - \mathcal{T}_\mathbf{x} s) + \frac{\beta}{2}||\mathbf{w}_i - \mathcal{T}_\mathbf{x} s||_2^2). \quad (16)$$

Here $\mathcal{T}_\mathbf{x} s$ is the gradient variations at location $\mathbf{x}$ in the depth map lattice Λ, $\mathbf{w}$ is an auxiliary variable, β is a weighting factor to ensure that the solution approximates that of Eq. (8) and numerical solution is of sufficient stability, and λ is the Lagrangian multiplier.

According to [26], the iterative alternating minimization of Eq. (16) can be calculated by

$$\begin{cases} \mathbf{w}^{k+1} \leftarrow \arg\min\limits_{\mathbf{w}} L_A(\mathbf{w}, s^k, \lambda^k) \\ s^{k+1} \leftarrow \arg\min\limits_{s} L_A(\mathbf{w}^{k+1}, s, \lambda^k) \\ \lambda^{k+1} \leftarrow \lambda^k - \beta(\mathbf{w}^{k+1} - Ds^{k+1}). \end{cases} \quad (17)$$

It can be observed that the minimization of $L_A(\mathbf{w}, s^k, \lambda^k)$ with respect to $\mathbf{w}$ is equivalent to the following optimization problems

$$\min_{\mathbf{w}_\mathbf{x}\in R^2} ||\mathbf{w}_\mathbf{x}||_2 + \frac{\beta}{2}||\mathbf{w}_\mathbf{x} - (\mathcal{T}_\mathbf{x} s^k + \frac{1}{\beta}\lambda_\mathbf{x}^k)||_2^2, \quad \forall \mathbf{x} \in \Lambda, \quad (18)$$

and the solution of which can be given explicitly by two-dimensional shrinkage

$$\mathbf{w}_i^{k+1} = \max\{||\mathcal{T}_\mathbf{x} s^k + \frac{1}{\beta}\lambda_\mathbf{x}^k||_2 - \frac{1}{\beta}, 0\} \frac{\mathcal{T}_\mathbf{x} s^k + \frac{1}{\beta}\lambda_\mathbf{x}^k}{||\mathcal{T}_\mathbf{x} s^k + \frac{1}{\beta}\lambda_\mathbf{x}^k||_2}, \quad \forall \mathbf{x} \in \Lambda, \quad (19)$$

where we assume $0 \cdot (0/0) = 0$.

Different from [26], the minimization of L_A with respect to s cannot be solved in closed form because of the complexity of the data term. We use the gradient flow method[21] for depth updation by minimizing

$$E(s) = \alpha E_d(s) + \int \lambda^k(\mathbf{y}) \nabla s(\mathbf{y}) d\mathbf{y} + \frac{\beta}{2} \int ||\mathbf{w}^{k+1}(\mathbf{y}) - \nabla s(\mathbf{y})||_2^2 d\mathbf{y}. \tag{20}$$

We introduce a pseudo-time variable τ, and update the depth map via gradient descending, i.e.,

$$s^{k+1} = s^k + \frac{\partial s}{\partial \tau} \triangle \tau, \tag{21}$$

in which $\partial s / \partial \tau$ is computed by using variational method to calculate $-E'(s)$:

$$E'(s) = \alpha M(s)(E_d(s))' - \nabla \cdot \lambda^k(\mathbf{y}) + \nabla \cdot \mathbf{w}^k(\mathbf{y}) - \nabla \cdot \nabla s^k(\mathbf{y}). \tag{22}$$

In the above equation,

$$\begin{aligned} E'_d(s) = & \int \delta(\Delta\sigma(\mathbf{y})) \cdot \frac{\partial \Delta\sigma(\mathbf{y})}{\partial s} (\widehat{I}_{2,\Delta\sigma}(\mathbf{y}) - I_1(\mathbf{y}))^2 dy \\ & + 2H(\Delta\sigma(\mathbf{y}))(\widehat{I}_{2,\Delta\sigma}(\mathbf{y}) - I_1(\mathbf{y})) \frac{\partial \widehat{I}_{2,\Delta\sigma}(\mathbf{y})}{\partial s} d\mathbf{y} \\ & - \int \delta(\Delta\sigma(\mathbf{y})) \cdot \frac{\partial \Delta\sigma(\mathbf{y})}{\partial s} (\widehat{I}_{1,\Delta\sigma}(\mathbf{y}) - I_2(\mathbf{y}))^2 d\mathbf{y} \\ & + 2(1 - H(\Delta\sigma(\mathbf{y})))(\widehat{I}_{1,\Delta\sigma}(\mathbf{y}) - I_2(\mathbf{y})) \frac{\partial \widehat{I}_{1,\Delta\sigma}(\mathbf{y})}{\partial s} d\mathbf{y}, \end{aligned} \tag{23}$$

and $M(s)$ is the preconditioning operation defined in a similar way as in [21]

$$M(s) = 1/(1 + 2H(\Delta\sigma(\mathbf{y}))I_1 |\frac{\partial \widehat{I}_{2,\Delta\sigma}(\mathbf{y})}{\partial s}| + 2(1 - H(\Delta\sigma(\mathbf{y})))I_2 |\frac{\partial \widehat{I}_{1,\Delta\sigma}(\mathbf{y})}{\partial s}|). \tag{24}$$

The iteration will diverge unless there is severely wrong depth initialization due to a calibration error or completely wrong camera setting. However, our approach promises sufficient initial results and gives a better estimation in the next iterations because most depth estimation errors occur in textureless regions. We model the mutual relationship between the blur kernels. In addition, our approach includes sparse priors of both AIF image and its depth map, as well as the numeric solution is robust to small errors. Therefore, the iterative refinement goes in the right direction and converges to the right solution, if no severe depth initialization error presents.

For clarity, we summarize the steps of the numerical solution as follows:

Algorithm 1. Numerical solution for depth estimation

1: Initialize the depth map as a plane $s^0 = \frac{(v_1+v_2)F}{v_1+v_2-2F}$ and $k = 1$;
2: Repeat following updating rules

- Update auxiliary variable $\mathbf{w}$ from Eq. (19) $w^{k+1} \leftarrow \arg\min_{w} L_A(w, s^k, \lambda^k)$;
- Update depth map s using Eq. (21) $s^{k+1} \leftarrow s^k + \frac{\partial s}{\partial \tau} t$;
- Update Lagrange multiplier λ according to $\lambda^{k+1} \leftarrow \lambda^k - \beta(w^{k+1} - Ds^{k+1})$;

3: Until converge or k attains the predefined maximum iteration number k_{max}.

5 Experiments

In this section, we verify the proposed algorithm on various synthetic and real data and comparison with state-of-the-arts are also provided.

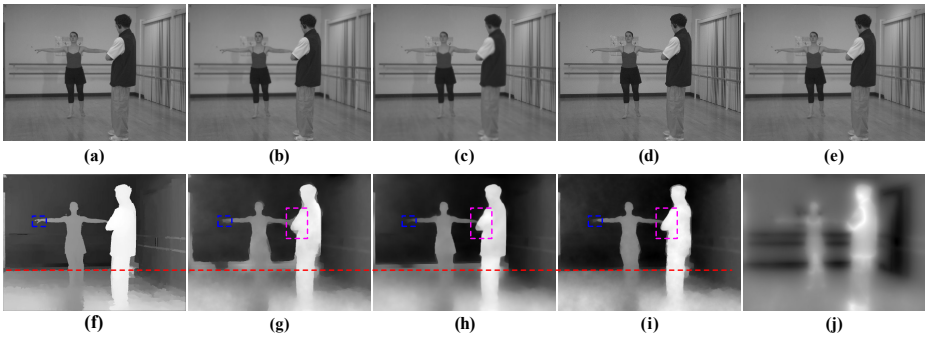


Fig. 3. The performance on Ballet dataset. (a) and (f) Ground truth of AIF image and depth map. (b) and (c) Synthetic defocused images. (d) and (i) AIF image and depth by our iterative algorithm. (g) Depth estimated by initialization method in Section 2. (h) Depth estimated by the algorithm proposed in [20]. (e) and (j) AIF image and depth by the algorithm proposed in [18].

5.1 Synthetic Data

The experiments in this section show the performance of depth estimation and AIF recovery on challenging synthetic data, and we also analyze the robustness to camera settings using a series of experiments.

A. Depth and AIF Estimation on Challenging Cases

One challenging example is the Ballet dataset published in [27], which provides the AIF images and corresponding high quality depth maps computed by stereo matching. We download the data from website and down sample them to resolution 512×384 pixels as the ground truth AIF image and depth, as shown in Figure 3(a) and (f) respectively. Empirically, we assume the depth varies from 2.00m

to 5.00m, the focal length of the virtual camera is 50mm and the $f\#$ is 8, calibration parameter $\kappa = 3e^4$, then two defocused images focusing at 2.00m and 5.00m respectively can be synthesized using the convolution model in Eq. (1), as shown in Figure 3(b)(c). In this experiment, considering the computational complexity, the maximum iteration number is set to 5. We implement our method with Matlab on a PC with an Intel 3.0G Hz Core2Duo CPU, and compare our performance with a state-of-the-art approach on the same hardware platform.

The results are as follows: it costs about 8 mins to obtain the initial depth map using the depth estimation method in Section 2, as shown in Figure 3(g); 12 mins is needed for the method proposed in [20], with the result shown in Figure 3(h); our iterative framework takes about 20 minutes to get the final AIF image and depth map, which are shown in Figure 3(d) and (i).

From the comparison one can see that, although the computational cost of our framework is higher, the depth estimate result is apparently more accurate than the initialization in Section 2 and state-of-the-art non-iterative method proposed in [20], especially in erroneous textureless regions, such as the regions over the red dashed line in figure 3. The non-local mean regularization term introduced in [20] assumes that pixels with similar colors are more likely to be in the same depth plane and can preserve the thin structure slightly better than ours, as shown in the blue marked region in Figure 3. However, it also causes estimation errors in some regions, e.g., the pink rectangle, where there exist abrupt color changes in the same depth plane, while our approach can overcome this problem well.

We also compare our result with one of the previous work estimating scene depth and AIF image simultaneously, and is with publicly available source codes. Quantitatively, the PSNR of recovered AIF image by our algorithm is 39.8dB while that of method proposed in [18] is 36.7dB, and visually the results in Figure 3(e) and (j) give higher quality image and depth respectively, these both validate the effectiveness of the proposed approach. The higher performance is mainly due to the reason that, iterative optimization reduces the number of unknowns compared to joint optimization, and by incorporating proper energy functions, the optimization of each target bears some robustness to the errors of the other one, so the iterative framework converges and outperforms the previous work in challenging cases, especially textureless regions.

B. Analysis on Sensitivity to Camera Parameters

To demonstrate that our framework is more robust to camera parameter changes than traditional non-iterative DFDs, we conduct three groups of experiments to test the effects from three main factors in DFD: $f/\#$, focal length and distance between two focal planes, results are respectively shown in Figure 4(a-c). For a pure analysis of effects from parameters, all the scene depth ranges in this subsection are set to be the same as that the Ballet data set, regardless of its semantic meaning or its true physical depth. Specifically, we change one camera parameter three times to test its effects on final performance while fixing the other two ones, and comparison with state-of-the-arts are also performed.

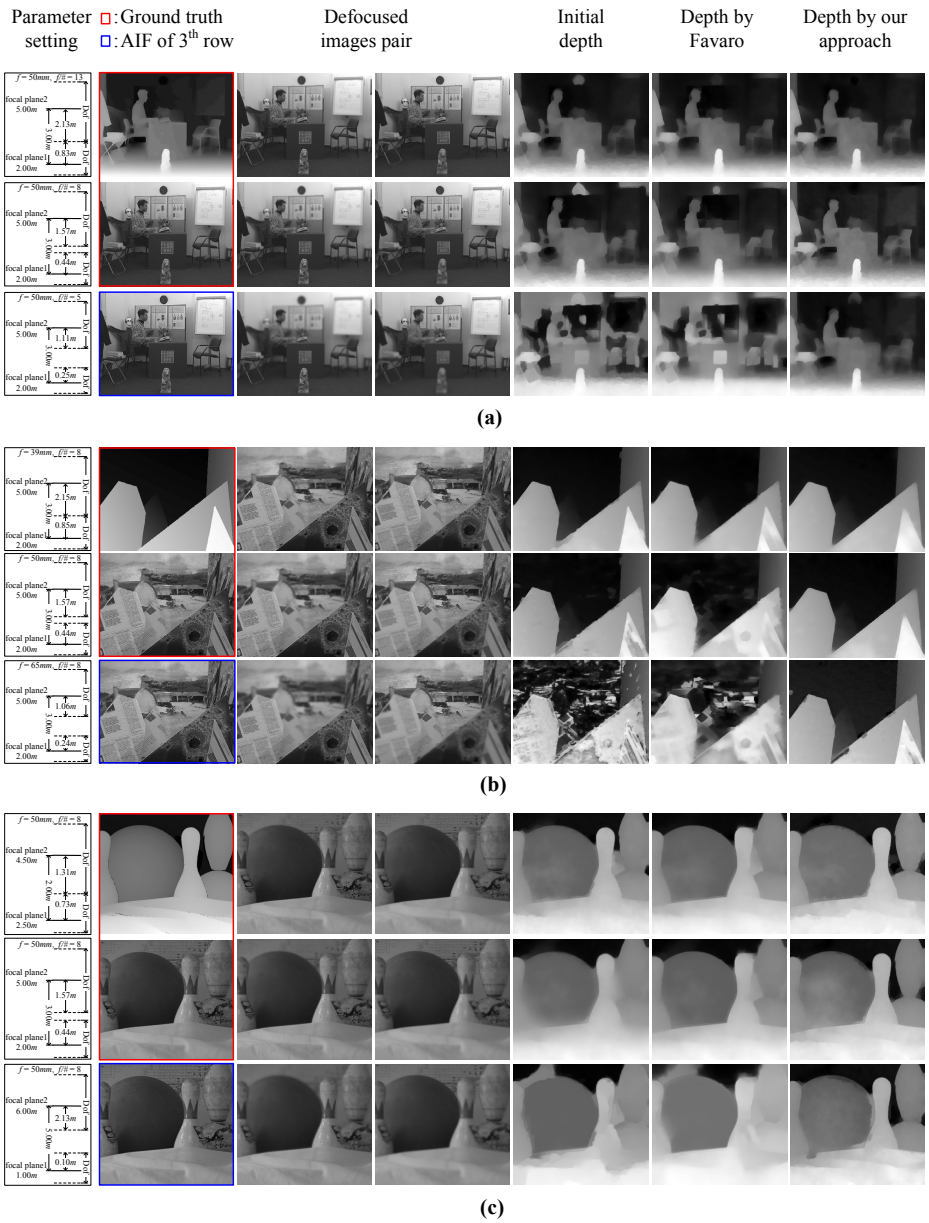


Fig. 4. Results at different parameter settings and performance comparison with a state-of-the-art algorithm[20]. (a) Result on BookArrival data set at different $f/\#$s. (b) Result on Barn data set at different focal lengths. (c) Results on Bowling data set at different focal plane intervals.

The first group of experiment is conducted on Bookarrival data set released by HHI(http://research.microsoft.com/en-us/um/people/sbkang/3dvideodownload/), we set the $f/\#$ of the camera to be 13, 8 and 5, and the results are shown in three rows of Figure 4(a), from top to down. The second and third groups of experiments are conducted respectively on Barn and Bowling data set downloaded from Middlebury(http://vision.middlebury.edu/stereo/data/). The former experiment changes focal length three times (39mm, 50mm and 65mm) and shows the experiment in Figure 4(b). The latter experiment sets the interval between two focal planes differently (2m, 3m and 5m) and the results are shown in Figure 4(c).

Under each parameter setting, we can compute the depth of fields of two defocused images according to projective geometry and CCD parameters. From [1], we know that the stability of DFD is closely related with the sampling of the axial position in DOF intervals. Generally, the robustness to perturbations is optimal when the focal plane interval equals to the union DOF of two defocused images and degenerates as their difference increases. From the experiment results in Figure 4, when the ratio between focal plane interval and the union depth of field of two defocused images within the interval is increasing, such as decreasing $f/\#$ (see Figure 4(a)), increasing focal length (see Figure 4(b)) or increasing

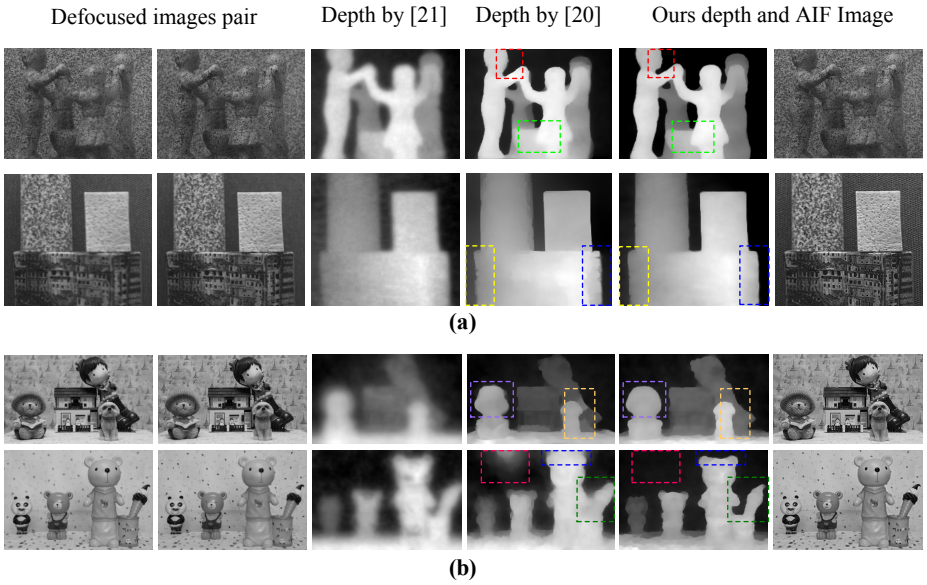


Fig. 5. Results and performance comparison on real captured data. (a) Results on two publicly available data sets. (b) Results on two image pairs captured using Canon EOS 5D.

the interval between two focal planes (see Figure 4(c)), the DFD algorithm is becoming more and more unstable, and this trend validates the conclusions in [1].

However, the results of three groups of experiments consistently reveal that the depth estimation performance by the proposed iterative framework is more insensitive to camera settings, compared to the initial depth in this paper and the algorithm proposed in [20], as shown in Figure 4.

5.2 Performance on Real Captured Data

We also test our algorithm on real data that is publicly available in [15] and [21] with camera parameters known, as shown in Figure 5(a).

From the two input images in the leftmost two columns, we display the depth maps generated by two state-of-the-art approaches proposed by Favaro et al.[20][21] (see 3rd and 4th column), in parallel with our results in 5th column. One can observe that the estimated depth results by [21] (3rd column) are over-smooth, while the results by [20] (4th column) are sharp but suffer from inaccurate edges. Our approach outputs depth maps of comparable sharpness to the latter but recovers more accurate edge structure. We also label out several regions with apparent improvements for better readability. The recovered AIF image by our algorithm is also provided, as shown in the rightmost column.

The publicly available data sets are all with rich texture, we capture some real data with more flat regions to test the algorithm performance further. Figure 5(b) displays the results and comparison, with the same sub-figure arrangements as Figure 5(a). We obtain two defocused images respectively focusing at 1m and 1.5m using Canon EOS 5D, with focal length being 50mm and $f/\#$ being 5.6. The captured images differentiate slightly in field of view and we adopt affine transformation referred in [6] for registration. Note that due to the simplicity of calibration method and the non-planar 3D surface of the scene, alignment is only reasonable but imperfect. However, our algorithm is of sufficient robustness to such slight misalignments and obtains promising results in both scenes, as shown in Figure 5(b). The comparison with state-of-the-arts gives similar conclusion as on publicly available data: the proposed approach outperforms or performs comparably with [21] and [20] in both scene with abundant texture (Figure 5(b) first row) and textureless scene (Figure 5(b) second row); the advantage is especially apparent on simple scenes with large flat regions.

6 Conclusions and Future Work

In this paper, an iterative feedback DFD method is presented to obtain all-in-focus image and more accurate depth simultaneously. The proposed method is able to achieve highly accurate depth estimation, especially in challenging cases and is more robust to the changes of camera parameters, at a slight expense of higher computational complexity than traditional algorithms. The algorithm can also be easily extended to multiple input images for further performance improvement and traditional DFD algorithms can be integrated into the flexible iterative framework.

The future works mainly focus on designing flexible imaging systems to capture multiple-focus images in a single exposure and perform depth estimation or refocusing on dynamic scenes.

References

1. Schechner, Y.Y., Kiryati, N.: Depth from defocus vs.stereo: How different really are they? International Journal of Computer Vision 39, 141–162 (2000)
2. Krotkov, E.: Focusing. International Journal of Computer Vision 1, 223–237 (1987)
3. Nayar, S.K., Nakagawa, Y.: Shape from focus. IEEE Transactions on Pattern Analysis and Machine Intelligence 16, 824–831 (1994)
4. Shoji, H., Shirai, K., Ikehara, M.: Shape from focus using color segmentation and bilateral filter. In: Proceedings of 4th Signal Processing Education Workshop, pp. 566–571 (2006)
5. Chaudhuri, S., Rajagopalan, A.N.: Depth from defocus: a real aperture imaging approach. Springer (1999)
6. Favaro, P., Soatto, S.: 3D shape reconstruction and image restoration: exploiting defocus and motion blur. Springer (2006)
7. Levin, A., Fergus, R., Durand, F., Freeman, W.: Image and depth from a conventional camera with a coded aperture. ACM Transactions on Graphics (Proc. SIGGRAPH) 26, 70–78 (2007)
8. Zhuo, S., Sim, T.: Recovering depth from a single defocused image (submitted to Pattern Recognition and online available)
9. Pentland, A.: A new sense for depth of field. IEEE Transactions on Pattern Analysis and Machine Intelligence 9, 523–531 (1987)
10. Subbarao, M., Surya, G.: Depth from defocus: A spatial domain approach. International Journal of Computer Vision 13, 271–294 (1994)
11. Favaro, P., Soatto, S.: Learning Shape from Defocus. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part II. LNCS, vol. 2351, pp. 735–745. Springer, Heidelberg (2002)
12. Subbarao, M.: Parallel depth recovery by changing camera aperture. In: IEEE International Conference on Computer Vision, pp. 149–155 (1988)
13. Gokstorp, M.: Computing depth from out-of-focus blur using a local frequency representation. In: International Conference on Pattern Recognition, pp. 153–158 (1994)
14. Rajagopalan, A., Chaudhuri, S.: A block shift-variant blur model for recovering depth from defocused images. In: International Conference on Image Processing, pp. 636–639 (1995)
15. Watanabe, M., Nayar, S.: Rational filters for passive depth from defocus. International Journal of Computer Vision 27, 203–225 (1998)
16. Rajagopalan, A., Chaudhuri, S.: An MRF model-based approach to simultaneous recovery of depth and restoration from defocused images. IEEE Transactions on Pattern Analysis and Machine Intelligence 21, 577 (1999)
17. Namboodiri, V.P., Chaudhuri, S., Hadap, S.: Regularized depth from defocus. In: IEEE International Conference on Image Processing, pp. 1520–1523 (2008)
18. Favaro, P., Mennucci, A., Soatto, S.: Observing shape from defocused images. International Journal of Computer Vision 52, 25–43 (2003)
19. Rajagopalan, A., Chaudhuri, S.: Space-variant approaches to recovery of depth from defocused images. Computer Vision and Image Understanding 68, 309–329 (1997)

20. Favaro, P.: Recovering thin structures via nonlocal-means regularization with application to depth from defocus. In: International Conference on Computer Vision and Pattern Recognition, pp. 1133–1140 (2010)
21. Favaro, P., Soatto, S., Burger, M., Osher, S.: Shape from defocus via diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence 30, 518–531 (2008)
22. Namboodiri, V., Chaudhuri, S.: On defocus, diffusion and depth estimation. Pattern Recognition Letters 28, 311–319 (2007)
23. Cai, J., Ji, H., Liu, C., Shen, Z.: High-quality curvelet based motion deblurring from an image pair. In: IEEE International Conference on Computer Vision and Pattern Recognition, pp. 1566–1573 (2009)
24. Chen, J., Yuan, L., Tang, C., Quan, L.: Robust dual motion deblurring. In: IEEE International Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
25. Rav-Acha, A., Peleg, S.: Two motion-based images are better than one. Pattern Recognition Letters 26, 311–317 (2005)
26. Tao, M., Yang, J.: Alternating direction algorithms for total variation deconvolution in image reconstruction. available at Optimization Online (2009)
27. Zitnick, C., Kang, S., Uyttendaele, M., Winder, S., Szeliski, R.: High-quality video view interpolation using a layered representation. ACM Transactions on Graphics (also Proc. SIGGRAPH) 23, 600–608 (2004)

Two-Image Perspective Photometric Stereo Using Shape-from-Shading

Roberto Mecca[1], Ariel Tankus[2], and Alfred Marcel Bruckstein[1]

[1] Department of Computer Science, Technion - Israel Institute of Technology
[2] Department of Biomedical Engineering, Technion - Israel Institute of Technology

Abstract. Shape-from-Shading and photometric stereo are two fundamental problems in Computer Vision aimed at reconstructing surface depth given either a single image taken under a known light source or multiple images taken under different illuminations, respectively. Whereas the former utilizes partial differential equation (PDE) techniques to solve the image irradiance equation, the latter can be expressed as a linear system of equations in surface derivatives when 3 or more images are given. It therefore seems that current photometric stereo techniques do not extract all possible depth information from each image by itself. This paper utilizes PDE techniques for the solution of the combined Shape-from-Shading and photometric stereo problem when only 2 images are available. Extending our previous results on this problem, we consider the more realistic perspective projection of surfaces during the photographic process. Under these assumptions, there is a unique weak (Lipschitz continuous) solution to the problem at hand, solving the well known convex/concave ambiguity of the Shape-from-Shading problem. We propose two approximation schemes for the numerical solution of this problem, an up-wind finite difference scheme and a Semi-Lagrangian scheme, and analyze their properties. We show that both schemes converge linearly and accurately reconstruct the original surfaces. In comparison with a similar method for the orthographic 2-image photometric stereo, the proposed perspective one outperforms the orthographic one. We also demonstrate the method on real-life images. Our results thus show that using methodologies common in the field of Shape-from-Shading it is possible to recover more depth information for the photometric stereo problem under the more realistic perspective projection assumption.

1 Introduction

Reconstruction of three dimensional surface shape is one of the most fundamental problems in Computer Vision. Two reconstruction approaches, both of which first introduced in the 1970s, are Shape-from-Shading (SfS) [1] and photometric stereo [2,3]. Shape-from-Shading is aimed at solving the image irradiance equation, which relates the reflectance map to image intensity. Photometric stereo is a monocular 3D shape reconstruction method based on several images of a scene taken from an identical viewpoint under different illumination conditions. The most common approach in the field divides the task into two: recovery of surface

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 110–121, 2013.

gradients and integration of the resultant gradient field to determine the 3D surface itself. The goal of the first part is to solve a system of image irradiance equations. When given 3 or more images, this system becomes linear. As such, the gradient field can be recovered analytically. For this reason, Shape-from-Shading and photometric stereo have very diverse methodologies, even though the latter is a generalization of the former.

A more recent development in the field of Shape-from-Shading is the transition from the assumption of an orthographic projection of the photographed surface onto the image plane to an assumption of perspective projection [4,5,6,7,8,9,10]. Perspective Shape-from-Shading algorithms were shown to outperform state-of-the-art orthographic techniques ([4]) and be applicable to real-life images ([7]).

Photometric stereo research has focused on reconstruction from three or more images (see [11] for a review). Conditions on the illumination and surface reflectance required to obtain uniqueness of solution for three light source photometric stereo are described by Okatani and Deguchi [12]. Even when the light source intensity and directions are unknown, Shashua [13] has shown that three or more images provide enough information to determine the scaled surface normals of an object up to an unknown linear transformation, which allows the reconstruction of the surface also under unknown lighting conditions (assuming distant light sources) [14].

For this reason, only few studies investigated the problem of 2-image photometric stereo (for example, [15,16,17]). A comprehensive work on existence and uniqueness in 2-image photometric stereo is that of Kozera [18]. Mecca and Falcone [16] extended some of the results of Kozera [18] and Onn and Bruckstein [15], proving a uniqueness result for weak (Lipschitz continuous) solutions. They also proposed two approximation schemes for the numerical solution of this problem: an up-wind finite difference scheme and a Semi-Lagrangian scheme.

Tankus and Kiryati [19] changed the common orthographic projection assumption in photometric stereo to a perspective one (similar to Tankus et al. [4] in Shape-from-Shading), and found an analytic linear solution for the gradient field of a 3-image perspective photometric stereo problem. Yoon et al. [20] employed a variational framework in their perspective photometric stereo algorithm, and demonstrated it using a large sets of input images (≥ 16).

Whereas 2-image orthographic photometric stereo has been investigated for extracting more information from each equation using Shape-from-Shading techniques, and 3-image perspective photometric stereo has an analytical solution for the gradient field, no information is available on the 2-image photometric stereo problem under the perspective projection model. The goal of this research is thus to utilize numerical schemes commonly used in the Shape-from-Shading realm also for 2-image photometric stereo under the perspective projection assumption, thus extracting additional information from each given image. We prove a uniqueness result for weak (Lipschitz continuous) solutions under the perspective projection model, and propose two numerical approxiamtion schemes: an up-wind finite difference scheme and a Semi-Lagrangian scheme. This paper

thus extends and combines three research directions, by Mecca and Falcone [16], Tankus and Kiryati [19], and Onn and Bruckstein [15].

The paper is organized as follows. Following the description of notations and assumptions (Sect. 2), we formulate the new differential model for the photometric stereo problem (Sect. 3) and we then prove the uniqueness of weak solution for the new differential model (Sect. 4). In Section 5 we suggest approximation schemes for the perspective photometric stereo-Shape-from-Shading problem. We demonstrate the performance of the suggested schemes by a comparison with the orthographic schemes [16] (Section 6). Concluding remarks appear in Section 7.

2 Notations and Assumptions

Let us fix the main ingredients for the formulation of the model for the Perspective Shape from Shading (PSfS) presented in [21]:

- the light source is given by a unit vector $\omega = (\omega_1, \omega_2, \omega_3)$ (with $\omega_3 < 0$);
- the surface in the real world is given by the analytical function $h(x, y) = (x, y, \hat{z}(x, y))$ (where the point (x, y) is in the image domain $\overline{\Omega} = \Omega \cup \partial\Omega$, on the optical plane);
- the associated perspective surface is given by the function $k(\xi, \eta) = (\xi, \eta, z(\xi, \eta))$ (where the point (ξ, η) is in the perspective image domain $\overline{\Omega^p} = \Omega^p \cup \partial\Omega^p$, on the focal plane, parallel to the optical one at a focal distance f);
- the transformation used to pass from one point in the optical plane (x, y) to the respective one in the focal plane is $\xi = -\frac{x}{\hat{z}(x,y)} f$, $\eta = -\frac{y}{\hat{z}(x,y)} f$. Then we have: $k(\xi, \eta) = (\xi, \eta, z(\xi, \eta)) = \big(-\frac{x}{\hat{z}(x,y)} f, -\frac{y}{\hat{z}(x,y)} f, \hat{z}(x, y)\big)$.

3 The New Photometric Stereo Differential Model

Now, considering the irradiance equation given by the inner product between the light source ω and the normal vector to the surface $k(\xi, \eta)$ [21], we have the following differential problem (non-linear PDE + Dirichlet boundary condition):

$$\begin{cases} \rho(\xi,\eta)\dfrac{-z_\xi(f\omega_1 + \xi\omega_3) - z_\eta(f\omega_2 + \eta\omega_3) - z\omega_3}{\sqrt{f^2(z_\xi^2 + z_\eta^2) + (z + \xi z_\xi + \eta z_\eta)^2}} = I(\xi,\eta), \text{ on } \Omega^p; \\ z(\xi,\eta) = g(\xi,\eta) \qquad\qquad\qquad\qquad\qquad\qquad \text{on } \partial\Omega^p; \end{cases} \tag{1}$$

which has no unique solution even if the albedo $\rho(\xi, \eta)$ is known.

Let us try to overpass the problem of uniqueness of solution considering the Photometric Stereo (PS) approach using two light sources defined by the unit vectors $\omega' = (\omega'_1, \omega'_2, \omega'_3)$ and $\omega'' = (\omega''_1, \omega''_2, \omega''_3)$ (with ω'_3, $\omega''_3 < 0$).

Using the information obtained by both images we can couple the two equations related to the irradiance equation in (1) obtaining the following system of non-linear PDE:

$$\begin{cases} \rho(\xi,\eta)\dfrac{-z_\xi(f\omega_1'+\xi\omega_3')-z_\eta(f\omega_2'+\eta\omega_3')-z\omega_3'}{\sqrt{f^2(z_\xi^2+z_\eta^2)+(z+\xi z_\xi+\eta z_\eta)^2}} = I_1(\xi,\eta), & \text{on } \Omega^p; \\ \rho(\xi,\eta)\dfrac{-z_\xi(f\omega_1''+\xi\omega_3'')-z_\eta(f\omega_2''+\eta\omega_3'')-z\omega_3''}{\sqrt{f^2(z_\xi^2+z_\eta^2)+(z+\xi z_\xi+\eta z_\eta)^2}} = I_2(\xi,\eta), & \text{on } \Omega^p; \\ z(\xi,\eta) = g(\xi,\eta) & \text{on } \partial\Omega^p. \end{cases} \tag{2}$$

Now, observing that the denominator of both equations is the same (i.e. it does not depend on the light source) and obviously always different from zero, we can explicit the non-linearity from the first equation for example

$$\sqrt{f^2(z_\xi^2+z_\eta^2)+(z+\xi z_\xi+\eta z_\eta)^2} = \frac{-z_\xi(f\omega_1'+\xi\omega_3')-z_\eta(f\omega_2'+\eta\omega_3')-z\omega_3'}{I_1(\xi,\eta)}\rho(\xi,\eta) \tag{3}$$

and replacing it in the other, we obtain the following linear problem

$$\begin{cases} b(\xi,\eta)\nabla z(\xi,\eta)+s(\xi,\eta)z(\xi,\eta)=0, & \text{on } \Omega^p; \\ z(\xi,\eta)=g(\xi,\eta) & \text{on } \partial\Omega^p. \end{cases} \tag{4}$$

Where:

$$\begin{aligned} b(\xi,\eta) = ((f\omega_1'+\xi\omega_3')I_2(\xi,\eta)-(f\omega_1''+\xi\omega_3'')I_1(\xi,\eta), \\ (f\omega_2'+\eta\omega_3')I_2(\xi,\eta)-(f\omega_2''+\eta\omega_3'')I_1(\xi,\eta)) \end{aligned} \tag{5}$$

and

$$s(\xi,\eta) = \omega_3' I_2(\xi,\eta) - \omega_3'' I_1(\xi,\eta). \tag{6}$$

It is clear that the albedo function disappears during the substitution of (3). This means that our new formulation of the PSfS-PS does not depend on the albedo, rather it is possible to compute it a posteriori.

4 Uniqueness of Weak Solution for the New Differential Model

With the aim to prove the uniqueness of weak (Lipschitz) solution of the differential problem (4) we start with the following:

Lemma 1. *If there are not any points $(\xi,\eta) \in \overline{\Omega}^p$ of black shadows for the image functions (i.e. $I_1(\xi,\eta) \neq 0$ and $I_2(\xi,\eta) \neq 0$), we have that $|b(\xi,\eta)| \neq 0$ (i.e. the vectorial function does not vanish in $\overline{\Omega}^p$).*

If we consider as a solution surface $\hat{z}(x,y)$ a Lipschitz one, and we consider the points where it is not differentiable as the family of regular curves $(\gamma_1(t),\ldots,\gamma_k(t))$ where t is the argument of the parametric representation, it is clear that this curve contains also the points of discontinuity of the image functions $I_1(\xi,\eta)$ and $I_2(\xi,\eta)$. Now, since the functions $b(\xi,\eta)$ and $s(\xi,\eta)$ depend directly on $I_1(\xi,\eta)$ and $I_2(\xi,\eta)$, the same family of curve represents the

discontinuity also for these coefficients of the PDE in (4). That is, if we consider our differential problem like an inverse problem of PSfS with photometric stereo technique, searching for a weak solution implies a study of the linear partial differential equation with discontinuous coefficients. Moreover there is a relation between the set of points of discontinuity of $b(\xi,\eta)$ and $s(\xi,\eta)$ and the set of points where the solution $\hat{z}(x,y)$ is not differentiable. In fact they are linked with a bijective correspondence due to the perspective. Another feature about the discontinuity type of $b(\xi,\eta)$ and $s(\xi,\eta)$ is always related to the fact that we are considering an inverse problem where it is proposed to find a Lipschitz solution. This means that it must be a jump discontinuity.

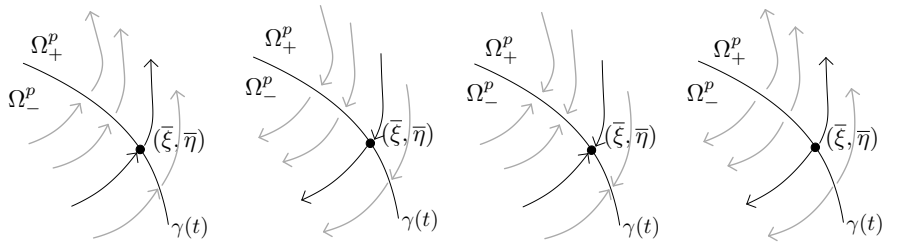


Fig. 1. All the possibles behaviors of the characteristic field close to the discontinuity curve $\gamma(t)$. The only admissible cases (that permits to the information to travel along the characteristic curves) are the first two from the left.

Theorem 2. *Let $\gamma(t)$ be a curve of discontinuity for the function $b(\xi,\eta)$ (and $f(\xi,\eta)$) and let $\overline{p} = (\overline{\xi},\overline{\eta})$ be a point of this curve. Let $n(\overline{\xi},\overline{\eta})$ be the outgoing normal with respect to the set Ω^p_+, than we have*

$$\left[\lim_{\substack{(\xi,\eta)\to(\overline{\xi},\overline{\eta})\\(\xi,\eta)\in\Omega^p_+}} b(\xi,\eta)\cdot n(\overline{\xi},\overline{\eta})\right]\left[\lim_{\substack{(\xi,\eta)\to(\overline{\xi},\overline{\eta})\\(\xi,\eta)\in\Omega^p_-}} b(\xi,\eta)\cdot n(\overline{\xi},\overline{\eta})\right] \geq 0 \tag{7}$$

A schematic explanation of the behavior of the vector field b described by this last theorem is represented in Fig. 1.

Theorem 3. *Let us consider the problem*

$$\begin{cases} b(\xi,\eta)\cdot\nabla z(\xi,\eta) + s(\xi,\eta)z(\xi,\eta) = 0, \text{ a.e. } (\xi,\eta)\in\Omega^p; \\ z(\xi,\eta) = g(\xi,\eta) \qquad\qquad\qquad\qquad \forall(\xi,\eta)\in\partial\Omega^p. \end{cases} \tag{8}$$

Let us suppose that $(\gamma_1(t),\ldots,\gamma_k(t))$, the family of discontinuity curves for $b(\xi,\eta)$ and $s(\xi,\eta)$, are not characteristic curves (with respect to the previous problem). Then there exists a unique Lipschitz solution of the problem.

A sketch of the proof of the previous and main theorem can easily obtained looking at Lemma 1 and Theorem 2. They can be considered as the main ingredients which permit to make travel the information stored on the boundary condition $g(\xi,\eta)$ across all the domain Ω^p. The trajectories followed are defined by the vector field $b(\xi,\eta)$ which has all the good properties to make Theorem 8 proved.

5 Some Approximation Schemes for the Perspective SfS-PS Linear Equation

For the numerical schemes we consider the domain $\overline{\Omega}^p = [a^p, b^p] \times [c^p, d^p]$. The discretization space steps are $\Delta_\xi = (b^p - a^p)/n$ and $\Delta_\eta = (c^p - d^p)/m$ where n and m are the number of intervals divide the sides of the rectangular domain (that is $\xi_i = a^p + i\Delta_\xi$, $\eta_j = c^p + j\Delta_\eta$ with $i = 0, \ldots, n$ and $j = 0, \ldots, m$). We will denote by $\overline{\Omega}^p_d$ all the points of the lattice belonging to $\overline{\Omega}^p$, by Ω^p_d all the internal points and by $\partial\Omega^p_d$ all the boundary points.

5.1 Finite Difference

Forward Up-Wind Scheme. In order to introduce a finite difference numerical scheme which does not need to consider a particular direction of the vector field b in order to be well defined, let us consider the following implicit up-wind scheme:

$$b^1_{i,j}\frac{Z^F_{i+1,j} - Z^F_{i-1,j}}{2\Delta_\xi} + b^2_{i,j}\frac{Z^F_{i,j+1} - Z^F_{i,j-1}}{2\Delta_\eta} + s_{i,j}Z^F_{i,j} = |b^1_{i,j}|\frac{Z^F_{i+1,j} - 2Z^F_{i,j} + Z^F_{i-1,j}}{2\Delta_\xi} + |b^2_{i,j}|\frac{Z^F_{i,j+1} - 2Z^F_{i,j} + Z^F_{i,j-1}}{2\Delta_\eta} \quad (9)$$

for $i = 1, \ldots, n-1$ and $j = 1, \ldots, m-1$. The artificial diffusion introduced in the right side of (9) allows to follow the vector field b considering the most appropriate discretization for the fist derivative in order to follow the characteristic lines ([22,23]).

The computation of Z^F consists of solve a global linear system where all the internal point of the grid are included. This means that the dimension of the system is $[(n-1)(m-1)] \times [(n-1)(m-1)]$. In order make understandable how we compute the matrix, we rewrite the (9) as follow:

$$Z^F_{i+1,j}\left(\frac{b^1_{i,j} - |b^1_{i,j}|}{2\Delta_\xi}\right) - Z^F_{i-1,j}\left(\frac{b^1_{i,j} + |b^1_{i,j}|}{2\Delta_\xi}\right) + Z^F_{i,j}\left(\frac{|b^1_{i,j}|}{\Delta_\xi} + \frac{|b^2_{i,j}|}{\Delta_\eta} + s_{i,j}\right) + Z^F_{i,j+1}\left(\frac{b^2_{i,j} - |b^2_{i,j}|}{2\Delta_\eta}\right) - Z^F_{i,j-1}\left(\frac{b^2_{i,j} + |b^2_{i,j}|}{2\Delta_\eta}\right) = 0. \quad (10)$$

This numerical scheme works forward with respect to the characteristics direction. This means that the information propagates starting from the inflow side of the boundary. In the numerical test are presented also results about the backward up-wind scheme.

5.2 Semi-lagrangian Discretization

A second numerical approach that permits to the solve equation (8) miming the propagation of the information along the characteristics is the following

semi-Lagrangian scheme. We pass then to consider the following equivalent equation obtained dividing the two sides of (8) by the norm of $b(\xi,\eta)$:

$$\nabla_\alpha z(\xi,\eta) + \frac{s(\xi,\eta)}{|b(\xi,\eta)|} z(\xi,\eta) = 0, \quad \forall (\xi,\eta) \in \Omega^p \tag{11}$$

with $\alpha(\xi,\eta) = \frac{b(\xi,\eta)}{|b(\xi,\eta)|}$.

We observe that the division by $|b(\xi,\eta)|$ doesn't involve any kind of difficulties for the numerical scheme (Lemma 1). Now, considering the definition of directional derivative, we can write:

$$\frac{z(\xi + h\alpha_1(\xi,\eta), \eta + h\alpha_2(\xi,\eta)) - z(\xi,\eta)}{h} + \frac{s(\xi,\eta)}{|b(\xi,\eta)|} z(\xi,\eta) \simeq 0, \quad \forall (\xi,\eta) \in \Omega^p \tag{12}$$

Considering a uniform discretization $\overline{\Omega}^p_d$ as in the previous section, we can finally write the semi-Lagrangian schemes:

$$z^{n+1}_{i,j} = z^n(\xi_i + h\alpha_1(\xi_i,\eta_j), \eta_j + h\alpha_2(\xi_i,\eta_j)) \frac{|b_{i,j}|}{|b_{i,j}| - hs_{i,j}} \quad \forall (\xi_i,\eta_j) \in \Omega^p_d \tag{13}$$

where $z^n(\xi_i,\eta_j) = z^n_{i,j}$ and $z^{n+1}(\xi_i,\eta_j) = z^{n+1}_{i,j}$ defined only on the grid nodes. In order to include the boundary condition on the scheme we assign an initial function $z^0_{i,j}$, such that $z^0(\xi_i,\eta_j) = g(\xi_i,\eta_j)\ \forall (\xi_i,\eta_j) \in \partial\Omega^p_d$.

This numerical scheme works backward with respect to the direction of the characteristics. This means that it will need of the boundary data on the outflow part of $\partial\Omega$. Also for this semi-Lagrangian scheme the forward version has been developed and the results are presented in the next section.

6 Numerical Tests

This section describes the experiments conducted with the proposed numerical schemes: the Semi-Lagrangian and the up-wind finite difference scheme, each in its forward and backward formulation.

For the numerical tests we utilized three surfaces (see Fig. 2), each with a different geometrical and analytical characteristics.

For each of these surfaces, we computed its perspective image under two light source directions according to the procedure described by Tankus et al. [4] (Fig. 3). We used a constant focal length $f = 1$ for all images. In all numerical tests, the albedo was set to $\rho = 1$ in all image domains except for the dark stripe in each of the images (see Fig. 3), where it was set to $\rho = 0.5$. We repeated the experiements with images of several sizes: 100×100, 200×200, 400×400, and 800×800 pixels.

Reconstruction by the suggested perspective Semi-Lagrangian and up-wind schemes is highly accurate, as demonstrated in Fig. 5.

We compared the suggested numerical methods for solving the perspective photometric stereo problem with the ones suggested by Mecca and Falcone [16]

for solving the equivalent orthographic photometric stereo problem. Whereas the orthographic methods converge and yield an accurate reconstruction of the aforementioned surfaces when images were generated by orthographic projection [16], the accuracy is compromised when required to reconstruct images generated by a more realistic photographic process: perspective projection (Fig. 4). The suggested perspective methods, on the other hand, faithfully reconstructed all surfaces, despite the irregularities (Fig. 5).

Three error measures comparing the reconstructed and true surfaces are provided for each scheme: the L^∞ norm in the perspective coordinate system $(\xi, \eta, z(\xi, \eta))$, root mean square error (RMSE) in the perspective coordinate system $(\xi, \eta, z(\xi, \eta))$, and RMSE in real-world coordinate system $(x, y, \hat{z}(x, y))$ (perspective: tables 1 and 2; orthographic: table 3). The L^∞ norm allows us to examine the convergence rate of the numerical schemes, showing that both perspective schemes converges linearly (i.e., with order 1), because doubling the number of grid nodes halves the error. The orthographic method did not converge on the surfaces examined. The RMSE measure, on the other hand, has the same units as surface depth, and therefore quantifies the mean error with respect to the original surface. This measure can be easily compared to the aforementioned ranges of $\hat{z}(x, y)$ values. The RMSE of the perspective reconstruction is an order of magnitude smaller than that of the orthographic reconstruction (cf. Tables 1 and 2 with Table 3).

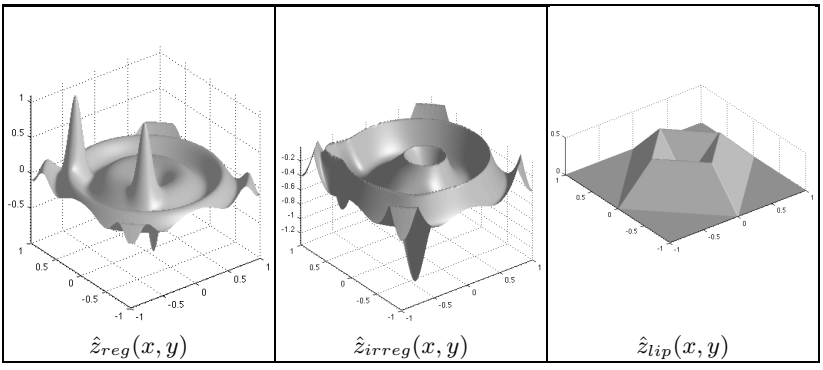


Fig. 2. Set of surfaces used for the numerical tests, each with different geometrical and analytical characteristics

In addition, we ran the algorithms on real-life images. Two pictures of Beethoven's bust were inputs to the backward and forward semi-Lagrangian schemes (Fig. 6). The backward reconstruction (Fig. 6c) emphasizes the reconstructed lips, right eye, hair and scarf, whereas the forward one (Fig. 6d), the three dimensional reconstructed nose with two distinguishable nostrils, left eye and shirt. Some inaccurate folds in the reconstructions seem to result from inaccurate boundary conditions and inaccurate measurement of camera parameters (focal length), leading to some accumulation of error.

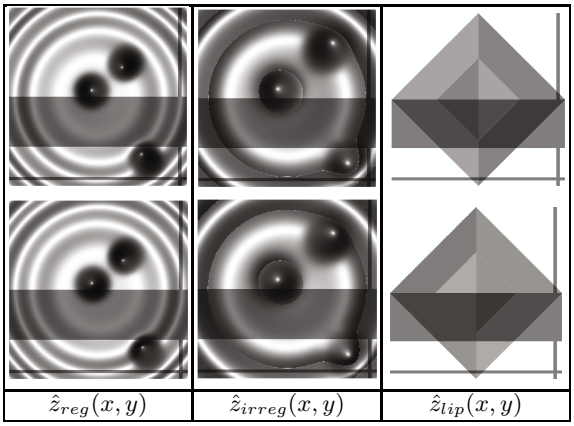


Fig. 3. Perspective images of the respective surfaces of Fig. 2, used as inputs to the algorithms. The light source directions, expressed in spherical coordinates $\omega = (\sin(\varphi)\cos(\theta), \sin(\varphi)\sin(\theta), \cos(\varphi))$, are $\varphi_1 = 0.1 + \pi$, $\theta_1 = 0.0$ for I_1 (in the first row) and $\varphi_2 = 0.1 + \pi$, $\theta_2 = \frac{3\pi}{4}$ for I_2 (in the second row). The albedo on the dark stripe of each image is $\rho = 0.5$; otherwise, $\rho = 1$.

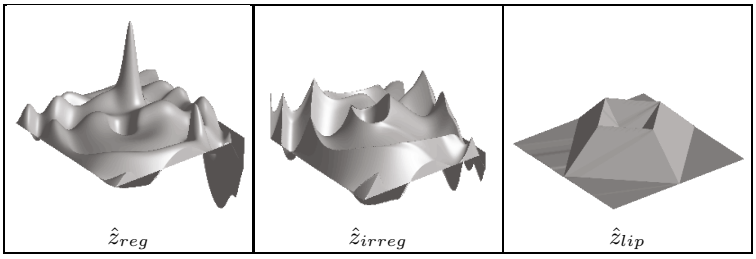


Fig. 4. Reconstruction by the orthographic backward semi-Lagrangian scheme presented in [16,17] using the input images of Fig. 3 (for original surfaces see Fig. 2). The reconstruction is inaccurate. We present the backward semi-Lagrangian reconstruction as it produced the best result among orthographic methods.

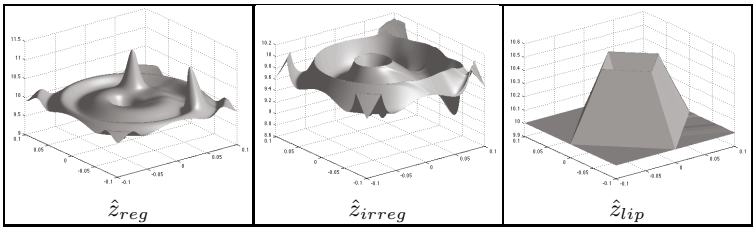


Fig. 5. Reconstruction by the proposed perspective backward semi-Lagrangian scheme, using the input images of Fig. 3. The surfaces are flipped compared to the original ones (Fig. 2) because of the perspective projection. The original surfaces were faithfully recovered.

Table 1. Convergence and accuracy of the forward numerical schemes for each surface of Fig. 2. For each surface we examined images of size $\Delta \times \Delta$ pixels, and computed three error measures: the L^∞ norm in the perspective coordinate system, RMSE in the perspective system, and RMSE in real-world coordinate system. The L^∞ norm shows convergence is linear (i.e., order 1). The RMSE measures quantify the accurate reconstruction with respect to the original surface.

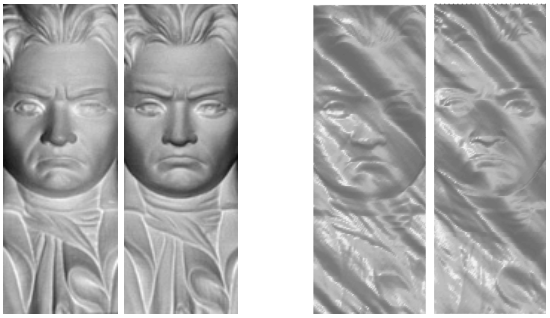
		semi-lag Forward			up-wind Forward		
	Δ	L^∞	MSE-persp	MSE-real	L^∞	MSE-persp	MSE-real
$\hat{z}_{reg}$	100	7.582×10^{-1}	0.079965	0.08006	6.780×10^{-1}	0.073121	0.073157
	200	3.543×10^{-1}	0.048369	0.04849	3.245×10^{-1}	0.046625	0.046775
	400	1.733×10^{-1}	0.027498	0.027577	1.631×10^{-1}	0.02839	0.028495
	800	8.567×10^{-2}	0.014932	0.014977	8.121×10^{-2}	0.016325	0.016389
$\hat{z}_{irreg}$	100	6.726×10^{-1}	0.11174	0.10957	4.693×10^{-1}	0.11507	0.11299
	200	4.977×10^{-1}	0.067081	0.068095	3.925×10^{-1}	0.078578	0.080503
	400	3.381×10^{-1}	0.037985	0.039874	2.664×10^{-1}	0.051528	0.054701
	800	2.174×10^{-1}	0.020888	0.02254	1.590×10^{-1}	0.035682	0.038287
$\hat{z}_{lip}$	100	1.136×10^{-1}	0.0037728	0.003863	1.165×10^{-1}	0.0036506	0.0037189
	200	5.723×10^{-2}	0.001627	0.0016577	6.459×10^{-2}	0.0016576	0.001682
	400	2.681×10^{-2}	0.0010702	0.0011037	3.069×10^{-2}	0.0010969	0.0011326
	800	1.280×10^{-2}	0.00048774	0.00049765	1.531×10^{-2}	0.00050909	0.00051989

Table 2. Convergence and accuracy of the backward numerical schemes for each surface of Fig. 2. The table is organized similarly to Table 1. The rate of convergence of the backward algorithm is the same as of the forward schemes (order 1). Accurate reconstruction is also achieved by the backward schemes.

		semi-lag Backward			up-wind Backward		
	Δ	L^∞	MSE-persp	MSE-real	L^∞	MSE-persp	MSE-real
$\hat{z}_{reg}$	100	7.582×10^{-1}	0.024478	0.025474	2.399×10^{-1}	0.0094924	0.011658
	200	1.789×10^{-1}	0.013978	0.014514	9.918×10^{-2}	0.0061304	0.007485
	400	8.494×10^{-2}	0.007591	0.0078847	4.662×10^{-2}	0.003698	0.0045026
	800	4.113×10^{-2}	0.003997	0.0041618	2.279×10^{-2}	0.0021912	0.0026803
$\hat{z}_{irreg}$	100	2.929×10^{-1}	0.040556	0.042861	1.673×10^{-1}	0.029009	0.028976
	200	2.952×10^{-1}	0.020974	0.023794	1.025×10^{-1}	0.019232	0.019162
	400	2.014×10^{-1}	0.012261	0.014336	8.878×10^{-2}	0.015983	0.016061
	800	1.697×10^{-1}	0.0072217	0.0086446	8.930×10^{-2}	0.014743	0.015029
$\hat{z}_{lip}$	100	1.136×10^{-1}	0.014512	0.015132	1.655×10^{-2}	0.014545	0.015172
	200	2.533×10^{-2}	0.0069519	0.0072408	4.790×10^{-3}	0.0069325	0.0072236
	400	2.681×10^{-2}	0.0035051	0.0036479	5.390×10^{-3}	0.003477	0.0036198
	800	4.600×10^{-3}	0.0017174	0.0017869	1.720×10^{-3}	0.0017049	0.0017746

Table 3. Convergence and accuracy of the orthographic semi-Lagrangian scheme [16] for each surface of Fig. 2. For each surface we examined images of size $\Delta \times \Delta$ pixels, and computed three error measures: the L^∞ norm in the perspective coordinate system, RMSE in the perspective system, and RMSE in real-world coordinate system. The L^∞ norm shows the scheme does not converge. The RMSE is at least an order of magnitude larger than with the proposed method (cf. Tables. 1 and 2).

		Forward			Backward		
	Δ	L^∞	MSE-persp	MSE-real	L^∞	MSE-persp	MSE-real
$\hat{z}_{reg}$	100	9.718×10^{-1}	0.13611	0.13944	1.014	0.13751	0.14127
	200	9.712×10^{-1}	0.13817	0.14188	1.032	0.14018	0.1441
	400	9.674×10^{-1}	0.13976	0.14366	1.037	0.14209	0.14618
	800	9.660×10^{-1}	0.14085	0.14485	1.038	0.14334	0.14757
$\hat{z}_{irreg}$	100	9.759×10^{-1}	0.192	0.18963	9.585×10^{-1}	0.19158	0.19105
	200	9.772×10^{-1}	0.19725	0.19511	9.386×10^{-1}	0.19695	0.19702
	400	9.761×10^{-1}	0.20061	0.19862	9.230×10^{-1}	0.20038	0.20093
	800	9.757×10^{-1}	0.20283	0.20091	9.181×10^{-1}	0.20267	0.20352
$\hat{z}_{lip}$	100	5.641×10^{-1}	0.2042	0.2091	5.658×10^{-1}	0.20459	0.20948
	200	5.703×10^{-2}	0.20555	0.21053	5.713×10^{-2}	0.20544	0.2104
	400	5.758×10^{-2}	0.20626	0.21125	5.753×10^{-2}	0.20594	0.21091
	800	5.787×10^{-2}	0.20661	0.21162	5.792×10^{-2}	0.20621	0.21119



(a) (b) (c) (d)

Fig. 6. Reconstruction of real-life images of the Beethoven bust (a, b) by the backward (c) and forward (d) semi-Lagrangian schemes (frontal view of the rendered reconstructed surfaces). Illumination directions: (a) $\varphi = 15.1^{\circ}, \theta = 72.5^{\circ}$, (b) $\varphi = 11.5^{\circ}, \theta = 184.9^{\circ}$. Focal length: $f = 100$.

7 Conclusions

This study utilized numerical schemes commonly used in the Shape-from-Shading literature also for the 2-image photometric stereo problem under the perspective projection assumption. We proved the uniqueness of the solution in the class of Lipschitz continuous surfaces given Dirichlet boundary conditions. We then extended the two numerical methods of Mecca and Falcone [16], the up-wind finite difference scheme and the Semi-Lagrangian scheme, for the solution of the 2-image *perspective* photometric stereo problem. We compared the suggested method with that of Mecca and Falcone [16] on synthetic examples, and showed that the suggested perspective semi-Lagrangian and up-wind schemes outperformed their method. As the method of Mecca and Falcone [16,17] can also reconstruct the albedo in a manner similar to the suggested perspective one, the inaccurate orthographic reconstruction is not due to the non-constant albedo, but rather a result of the more realistic set of assumptions of a perspective projection in the proposed algorithms. We also demonstrated the ability of the our method to reconstruct real-life images. Our results thus demonstrate that numerical methods of the type common in the Shape-from-Shading literature may provide additional information for solving a perspective photometric stereo problem, as presented here for a 2-image input problem.

References

1. Horn, B.K.P.: Image intensity understanding. Artificial Intelligence 8, 201–231 (1977)
2. Woodham, R.J.: Photometric stereo: A reflectance map technique for determining surface orientation from a single view. In: Proc. SPIE Annual Technical Symposium on Image Understanding Systems and Industrial Applications, San Diego, CA, pp. 136–143 (1978)

3. Woodham, R.J.: Photometric method for determining surface orientation from multiple images. Optical Engineering 19, 139–144 (1980)
4. Tankus, A., Sochen, N., Yeshurun, Y.: Shape-from-Shading under perspective projection. International Journal of Computer Vision 63, 21–43 (2005)
5. Tankus, A., Sochen, N., Yeshurun, Y.: A new perspective [on] Shape-from-Shading. In: Proceedings of the 9th IEEE International Conference on Computer Vision, Nice, France, vol. II, pp. 862–869 (2003)
6. Tankus, A., Sochen, N., Yeshurun, Y.: Perspective Shape-from-Shading by Fast Marching. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, vol. I, pp. 43–49 (2004)
7. Tankus, A., Sochen, N., Yeshurun, Y.: Reconstruction of medical images by perspective Shape-from-Shading. In: Proceedings of the International Conference on Pattern Recognition, Cambridge, UK, vol. 3, pp. 778–781 (2004)
8. Prados, E., Faugeras, O.D.: "Perspective shape from shading" and viscosity solutions. In: ICCV, pp. 826–831. IEEE Computer Society (2003)
9. Prados, E., Soatto, S.: Fast Marching Method for Generic Shape from Shading. In: Paragios, N., Faugeras, O., Chan, T., Schnörr, C. (eds.) VLSM 2005. LNCS, vol. 3752, pp. 320–331. Springer, Heidelberg (2005)
10. Courteille, F., Crouzil, A., Durou, J.D., Gurdjos, P.: Towards shape from shading under realistic photographic conditions. In: ICPR (2), pp. 277–280 (2004)
11. Argyriou, V., Petrou, M., Hawkes, P.W.: Chapter 1 photometric stereo: An overview. In: Advances in Imaging and Electron Physics, vol. 156, pp. 1–54. Elsevier (2009)
12. Okatani, T., Deguchi, K.: On uniqueness of solutions of the three-light-source photometric stereo: Conditions on illumination configuration and surface reflectance. CVIU 81, 211–226 (2001)
13. Shashua, A.: On photometric issues in 3D visual recognition from a single 2D image. International Journal of Computer Vision 21, 99–122 (1997)
14. Basri, R., Jacobs, D., Kemelmacher, I.: Photometric stereo with general, unknown lighting. International Journal of Computer Vision 72, 239–257 (2007)
15. Onn, R., Bruckstein, A.M.: Integrability Disambiguates Surface Recovery in Two-Image Photometric Stereo. International Journal of Computer Vision 5, 105–113 (1990)
16. Mecca, R., Falcone, M.: Uniqueness and approximation of a photometric shape-from-shading model. SIAM Journal on Imaging Sciences (2012) (submitted)
17. Mecca, R.: Uniqueness for shape from shading via photometric stereo technique. In: Macq, B., Schelkens, P. (eds.) IEEE ICIP, pp. 2933–2936 (2011)
18. Kozera, R.: Existence and uniqueness in photometric stereo. Applied Mathematics and Computation 44, 103 (1991)
19. Tankus, A., Kiryati, N.: Photometric stereo under perspective projection. In: Proceedings of the Tenth International Conference on Computer Vision, Beijing, China (2005)
20. Yoon, K.J., Prados, E., Sturm, P.: Generic Scene Recovery using Multiple Images. In: Tai, X.-C., Mørken, K., Lysaker, M., Lie, K.-A. (eds.) SSVM 2009. LNCS, vol. 5567, pp. 745–757. Springer, Heidelberg (2009)
21. Tankus, A., Sochen, N.A., Yeshurun, Y.: Shape-from-shading under perspective projection. International Journal of Computer Vision 63(1), 21–43 (2005)
22. Quarteroni, A., Valli, A.: Numerical Approximation of Partial Differential Equations. Springer (1994)
23. Strickwerda, J.: Finite Difference Schemes and PDE. Wadsworth Brooks/Cole (1989)

Stable Two View Reconstruction Using the Six-Point Algorithm

Kazuki Nozawa, Akihiko Torii, and Masatoshi Okutomi

Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo, 152-8550, Japan

Abstract. We propose a practical scheme for selecting a pair of images which can be a good initial seed for incremental SfM to accomplish a feasible reconstruction from input images with no external camera information such as EXIF. The key idea is the effective use of the 6-point algorithm by detecting infeasible pairs of images due to the degenerate configurations as well as the other conditions. We deeply analyze all the degenerate configurations of the 6-point algorithm and derive the algorithms for detecting image pairs fallen into those degenerate configurations. Further, we implement an efficient pipeline for selecting the initial pair, which can be easily plugged into the standard incremental SfM systems. Our experimental results on synthetic and real data show that our algorithms successfully detect and reject the pairs of images which are infeasible for 3D reconstruction. Further, we demonstrate 3D reconstruction by plugging our infeasible pair detection algorithm into the standard SfM pipeline.

1 Introduction

Incremental Structure-from-Motion (SfM) has achieved great successes for 3D reconstruction from photo collections [23] as well as sequential images [2], even for extremely large scale [1,7]. The resulted camera poses and scene structures (sparse 3D point clouds) are used for various applications, e.g. virtual navigation [17], camera localization [16,21], and dense reconstruction [8,3].

Since typical incremental SfM computes camera motions and scene structures by a seed-and-grow manner, it is critical to have an accurate and a stable initial seed reconstruction from a pair or a tuple of images. The initial seeds are determined by evaluating several conditions obtained from the results of pairwise image matching. For example, the commonly used conditions are quality of feature correspondences, i.e. number of matched features, and the geometric relationship among cameras and scene structures [23,10]. In this paper, we focus on the use of a pair of images as the initial seed of SfM in order to keep the simplicity and generality of the pipeline in contrast to [12].

For the pair of images selected as the initial seed, two-view reconstruction can be performed by using the 5-point algorithm [18] combining with RANSAC [6] (or its variants [4,19,20]). One of the advantages in this technique is that it has higher probability to hit a hypothesis not contaminated by outliers, i.e. robust

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 122–135, 2013.

against mismatches in feature correspondences, than the 7- or the 8-point algorithm [11] since it requires fewer samples for computing each hypothesis. Another advantage is that the 5-point algorithm itself has only one degenerate configuration of cameras, which is pure rotation, thanks to the minimal computation of the essential matrix that encodes relative rotation and translation only. The natural drawback of the 5-point algorithm is to require camera intrinsic parameters by some other methods.

For most of the recent cameras, it is possible to assume zero skew, a known aspect ratio (set to 1), and a known optical center (center of an image) [25], in contrast, a focal length widely changes on every image by zooming. In order to obtain focal lengths, the popular SfM pipelines [22,28] use EXIF tags and camera manufacture specifications or, if such external information is unavailable, simply assume a certain preset such as a 60-degree field of view. With this approach, if the focal lengths are estimated with large errors, the quality of initial 3D reconstruction is very unpredictable: we cannot predict whether the errors might be compensated in bundle adjustment or the reconstruction could end up in a complete failure.

In this work, we choose the 6-point algorithm [24,15] which can compute the camera motion (fundamental matrix) and focal length from a six-tuple of correspondences. This is a natural extension of the 5-point algorithm [18] which solves minimal problems based on Gröbner basis [18] or polynomial eigenvalue problem [15]. Even though the 6-point algorithm can give the focal length estimate with only one additional correspondence w.r.t. the 5-point case, it has not been spotlighted since it requires careful treatments to degenerate configurations of camera pairs and scenes. All of the image pair becomes degenerated in some particular cases, i.e. a poster on single planar wall and nothing else is taken ("planar scene", described in Section 2.1), turntable sequences taken under the condition that the optical axis is intersected to the rotation axis of the turntable ("equidistant intersecting optical axes", Section 2.3), video sequences acquired by a vehicle-mounted camera running with no turn ("parallel axes", Section 2.4). In those cases, stable reconstruction cannot be achieved by starting an initial reconstruction using the focal length and the relative camera motion obtained by the 6-point algorithm. Torii et. al. [27] tackled this problem by detecting pairs of images acquired with degenerate configurations by adopting singular value ratio test (Section 3).

In this paper, we propose a practical scheme for selecting a pair of images which can be a good initial seed for incremental SfM to accomplish a feasible reconstruction from input images with no external camera information. The key idea is the effective use of the 6-point algorithm which gives camera motion (fundamental matrix) and focal length by efficiently detecting infeasible pairs of images due to the degenerate configurations (Section 2) as well as the other conditions (Section 3). The main contribution w.r.t. the most related work [27] is that we deeply analyze all the degenerate configurations of the 6-point algorithm, of which two are not considered in [27], and derive the algorithms for detecting image pairs fallen into those degenerate configurations. Further, we implement

an efficient pipeline for selecting the initial pair, which can be easily plugged into the standard incremental SfM systems.

Related Works: Although most of the related works are already described, we summarize a few more strongly related works. Kanatani et.al. [13,14] presented a method for computing focal lengths from a fundamental matrix computed by the 7- or 8-point algorithm in a closed form [13] and further extended its stability including the detection of degeneracy conditions in their case [14]. The theory and algorithms presented in [14] are concrete but unfortunately, it is hard to assess practical performances in challenging dataset due to the experimental validations with limited examples.

Gherardi and Fusiello [9] proposed a practical autocalibration approach which repeats update of an initial guess of intrinsic parameters of an image pair by searching an inherently bounded parameter space and by scoring likelihood of the estimated intrinsic parameters using the other cameras. Due to its nature of estimating all intrinsic parameters, the stable estimation can be achieved with more than two cameras as they demonstrated in the experiments.

In this paper, we focus on the use of the 6-point algorithm for estimating focal length from a pair of images according to its efficiency when bundled with a RANSAC scheme and its simplicity to plug into incremental SfM pipelines.

2 Detection of Degenerated Image Pairs

In this chapter, we describe four types of degeneracy underlying the computation of a relative camera motion and a focal length from an image pair using the 6-point algorithm. One is due to the degenerate scene and the others are to the degenerate camera configurations.

In most of practical situations, it is hard to classify whether the images are degenerated by using the inlier ratio resulted from RANSAC. This is because RANSAC returns the best hypothesis arbitrary fitting to an inlier set even for degenerate configurations according to noisy measurements. Even worse, the hypothesis with degeneracy often gives high inlier ratio. Therefore, we develop the algorithms which are optimized for detecting degeneracy.

2.1 Planar Scene

The degenerated scene is "planar scene"; all feature points seen by two cameras lie on a plane in a 3D space, i.e. coplanar. The 6-point algorithm can neither obtain a valid fundamental matrix nor a focal length when the six-tuple corresponding is coplanar. Planar scene is also degeneracy for the 7- and the 8-point algorithms, so that the detection algorithm is well known [5]. We can detect this degeneracy by explicitly computing a homography from the corresponding feature points.

As the similar way proposed in DEGENSAC [5], the image pairs degenerated by planar scene is quickly detected by verifying whether a six-tuple of points used

for computing the fundamental matrix is related by a homography. In detail, we compute a homography using a four-tuple out of the six and verify whether the remaining two fit in the homography. Instead of testing this degeneracy for samples on each RANSAC loop as [5], we test 15 homographies obtained from the six-tuple resulted by RANSAC. The important idea in DEGENSAC [5] is to find the stable hypothesis even from the scene dominated by a plane using the plane-plus-parallax. In contrast, we simply reject such an image pair by assuming we have sufficiently large dataset and better seeds exist for the following SfM.

Additionally, we check whether the scene is dominated by a single homography by using the standard 4-point RANSAC for all of the input correspondences. If the scene is actually planar, the number of inliers resulted by RANSAC on a homography hypothesis increase. Therefore, we can detect the degeneracy by comparing the number of inliers which support the fundamental matrix and the homography. This is computationally costly but more robust to the noise than checking six-tuple of correspondences only.

2.2 Pure Rotation

The most simple degenerate camera configuration is "pure rotation"; two cameras are configured without translation. In this case, the 6-point algorithm fails to estimate both the fundamental matrix and the focal length. This is also detected by using the same algorithm based on homography as described in Section 2.1. Note that this is also degeneracy for the 5-point algorithm as well as the 7- and the 8-point algorithms.

2.3 Equidistant Intersecting Optical Axes

The third degeneracy is that the optical axes of the two cameras intersect and the two distances between the camera center and the intersection point are the same. This configuration can be interpreted as cameras lie on a sphere and their optical axes are oriented to the center of the sphere. This degeneracy often occurs in practice, for instance, if images taken by a fixed camera while a target object moved on a turntable and the optical axis intersected to the rotation axis of the turntable, the configuration falls into this degeneracy.

Further, it is impossible to obtain the correct focal length but the estimated fundamental matrix is still valid [24].[1] Therefore, we can detect this configuration by evaluating the projection of the optical axes. The detection is composed by two steps of evaluating the necessary conditions: (i) detection of coplanar optical axes; (ii) detection of isosceles triangle composed of the camera centers and the intersection point.

Step1: Detecting Coplanar Optical Axes. One of the necessary conditions of this degeneracy is that the optical axes must lie on the same plane in 3D space.

[1] This degenerate configuration gives 3D scene reconstruction by arbitrary but common focal length due to the remaining projective ambiguity.

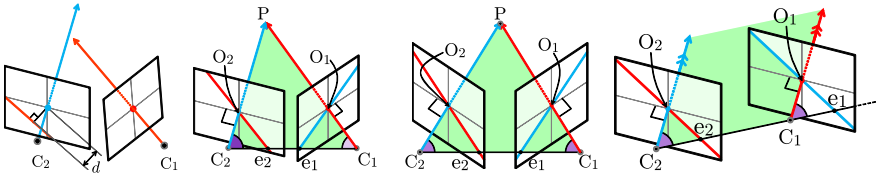


Fig. 1. (a), **(b)**: The optical axes are not coplanar or intersect at the difference distance from the cameras. The 6-point algorithm can estimate the focal length. **(c)**, **(d)**: The optical axes intersect at the same distance from the cameras or are parallel so that focal length cannot be recovered.

We consider a pair of cameras; camera 1 and camera 2. On the image plane of camera 2, we draw the epipolar line corresponding to the image center of camera 1. The epipolar line is represented as:

$$\begin{pmatrix} x & y & 1 \end{pmatrix} \boldsymbol{F} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \boldsymbol{F}_{13}x + \boldsymbol{F}_{23}y + \boldsymbol{F}_{33} = 0 \tag{1}$$

This line passes through the image center only when the optical axes intersect or parallel (Fig. 1 (b), (c), (d)). It is possible to detect if the image pair has intersectional or parallel optical axes by measuring the distance d between the image center and the epipolar line corresponding to the image center of the other,

$$d = \frac{|\boldsymbol{F}_{33}|}{\sqrt{\boldsymbol{F}_{13}^2 + \boldsymbol{F}_{23}^2}} \tag{2}$$

The epipolar line on the other image is also verified. This detection is similar to the detection described in [14].

Step2: Detecting Isosceles Triangle. There is another necessary condition for the degeneracy of equidistant intersecting optical axes (Fig. 1 (c)). If the optical axes intersect equidistantly, the triangle composed of the camera centers and the intersection point is isosceles. This degeneracy could be detected by evaluating whether the triangle is isosceles or not. However, of course, the angles between the optical axes and the epipole cannot be computed in Euclidean space because the correct focal lengths are not estimated from the 6-point algorithm due to the degeneracy.

This degeneracy can be still detected by using the following geometrical relationships assuming an unknown but a common focal length. Let us consider the two triangles $\triangle(\mathrm{O}_1, \mathrm{C}_1, \mathrm{e}_1)$ and $\triangle(\mathrm{O}_2, \mathrm{C}_2, \mathrm{e}_2)$, where C_1 and C_2 are the camera centers, O_1 and O_2 are the image centers, e_1 and e_2 are the epipoles, respectively. Note that the angles $\angle(\mathrm{C}_1, \mathrm{O}_1, \mathrm{e}_1) = \angle(\mathrm{C}_2, \mathrm{O}_2, \mathrm{e}_2) = \pi/2$ because they are the intersection of optical axes and the image planes. Further, the lengths

O_1C_1 and O_2C_2 are equal since we assumed a common focal length. Consequently, if the lengths O_1e_1 and O_2e_2 are equal, the two triangles $\triangle(O_1, C_1, e_1)$ and $\triangle(O_2, C_2, e_2)$ are congruent, and the angles $\angle(O_1, C_1, e_1)$ and $\angle(O_2, C_2, e_2)$ are the same. Thus, the triangle $\triangle(P, C_1, C_2)$ is isosceles. Since the distances O_1e_1 and O_2e_2 can be computed on the image planes, it is possible to use these distances for detecting equidistant intersecting optical axes.

2.4 Parallel Optical Axes

The last case is when two optical axes are parallel. Unfortunately, this configuration also occurs frequently since the camera motion under pure translation plus rotation around the optical axis is included in this degeneracy.

As in the case of the intersectional and equidistant optical axes, focal length cannot be recovered; on the other hand, the estimated fundamental matrix is still valid. When the optical axes are parallel, they are coplanar and the corresponding angles consisted by the two optical axes and the epipole are equivalent (Fig. 1 (d)). Note that the parallel optical axes can be considered as intersecting at the point at infinity. Therefore, we can detect them using the same algorithm as detecting the intersectional and equidistant optical axes.

3 Detection of Invalid Essential Matrices

For the practical use of the 6-point algorithm with RANSAC, the degenerate configurations of cameras and scenes are not the only reason for contaminating the estimation of a relative camera motion and a focal length. The estimation fails if the measurements of features are too noisy or if some pairs of images with different focal lengths are included in the image set. Torii et al. [27] found that the quality of the estimation of the algorithm is correlated with the ratio of the two non-zero singular values of the essential matrix and thus the ratio can be a criterion of evaluating the validity of estimation.

Singular Value Test (SVT). The two non-zero singular values of an essential matrix is ideally equivalent. The 6-point algorithm uses this property as one of the constraint for minimal solution, so that the ratio of two non-zero singular values of essential matrix decomposed from the fundamental matrix $\boldsymbol{F}_{6pt}$ using the focal length f_{6pt} is always one. Here, if the estimation of a fundamental matrix and a focal length is successful and the inlier set $\text{inl}_{6\text{pt}}$ supporting them is geometrically correct, the fundamental matrix $\boldsymbol{F}_{LS}$ re-estimated from $\text{inl}_{6\text{pt}}$ using least squares should be valid. Then, the essential matrix $\boldsymbol{E}_{LS}$ obtained by factorizing the fundamental matrix $\boldsymbol{F}_{LS}$ using f_{6pt} should have the ratio of two non-zero singular values s_1 and s_2 to be one. We use the singular value ratio (SVR) $\tau = s_2/s_1$, where $s_1 \geq s_2$, of $\boldsymbol{E}_{LS}$ for classifying if the estimation is valid.

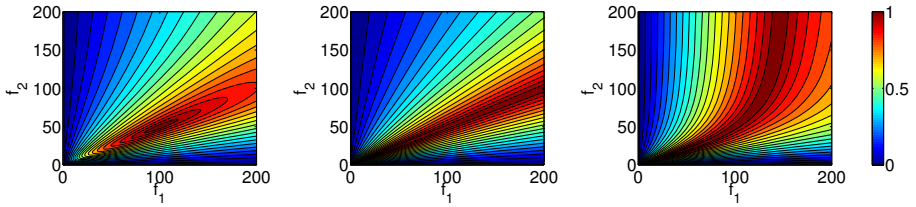


Fig. 2. The SVR of essential matrix decomposed from valid fundamental matrix using various focal lengths. The ground truth is $f_{t1} = 100, f_{t2} = 50$. (a) The camera configuration is non-degenerate. SVR is high only when it is computed with the correct focal lengths. (b) The optical axes intersect at the same distance from the cameras or are parallel. SVR can be high when $f_1 : f_2$ is equivalent to $f_{t1} : f_{t2}$. The algorithm described in Section 2.3 can detect this case. (c) The optical axes intersect at the different distances from the cameras. SVR can be high even though the focal length estimation via the 6-point algorithm fails.

Limitation of the SVT. For the scene and the camera configuration with non-degeneracy, SVR is close to one only if the fundamental matrix is factorized using correct focal lengths (Fig. 2 **(a)**). Therefore, SVT can detect the invalid estimation of the 6-point algorithm due to the measurement noise or input of image pairs with different focal lengths.

Meanwhile, SVT is confused by two conditions: (i) the optical axes of the cameras are parallel or intersect equidistantly; (ii) the optical axes intersect at the different distance from the cameras.

In the case of (i), note that this is degenerate condition described in Section 2.3 and Section 2.4, SVR is high when $f_1 : f_2$ is correctly estimated as well as f_1 and f_2 respectively are (Fig. 2 **(b)**). When the genuine focal lengths are equivalent each other, SVT cannot detect the failure of focal length estimation because the solution of the 6-point algorithm always satisfies $f_1 : f_2 = f_{6pt} : f_{6pt} (= 1 : 1)$ and the SVR is one. However, this case can be detected by checking the projections of the optical axes as Section 2.3 because fundamental matrix is estimated correctly when the images have a common focal length. Besides, if the focal lengths are different, the solution is always to be $f_1 : f_2 \neq f_{6pt} : f_{6pt}$, so the SVR decrease.

The problematic case is (ii): SVR is high with certain combination of the focal lengths (Fig. 2 **(c)**). This combination is associated by the focal lengths and the distances to the intersection point. In this situation, we cannot evaluate the validity of estimation using SVR. Moreover, the 6-pooint algorithm can estimate neither the focal length nor the fundamental matrix from the image pair with various focal lengths. We cannot detect this case by the detectors we described above without the fundamental matrix and the reliability of SVT, so that we need another algorithm to detect it.

In order to detect the case (ii), we re-estimate fundamental matrix via linear 8-point algorithm with RANSACing all the tentative matches. All the image pairs which deteriorate the 8-point algorithm are detected by the detector with

homography, so that the estimation will be successful. We can detect the case by checking the co-planarity of the optical axes in Section 2.3 using the correct fundamental matrix computed by the 8-point algorithm.

4 The Pipelines of Initial Image Pair Selection

We consider two types of pipelines to select an initial seed for stable initial reconstruction. The first pipeline assumes the input images have a fixed focal length. The second one assumes more general situation, i.e. the input images with various focal lengths. We combine all the detectors described above while taking into account the efficiency.

4.1 A Fixed Focal Length

1. Pick a pair of input images.
2. Estimate the focal length f_6 and the fundamental matrix $\boldsymbol{F}_6$ using the 6-point algorithm with RANSAC and obtain the set of inliers inl_6 which support $\boldsymbol{F}_6$.
3. If the number of inliers inl_6 is less than a threshold (30 in the experiments) the pair is rejected from initial pair candidate then go to step1. This is the detection of image pairs which have no or small common field of view.
4. Compute a homography $\boldsymbol{H}_6$ from the four-tuple out of the 6 points used for computing $\boldsymbol{F}_6$ and f_6.
5. If the six-tuple of points is coplanar, then go to step1.
6. Re-estimate the fundamental matrix $\boldsymbol{F}_{LS}$ from inl_6 using the least squares.
7. Draw the epipolar line corresponding to the center of the other image.
8. If the distance is smaller than the threshold (5% of image width in the experiments), they have coplanar optical axes, move to step9, otherwise move to step1.
9. Calculate the distances between the image center and the epipole.
10. If the distances are almost the same, reject the pair and go to step1.
11. Compute SVR from the essential matrix $\boldsymbol{E}_{LS}$ obtained from $\boldsymbol{F}_{LS}$ and f_6.
12. If the SVR is smaller than threshold (0.98 in the experiments) reject the pair and move to step1.
13. Decompose $\boldsymbol{E}_{LS}$ into a rotation matrix and a translation vector using the result of SVD.
14. Triangulate all the correspondents which support the estimation.
15. Compute the dominant apical angle [26] of the reconstruction points.
16. If the DAA is small (smaller than 0.1 deg), reject the pair and go to step1. This is because a point whose apical angle is extremely small tends to magnify the noise.
17. Run four-point-sampling RANSAC and obtain the likely homography and the inlier set inl_H which support them.
18. If the $\text{inl}_\text{H} > \text{inl}_6$, reject the pair and move to step1
19. If this is the last pair, quit the procedure; if not, move back to step1

When all the image pairs have been gone through this procedure, we can start the initial reconstruction from one of the initial pair candidates. If there is no pair available for stable reconstruction with the 6-point algorithm, our pipeline successfully selects "none" as an initial seed and avoids performing meaningless reconstruction.

4.2 Various Focal Lengths

When we include images taken with different focal lengths, we have to add a few more steps to the algorithm in Section 4.1 since it is necessary to detect and to reject the image pairs with different focal lengths.

As we described in Section 3, the image pair with coplanar optical axes must be simply detected and rejected, thus the step8 on the pipeline is replaced with

8. If the distance is smaller than the threshold, reject the pair and go to step1.

so the step9 and 10 in the pipeline in Section 4.1 are skipped.

Finally, we add the following steps,

19. Re-estimate the fundamental matrix $\boldsymbol{F}_{L8}$ via linear 8-point algorithm with RANSACing all the tentative matches.
20. Re-execute the step7 and 8 using $\boldsymbol{F}_{L8}$
21. Re-execute the step11 and 12 using $\boldsymbol{F}_{L8}$
22. If this is the last pair, quit the procedure; if not, move back to step1

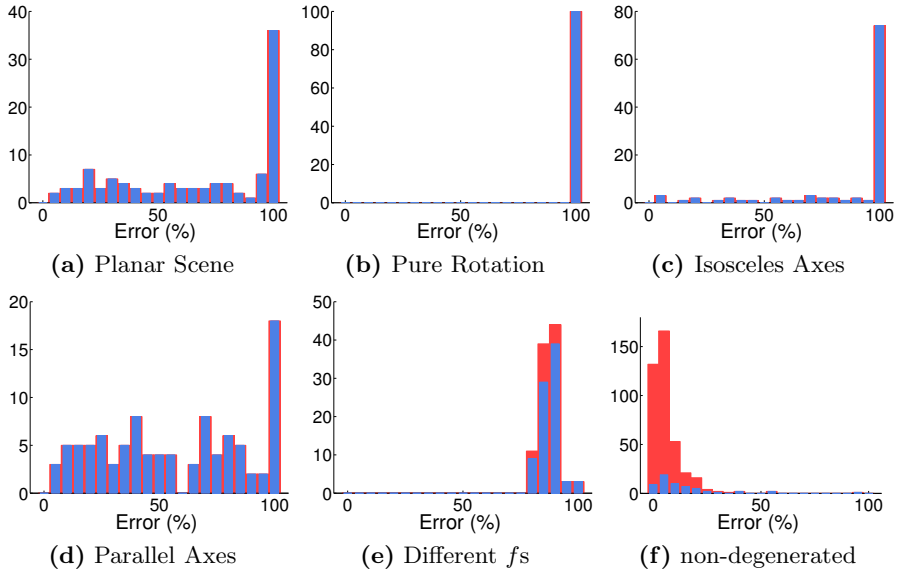
Since the additional process using the 8-point algorithm with RANSAC is costly, the pipelines are explicitly separated.

5 Experiments

5.1 Synthetic Data

We demonstrate the proposed degeneracy detection on synthetic data. First, we consider the six different cases, as shown in Fig. 3. There are four degenerated sets: (a) planar scene, (b) pure rotation, (c) isosceles axes, and (d) parallel optical axes. The other case is (e) different fs and (f) non-degenerated which is designed absolutely not to degenerate.

The image size is 2288×1520 and Gaussian noise ($\sigma = 3$) is added on the images. We reconstructed the scene with the 6-point algorithm and analyzed the "error". Here, an "error" is defined for each image pair as the average distance between the reconstructed point and the ground truth. The distance is normalized by the distance from the camera. For (a)-(e), we applied a detector which is related to the degeneracy or infeasible condition. For (f), we applied all the detectors to the non-degenerated dataset. Figure. 3 shows that the reconstruction from all the dataset but the non-degenerated one is unreliable. However, all the degenerated pairs and most of the different focal lengths pairs are detected. There are few false positives in well-reconstructed dataset, the non-degenerated one.



(a) Planar Scene **(b)** Pure Rotation **(c)** Isosceles Axes

(d) Parallel Axes **(e)** Different fs **(f)** non-degenerated

Fig. 3. The histograms of the average error of the 3D reconstruction using the focal length estimated with the 6-point algorithm. The error is defined as the distance from the reconstructed point to the original position divided by the distance from the camera as normalization. **(a)** – **(e)** Each graph represents the 100 image pairs satisfies different kinds of infeasible condition. (*Red*) all the image pairs. (*Blue*) image pairs detected as infeasible for reconstruction. **(f)** 400 non-degenerate pairs. (*Red*) all the image pairs. (*Blue*) the image pairs failed at least one examination.

5.2 Real Data

Next, we evaluate the performance of the proposed pipeline on the real photographs. We make two real image datasets. "Campus" dataset is composed of 200 pictures of a campus building with a fixed focal length, which gives 19900 image pairs. "Trevi" dataset is composed of 46 images available on Flickr, which gives 1035 pairs. We reject all the image pairs which deteriorating 3D reconstruction as we described in Section 4.

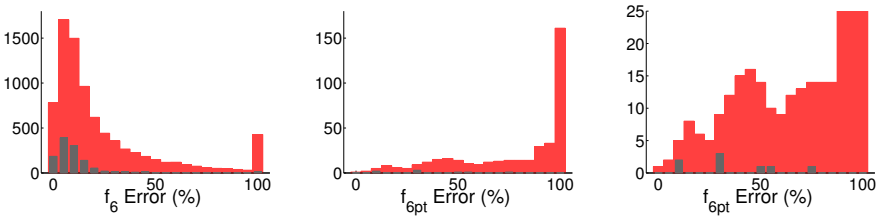
The histograms in Fig. 4 (a), (b) show the results of focal length estimation. We define the error as the difference between the estimated focal lengths and the ones obtained from the EXIF tag (by regarding as the ground truth) normalized by the ground truth. The errors of estimation from the image pairs without common field of view are meaningless so that they, few matching pairs, are rejected from the evaluation. Note that the gray histogram is made up with the image pairs which are NOT detected. There are a large number of image pairs that cannot provide the accurate focal lengths with the 6-point algorithm. Specifically, the focal length estimation from Trevi dataset is very difficult because most of them are compound different focal lengths pair. We can see that most of the image pairs which provide large error are detected. Although there

are some false positives remained, the initial seed selection is not contaminated combining with the scoring process after the removal of most of the infeasible image pairs.

We score the image pairs by

$$s = \frac{s_1 + s_2 + s_3 + s_4}{4}, \qquad \text{where} \begin{cases} s_1 = 1 - \dfrac{|\boldsymbol{m}^{\mathrm{H}}|}{|\boldsymbol{m}|} \\ s_2 = \dfrac{|\boldsymbol{m}|}{500} \\ s_3 = 1 - \dfrac{1 - SVR}{1 - 0.98} \\ s_4 = \dfrac{CH(C_i) + CH(C_j)}{A_i + A_j} \end{cases} \tag{3}$$

First three terms are described on [27]. $|\boldsymbol{m}|$ represents the number of the inlier supporting the fundamental matrix. $|\boldsymbol{m}^{\mathrm{H}}|$ denotes the number of the inlier supporting the homography obtained via RANSAC. SVR of the s_3 denotes the singular value ratio of the obtained essential matrix. s_4 refer a part of the score described in [10]. $CH(\cdot)$ is the area of the convex hull of a set of points and $C_{\cdot}$ represents the point that corresponding is confirmed. $A_{\cdot}$ is the area of the image. Then we make Bundler [22] start initial reconstruction from the pair with best score using the focal length estimated with the 6-point algorithm. We qualitatively compare the result with the reconstruction by Bundler with and without EXIF tag. The reconstructed point clouds are shown in Fig. 5 and the numerical data are on Table 1. Without EXIF, bundler obtains the focal lengths of the initial pair supposing the angle of view but sometimes, like these cases, it does not work well. The reconstruction of campus without EXIF (Fig. 5 (c)) is damaged. On the other hand, the result of Trevi without EXIF (Fig. 5 (d)) still keeps a shape of facade but it is skewed and the number of reconstructed points are much less than the other two (Figs. 5 (f), (h)). Using the 6-point algorithm with our proposed criteria, 3D reconstruction succeeds without any external information and the output is promising.

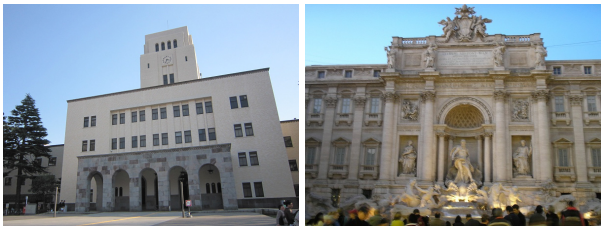


(a) **(b)** **(c)**

Fig. 4. The histogram of the error of focal length estimation which are composed of the results from (*Red*) image pairs whose enough numbers of matches are counted. (*Gray*) image pairs that pass all the examinations. (a) Campus; fixed focal length. (b) Trevi; images from Flickr. (c) is a close-up of (b) at lower frequency.

Table 1. The number of cameras whose pose is estimated, reconstructed points, and the comparison of (estimated) focal lengths of initial pair and that on the EXIF as ground truth

	Campus		Initial f	Trevi		Initial fs
	Cameras	Points	(f on EXIF)	Cameras	Points	(fs on EXIF)
Bundler (without EXIF)	200	31,217	532 (1026)	22	3,077	532 (1591) 532 (1818)
Bundler (with EXIF)	200	66,281	1026 (1026)	24	6,149	1591 (1591) 1818 (1818)
Proposed + Bundler (without EXIF)	200	66,577	1039 (1026)	24	6,227	2013 (1652) 2013 (1624)



(A) **(a)**

(B) **(C)** **(D)**

(b) **(c)** **(d)**

Fig. 5. (A): The picture of the campus building. **(a)**: The picture of Trevi Fountain downloaded from Flickr. **(B)**,**(b)**: Reconstruction by Bundler without EXIF. **(C)**,**(c)**: Reconstruction by Bundler using EXIF. **(D)**,**(d)**: Reconstruction by Bundler without EXIF starting from the image pair and using the focal length which are obtained by our proposal.

6 Conclusions

We list up all the conditions deteriorating the 6-point algorithm and proposed the criteria for detecting the entire image pairs infeasible for it; especially, in the focal length estimation, the intersection of optical axes is very critical so we showed how to detect the intersection. We showed the performance of our tests and that we can start initial reconstruction stably without any ancillary information or invalid assumptions on angle of view.

Acknowledgement. This work was partly supported by Grant-in-Aid for Scientific Research (21240015) from the Japan Society for the Promotion of Science.

References

1. Agarwal, S., Snavely, N., Simon, I., Seitz, S.M., Szeliski, R.: Building rome in a day. In: Proc. IEEE 12th Int. Computer Vision Conf., pp. 72–79 (2009)
2. Akbarzadeh, A., Frahm, J.M., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Merrell, P., Phelps, M., Sinha, S., Talton, B., Wang, L., Yang, Q., Stewénius, H., Yang, R., Welch, G., Towles, H., Nistér, D., Pollefeys, M.: Towards urban 3D reconstruction from video. In: Proc. 3DPVT (2006)
3. Chauve, A.L., Labatut, P., Pons, J.P.: Robust piecewise-planar 3d reconstruction and completion from large-scale unstructured point data. Proc. CVPR, 1261–1268 (2010)
4. Chum, O., Matas, J.: Matching with prosac: Progressive sample consensus. In: Proc. CVPR, pp. I:220–I:226 (2005)
5. Chum, O., Werner, T., Matas, J.: Two-view geometry estimation unaffected by a dominant plane. In: Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 772–779 (2005)
6. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Comm. ACM 24, 381–395 (1981)
7. Frahm, J.-M., Fite-Georgel, P., Gallup, D., Johnson, T., Raguram, R., Wu, C., Jen, Y.-H., Dunn, E., Clipp, B., Lazebnik, S., Pollefeys, M.: Building Rome on a Cloudless Day. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 368–381. Springer, Heidelberg (2010)
8. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. PAMI 32, 1362–1376 (2010)
9. Gherardi, R., Fusiello, A.: Practical Autocalibration. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 790–801. Springer, Heidelberg (2010)
10. Gherardi, R., Toldo, R., Farenzena, M., Fusiello, A.: Samantha: Towards automatic image-based model acquisition. In: Proc. Conf. Visual Media Production, CVMP, pp. 161–170 (2010)
11. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2000) ISBN: 0521623049
12. Havlena, M., Torii, A., Pajdla, T.: Efficient structure from motion by graph optimization. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 100–113. Springer, Heidelberg (2010)

13. Kanatani, K., Matsunaga, C.: Closed-form expression for focal lengths from the fundamental matrix. In: Proc. 4th Asian Conf. Computer Vision, pp. 128–133 (2000)
14. Kanatani, K., Nakatsuji, A., Sugaya, Y.: Stabilizing the focal length computation for 3-d reconstruction from two uncalibrated views. Int. J. Comput. Vision 66, 109–122 (2006)
15. Kukelova, Z., Bujnak, M., Pajdla, T.: Polynomial eigenvalue solutions to the 5-pt and 6-pt relative pose problems. In: Proc. BMVC 2008 (2008)
16. Li, Y., Snavely, N., Huttenlocher, D.P.: Location Recognition Using Prioritized Feature Matching. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 791–804. Springer, Heidelberg (2010)
17. Microsoft: Photosynth (2008), http://livelabs.com/photosynth
18. Nistér, D.: An efficient solution to the five-point relative pose problem. IEEE Transactions on Pattern Analysis and Machine Intelligence
19. of RANSAC, Y.: 26, 756–770 (2004) (2008), http://cmp.felk.cvut.cz/ransac-cvpr2006/
20. Raguram, R., Frahm, J.-M., Pollefeys, M.: A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Random Sample Consensus. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 500–513. Springer, Heidelberg (2008)
21. Sattler, T., Leibe, B., Kobbelt, L.: Fast image-based localization using direct 2d-to-3d matching. In: ICCV, pp. 667–674 (2011)
22. Snavely, N.: Bundler: Structure from motion (sfm) for unordered image collections (2008), http://phototour.cs.washington.edu/bundler/
23. Snavely, N., Seitz, S., Szeliski, R.: Modeling the world from internet photo collections. International Journal of Computer Vision 80, 189–210 (2008)
24. Stewenius, H., Nister, D., Kahl, F., Schaffalitzky, F.: A minimal solution for relative pose with unknown focal length. In: Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, CVPR 2005, vol. 2, pp. 789–794 (2005)
25. Szeliski, R.: Computer vision: algorithms and applications. Springer, New York (2010)
26. Torii, A., Havlena, M., Pajdla, T., Leibe, B.: Measuring camera translation by the dominant apical angle. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2008, 1–7 (2008)
27. Torii, A., Kukelova, Z., Bujnak, M., Pajdla, T.: The Six Point Algorithm Revisited. In: Koch, R., Huang, F. (eds.) ACCV Workshops 2010, Part II. LNCS, vol. 6469, pp. 184–193. Springer, Heidelberg (2011)
28. Wu, C.: Visualsfm: A visual structure from motion system (2011), http://www.cs.washington.edu/homes/ccwu/vsfm/

Unknown Radial Distortion Centers in Multiple View Geometry Problems

José Henrique Brito[1,2], Roland Angst[3], Kevin Köser[3], Christopher Zach[4], Pedro Branco[2], Manuel João Ferreira[2], and Marc Pollefeys[3]

[1] Instituto Politécnico do Cávado e do Ave, Barcelos, Portugal
[2] Centro Algoritmi, Universidade do Minho, Guimarães, Portugal
[3] Computer Vision and Geometry Group, ETH Zürich, Switzerland
[4] Microsoft Research, Cambridge, UK

Abstract. The radial undistortion model proposed by Fitzgibbon and the radial fundamental matrix were early steps to extend classical epipolar geometry to distorted cameras. Later minimal solvers have been proposed to find relative pose and radial distortion, given point correspondences between images. However, a big drawback of all these approaches is that they require the distortion center to be exactly known. In this paper we show how the distortion center can be absorbed into a new radial fundamental matrix. This new formulation is much more practical in reality as it allows also digital zoom, cropped images and camera-lens systems where the distortion center does not exactly coincide with the image center. In particular we start from the setting where only one of the two images contains radial distortion, analyze the structure of the particular radial fundamental matrix and show that the technique also generalizes to other linear multi-view relationships like trifocal tensor and homography. For the new radial fundamental matrix we propose different estimation algorithms from 9,10 and 11 points. We show how to extract the epipoles and prove the practical applicability on several epipolar geometry image pairs with strong distortion that - to the best of our knowledge - no other existing algorithm can handle properly.

1 Introduction

When trying to relate images, the robust estimation of the fundamental matrix based on local feature correspondences is a very powerful approach. Stochastic estimation algorithms such as RANSAC can find the correct two-view relation with high probability and at the same time distinguish inliers and outliers to the model (i.e. mismatches). However, this approach relies on the appropriateness of the model, i.e. it assumes that the images strictly obey to the pinhole camera model. In practice however, images can contain significant distortion induced by the lens (system) of a real camera. Consequently, in the literature several camera models and techniques have been proposed to model such distortion [4,7,17,1,18,5,10,3]. However, for automatic registration of images obtained from internet sources or archives, an offline camera calibration phase is not feasible. In such cases lens distortion has to be considered directly in the multi-view

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 136–149, 2013.

geometry estimation stage. This was the idea of the undistortion model proposed by Fitzgibbon [7] that has been extended to the radial fundamental matrix by Barreto and Daniilidis [1]. The assumption is that undistortion can be modeled in a radial fashion with respect to a distortion center. The main drawback in both formulations is that the distortion center must be known in advance, which we argue is not practical when images stem from sources like archives or internet photo collections. Using a wrong distortion center renders the whole concept of radial distortion meaningless, although assuming the distortion center to be at the center of the image can sometimes still be a valid approximation. However, in the case of cropped images or images taken with digital zoom no heuristics exist where to place the distortion center. Consequently, in this contribution we generalize the radial fundamental matrix (and all other multilinear multiple view relations) to unknown distortion centers. This is very analogue to the ideal pinhole case where the essential matrix was generalized to the fundamental matrix [6] that could then account for any principal point. Also in the case of the radial fundamental the dimensions of the matrix do not change once the distortion center is considered and linear algorithms require the same number of points for estimating it.

For clarity of presentation we start from the setting where only one of the two images contains radial distortion and analyze the structure of the particular radial fundamental matrix. It will turn out that a change of distortion center acts linearly on the lifted point representation, allowing to do the same generalization for other multi-view geometry relations like homograpy, trifocal tensor and so forth. We then continue to derive different estimation algorithms for our radial fundamental matrix from 9,10 and 11 points that exploit the specific algebraic structure and show how to extract the epipoles. Finally, we prove the practical applicability of the new theory on several epipolar geometry image pairs with strong distortion that - to the best of our knowledge - no other existing algorithm can handle properly.

2 Previous Work

For ideal pinhole cameras the essential matrix has been introduced by Longuet-Higgins [15] and it allowed efficient computation of the relative pose between two views. However, pre-calibration of these views was mandatory and prevented using this technique for images with unknown calibration parameters since one had to know e.g. focal length and principal point of both cameras. Much later, the introduction of the fundamental matrix [6,16] removed this restriction and allowed to work with unknown images, zoom cameras and led to a whole theory of auto-calibration from images and projective reconstruction (cf. to [11]). Practically, already the original 8-point algorithm from [15] could have been applied to the uncalibrated setting, but due to notation and for historic reasons this was not clear before the proposal of the fundamental matrix. Nowadays, the core of Longuet-Higgins algorithm is known as the 8- point algorithm for fundamental matrix estimation [11].

The fundamental matrix applies to ideal pinhole cameras, but real cameras have lenses that sometimes result in distortion of the image and, due to the shape of the lens, this distortion is typically radially-symmetric with respect to a distortion center. Many formulations exist to cope with this problem (e.g. [7,17,1,18,5,10]. According to one of the classical distortion models [4] the deformation of an undistorted point into a distorted point (as caused by the lens) is represented by a polynomial equation, but due to the nature of the distortion function it was not easily possible to estimate the inverse of the distortion directly from point correspondences. Fitzgibbon[7] has suggested to directly model the undistortion of a point rather than the distortion and argued that earlier *distortion models* were as good or bad empiric approximations to the true lens behaviour as an *undistortion model* might be. Having an undistortion model has the advantage that one can directly work with distorted coordinates, which is what is measured in an image.

However, similarly as in the derivation of the essential matrix of Longuet-Higgins, now Fitzgibbon assumed that the distortion center is known beforehand. Later, his model was reformulated into the radial fundamental matrix by Barreto and Daniilidis[1]. They proposed a linear 15 point method to estimate the matrix and recently it has been shown by Kukelova et al. [13] that this view relation can be estimated actually from only 9 correspondences in a minimal solver. All of the above mentioned papers kept the strong requirement that the distortion center needs to be known in advance, which practically prevented the use of these techniques for unknown, cropped images or in the case of (digital) zoom. Li et al. [14] addressed the unknown distortion center problem, but they need a calibration grid or a very high number of noise-free point correspondences, among other restrictions.

In this paper we will show that the position of the distortion center can be absorbed into the radial fundamental matrix in very much the same way as the principal point is absorbed into the fundamental matrix.

3 The Lifting-Trick for Radial Distortion

3.1 Second-Order Radial Distortion Models

The traditionally used second-order distortion model in computer vision with unknown center of distortion $(d_x, d_y)^T \in \mathbb{R}^2$ describes the radial distortion as

$$\begin{pmatrix} x_d \\ y_d \end{pmatrix} = \begin{pmatrix} x_u \\ y_u \end{pmatrix} + \lambda \tilde{r}^2 \left(\begin{pmatrix} x_u \\ y_u \end{pmatrix} - \begin{pmatrix} d_x \\ d_y \end{pmatrix} \right), \tag{1}$$

where $(x_d, y_d)^T \in \mathbb{R}^2$ and $(x_u, y_u)^T \in \mathbb{R}^2$ are the distorted and the undistorted point, respectively, whereas $\lambda \in \mathbb{R}$ is the distortion coefficient and $\tilde{r}^2 = \left\| (x_u, y_u)^T - (d_x, d_y)^T \right\|^2$ is the squared Euclidean distance between the center of distortion and the *undistorted* point. Eq. 1 is a *distortion* model since it actually describes the distorted point in explicit form: given the undistorted point

$(x_u, y_u)^T$ and the distortion parameters λ and $(d_x, d_y)^T$, the distorted point can be computed easily by evaluating the right-hand side of Eq. 1.

Fitzgibbon [7] has proposed a slightly different model, which he showed to be equivalently powerful as the model above, i.e. it provides the same approximation accuracy to the underlying true distortion. However, his model enjoys an interesting property. Specifically, this radial distortion model can conveniently be expressed with homogeneous coordinates

$$p_u = \begin{pmatrix} x_u \\ y_u \\ 1 \end{pmatrix} \cong \begin{pmatrix} x_d \\ y_d \\ 1 + \lambda r^2 \end{pmatrix}, \tag{2}$$

with $r^2 = x_d^2 + y_d^2$ and where $\cong$ denotes equality up to a scalar multiple. In this paper, we extend his formulation to the case where not only the distortion coefficient λ is unknown, but the center of radial distortion $(d_x, d_y)^T$ as well. In this case, his model can be extended by starting with

$$\begin{pmatrix} x_d \\ y_d \end{pmatrix} = \begin{pmatrix} x_u \\ y_u \end{pmatrix} + \lambda r^2 \left(\begin{pmatrix} x_u \\ y_u \end{pmatrix} - \begin{pmatrix} d_x \\ d_y \end{pmatrix} \right), \tag{3}$$

where $r^2 = \left\| (x_d, y_d)^T - (d_x, d_y)^T \right\|^2$. The only distinction to the model in Eq. 1 is that the distance is now measured between the *distorted* point and the center of radial distortion. In contrast to the distortion model in Eq. 1 however, Fitzgibbon's model actually is an *undistortion* model: the right-hand side of Eq. 3 is linear in the undistorted point $(x_u, y_u)^T$ and hence one can compute an explicit form for this undistorted point given the distorted point (x_d, y_d) and the distortion parameters λ and $(d_x, d_y)^T$. In the following section, we are going to show how this more complex formulation can be conveniently handled with a lifting trick.

3.2 Lifting to 4D Space

Lifting is a process in polynomial algebra which embeds a problem with nonlinear polynomial terms in a higher dimensional linear space. In our case, radially distorted points in the projective 2-plane $\mathbb{P}^2$ will be mapped to points in projective 3-space $\mathbb{P}^3$. A distorted point with homogeneous coordinates $p_d = (x_d, y_d, z_d)^T \in \mathbb{P}^2$ will be mapped to the point $(x_d z_d, y_d z_d, z_d^2, x_d^2 + y_d^2)^T \in \mathbb{P}^3$. Hence, the projective 2-plane $\mathbb{P}^2$ is mapped to a quadric surface in $\mathbb{P}^3$ defined through $\left\{(x, y, z, w) \in \mathbb{P}^3 | zw - x^2 - y^2 = 0\right\}$[1]. Interestingly and most importantly, the lifted distorted points can be mapped to the undistorted points by a fixed linear transformation, as we will derive shortly. Note that the same lifting scheme has been proposed by Barreto and Daniilidis [1] (see Eq. 7 in their paper). Their derivation is however closely linked to the fundamental matrix, but we would like to highlight that this lifting trick can be applied independently

[1] Points of the form $(x_d z_d, y_d z_d, z_d, x_d^2 + y_d^2)^T$ fulfill this equation $zw - x^2 - y^2 = 0$ as can easily be verified by setting $x = x_d z_d$, $y = y_d z_d$, $z = z_d^2$, $w = x_d^2 + y_d^2$.

of the type of multiple view constraint, i.e. it applies to homographies, trifocal tensors, etc as well. Furthermore, Barreto and Daniilidis assumed a known center of radial distortion. In the following, we show in detail how the same lifting scheme can be generalized to the case of unknown distortion center, resulting in a different linear transformation matrix than the one derived in [1], though.

Let us now present this lifting trick in detail, starting from the distortion model in Eq. 3. Simple algebraic manipulation of Eq. 3 leads to

$$\begin{pmatrix} x_d \\ y_d \end{pmatrix} + \lambda r^2 \begin{pmatrix} d_x \\ d_y \end{pmatrix} = \left(1 + \lambda r^2\right) \begin{pmatrix} x_u \\ y_u \end{pmatrix}, \tag{4}$$

which shows that the undistorted point $(x_u, y_u)^T$ is a scalar multiple of $(x_d, y_d)^T + \lambda r^2 (d_x, d_y)^T$. The scalar factor can be absorbed with a homogeneous representation

$$p_u = \begin{pmatrix} x_u \\ y_u \\ 1 \end{pmatrix} \cong \begin{pmatrix} x_d + \lambda r^2 d_x \\ y_d + \lambda r^2 d_y \\ 1 + \lambda r^2 \end{pmatrix} = \begin{bmatrix} 1 & & d_x \\ & 1 & d_y \\ & & 1 \end{bmatrix} \begin{pmatrix} x_d - d_x \\ y_d - d_y \\ 1 + \lambda r^2 \end{pmatrix}. \tag{5}$$

For additional generality and in order to stay closer to [1], let us represent the distorted point $p_d = (x_d, y_d, z_d) \in \mathbb{P}^2$ as an element of projective 2-space. The previous equation Eq. 5 then becomes

$$p_u \cong \begin{bmatrix} 1 & & d_x \\ & 1 & d_y \\ & & 1 \end{bmatrix} \begin{pmatrix} x_d z_d^{-1} - d_x \\ y_d z_d^{-1} - d_y \\ 1 + \lambda r^2 \end{pmatrix}, \tag{6}$$

with $r^2 = \left(x_d z_d^{-1} - d_x\right)^2 + \left(y_d z_d^{-1} - d_y\right)^2$. Some further algebraic manipulations allow us to expose all the components due to the distorted point on the right hand side

$$p_u \cong \begin{bmatrix} 1 & & d_x \\ & 1 & d_y \\ & & 1 \end{bmatrix} \begin{pmatrix} x_d z_d^{-1} - d_x \\ y_d z_d^{-1} - d_y \\ 1 + \lambda \left(\left(x_d z_d^{-1} - d_x\right)^2 + \left(y_d z_d^{-1} - d_y\right)^2 \right) \end{pmatrix} \tag{7}$$

$$= \begin{bmatrix} 1 & & d_x & \lambda d_x \\ & 1 & d_y & \lambda d_y \\ & & 1 & \lambda \end{bmatrix} \begin{pmatrix} x_d z_d^{-1} - d_x \\ y_d z_d^{-1} - d_y \\ 1 \\ \left(x_d z_d^{-1} - d_x\right)^2 + \left(y_d z_d^{-1} - d_y\right)^2 \end{pmatrix} \tag{8}$$

$$\cong \underbrace{\begin{bmatrix} 1 & & d_x & d_x \\ & 1 & d_y & d_y \\ & & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & \lambda \end{bmatrix} \begin{bmatrix} 1 & & -d_x & \\ & 1 & -d_y & \\ & & 1 & \\ -2d_x & -2d_y & d_x^2 + d_y^2 & 1 \end{bmatrix}}_{=L \in \mathbb{R}^{3\times 4}} \begin{pmatrix} x_d z_d \\ y_d z_d \\ z_d^2 \\ x_d^2 + y_d^2 \end{pmatrix}, \tag{9}$$

where in Eq. 8 the lifting trick has been used and Eq. 9 is equal to Eq. 8 up to a scale factor of z_d^2 which does not matter since $p_u \in \mathbb{P}^2$ is an element of projective

2-space[2]. This derivation provides an important insight and leads to one of the main contributions of this paper. Eq. 9 shows that the undistorted homogeneous coordinates p_u can be expressed by a 3×4 linear transformation L applied from the left to the *lifted* data vector $\left(x_d z_d, y_d z_d, z_d^2, x_d^2 + y_d^2\right)^T \in \mathbb{P}^4$. This linear algebraic representation has far reaching consequences. All multiple-view geometry entities, such as homographies or fundamental matrices, act on homogeneous coordinates of undistorted points. Unfortunately, if the input images are radially distorted, these entities are no longer applicable. However, these entities can be lifted to a higher dimensional space by multiplying them (either from the left and/or the right) with the 3-by-4 matrix L thereby acting on radially distorted lifted coordinates. The matrix L is a function of the radial distortion parameters and therefore also unknown. However, given sufficiently many *distorted* image observations, the lifted multiple view entities can be estimated nonetheless. This will be demonstrated in the following sections with the fundamental matrix.

4 Single-Sided Radial Fundamental Matrix

The fundamental matrix captures the projective relation between two camera views [11]. Given a homogeneous point correspondence p_u and q_u between two images of the same 3D point, the fundamental matrix relates these points by the constraint $q_u^T F p_u = 0$. The fundamental matrix actually maps a point in one image to an epipolar line in the other image. Since neither q_u nor p_u can be the zero vector, F has a non-trivial left and right nullspace. These nullspaces actually correspond to the two epipoles. The next section shows how the fundamental matrix can be extended to handle a radially distorted point measurement p_d instead of an undistorted measurement p_u.

4.1 Derivation of the Single-Sided Radial Fundamental Matrix

Let us now assume that one of the two images is radially distorted, say the one where feature point p_u has been observed. This means that only the radially distorted point $(x_d, y_d)^T$ is known. Thanks to the derivation in Sec. 3.2, we know how to handle this situation. A simple right-multiplication by L lifts the fundamental matrix (on one side) to a 4D projective space which allows to use the radially distorted measurements

$$0 = q_u^T F p_u = q_u^T F L \begin{pmatrix} x_d z_d \\ y_d z_d \\ z_d^2 \\ x_d^2 + y_d^2 \end{pmatrix} = q_u^T \hat{F} \begin{pmatrix} x_d z_d \\ y_d z_d \\ z_d^2 \\ x_d^2 + y_d^2 \end{pmatrix}, \tag{10}$$

where the *single-sided radial fundamental matrix* $\hat{F} = FL \in \mathbb{R}^{3\times 4}$ has been introduced. The decomposition $L = [I \mid 0] + \lambda \left(d_x, d_y, 1\right)^T \left(-2d_x, -2d_y, d_x^2 + d_y^2, 1\right)$

[2] Of course in practice, the measurement will be normalized such that $z_d = 1$ and the formulas simplify slightly. Nevertheless, the more general representation is easier to interpret in terms of mappings between projective spaces.

leads to another interesting representation of the single-sided radial fundamental matrix

$$\hat{F} = FL = \left([F \mid 0] + F\lambda \begin{pmatrix} d_x \\ d_y \\ 1 \end{pmatrix} \begin{pmatrix} -2d_x & -2d_y & d_x^2 + d_y^2 & 1 \end{pmatrix} \right). \tag{11}$$

4.2 Properties of the Single-Sided Radial Fundamental Matrix

Since the single-sided radial fundamental matrix $\hat{F} = FL$ is given as the product between the ordinary rank-2 fundamental matrix F and the matrix L, its rank equals 2. As a 3×4 matrix of rank 2, the single-sided radial fundamental matrix has $3 \cdot 2 + 2 \cdot 4 - 2 \cdot 2 - 1 = 9$ degrees of freedom (minus one due to the scale ambiguity)[3].

Unfortunately, the number of parameters we are looking for equals 7 for the standard fundamental matrix plus 3 for the radial distortion parameters. Hence, there are 10 parameters but only 9 degrees of freedom in the single-sided radial fundamental matrix. This implies that there is a one-parametric family of perfectly valid solutions. Hence, given a single-sided radial fundamental matrix, it is not possible to uniquely extract the underlying fundamental matrix and the 3 radial distortion parameters. This is in contrast to previous work [1,13] which assumed a known radial distortion center which decreased the number of parameters by 2. This allowed the unique extraction of all the 8 parameters. Nevertheless, in the remainder of this section, we will show that the epipoles are unique and how they can be extracted from the single-sided radial fundamental matrix even if the radial distortion center is unknown.

The extraction of the left epipole e' from the rank-2 matrix $\hat{F}$ is easy: Since $\hat{F} = FL$, both $\hat{F}$ and F share the same left-nullspace and hence e' equals the left nullspace of $\hat{F}$. This nullspace can be easily computed e.g. with the singular-value decomposition of $\hat{F}$. The right epipole is more tricky since there is a two-dimensional right nullspace $N = [n_1, n_2] \in \mathbb{R}^{4\times 2}$ of $\hat{F} \in \mathbb{R}^{3\times 4}$, i.e. $\hat{F}N = 0 \in \mathbb{R}^{3\times 2}$. This nullspace can again be computed with the singular-value decomposition of $\hat{F}$. The lifted coordinates of the distorted right epipole $e \in \mathbb{P}^3$ must lie in this nullspace since the undistorted epipole lies in the right nullspace of the standard fundamental matrix F which is a factor of $\hat{F} = FL$. Hence, due to this fact and since the distorted coordinates are only defined up to scale, the lifted coordinates of the distorted epipole $e(\alpha) = \alpha n_1 + (1-\alpha)n_2$ can be parametrized with one parameter $\alpha \in \mathbb{R}$. As described at the beginning of Sec. 3.2, valid points $(x, y, z, w) \in \mathbb{P}^3$ in the lifted space are restricted to a quadric surface defined through the equation $zw - x^2 - y^2 = 0$. Plugging the one-parametric representation $e(\alpha)$ into this quadric equation yields a quadratic equation in α which can be solved easily in closed form. This results in two

[3] A matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ of rank r can be factorized $\mathbf{A} = \mathbf{BC}$ with $\mathbf{B} \in \mathbb{R}^{m\times r}$ and $\mathbf{C} \in \mathbb{R}^{r\times n}$. The matrix factors are unique up to a multiplication with a regular matrix $\mathbf{Q} \in \mathbb{R}^{r\times r}$, i.e. $\mathbf{A} = \mathbf{BQQ}^{-1}\mathbf{C}$ and as such $\mathbf{A}$ has $mr + rn - r^2$ degrees of freedom.

equally valid solutions for the distorted coordinates of the right epipole. Note that this is an inherent characteristics of the Fitzgibbon distortion model which always provides two possible distorted points, given the undistorted point and the radial distortion parameters[4].

4.3 Further Examples - Two-Sided Radial Fundamental and Homographies

As already previously mentioned, the same lifting trick can be applied to other entities in multiple view geometry in the presence of radial distortion with unknown center of radial distortion. For example, the two-sided radial fundamental matrix where both images are radially distorted is given by left- and right-multiplying the standard fundamental matrix with the transformations mapping lifted points to undistorted points, i.e.

$$\left(x'_d, y'_d, 1, {x'_d}^2 + {y'_d}^2\right) {L'}^T F L \left(x_d, y_d, 1, x_d^2 + y_d^2\right)^T = 0 \tag{12}$$

If both images have the same radial distortion, then $L = L'$. This results in a 4×4 two-sided radial fundamental matrix which is again of rank 2 and has therefore $4 \cdot 2 + 2 \cdot 4 - 2 \cdot 2 - 1 = 11$ degrees of freedom. There are $7 + 3 + 3 = 13$ parameters (7 due to the standard fundamental matrix and twice times 3 parameters for the two distortion models), and again, there is no unique solution for the parameters. However, the two epipoles can be extracted analogously to the single-sided radial fundamental matrix.

Another example is given by a one-sided radial homography. Again multiplying the lifted coordinates $\tilde{x}$ of the distorted image from the left by L yields $x' \cong HL\tilde{x}$ and hence the one-sided radial homography HL is a full rank 3×4 matrix. Both the two-sided radial fundamental matrix and the one-sided radial homography can be estimated with linear methods analogously to the algorithms presented next for the single-sided radial fundamental matrix.

5 Single-Sided Radial Fundamental Matrix Estimation

The algebraic epipolar constraint

$$q^T \hat{F} \left(x_d \;\; y_d \;\; 1 \;\; x_d^2 + y_d^2\right)^T = 0 \tag{13}$$

can be rewritten using kronecker products [8] as

$$\underbrace{\left(x_d \;\; y_d \;\; 1 \;\; x_d^2 + y_d^2 \;\; \right) \otimes q^T}_{A} \underbrace{\mathrm{vec}(\hat{F})}_{f} = 0 \tag{14}$$

[4] Solving Eq. 3 for the distorted coordinates $(x_d, y_d)^T$ given $(x_u, y_u)^T$, λ, and $(d_x, d_y)^T$ asks for intersecting two conics which in this specific instance can have up to two solutions.

From each correspondence, we obtain a different row vector A_i. Stacking 11 of these equations on top of each other we obtain an 11×12 matrix and f must lie in the null space of that matrix, like in the 8-point algorithm for estimating the fundamental matrix. Similarly, rank two of the resulting matrix can be enforced via a singular value decomposition afterwards.

The Ten Point Algorithm. In an analogous way to the 7-point-algorithm for classical fundamental matrix estimation, we use one correspondence less than is required for the linear solution above and obtain a two-dimensional null-space spanned by f_1 and f_2. The true f must thus be a linear combination of both, where we can fix one of the coefficients, since f is only defined up to scale.

$$f = \alpha f_1 + f_2 \tag{15}$$

We now perform the inverse operation to vectorization and reassemble the matrix $\hat{F}$ from the vector f, and for convenience of notation, explicitely write down the columns:

$$\hat{F} = \left(\hat{F}_1 \; \hat{F}_2 \; \hat{F}_3 \; \hat{F}_4\right) \tag{16}$$

We now choose alpha such that

$$\det\left(\hat{F}_1 \; \hat{F}_2 \; \hat{F}_3\right) = 0 \tag{17}$$

which is the same step as in the standard seven-point algorithm. Thus, from ten correspondences and one cubic determinant constraint we estimate the matrix $\hat{F}$. However, in the presence of noise, it is however not guaranteed that $\hat{F}$ will have rank two, since the last column of $\hat{F}$ can vary freely. Again, rank two of the resulting matrix can be enforced via SVD afterwards.

The Nine Point Algorithm. As mentioned above, in the ten-point-algorithm only the first three columns of F are forced to be in a 2D subspace, however, the last column could still vary freely in the presence of noise. Consequently, we might enforce also the last three columns of F to be linearly dependent. To start, we can use only nine correspondences and obtain a 3D nullspace

$$f = \alpha f_1 + \beta f_2 + f_3 \tag{18}$$

We now choose α and β such that

$$\det\left(\hat{F}_1 \; \hat{F}_2 \; \hat{F}_3\right) = 0 \quad \wedge \quad \det\left(\hat{F}_2 \; \hat{F}_3 \; \hat{F}_4\right) = 0 \tag{19}$$

These are two cubic equations in α and β and according to Bezout's theorem there cannot be more than nine discrete solutions. The derivation of the exact solution is out of the scope of this paper, however the interested reader is refered to Groebner basis methods [12]. As argued before, there are nine degrees of freedom in $\hat{F}$ and so there can be no solution based on less than nine points.

6 Experiments

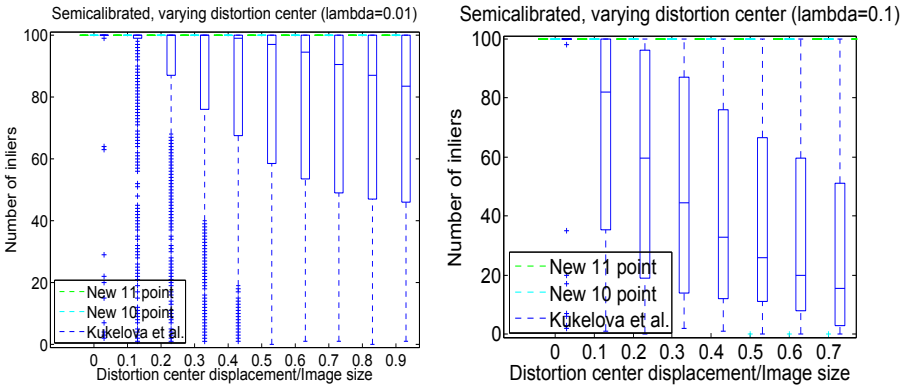
In this section we demonstrate the usefulness of the presented formulation and prove empirically that the new model can cope with arbitrary distortion centers while earlier methods cannot. We first analyse this using synthetic data and then with real images. In the experiments, we use image pairs in which one image has known instrinsics and the other image has unknown focal length, radial distortion center and radial distortion coefficients. In the experiments with synthetic data we use random camera configurations, different distortion centers and different distortion parameters. Due to the lack of earlier methods for our setting, the results are compared to those obtained with the state of the art radial distortion solver from Kukelova et al.[13], although this latter method assumes the distortion center to be at the center of the image and also estimates distortion for both cameras. In contrast, in our setting, one of the images in each image pair has known intrinsics and the distortion center is not at the center of the image. We then test the algorithms with real world images which were taken with cameras that exhibit a significant level of distortion. We generated cropped versions of these images, so that the center of distortion would not lie at the center of the image.

6.1 Evaluation with Synthetic Data

The first set of tests for the semicalibrated case was performed using synthetic data. All tests with synthetic data were performed with a set of 100 random 3D points and 1000 generated random camera poses. The first camera was placed at the origin, with fixed parameters, pointed towards the set of 3D points. The 1000 random poses were generated for the second camera, by generating random translations, random rotations and random focal lengths, varying between 1/2 and 2x the focal length of the first camera. For each camera pose we projected the 3D points on both cameras, distorted the points on the image of the second camera according to the distortion model in Eq. 3, setting the displacement of the distortion center to vary between 0 and the width/height of the image and using different values for the distortion coefficient. For each setting we computed the number of inliers with each algorithm. Results are presented in Fig. 1. We can see that, as the center of distortion is placed further away from the image center, the number of identified correspondences is constant for both implementations of our method, whereas method [13] increasingly fails to correctly identify the correspondences, as it does not correctly model the position of the distortion center.

6.2 Test with Real Images

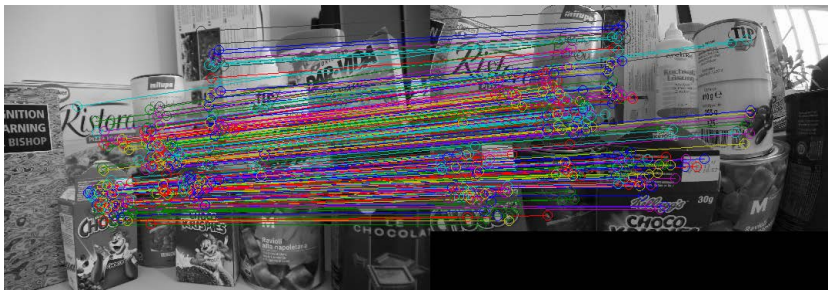
To test the theory on real images we first matched a set of uncalibrated, distorted, cropped images to an image with known calibration parameters using different datasets. The undistorted images were cropped in such a way that the center of distortion would be located away from the center of the resulting image.



(a) Distortion parameter of $\lambda = 0.1$. **(b)** Distortion parameter of $\lambda = 0.01$.

Fig. 1. Boxplots of the number of inliers for 1000 randomly generated camera poses with varying distortion center. Note that our 11- and 10-point algorithms nearly always find all the 100 inliers.

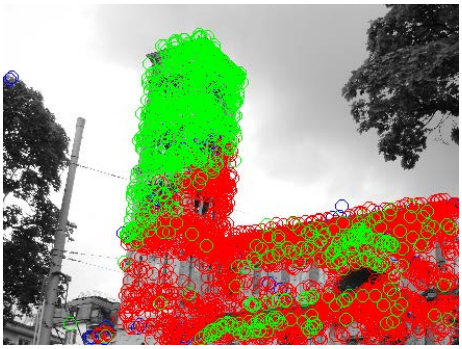
(a) 146 inliers identified by the method from Kukelova et al. [13]



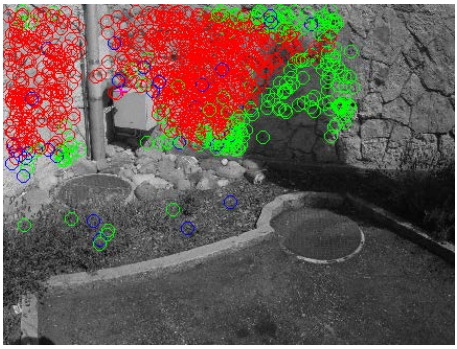
(b) 294 inliers identified by our new method.

Fig. 2. Results for dataset 'Shopping'

(a) Dataset 'Shopping'



(b) dataset 'Church'



(c) dataset 'Corner'

Fig. 3. Comparison between our method and [13]: red are the inliers found by both methods; green are the extra inliers found by our method; blue are inliers found by the method from Kukelova et al. not found by our method

To extract features in the images we used SURF[2], and then we computed a number of putative matches in each image pair by standard feature space matching. This produced a number of matches for each image pair, not all of which were correct correspondences. We then ran both the 11 point and 10 point implementations of our algorithm and [13], in a RANSAC framework with same parameters and constructing hypotheses on the same sample sets. In the end we computed the number of inliers with a threshold of 3 pixels. To obtain the epipolar error (used for classifying outliers) we computed the distance in pixels between a point and the epipolar line in the undistorted image. Before applying the point correspondences to the different algorithms, we normalise the image measurements similarly to the 8-point algorithm [9]. For the calibrated image we use the inverse of the camera intrinsics for the normalization, and for the uncalibrated/distorted/cropped one we use an initial estimate of the focal length, $f_{\text{guess}} = \frac{W/2}{\tan(fov_{\text{guess}}/2)}$, where W is the image width and $fov_{\text{guess}} = 50°$ is an a-priori estimate of the field of view. Furthermore, the image points are normalised with respect to the center of the image. As the normalisation is only performed

to enhance the conditioning of the system, any similarity transformation is a valid normalisation in our formulation.

Results for one of the tested datasets are shown in Fig. 2 where we can see the inliers identified by method of [13] and our new method in an image pair where the image on the left has been previously undistorted with an offline camera calibration phase and the image on the right has unknown distortion parameters and was also cropped so that the distortion center is now in the upper part of the image. One can visualy see that our method is able to identify a higher number of inliers, especially in areas away from the distortion center. Fig. 3a shows a direct comparison of which inliers are identified by our new method and [13]. Again we can see that both methods identify inliers close to the center of distortion but our method identifies extra inliers away from the distortion center. Similar results can be obtained for different datasets in Fig. 3b and Fig. 3c. For the image pair in Fig. 3b the distorted image was cropped so that the distortion center was placed in the bottom right region of the image. For the image pair in Fig. 3c the distorted image was cropped so that the distortion center was placed in the top left region of the image. Also for these image pairs, the method from [13] found only a spatially confined set of correspondences near the center of distortion, whereas our method would be able to use more correspondences also far away from the center of distortion, where radial distortion is more severe. The inliers found by the method of [13] must be explained as an algebraic fit to the data, because the algorithm was not geometrically designed to cope with an unknown distortion center. To the best of our knowledge, the approach presented in this paper is the only one designed to handle epipolar geometry problems with fully unknown radial distortion.

7 Conclusion

We have shown that the lifting of image points into 4-space can consider the distortion center in a linear way. This allows for instance to generalize the radial fundamental matrix to the case of unknown distortion centers, facilitating now practical use of the radial fundamental matrix even with cropped or zoomed images or more generally with images where the center of distortion is unknown. We have proven this by devising different algorithms to estimate the matrix from point correspondences and have shown results on real images that we believe cannot be obtained with any other existing framework. Furthermore, since a change of distortion center can be expressed linearly in 4-space, now the radial distortion model with unknown center can be applied to all multilinear multiple view relations, such as the trifocal tensor and homographies. Besides this, the insight about the distortion center might pave the way for a series of new minimal solvers with unknown distortion center. On top of this we believe that the new radial fundamental matrix can open the door to a theory of radial distortion self calibration, i.e. on top of focal length and principal point one could now look for the distortion coefficient and the distortion center when given multiple image pairs or image sequences, enforce some constraints (e.g. constant distortion center throughout a sequence) and so forth.

References

1. Barreto, J.P., Daniilidis, K.: Fundamental Matrix for Cameras with Radial Distortion. In: International Conference on Computer Vision, ICCV, pp. 625–632 (2005)
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. Computer Vision and Image Understanding, CVIU 3(110), 346–359 (2008)
3. Brito, J.H., Zach, C., Köser, K., Ferreira, M.J., Pollefeys, M.: One-sided Radial Fundamental Matrix Estimation. British Machine Vision Conference, BMVC (2012)
4. Brown, D.C.: Close-range camera calibration. Photogrammetric Engineering 8 (37), 855–866 (1971)
5. Claus, D., Fitzgibbon, A.W.: A Rational Function Lens Distortion Model for General Cameras. Computer Vision and Pattern Recognition, CVPR, 213–219 (2005)
6. Faugeras, O., Luong, Q., Maybank, S.: Camera self-calibration: Theory and experiments. In: Sandini, G. (ed.) ECCV 1992. LNCS, vol. 588, pp. 321–334. Springer, Heidelberg (1992)
7. Fitzgibbon, A.W.: Simultaneous linear estimation of multiple view geometry and lens distortion. In: Computer Vision and Pattern Recognition, CVPR, vol. 1, pp. 125–132 (2001)
8. Fusiello, A.: A matter of notation: Several uses of the Kronecker product in 3D computer vision. Pattern Recognition Letters 15 (28), 2127–2132 (2007)
9. Hartley, R.: In defense of the eight-point algorithm. Pattern Analysis and Machine Intelligence, PAMI 6 (19), 580–593 (1997)
10. Hartley, R., Kang, S.B.: Parameter-Free Radial Distortion Correction with Center of Distortion Estimation. Pattern Analysis and Machine Intelligence, PAMI 8 (29), 1309–1321 (2007)
11. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2004)
12. Kukelova, Z., Bujnak, M., Pajdla, T.: Automatic Generator of Minimal Problem Solvers. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 302–315. Springer, Heidelberg (2008)
13. Kukelova, Z., Byröd, M., Josephson, K., Pajdla, T., Aström, K.: Fast and robust numerical solutions to minimal problems for cameras with radial distortion. Computer Vision and Image Understanding, CVIU 2 (114), 234–244 (2010)
14. Li, H., Hartley, R.: A Non-iterative Method for Correcting Lens Distortion from Nine-Point Correspondences. In: Proc. OmniVision 2005, ICCV-Workshop (2005)
15. Longuet, A.: computer algorithm for reconstructing a scene from two projections. Nature 293, 133–135 (1981)
16. Luong, Q.-T., Faugeras, O.D.: The fundamental matrix: Theory, algorithms, and stability analysis. International Journal of Computer Vision, IJCV 1 (17), 43–75 (1996)
17. Micusik, B., Pajdla, T.: Estimation of omnidirectional camera model from epipolar geometry. Computer Vision and Pattern Recognition, CVPR 1, 485–490 (2003)
18. Thirthala, S., Pollefeys, M.: Pollefeys, Marc: Multi-view geometry of 1D radial cameras and its application to omnidirectional camera calibration. In: International Conference on Computer Vision, ICCV, vol. 2, pp. 1539–1546 (2005)

Depth-Estimation-Free Condition for Projective Factorization and Its Application to 3D Reconstruction

Yohei Murakami[1], Takeshi Endo[2], Yoshimichi Ito[1], and Noboru Babaguchi[1]

[1] Graduate School of Engineering, Osaka University, Osaka Japan
[2] School of Engineering, Osaka University, Osaka Japan

Abstract. This paper concerns depth-estimation-free conditions for projective factorization. We first show that, using an algebraic approach, the estimation of the projective depth is avoidable if and only if the origins of all camera coordinate systems are lying on a single plane, and optical axes of the coordinate systems point the same direction that is perpendicular to the plane. Next, we generalize the result to the case where the points are possibly restricted on a plane or on a line. The result clearly reveals the trade-off between the freedom of camera motion and that of point location. We also give a least-square-based method for Euclidean reconstruction from the result of the projective reconstruction. The proposed method is evaluated through simulation from the viewpoint of computational time.

1 Introduction

Since Sturm and Triggs [1] first proposed projective factorization-based approach to 3D reconstruction, extensive studies have been made for projective factorization. In projective factorization, the estimation of the projective depth plays a central role, and various methods have been proposed such as using epipolar geometry [1], and using iterative computation [2,3,4,5,6,7]. Now, fundamental tools for estimating projective depth have been already established.

However, the projective depth should be estimated for all feature points on all images, it requires large amount of computational loads. Therefore, if the condition for avoiding projective depth estimation is clarified, we can reduce the computational costs for 3D reconstruction by satisfying the condition. Concerning the condition for avoiding projective depth estimation, several results have been presented in a fragmented manner in the existing researches. In [8], Hartley introduces some interesting examples, but a systematic analysis is not made for clarifying the condition. In [9], Triggs derives several conditions for avoiding projective depth estimation in projective space using a geometric approach. His derivation is systematic and elegant, but slightly difficult.

In this paper, we give a comprehensive description of the problem for the depth-estimation-free condition in Euclidean space, and derive a necessary and sufficient condition for depth-estimation-free projective factorization using an

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 150–162, 2013.

algebraic approach. Since the proof is based on elementary linear algebra, it is easy to understand. Next, we generalize the result to the case where the points are possibly restricted on a plane or on a line. The condition obtained clearly reveals the trade-off between the freedom of camera motion and that of point location, which is one of the most important contribution of this paper.

Based on the depth-estimation-free condition, we propose a least-square-based method for Euclidean reconstruction from the result of the projective reconstruction. The proposed method is evaluated through simulation from the viewpoint of the computational time.

2 Necessary and Sufficient Condition for Depth-Estimation-Free Projective Factorization

2.1 Preliminary

Consider the situation that N 3D points are projected on F images. Let $\mathbf{X}_j$ $(j = 1, \cdots, N)$ and $\mathbf{x}_{ij}$ $(i = 1, \cdots, F; j = 1, \cdots, N)$ be, respectively, the homogeneous coordinate vector of the j-th 3D point and the homogeneous coordinate vector of the image point of the j-th 3D point projected on the i-th image, which are given by

$$\mathbf{X}_j = \begin{bmatrix} X_j & Y_j & Z_j & 1 \end{bmatrix}^\top, \quad \mathbf{x}_{ij} = \begin{bmatrix} u_{ij} & v_{ij} & 1 \end{bmatrix}^\top. \tag{1}$$

Let $\mathbf{P}_i$ $(i = 1, \cdots, F)$ be the image projection matrix associated with the i-th frame given by

$$\mathbf{P}_i = \mathbf{K}_i \underline{\mathbf{P}}_f \mathbf{M}_w^i, \tag{2}$$

where $\mathbf{K}_i$, $\underline{\mathbf{P}}_f$, and $\mathbf{M}_w^i$ are, respectively, the camera calibration matrix, the perspective projection matrix, and the camera motion matrix associated with the i-th image, which are given, respectively, by

$$\mathbf{K}_i = \begin{bmatrix} k_{11}^i & k_{12}^i & k_{13}^i \\ 0 & k_{22}^i & k_{23}^i \\ 0 & 0 & 1 \end{bmatrix}, \quad \underline{\mathbf{P}}_f = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{M}_w^i = \begin{bmatrix} \mathbf{R}_i & \mathbf{t}_i \\ \mathbf{0}^\top & 1 \end{bmatrix}, \tag{3}$$

$$\mathbf{R}_i = \begin{bmatrix} r_{11}^i & r_{12}^i & r_{13}^i \\ r_{21}^i & r_{22}^i & r_{23}^i \\ r_{31}^i & r_{32}^i & r_{33}^i \end{bmatrix}, \quad \mathbf{t}_i = \begin{bmatrix} t_1^i \\ t_2^i \\ t_3^i \end{bmatrix} = -\mathbf{R}_i \mathbf{T}_i, \quad \mathbf{T}_i = \begin{bmatrix} t_x^i \\ t_y^i \\ t_z^i \end{bmatrix}, \tag{4}$$

and $\mathbf{R}_i$ and $\mathbf{T}_i$ are, respectively, the rotation matrix representing the orientation of the camera coordinate system associated with the i-th image, and the coordinates of the camera center associated with the i-th image in the world coordinate system. The 3D point $\mathbf{X}_j$ and its projection onto i-th frame $\mathbf{x}_{ij}$ are related by the following equation:

$$\lambda_{ij} \mathbf{x}_{ij} = \mathbf{P}_i \mathbf{X}_j, \tag{5}$$

where λ_{ij} is referred to as the projective depth associated with $\mathbf{x}_{ij}$. By arranging (5) into a large matrix $\mathbf{W}$ whose i-j component block is $\lambda_{ij}\mathbf{x}_{ij}$, we have

$$\mathbf{W} = \mathbf{M}\mathbf{S} \tag{6}$$

where $\mathbf{W} \in \mathcal{R}^{3F\times N}$, $\mathbf{M} \in \mathcal{R}^{3F\times 4}$, and $\mathbf{S} \in \mathcal{R}^{4\times N}$ are given by

$$\mathbf{W}=\begin{bmatrix} \lambda_{11}\mathbf{x}_{11} & \lambda_{12}\mathbf{x}_{12} & \dots & \lambda_{1N}\mathbf{x}_{1N} \\ \lambda_{21}\mathbf{x}_{21} & \lambda_{22}\mathbf{x}_{22} & \dots & \lambda_{2N}\mathbf{x}_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{F1}\mathbf{x}_{F1} & \lambda_{F2}\mathbf{x}_{F2} & \dots & \lambda_{FN}\mathbf{x}_{FN} \end{bmatrix},\ \mathbf{M}=\begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \vdots \\ \mathbf{P}_F \end{bmatrix},\ \mathbf{S}=\begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \dots & \mathbf{X}_N \end{bmatrix}, \quad (7)$$

respectively.

Here, we introduce some basic notions for considering conditions for depth-estimation-free projective factorization.

Definition 1. *A matrix $\mathbf{\Lambda} \in \mathcal{R}^{F\times N}$ whose i-j entry is a projective depth λ_{ij}, that is,*

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_{11} & \lambda_{12} & \dots & \lambda_{1N} \\ \lambda_{21} & \lambda_{22} & \dots & \lambda_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{F1} & \lambda_{F2} & \dots & \lambda_{FN} \end{bmatrix} \quad (8)$$

is called a projective depth matrix.

It is well-known that λ_{ij} can be replaced by $\alpha_i\beta_j\lambda_{ij}$ for nonzero α_i and β_j because homogeneous coordinate vectors $\mathbf{X}_j$ and $\mathbf{x}_{ij}$ (and thus, $\mathbf{P}_i$) are defined only up to an arbitrary nonzero scaling. Therefore, if we can make each entry of $\mathbf{\Lambda}$ equal to 1 by nonzero multiplications of each row and each column of projective depth matrix, the estimation of the projective depth is not required at all. As we will see later, it depends on the camera motion matrices $\mathbf{M}_w^i$ $(i = 1, \cdots, F)$. We refer to the set of the camera motion matrices as *camera motion* and denote it by $\mathcal{M}$. Summarizing the above, we give the following definition concerning the depth-estimation-free projective factorization.

Definition 2. *If every entry λ_{ij} of a projective depth matrix $\mathbf{\Lambda}$ can be made equal to 1 by nonzero multiplications of each row and each column of projective depth matrix, we say that camera motion $\mathcal{M}$ is depth-estimation-free. In such a situation, we refer to the projective factorization as the depth-estimation-free projective factorization.*

In what follows, we proceed our consideration under the following assumptions.

Assumption 1. *The camera coordinate system associated with the first image coincides with the world coordinate system.*

It should be noted we can set this assumption without loss of generality. By this assumption, we can set $\mathbf{R}_1 = \mathbf{I}_3$ and $\mathbf{t}_1 = \mathbf{T}_1 = \mathbf{0}$ where $\mathbf{I}_n$ denotes the identity matrix of size n. We also make the following assumption, which is usually introduced when factorization method is considered.

Assumption 2. *All 3D points $\mathbf{X}_j$ $(j = 1, \cdots, N)$ are viewed by all cameras.*

(2), (3), and (5) imply that the optical axis of each camera coincides with Z-axis of camera coordinate system. Form this, together with Assumptions 1 and 2, we obtain

$$\lambda_{ij} > 0 \ (i = 1, \cdots, F; j = 1, \cdots, N), \ Z_j > 0 \ (j = 1, \cdots, N). \tag{9}$$

2.2 Derivation of Depth-Estimation-Free Condition

We begin with the following lemma, which is a direct consequence of Definition 2.

Lemma 1. *Camera motion $\mathcal{M}$ is depth-estimation-free if and only if the projective depth matrix $\mathbf{\Lambda}$ is expressed as follows:*

$$\mathbf{\Lambda} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_F \end{bmatrix} \begin{bmatrix} \beta_1 \ \beta_2 \ \dots \ \beta_N \end{bmatrix}, \tag{10}$$

$$\alpha_i \neq 0 \, (i = 1, \cdots, F), \ \beta_j \neq 0 \, (j = 1, \cdots, N). \tag{11}$$

Proof of Lemma 1. From Definition 2, camera motion $\mathcal{M}$ is depth-estimation-free if and only if there exist nonsingular diagonal matrices $\mathbf{A} \in \mathcal{R}^{F \times F}$ and $\mathbf{B} \in \mathcal{R}^{N \times N}$ satisfying the following equation.

$$\mathbf{A}\mathbf{\Lambda}\mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \begin{bmatrix} 1 \ 1 \ \dots \ 1 \end{bmatrix}. \tag{12}$$

The sufficiency of conditions (10) and (11) is straightforward because if we choose

$$\mathbf{A} = \mathrm{diag}[1/\alpha_1, 1/\alpha_2, \cdots, 1/\alpha_F], \ \mathbf{B} = \mathrm{diag}[1/\beta_1, 1/\beta_2, \cdots, 1/\beta_N], \tag{13}$$

condition (12) is satisfied, where $\mathrm{diag}[a, b, c]$ denotes the diagonal matrix whose diagonal entries are a, b, and c in order. In what follows, we show the necessity part.

Since $\mathbf{A}$ and $\mathbf{B}$ in (12) are nonsingular, $\mathrm{rank}\,(\mathbf{\Lambda}) = \mathrm{rank}\,(\mathbf{A}\mathbf{\Lambda}\mathbf{B})$ holds where $\mathrm{rank}(\mathbf{A})$ is the rank of $\mathbf{A}$. Therefore, the rank of $\mathbf{\Lambda}$ should be one since the rank of the matrix of the righthand-side of (12) is one. Thus, $\mathbf{\Lambda}$ must be expressed as in (10). In (10), if $\alpha_i = 0$ for some i, the i-th row of the projective depth matrix becomes zero no matter how nonsingular diagonal matrices $\mathbf{A}$ and $\mathbf{B}$ are selected. Therefore, α_i must be nonzero for $i = 1, \cdots, F$. In a similar manner, we have β_j must be nonzero for $j = 1, \cdots, N$. This completes the proof of Lemma 1. □

Based on Lemma 1, we derive a necessary and sufficient condition for the depth-estimation-free projective factorization in Euclidean space.

Theorem 1. *The following three conditions are equivalent.*

(i) Camera motion $\mathcal{M}$ is depth-estimation-free.
(ii) $\mathbf{R}_i$ and $\mathbf{T}_i$ $(i = 1, \cdots, F)$ have the following structures.

$$\mathbf{R}_i = \begin{bmatrix} r_{11}^i & r_{12}^i & 0 \\ r_{21}^i & r_{22}^i & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_i = \begin{bmatrix} t_x^i \\ t_y^i \\ 0 \end{bmatrix}. \tag{14}$$

(iii) Origins of all camera coordinate systems are lying on a single plane, and optical axes of the coordinate systems point the same direction that is perpendicular to the plane.

Proof of Theorem 1. The implication of condition (ii) is nothing but condition (iii). Here, we show the equivalence of condition (i) and condition (ii). First, we show (i) implies (ii).

By (1)-(5), projective depth matrix $\mathbf{\Lambda}$ is rewritten as

$$\mathbf{\Lambda} = \mathbf{M}_{3s}\mathbf{X}_s, \ \mathbf{M}_{3s} = \begin{bmatrix} r_{31}^1 & r_{32}^1 & r_{33}^1 & t_3^1 \\ r_{31}^2 & r_{32}^2 & r_{33}^2 & t_3^2 \\ \vdots & \vdots & \vdots & \vdots \\ r_{31}^F & r_{32}^F & r_{33}^F & t_3^F \end{bmatrix}, \ \mathbf{X}_s = \begin{bmatrix} X_1 & X_2 & \dots & X_N \\ Y_1 & Y_2 & \dots & Y_N \\ Z_1 & Z_2 & \dots & Z_N \\ 1 & 1 & \dots & 1 \end{bmatrix}. \tag{15}$$

If camera motion $\mathcal{M}$ is depth-estimation-free, $\mathrm{rank}(\mathbf{\Lambda}) = 1$ because $\mathbf{\Lambda}$ satisfies condition (10) by Lemma 1. In generic situation, points are located arbitrarily in 3D space. Therefore, $\mathbf{X}_s$ is a row full rank matrix when N is large enough $(N \geq 4)$. Therefore, $\mathrm{rank}\,(\mathbf{M}_{3s}) = \mathrm{rank}\,(\mathbf{M}_{3s}\mathbf{X}_s) = \mathrm{rank}\,(\mathbf{\Lambda})$ holds [10], and thus, $\mathrm{rank}(\mathbf{M}_{3s}) = 1$.

Here, note that $\begin{bmatrix} r_{31}^1 & r_{32}^1 & r_{33}^1 & t_3^1 \end{bmatrix} = [0\ 0\ 1\ 0]$ because $\mathbf{R}_1 = \mathbf{I}$ and $\mathbf{t}_1 = \mathbf{0}$ from Assumption 1. Therefore, $\begin{bmatrix} r_{31}^i & r_{32}^i & r_{33}^i & t_3^i \end{bmatrix}$ $(i = 2, \cdots, F)$ is a nonzero scalar multiplication of $\begin{bmatrix} r_{31}^1 & r_{32}^1 & r_{33}^1 & t_3^1 \end{bmatrix}$ because $\mathrm{rank}(\mathbf{M}_{3s}) = 1$. Note also that the norm of the row vector $\begin{bmatrix} r_{31}^i & r_{32}^i & r_{33}^i \end{bmatrix}$ is equal to one because it is a row vector of a rotation matrix $\mathbf{R}_i$. Therefore, we have

$$\begin{bmatrix} r_{31}^i & r_{32}^i & r_{33}^i & t_3^i \end{bmatrix} = \begin{bmatrix} 0\ 0\ \pm 1\ 0 \end{bmatrix} \ (i = 2, \cdots, F). \tag{16}$$

From this, we obtain the structures of $\mathbf{R}_i$, $\mathbf{t}_i$, and $\mathbf{T}_i$, as

$$\mathbf{R}_i = \begin{bmatrix} r_{11}^i & r_{12}^i & 0 \\ r_{21}^i & r_{22}^i & 0 \\ 0 & 0 & \pm 1 \end{bmatrix}, \ \mathbf{t}_i = \begin{bmatrix} t_1^i \\ t_1^i \\ 0 \end{bmatrix}, \ \mathbf{T}_i = \begin{bmatrix} t_x^i \\ t_y^i \\ 0 \end{bmatrix}, \tag{17}$$

respectively. Here, if $r_{33}^i = -1$, the optical axis of the camera associated with the first image and that with the i-th image point opposite direction each other. This contradicts Assumption 2. Thus, $r_{33}^i = 1$ is obtained, and we have shown that condition (i) implies condition (ii).

Next, we show (ii) implies (i). From (15), together with $\begin{bmatrix} r^1_{31} & r^1_{32} & r^1_{33} & t^1_3 \end{bmatrix} = [0\ 0\ 1\ 0]$, we have $\lambda_{ij} = Z_j$. Note that $Z_j \neq 0$ because of (9). This implies that λ_{ij} satisfies the condition of Lemma 1 ($\alpha_i = 1$, $\beta_j = Z_j$), and thus, we have shown that (ii) implies (i). This completes the proof of Theorem 1. □

In [8], it is pointed out that if the points are located at different depths, but each point remains the same depth from the cameras through the whole sequence, each depth can be set equal to 1. The condition is nothing but condition (iii) of Theorem 1, and thus, the description implies that condition (iii) is a sufficient condition for depth-estimation-free projective factorization. The importance of Theorem 1 is that the condition is not only sufficient but also necessary.

In [9], the notion of *pseudo-affine*, that is, optical planes of all cameras coincide, is introduced, and it is shown in projective space that pseudo-affine condition is a necessary and sufficient condition for avoiding depth estimation. The pseudo-affine condition corresponds to condition (iii) of Theorem 1, and therefore, Theorem 1 can be regarded as a Euclidean space counterpart of the result in [9].

Theorem 1 gives a necessary and sufficient condition for depth-estimation-free projective factorization in generic situation. Next, we consider the case that points location is possibly restricted to a plane or on a line. In both cases, the restricted region of the points are described by a set of $\mathbf{X}$ satisfying $\mathbf{\Pi X} = \mathbf{0}$, that is, a null space of $\mathbf{\Pi}$, where $\mathbf{\Pi}$ is a row full rank matrix. By using this expression for the restricted region, we obtain a necessary and sufficient condition for depth-estimation-free projective factorization for the case that points location is possibly restricted.

Theorem 2. *Suppose that* $\mathbf{\Pi} \in \mathcal{R}^{c\times 4}$ *is a row full rank matrix, and let* $\mathbf{p}^{i\top}_3 = \begin{bmatrix} r^i_{31} & r^i_{32} & r^i_{33} & t^i_3 \end{bmatrix}$ *and* $\mathbf{e}^\top_3 = \begin{bmatrix} 0\ 0\ 1\ 0 \end{bmatrix}$. *Also suppose that the points are restricted in the null space of* $\mathbf{\Pi}$. *Then, camera motion* $\mathcal{M}$ *is depth-estimation-free if and only if there exist* $a_i > 0$, $\mathbf{b}_i \in \mathcal{R}^c$ $(i = 2, \cdots, F)$ *satisfying the following conditions.*

$$\mathbf{p}^{i\top}_3 = a_i \mathbf{e}^\top_3 + \mathbf{b}^\top_i \mathbf{\Pi}, \tag{18}$$

$$\mathbf{p}^{i\top}_3 \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0}^\top & 0 \end{bmatrix} \mathbf{p}^i_3 = 1. \tag{19}$$

Here, we introduce the singular value decomposition of $\mathbf{\Pi} \in \mathcal{R}^{c\times 4}$ as follows:

$$\mathbf{\Pi} = \mathbf{U} \begin{bmatrix} \mathbf{\Sigma} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{V}^\top_1 \\ \mathbf{V}^\top_2 \end{bmatrix} \tag{20}$$

where $\mathbf{U} \in \mathcal{R}^{c\times c}$ is an orthogonal matrix, $\mathbf{\Sigma} \in \mathcal{R}^{c\times c}$ is a diagonal matrix consisting of singular values, and $\mathbf{O} \in \mathcal{R}^{c\times(4-c)}$ is a zero matrix. $\begin{bmatrix} \mathbf{V}_1 & \mathbf{V}_2 \end{bmatrix} \in \mathcal{R}^{4\times 4}$ is an orthogonal matrix composed by $\mathbf{V}_1 \in \mathcal{R}^{4\times c}$ and $\mathbf{V}_2 \in \mathcal{R}^{4\times(4-c)}$, which satisfies the following equations.

$$\begin{bmatrix} \mathbf{V}_1 & \mathbf{V}_2 \end{bmatrix} \begin{bmatrix} \mathbf{V}^\top_1 \\ \mathbf{V}^\top_2 \end{bmatrix} = \mathbf{V}_1\mathbf{V}^\top_1 + \mathbf{V}_2\mathbf{V}^\top_2 = \mathbf{I}_4, \tag{21}$$

$$\begin{bmatrix} \mathbf{V}^\top_1 \\ \mathbf{V}^\top_2 \end{bmatrix} \begin{bmatrix} \mathbf{V}_1 & \mathbf{V}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{I}_c & \mathbf{O} \\ \mathbf{O}^\top & \mathbf{I}_{4-c} \end{bmatrix}. \tag{22}$$

It is noted that the null space of $\mathbf{\Pi}$ coincides with the image of $\mathbf{V}_2$. Therefore, points $\mathbf{X}_j$ satisfying $\mathbf{\Pi}\mathbf{X}_j = \mathbf{0}$ is expressed as $\mathbf{X}_j = \mathbf{V}_2\bar{\mathbf{X}}_j$ for some $\bar{\mathbf{X}}_j \in \mathcal{R}^{4-c}$. Furthermore, $\mathbf{\Pi}$ is expressed as

$$\mathbf{\Pi} = \mathbf{T}\mathbf{V}_1^\top, \tag{23}$$

where $\mathbf{T} = \mathbf{U}\mathbf{\Sigma}$. From (22), we have $\mathbf{\Pi}\mathbf{V}_2 = \mathbf{T}\mathbf{V}_1^\top\mathbf{V}_2 = \mathbf{O}$.

Before going to the proof of Theorem 2, we show the following lemma.

Lemma 2. *If Assumption 2 is satisfied and* $\mathbf{\Pi}$ *is of row full rank,* $\mathbf{e}_3^\top\mathbf{V}_2 \neq \mathbf{0}^\top$.

Proof of Lemma 2. We first show that if Assumption 2 is satisfied and $\mathbf{\Pi}$ is of row full rank, $\begin{bmatrix}\mathbf{e}_3^\top \\ \mathbf{\Pi}\end{bmatrix}$ is also of row full rank. Next, we show that if $\begin{bmatrix}\mathbf{e}_3^\top \\ \mathbf{\Pi}\end{bmatrix}$ of row full rank, $\mathbf{e}_3^\top\mathbf{V}_2 \neq \mathbf{0}^\top$ holds.

Now, suppose that $\begin{bmatrix}\mathbf{e}_3^\top \\ \mathbf{\Pi}\end{bmatrix}$ is not of row-full rank. Then, $\mathbf{e}_3^\top$ is expressed as a linear combination of row vectors of $\mathbf{\Pi}$ because $\mathbf{\Pi}$ is of row-full rank. Therefore, points $\mathbf{X}$ satisfying $\mathbf{\Pi}\mathbf{X} = \mathbf{0}$ also satisfy $\mathbf{e}_3^\top\mathbf{X} = Z = 0$. This implies that points $\mathbf{X}$ satisfying $\mathbf{\Pi}\mathbf{X} = \mathbf{0}$ is lying on the plane given by $Z = 0$. This contradicts condition (9), and thus, Assumption 2 is not satisfied. This implies that Assumption 2 and row-full rankness of $\mathbf{\Pi}$ yield row-full rankness of $\begin{bmatrix}\mathbf{e}_3^\top \\ \mathbf{\Pi}\end{bmatrix}$.

Next, we show that $\mathbf{e}_3^\top\mathbf{V}_2 = \mathbf{0}^\top$ yields contradiction when $\begin{bmatrix}\mathbf{e}_3^\top \\ \mathbf{\Pi}\end{bmatrix}$ is of row-full rank. Since the set of row vectors of $\begin{bmatrix}\mathbf{V}_1^\top \\ \mathbf{V}_2^\top\end{bmatrix}$ is a basis for four dimensional space, $\mathbf{e}_3^\top$ is expressed as a linear combination of the basis. Therefore, $\mathbf{e}_3^\top$ is expressed as $\mathbf{e}_3^\top = \mathbf{k}_1^\top\mathbf{V}_1^\top + \mathbf{l}_1^\top\mathbf{V}_2^\top$ for some $\mathbf{k}_1 \in \mathcal{R}^c$ and $\mathbf{l}_1 \in \mathcal{R}^{4-c}$. By post-multiplying $\mathbf{V}_2$ for both sides of this equation, we have

$$\mathbf{e}_3^\top\mathbf{V}_2 = \mathbf{l}_1^\top \tag{24}$$

because of (22). Therefore, $\mathbf{e}_3^\top\mathbf{V}_2 = \mathbf{0}^\top$ yields $\mathbf{l}_1^\top = \mathbf{0}^\top$, and thus, we obtain $\mathbf{e}_3^\top = \mathbf{k}_1^\top\mathbf{V}_1^\top$. This, together with (23), we obtain $\begin{bmatrix}\mathbf{e}_3^\top \\ \mathbf{\Pi}\end{bmatrix} = \begin{bmatrix}\mathbf{k}_1^\top \\ \mathbf{T}\end{bmatrix}\mathbf{V}_1^\top$, which contradicts the row-full rankness of $\begin{bmatrix}\mathbf{e}_3^\top \\ \mathbf{\Pi}\end{bmatrix}$ because $\mathbf{a}^\top = \begin{bmatrix}1 & -\mathbf{k}_1^\top\mathbf{T}^{-1}\end{bmatrix} \neq 0$ yields $\mathbf{a}^\top\begin{bmatrix}\mathbf{e}_3^\top \\ \mathbf{\Pi}\end{bmatrix} = 0$. Thus, we have shown that row-full rankness of $\begin{bmatrix}\mathbf{e}_3^\top \\ \mathbf{\Pi}\end{bmatrix}$ implies $\mathbf{e}_3^\top\mathbf{V}_2 \neq \mathbf{0}^\top$, and the proof of Lemma 2 is now completed. □

Now, we are in a position to prove Theorem 2.

Proof of Theorem 2. Since $\begin{bmatrix} r_{31}^i & r_{32}^i & r_{33}^i \end{bmatrix}$ is a row vector of rotation matrix $\mathbf{R}_i$, its norm must be one, which is equivalent to condition (19) of Theorem 2. In what follows, we show that, under condition (19), camera motion $\mathcal{M}$ is depth-estimation-free if and only if condition (18) is satisfied for some $a_i > 0$ and $\mathbf{b}_i \in \mathcal{R}^c$ $(i = 2, \cdots, F)$.

First, we show the sufficiency part. If there exist $a_i > 0$ and $\mathbf{b}_i \in \mathcal{R}^c$ $(i = 2, \cdots, F)$ satisfying condition (18), $\mathbf{\Lambda}$ is rewritten as

$$\mathbf{\Lambda} = \mathbf{M}_{3s}\mathbf{X}_s = \begin{bmatrix} \mathbf{e}_3^\top \\ \mathbf{p}_3^{2\top} \\ \vdots \\ \mathbf{p}_3^{F\top} \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 \cdots \mathbf{X}_N \end{bmatrix} = \begin{bmatrix} \mathbf{e}_3^\top \\ a_2\mathbf{e}_3^\top + \mathbf{b}_2^\top\mathbf{\Pi} \\ \vdots \\ a_F\mathbf{e}_3^\top + \mathbf{b}_F^\top\mathbf{\Pi} \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 \cdots \mathbf{X}_N \end{bmatrix}, \quad (25)$$

$$= \begin{bmatrix} 1 \\ a_2 \\ \vdots \\ a_F \end{bmatrix} \begin{bmatrix} Z_1 \cdots Z_N \end{bmatrix}, \quad (26)$$

where we use (15), and relations $\mathbf{e}_3^\top\mathbf{X}_j = Z_j$ and $\mathbf{\Pi}\mathbf{X}_j = \mathbf{0}$. Since $Z_j > 0$ $(j = 1, \cdots, N)$ from (9), and $a_i > 0$ $(i = 2, \cdots, F)$, $\mathbf{\Lambda}$ satisfies the condition of Lemma 1, and thus, camera motion $\mathcal{M}$ is depth-estimation-free. This completes the proof of the sufficiency part.

Next, we show the necessity part. Since the set of row vectors of $\begin{bmatrix} \mathbf{V}_1^\top \\ \mathbf{V}_2^\top \end{bmatrix}$ is a basis for four dimensional space, $\mathbf{e}_3^\top$ and $\mathbf{p}_3^{i\top}$ $(i = 2, \cdots, F)$ are expressed as follows with some $\mathbf{k}_i \in \mathcal{R}^c$ and $\mathbf{l}_i \in \mathcal{R}^{4-c}$ $(i = 1, \cdots, F)$.

$$\begin{bmatrix} \mathbf{e}_3^\top \\ \mathbf{p}_3^{2\top} \\ \vdots \\ \mathbf{p}_3^{F\top} \end{bmatrix} = \begin{bmatrix} \mathbf{K}^\top \ \mathbf{L}^\top \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^\top \\ \mathbf{V}_2^\top \end{bmatrix}, \quad \mathbf{K}^\top = \begin{bmatrix} \mathbf{k}_1^\top \\ \mathbf{k}_2^\top \\ \vdots \\ \mathbf{k}_F^\top \end{bmatrix}, \quad \mathbf{L}^\top = \begin{bmatrix} \mathbf{l}_1^\top \\ \mathbf{l}_2^\top \\ \vdots \\ \mathbf{l}_F^\top \end{bmatrix}. \quad (27)$$

From (15) and (22), together with the fact that point $\mathbf{X}_j$ satisfying $\mathbf{\Pi}\mathbf{X}_j = \mathbf{0}$ is expressed as $\mathbf{X}_j = \mathbf{V}_2\bar{\mathbf{X}}_j$ for some $\bar{\mathbf{X}}_j \in \mathcal{R}^{4-c}$, $\mathbf{\Lambda}$ is rewritten as

$$\mathbf{\Lambda} = \mathbf{M}_{3s}\mathbf{X}_s = \begin{bmatrix} \mathbf{e}_3^\top \\ \mathbf{p}_3^{2\top} \\ \vdots \\ \mathbf{p}_3^{F\top} \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 \cdots \mathbf{X}_N \end{bmatrix} = \begin{bmatrix} \mathbf{K}^\top \ \mathbf{L}^\top \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^\top \\ \mathbf{V}_2^\top \end{bmatrix} \mathbf{V}_2 \begin{bmatrix} \bar{\mathbf{X}}_1 \cdots \bar{\mathbf{X}}_N \end{bmatrix}, \quad (28)$$

$$= \mathbf{L}^\top \begin{bmatrix} \bar{\mathbf{X}}_1 \cdots \bar{\mathbf{X}}_N \end{bmatrix}. \quad (29)$$

Here, note that the rank of $\mathbf{L}^\top$ must be one because $\mathrm{rank}(\mathbf{\Lambda}) = 1$ from Theorem 1 and the matrix $\begin{bmatrix} \bar{\mathbf{X}}_1 \cdots \bar{\mathbf{X}}_N \end{bmatrix}$ is arbitrary. On the other hand, from (24) and Lemma 2, $\mathbf{l}_1^\top = \mathbf{e}_3^\top\mathbf{V}_2 \neq 0$ from assumptions. This implies that $\mathbf{l}_i^\top$ is expressed as

$$\mathbf{l}_i^\top = a_i\mathbf{l}_1^\top = a_i\mathbf{e}_3^\top\mathbf{V}_2 \ (i = 2, \cdots, F) \quad (30)$$

for some a_i. From (21), (23), (27), and (30), $\mathbf{p}_3^{i\top}$ is rewritten as

$$\mathbf{p}_3^{i\top} = \mathbf{k}_i^\top \mathbf{V}_1^\top + \mathbf{l}_i^\top \mathbf{V}_2^\top = \mathbf{k}_i^\top \mathbf{V}_1^\top + a_i \mathbf{e}_3^\top \mathbf{V}_2 \mathbf{V}_2^\top, \tag{31}$$

$$= \mathbf{k}_i^\top \mathbf{V}_1^\top + a_i \mathbf{e}_3^\top (\mathbf{I} - \mathbf{V}_1 \mathbf{V}_1^\top) = a_i \mathbf{e}_3^\top + (\mathbf{k}_i^\top - a_i \mathbf{e}_3^\top \mathbf{V}_1) \mathbf{V}_1^\top, \tag{32}$$

$$= a_i \mathbf{e}_3^\top + (\mathbf{k}_i^\top - a_i \mathbf{e}_3^\top \mathbf{V}_1) \mathbf{T}^{-1} \mathbf{\Pi} = a_i \mathbf{e}_3^\top + \mathbf{b}_i^\top \mathbf{\Pi}, \tag{33}$$

where we put $\mathbf{b}_i^\top = (\mathbf{k}_i^\top - a_i \mathbf{e}_3^\top \mathbf{V}_1)\mathbf{T}^{-1}$ in (33). Thus, we have shown that $\mathbf{p}_3^{i\top}$ is expressed as (18). In this case, $\mathbf{\Lambda}$ is rewritten as (26). From (26), it is shown that $a_i > 0$ $(i = 2, \cdots, F)$ because of condition (9). This completes the proof of the necessity part of Theorem 2. □

As a special case of Theorem 2, we consider the case that points are restricted on the plane given by $z = 10$. In this case, we can select $\mathbf{\Pi}$ of $\mathbf{\Pi X} = 0$ as $\mathbf{\Pi} = \begin{bmatrix} 0 & 0 & 1 & -10 \end{bmatrix}$. From condition (9) and (19), $a_i + b_i = 1$, $r_{31}^i = 0$, $r_{32}^i = 0$, $r_{33}^i = 1$, and $t_z^i = -t_3^i = 10(1 - a_i)$ are necessary. Therefore, if $a_i > 0$, conditions of Theorem 2 are satisfied. This implies that camera motion $\mathcal{M}$ is depth-estimation-free provided that optical axes of all cameras are perpendicular to the plane. Such an example is also shown in [8].

It is straightforward to show that the conditions of Theorem 2 are reduced to those of Theorem 1, when no restriction to the points location exists. Therefore, Theorem 2 can be regarded as a generalization of Theorem 1. From condition (18) of Theorem 2, we can observe that the freedom of choosing $\mathbf{p}_3^{i\top}$ increases as the freedom of point location decreases. Therefore, Theorem 2 describe the trade-off between the freedom of camera motion and that of point location.

3 Euclidean Reconstruction under Depth-Estimation-Free Condition

In this section, we give a method for Euclidean reconstruction from the projective reconstruction under the depth-estimation-free condition. In particular, we give a method under the following assumption.

Assumption 3. *All cameras satisfy condition (iii) of Theorem 1. Furthermore, all camera has the same orientation.*

From this assumption, the rotation matrix of each camera becomes identity, that is, $\mathbf{R}_i = \mathbf{I}$ $(i = 1, \cdots, F)$. However, in this case, it is known that self-calibration is impossible, that is, we cannot obtain internal camera parameters $\mathbf{K}_i$ and external camera parameters $\mathbf{M}_w^i$ in (2) at the same time [11]. Therefore, we assume the following condition.

Assumption 4. *Internal parameters $\mathbf{K}_i$ of all cameras are known.*

In what follows, we consider a method for finding nonsingular matrix $\mathbf{H} \in \mathcal{R}^{4\times 4}$ that attains a Euclidean reconstruction

$$\mathbf{M} = \mathbf{M}_s \mathbf{H}, \quad \mathbf{S} = \mathbf{H}^{-1} \mathbf{S}_s, \tag{34}$$

from a result of a projective reconstruction $\mathbf{W}' = \mathbf{M}_s\mathbf{S}_s$, where

$$\mathbf{W}' = \begin{bmatrix} \mathbf{x}_{11} & \mathbf{x}_{12} & \dots & \mathbf{x}_{1N} \\ \mathbf{x}_{21} & \mathbf{x}_{22} & \dots & \mathbf{x}_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_{F1} & \mathbf{x}_{F2} & \dots & \mathbf{x}_{FN} \end{bmatrix}, \ \mathbf{M}_s = \begin{bmatrix} \mathbf{P}_s^1 \\ \mathbf{P}_s^2 \\ \vdots \\ \mathbf{P}_s^F \end{bmatrix}, \ \mathbf{S}_s = \begin{bmatrix} \mathbf{X}_s^1 & \mathbf{X}_s^2 & \dots & \mathbf{X}_s^N \end{bmatrix}, \tag{35}$$

and $\mathbf{P}_s^i \in \mathcal{R}^{3\times4}$, $\mathbf{X}_s^j \in \mathcal{R}^{4\times1}$. The validity of using $\mathbf{W}'$, instead of using $\mathbf{W}$ in (7), will be explained later. Here, we assume that $\mathbf{P}_s^1$ is set to be $\mathbf{P}_s^1 = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \end{bmatrix}$ by pre-processing. Let $\mathbf{P}_i$ and $\mathbf{X}_j$ be, respectively, the projection matrix of the camera associated with the i-th image and the homogeneous coordinate vector of the j-th 3D point after Euclidean reconstruction. Then, from (34) and (35), we have

$$\mathbf{P}_i = \mathbf{P}_s^i\mathbf{H}, \ \mathbf{X}_j = \mathbf{H}^{-1}\mathbf{X}_s^j. \tag{36}$$

From (2) and (3), and Assumption 1, we have $\mathbf{P}_1 = [\mathbf{K}_1 \ \mathbf{0}]$. By this, together with (36) and $\mathbf{P}_s^1 = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \end{bmatrix}$, we obtain $\begin{bmatrix} \mathbf{K}_1 & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \end{bmatrix} \mathbf{H}$. Thus, we obtain

$$\mathbf{H} = \begin{bmatrix} \mathbf{K}_1 & \mathbf{0} \\ \mathbf{v}^\top & k \end{bmatrix} \tag{37}$$

where $\mathbf{v} \in \mathcal{R}^3$ and k is a nonzero constant. Since k is only affect the scale of the projective reconstruction, we can set $k = 1$ without loss of generality. Therefore, Finding $\mathbf{v}$ is equivalent to finding $\mathbf{H}$.

From (2)-(4) and Assumption 3, we have $\mathbf{P}_i = \mathbf{K}_i \begin{bmatrix} \mathbf{I} & -\mathbf{T}_i \end{bmatrix}$. By substituting this and (37) into the first equation of (36), we obtain

$$\mathbf{K}_i \begin{bmatrix} \mathbf{I} & -\mathbf{T}_i \end{bmatrix} = \begin{bmatrix} \mathbf{A}_s^i & \mathbf{B}_s^i \end{bmatrix} \begin{bmatrix} \mathbf{K}_1 & \mathbf{0} \\ \mathbf{v}^\top & 1 \end{bmatrix}, \tag{38}$$

where $\begin{bmatrix} \mathbf{A}_s^i & \mathbf{B}_s^i \end{bmatrix} = \mathbf{P}_s^i$, $\mathbf{A}_s^i \in \mathcal{R}^{3\times3}$, and $\mathbf{B}_s^i \in \mathcal{R}^3$. From (38) we obtain the following equation with respect to $\mathbf{v}$.

$$\mathbf{K}_i - \mathbf{A}_s^i\mathbf{K}_1 = \mathbf{B}_s^i\mathbf{v}^\top \ \ (i = 1, \cdots, F). \tag{39}$$

Here, we consider $\mathbf{v}$ that minimizes the following cost function J_1.

$$J_1 = \sum_{i=1}^{F} \left\| \mathbf{K}_i - \mathbf{A}_s^i\mathbf{K}_1 - \mathbf{B}_s^i\mathbf{v}^\top \right\|^2, \tag{40}$$

where $\left\| \mathbf{A} \right\|$ stands for the Frobenius norm of a matrix $\mathbf{A}$. The solution is given as follows:

$$\mathbf{v} = \left(\sum_{i=1}^{F} \left(\mathbf{K}_i^\top - \mathbf{K}_1^\top \mathbf{A}_s^{i\top} \right) \mathbf{B}_s^i \right) \left(\sum_{i=1}^{F} \mathbf{B}_s^{i\top} \mathbf{B}_s^i \right)^{-1}. \tag{41}$$

The derivation of (41) is omitted because this is a typical application of the least square method. Matrix $\mathbf{H}$ is obtained by substituting $\mathbf{v}$ given by (41) into (37),

and thus, Euclidean reconstruction is attained by calculating $\mathbf{M}$ and $\mathbf{S}$ according to (34).

Now, we show the validity of the use of $\mathbf{W}'$ instead of $\mathbf{W}$. Under the condition (iii) of Theorem 1, each λ_{ij} does not depend on i, and thus, we can set $\lambda_{ij} = d_j$. Let the matrix $\mathbf{D}$ be $\mathbf{D} = \mathrm{diag}[d_1, \cdots, d_N]$. In this case, $\mathbf{W}(= \mathbf{MS})$ in (7) and $\mathbf{W}'$ are related by $\mathbf{W}' = \mathbf{WD}^{-1} = \mathbf{MS}'$, where $\mathbf{S}' = \mathbf{SD}^{-1} = [\mathbf{X}_1/d_1, \cdots, \mathbf{X}_N/d_N]$. Since $\mathbf{X}_i/d_i$ and $\mathbf{X}_i$ indicate the same 3D point, we do not have to estimate the exact values of d_i and $\mathbf{X}_i$ (only directions are needed), and thus, we can set $d_i = 1$ without loss of generality. Therefore, we can use $\mathbf{W}'$ instead of $\mathbf{W}$, provided that depth-estimation-free condition is satisfied.

4 Simulation

In this section, we compare the computational time of our proposed method and that of an existing method for 3D reconstruction through simulation. We use three types of simulated image sequences as shown in Fig. 1, where we call them box, cylinder, and sphere from the top, respectively.

The image sequence called box includes 100 points that are randomly placed in the cube with edge length 100 centered at $(50, 50, 150)$. The image sequence called cylinder includes 100 points that are placed on the side surface of the cylinder whose top and bottom surfaces are circle with radius 50 that are placed on the planes $Y = 100$ and $Y = 0$, respectively, and centered at $(50, 100, 150)$ and $(50, 0, 150)$, respectively. The image sequence called sphere includes 100 points that are placed on the spherical surface with radius 50 centered at $(50, 50, 150)$.

We suppose that the internal parameters of the camera for the simulation are as follows: focal length: 600; image center: (240, 160); skew: 0. We move the camera from $(0, 0, 0)$ to $(100, 0, 0)$ along X-axis of the world coordinate. During the movement, the optical axis of the camera always points Z-axis of the world coordinate so as to satisfy Assumption 4. The number of frames of each image sequence is 101.

As an existing method, we apply the method proposed in [6], which we think one of the most efficient method that have ever been proposed. The specification of the computer for the simulation is as follows: CPU: Inter(R)Core(TM): 7-2600CPU 3.40GHz; memory size: 3.49GB. For 3D reconstruction, we first find a projective reconstruction, and then, based on the result, we find a Euclidean reconstruction.

The results are shown in Table 1, where each computational time is the average of 100 trials. In Table 1, projective r., Euclidean r., and total are, respectively, the average of the computational time for the projective reconstruction, that for the Euclidean reconstruction, and sum of them. From Table 1, comparing with the existing method, we observe that the proposed method significantly reduces the computational time for 3D reconstruction because the camera is moved so as to satisfy the depth-estimation-free condition.

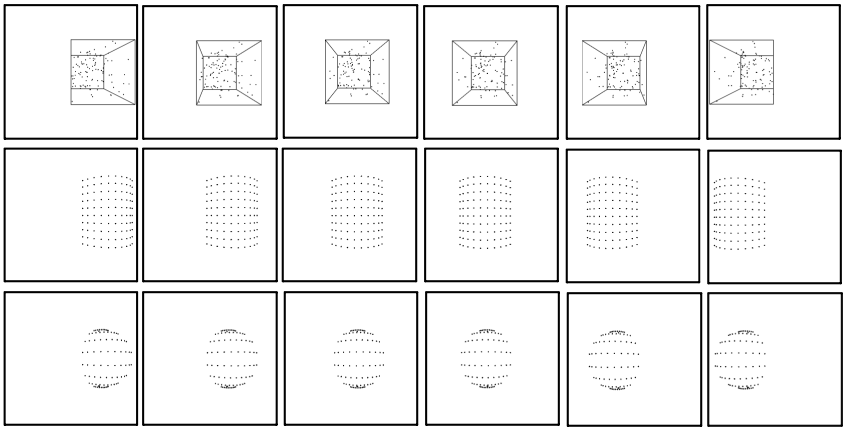


Fig. 1. Simulation data (top: box, middle: cylinder, bottom: sphere)

Table 1. Comparison of computational time (sec)

	existing method			proposed method		
	projective r.	Euclidean r.	total	projective r.	Euclidean r.	total
box	3.5402	0.0009	3.5411	0.0104	0.0004	0.0108
cylinder	2.4684	0.0010	2.4694	0.0070	0.0003	0.0073
sphere	2.8033	0.0009	2.8042	0.0081	0.0003	0.0084

5 Conclusion

In this paper, we have given a comprehensive description of the problem for the depth-estimation-free condition, which had been considered in a fragmented manner in the existing researches, and have derived a necessary and sufficient condition for depth-estimation-free projective factorization using an algebraic approach. In generic situation, the condition is as follows: origins of all camera coordinate systems are lying on a single plane, and optical axes of the coordinate systems point the same direction that is perpendicular to the plane. This condition is closely related to the conditions obtained in [8] and [9].

Furthermore, we have extended the condition to the case where points location is possibly restricted on a plane or on a line, and have obtained a generalized version of the above condition. The condition clearly reveals the trade-off between the freedom of camera motion and that of point location. In deriving the condition, the idea that restricted area is expressed by the null space of a matrix plays a crucial role.

Based on the condition, we have given a method for a Euclidean reconstruction from the result of the projective factorization. Furthermore, we have evaluated the proposed method through simulation from the viewpoint of computational time, and have shown that the proposed method significantly reduces the

computational time for 3D reconstruction compared with one of the most efficient existing method. This work is supported in part by a Grant-in-Aid for scientific research from the Japan Society of the Promotion of Science.

References

1. Sturm, P., Triggs, W.: A Factorization Based Algorithm for Multi-Image Projective Structure and Motion. In: Buxton, B.F., Cipolla, R. (eds.) ECCV 1996. LNCS, vol. 1065, pp. 709–720. Springer, Heidelberg (1996)
2. Ueshiba, T., Tomita, F.: A Factorization Method for Projective and Euclidean Reconstruction from Multiple Perspective Views via Iterative Depth Estimation. In: Burkhardt, H.-J., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1406, pp. 296–310. Springer, Heidelberg (1998)
3. Heyden, A.: An iterative factorization method for projective structure and motion from image sequences. Image and Vision Computing 17, 981–991 (1999)
4. Mahamud, S., Hebert, M.: Iterative projective reconstruction from multiple views. In: Proceedings of the 14th IEEE Conference on Computer Vision and Pattern Recognition, pp. 430–437 (2000)
5. Han, M., Kanade, T.: Multiple motion scene reconstruction from uncalibrated views. In: Proceedings of the 8th IEEE International Conference on Computer Vision, pp. 163–170 (2001)
6. Ackermann, H., Kanatani, K.: Fast projective reconstruction: Toward ultimate efficiency. IPSJ Transactions on Computer Vision and Image Media 49, 68–78 (2008)
7. Dai, Y., Li, H., He, M.: Element-Wise Factorization for N-View Projective Reconstruction. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 396–409. Springer, Heidelberg (2010)
8. Hartley, R., Zisserman, Z.: Multiple View Geometry in Computer Vision. Cambridge University Press (2004)
9. Triggs, B.: Some notes on factorization methods for projective structure and motion (1998) (unpublished)
10. Harville, D.A.: Matrix Algebra From a Statistician's Perspective. Springer (2008)
11. Sturm, P.: Critical motion sequences for monocular self-calibration and uncalibrated euclidean reconstruction. In: Proceedings of the 11th IEEE Conference on Computer Vision and Pattern Recognition, pp. 1100–1105 (1997)

Epipolar Geometry Estimation for Urban Scenes with Repetitive Structures

Maria Kushnir and Ilan Shimshoni

Department of Information Systems
University of Haifa, Israel
mkushn01@campus.haifa.ac.il, ishimshoni@mis.haifa.ac.il

Abstract. Algorithms for the estimation of epipolar geometry from a pair of images have been very successful in recent years, being able to deal with wide baseline images. The algorithms succeed even when the percentage of correct matches from the initial set of matches is very low. In this paper the problem of scenes with repeated structures is addressed, concentrating on the common case of building facades. In these cases a large number of repeated features is found and can not be matched initially, causing state-of-the-art algorithms to fail. Our algorithm therefore clusters similar features in each of the two images and matches clusters of features. From these cluster pairs, a set of hypothesized homographies of the building facade are generated and ranked mainly according the support of matches of non-repeating features. Then in a separate step the epipole is recovered yielding the fundamental matrix. The algorithm then decides whether the fundamental matrix has been recovered reliably enough and if not returns only the homography. The algorithm has been tested successfully on a large number of pairs of images of buildings from the benchmark ZuBuD database for which several state-of-the-art algorithms nearly always fail.

1 Introduction

Repeated structures are commonly seen in many types of scenes. They are especially prevalent in man made scenes such as buildings as can be seen for example in Fig. 1. For reasons which will be explained shortly, algorithms for epipolar geometry estimation from two images tend to fail on such scenes. The goal of this paper is to present an algorithm to deal with these cases. In this paper we will concentrate on building facades which are one of the most common cases of repeated structures.

In recent years there has been significant progress in developing algorithms for epipolar geometry estimation for wide baseline image pairs. Generally speaking, the algorithm is given as input two images. On both images a feature detection algorithm is run yielding a set of features and their associated descriptors (e.g., SIFT [1]). The two feature sets are then matched yielding a set of pairs of similar features from the two images. On this set of putative matches a robust algorithm from the RANSAC [2] family is run resulting in a model which in some

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 163–176, 2013.

cases is the fundamental matrix or an homography in others. The matches are also classified as inliers or outliers. On this general framework many advances have been made. The LO-RANSAC [3] algorithm performs local optimization on candidate solutions suggested by RANSAC, reducing the number of iterations. Other algorithms suggest methods to guide the selection of subsets selected by the RANSAC process [4,5,6,7]. Finally, methods were suggested to reduce the number of putative matches selected at each iteration resulting in a much faster algorithm which can deal with a much higher percentage of outliers [8,9,10].

(a) (b)

Fig. 1. Possible cases of images with repeating structures. (a) A building with repeating elements appearing on the same vertical and horizontal lines. (b) A building with repeating elements appearing periodically in a grid structure.

As a result of all these advances wide baseline stereo image registration systems are successful in many hard cases with very low inlier match percentage. However, for scenes with repeated structures they often fail. The reason for this is that repeated structures yield similar sets of local features for which humans and automated systems fail to match correctly. In most cases the algorithm is able to recognize that there are several very similar matches to such a feature and it therefore discards the feature altogether. As a result, when the overlap between the two scenes contains mainly repeated structures, the alignment algorithms tend to fail.

1.1 Related Work on Repeated Elements

In this work we will be dealing with image registration but repeated elements have been extensively studied in different contexts such as detection and grouping of similar elements [11,12,13], classification and identification [14], matching [15,16], geo-tagging and location recognition [17,18,19] as well as structure from motion methods [20].

From these works we would like to elaborate on several papers. In [20] the problem addressed was of recovering the structure from a large number of images (SfM) when the scene contains multiple instances of the same object. The

challenge which is addressed in that paper is to eliminate the incorrect fundamental matrices from the set of fundamental matrices recovered from matching all image pairs. This is done using geometric and image-based cues.

Perhaps the papers most closely related to our work, dealing with image registration and repeated patterns are [16,17,19,21,22,23,24], as they present different approaches for matching images of building facades, without analyzing or modeling of the entire structure as in [11,12,13].

In [24] a guided RANSAC algorithm is presented. A large number of putative matches is generated by matching all possible similar points but giving repeated features low probabilities. Thus, they are not used in the model generation step but only in the verification step. When the number of correct unique correspondences is small the running time of the algorithm can be long.

In [16] it is assumed that the objects investigated are comprised of planar quadrilaterals bounded by straight lines. For each hypothesized match between a pair of quadrilaterals, the homography between images is calculated. The score of the homography is given by counting the number of corresponding Harris corners within the region. It should be noted that there is no descriptor extraction for the detected Harris points, and that this method results with a projective homography, that matches two building facades without any estimation of the epipolar geometry. [19] recovers the position of a mobile robot by matching building facades. The algorithm exploits the fact that the views were obtained from similar heights, and thus matches are restricted to a narrow margin surrounding a 1D scan line. Similarly, in [21] invalid correspondences are eliminated based on geometric constraints generated from approximate knowledge of internal and external camera calibration parameters. [22,23] deal with scenes with multiple objects using an a-contrario approach. They concentrate on the post-processing step in which the algorithm has to decide which of the matches belong to the current solution. Finally, [17] extracts calibrated images from an existing database and matches it to an input image. The transformation supported by the maximal number of matches is returned. It is therefore possible that a shifted solution will be returned by the algorithm.

1.2 Our Approach

In this paper we suggest an algorithm to deal with the case of repeated structures placed on planar or close to planar surfaces. The main application of such an algorithm is for images containing mainly building facades. Without such an algorithm, systems with an image registration component will fail from time to time unexpectedly when the overlap between the images are mainly building facades.

The algorithm exploits three important characteristics of the scenes that we are dealing with. First, a large number of repeated structures lie on a planar surface in an ordered fashion and second that the local feature descriptors detected in the image can be clustered and the clusters between the two images can be matched without determining initially how the individual members of a matched cluster are matched. Using the repeated features usually several possible

solutions are generated with similar support. We therefore extract the small number of regular unique matches in order to select the correct solution, in contrast to the methods described above.

The algorithm divides the task of epipolar geometry estimation into two steps. It first recovers the homography associated with the building's facade and then recovers the epipole. Finally it decides whether the fundamental matrix is reliable and if not returns only the homography.

The paper continues as follows. In the next section we will present our general approach. In Section 3 we present experimental results run on challenging image pairs from the ZuBuD database of images of buildings from Zurich [25] for which general purpose state of the art algorithms usually fail. We compare our method to a SIFT [1] matching step followed by a standard RANSAC [2] and to two state-of-the-art wide baseline registration algorithms BEEM [10] and BLOGS [7]. Conclusions and plans for future work are discussed in Section 4.

2 The Algorithm

Scenes with repeated structures are very common. In this paper we will concentrate on the special case where most of the repeated elements lie on planar surfaces or close to planar surfaces such as building facades.

In our algorithm we consider two cases. In the first case we only assume that the repeated objects are partially organized horizontally or vertically (Fig. 1(a)). In the second case we assume that there exists a grid of repeated objects (Fig. 1(b)). We will first describe the algorithm which deals with the non-periodic case and then in Section 2.4 the modifications required to deal with grids of repeated objects will be presented.

When two images containing a planar surface with repeated objects are given, the first step of the algorithm (described in Section 2.1) is to find for each image an homography which will transform the image into a fronto parallel view. This step is performed for two reasons. First, eliminating the projective distortion makes the descriptors recovered from the repeated features more similar and thus easier to cluster. Second, when given two fronto-parallel images of a planar surface, the transformation between them is much simpler. All that has to be recovered, is the 2D translation and the scale factor.

On each of the rectified images SIFT features are extracted and features with similar descriptors are clustered. We then match pairs of clusters from the two images. There are of course features which do not cluster and will be called non-repeating features.

In Section 2.2 we generate a set of hypothesized transformations of the plane appearing in both rectified images. This is done by matching minimal subsets of features from a cluster generated from the first image to a subset of features from its corresponding cluster from the second image. The hypothesized transformations are ranked by the number of matched features that satisfy $\mathbf{x}' = H\mathbf{x}$.

In Section 2.3 we exploit the fact that the fundamental matrix F can be factored into $F = [\mathbf{e}']_{\times} H$. Therefore it can be computed by estimating the

epipole $\mathbf{e}'$ for a given homography. Once F has been found the algorithm decides whether there is enough evidence to support it, and if not it returns only H.

2.1 Image Rectification

In our algorithm, we use the Canny edge detector, to detect edges and from them extract line segments in the image. Then, we apply RANSAC [2] twice to find the vertical vanishing point $\mathbf{V}_{\mathbf{p}_v}$ and the horizontal vanishing point $\mathbf{V}_{\mathbf{p}_h}$, although as will be shown later, our method can handle a swap in those directions.

Under the standard assumptions of square pixels, zero skew, and that the principal point is at the image center, the internal calibration matrix K and the rotation matrix R can be recovered [26, Chapter 8]. Consequently, the original image is rectified by: $H = KRK^{-1}$, resulting in a fronto-parallel view. An example of the results of this procedure can be seen in Fig. 2.

(a) (b)

Fig. 2. Image rectification. (a) The original image with detected line segments, that are consistent with the two vanishing points. (b) Fronto-parallel rectified image.

From the rectified images we extract SIFT features and descriptors (using the implementation provided by [27]). This step is performed on the rectified images, since in the case of repeated features, descriptors are more similar due to elimination of the projective distortion. In general, each SIFT key-point can be assigned with an orientation, based on the local image gradient direction, which is the key step in achieving invariance to rotation. In our case, we use upright SIFT key-points, for which the key-point orientation is set to be vertical. The single fixed orientation for all features is a natural choice, given that the rotation is compensated through the rectification. Moreover, it prevents features such as for example window corners of different orientations to be considered as the same feature.

We then cluster the SIFT key-points within a single image. Repeating points are identified and clustered if their appearances are similar, i.e., their normalized cross correlation is larger than a threshold (in our experiments 0.9). For each cluster, we select the medoid of the repeating points' descriptors as the cluster descriptor. The result of this step is not perfect. Not all clusters represent real

repeating objects and not all repeating objects are represented by a feature in a cluster of a repeating feature. Still as can be seen in the supplementary material, the number of recovered clusters can be used to differentiate between images with or without repeating structures.

2.2 Planar Homography Estimation

We start the image registration process by searching for a specific transformation H, induced by the rectified plane with repeating elements on it, that maps one rectified image to another. For that purpose we assume that repeating key-points, appear on the same vertical and horizontal lines, without specific requirements about distances or periodicity. We build a list of candidate transformations.

As a first step, we match key-point clusters from the rectification stage. We check all possible cluster pairs from two images and compute the Euclidean distance between the cluster descriptor vectors of each pair. Similar to Lowe's approach, we define the best match as the one with minimal distance. We determine the probability that a cluster match is correct, by taking the ratio of distance from the closest neighbor to the distance of the second closest. Small clusters (smaller than 5 points), or those that do not have any good match (distance ratio larger than 0.8) are discarded.

When searching for all possible homographies, we use the rectified images extracted previously. Due to that, both transformed images are fronto-parallel. As a result, instead of searching for eight degrees of freedom of a general projective transformation H, we are left with only three, namely the two coordinates of a relative translation t_x,t_y and the relative scale s

$$H = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \implies \begin{bmatrix} s & 0 & t_x \\ 0 & s & t_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{1}$$

To detect H candidates, we check all the feature points from two images. For each point of the first image $\mathbf{x}_c$ we try to find its approximate vertical $\mathbf{x}_v$ and horizontal $\mathbf{x}_h$ nearest neighbors within the same cluster if they exist. Such a point triplet will be denoted $\mathcal{T}$. We perform an identical procedure on the second image yielding point triplets, each denoted $\mathcal{T}'$. We then match pairs of point triplets from the two rectified images, belonging to matched clusters. Every such matched triplet $\mathcal{MT} = \{\mathcal{T}, \mathcal{T}'\}$ is used to compute a transformation H. In general two feature points from each image would be sufficient for transformation estimation, but relying on triplets gives rise to less candidates to handle and much more accurate results. Exploiting the scale constraint, we eliminate transformations that do not satisfy it. When the triplet strategy fails, we resort to using pairs of feature points from each image. In general the scale ratio is not always accurately estimated. In this case there are two scales s_x s_y instead of one s. We have implemented both versions dealing with two/one scale and they both always succeeded.

For each candidate transformation, other feature point pairs from different clusters which satisfy the transformation relation are accumulated and are

considered point matches which support the transformation. Based on them, we improve the accuracy of H using LO-RANSAC [3] as follows. We iteratively calculate an homography based on randomly selecting half of supporting point matches, and compute H using a non-linear method which minimizes Sampson's approximation to the geometric re-projection error. As a result we obtain, an accurate homography, which relies on many points, instead of the single localized triplet and is immune to the inaccuracies of the rectification procedure and the proximity between the point triplet. The result of this step is a list of candidate homographies.

(a) (b)

Fig. 3. Typical results, when building a list of all possible homographies. (a) Blue stars: An arbitrary point with its vertical and horizontal nearest neighbors within the same cluster. Red points: Additional points from the same cluster, that support the same H. (b) Blue stars: correct point match and its nearest neighbors. Green squares: An alternative point match.

To illustrate the process we present in Fig. 3 a typical result. In both images two point triplets are marked by blue stars. These two point triplets can be used to compute the correct H. The red points in Fig. 3(a) indicate additional points that support that H. The green squares in Fig. 3(b) represent an alternative point triplet from which an additional H candidate is computed.

Homography Ranking. Once the set of homographies has been generated the next task is to rank them. In order to better deal with this issue, we assume that not only repeating elements appear on the plane, but also several unique key-points, which we plan to exploit to break the symmetry. We therefore match the SIFT key-points from the two rectified images, by the standard technique, proposed by Lowe.

We rank each homography from the list, by the number of key-point matches that are consistent with it. If the homography H is the correct one, it should return not only repeating key-points from several clusters, but also corresponding locations of unique SIFT key-points. Therefore, we sort homographies based on the following score:

$$S_H = N_{rep} + \alpha N_{non-rep}, \tag{2}$$

where α is a weight constant and N_{rep} and $N_{non-rep}$ are the numbers of supporting correspondences from repeating and non-repeating key-points respectively. In our experiments we set $\alpha = 100$ (the algorithm works for $\alpha > 10$), to emphasize that we mostly rely on the small number of matches of unique key-points, to rank the homographies. When sorted, we iteratively check the homography with the maximal S_H until no improvement is reached.

One of the advantages of our method is that we can empirically tune this weight constant α, by changing the preference of one type of key-points over the other. In [16,17,19], this would be impossible, since only the number of matches is counted. As a result, for an image pair with partial occlusion of the repeating elements, the homography H having the maximal overlap would be chosen, as there are naturally more repeating key-points in the images. In our method on the other side a few highly weighted non-repeating key-points would be sufficient to detect the correct H, regardless of occlusion or a partially non-overlapping scene.

In addition, there always is a possibility that when rectifying one of the images, horizontal and vertical directions were swapped. This is especially common when the original images were taken with a roll angle of approximately 90°, as shown in Fig. 2(b). We therefore keep the first rectified image unchanged and check for three possible alignments of the second image: the one obtained from its rectification and the two rotations by $\pm 90°$. We also change all the key-point locations and descriptors respectively. For each one of the three alignments, we rank the homographies as described above.

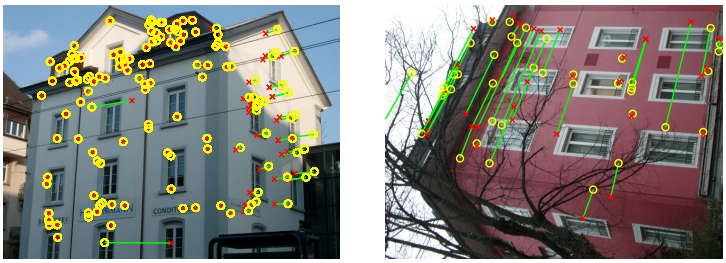
2.3 Image Registration

After the homographies have been ranked, for each one of them a RANSAC process will be run to estimate the epipole $\mathbf{e}'$. Combining it with the homography H yields F. When looking for the correct fundamental matrix F, we assume that the repeating elements are bounded to the underlying plane, and therefore they are not considered in this step. Matching correspondences of the non-repeating key-points however, can appear on, as well as off the plane. Thus, we select non-repeating key-points, that can contribute to the estimation of F. Those point pairs $\{\mathbf{x}_i, \mathbf{x}'_i\}$ must satisfy:

$$\|H\mathbf{x}_i - \mathbf{x}'_i\| = \|\mathbf{x}''_i - \mathbf{x}'_i\| \propto |\rho_i| > d_{proj}, \tag{3}$$

where $\mathbf{x}''_i = H\mathbf{x}_i$, ρ_i is the projective depth, relative to the underlying plane, and d_{proj} is a constant distance threshold. In our experiments d_{proj} was set to five pixels.

In Fig. 4(a) we show non-repeating key-points $\mathbf{x}'_i$ marked by yellow circles and $\mathbf{x}''_i$ by red crosses. The green lines are proportional to the projective depth ρ_i of the pairs. For most of the in-plane key-points, we can see the yellow circles merge with the red crosses, which indicates zero projective depth. There are only two mistaken matched pairs, that are associated with a visible green line, despite being on the plane. In addition, we can observe for the off-plane points, that the further the point is from the plane, the longer is the green line associated with it.



(a) (b)

Fig. 4. (a) Matching correspondences of the non-repeating key-points. Original image with non-repeating key-points $\mathbf{x}'_i$ marked by yellow circles and $\mathbf{x}''_i$ by red crosses. (b) Wrong matches of non-repeating key-points, that result from repetitive elements. Original image with non-repeating key-points $\mathbf{x}'_i$ marked by yellow circles and $\mathbf{x}''_i$ by red crosses. Green lines are proportional to the projective depth ρ_i.

Another problem demonstrated in Fig. 4(b) is of putative matches, which are due to incorrect matches between repetitive features that were not detected as such during the clustering phase. In general RANSAC is able to deal with outliers. However, when the feature pairs lie on a horizontal or vertical line on the facade as can be seen in the figure, these incorrect matches will vote together for an incorrect epipole, the horizontal or vertical vanishing point which in many cases will produce an incorrect solution. We therefore remove these putative matches from consideration. These removed matches satisfy $(H\mathbf{x}-\mathbf{x}')\times\mathbf{V_p}\approx 0$.

All the remaining matches, termed candidate F supporters, are used in the RANSAC step to recover the epipole $\mathbf{e}'$. The candidate H and the recovered $\mathbf{e}'$ will then be combined to yield the fundamental matrix $F=[\mathbf{e}']_\times H$. In this step the putative matches come from two sources: matched features extracted from the rectified images which mainly come from the parallel planes consisting of the building's facade and matched features extracted from the original images. These matches usually come from off-plane 3D points, since they become too distorted in the rectification process to be matched using the rectified images.

Once the RANSAC step has been completed all the matches that support the fundamental matrix F (including the ones that support the homography) are given to a final RANSAC step which recovers the homography H accurately.

The question that remains is whether the algorithm should return F or that there is not enough evidence to support a fundamental matrix (when for example the overlap between the two images is close to planar) and only H should be returned. We answer this question by counting the number of matches that support F and do not support H. If there are more than a certain number of supporters (10 in our experiments) F is returned by the algorithm and if not, only H is returned.

2.4 Grid of Repeating Structures

In this section we switch to a more demanding case than discussed earlier, the case of periodic repeating elements. We describe here the additional steps and required changes to the full algorithm flow, previously presented.

During the first step of image rectification and key-points extraction, additional information can be evaluated. Assuming periodicity, we estimate optimal horizontal and vertical repetition intervals separately for each rectified image. We define the difference between every pair of intersections, of all the detected line segments with one of the axes, as a possible horizontal or vertical repetition interval. In other words, if two intersection points x_1 and x_2 support an interval I then $x_1 - x_2 = kI$ for some integer k or $\mod(x_1, I) = \mod(x_2, I)$.

Therefore, for every possible interval I, we build a histogram h_n at a resolution of one pixel, of $\mod(x_i, I)$. Thus, the number of supporting pairs of lines for an interval will be

$$N_I = \sum_{n=0}^{\lceil I-1 \rceil} h_n(I)(h_n(I) - 1)/2. \tag{4}$$

If an interval I is a good candidate, we expect to have sharp peaks in the histogram, coming from the unification of repeating lines' intersection. An example of such a histogram for a good vs. bad candidate can be seen in Fig. 5. However, for intervals $\{\frac{I}{2}, \frac{I}{3}, \frac{I}{4} \ldots\}$, it is expected to have even sharper histograms.

Thus, the score for interval is set to $S_I = N_I I$ to induce a preference for I and not for its fractions. The algorithm builds a list of several (three in our implementation) candidates for I with the maximal scores. The value of $S_I(I_{max})$ can be used to detect images with a grid of repeating structures, as can be seen in the submitted supplementary material.

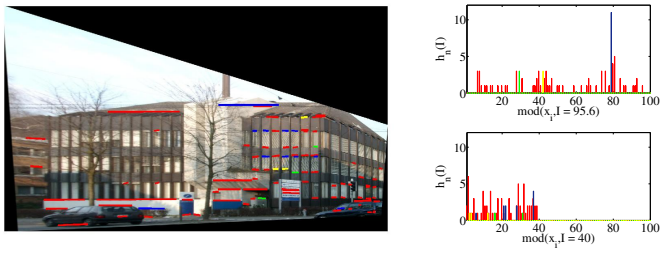


Fig. 5. Estimation of optimal I_y. Rectified image with all the detected horizontal line segments and histograms for good ($I_y = 95.6$) and bad ($I_y = 40$) candidates of I_y.

It is during the next step, when building a list of all possible transformations between the two rectified images, we exploit the list of optimal horizontal and vertical repetition intervals extracted previously. We compute the relative scale s from Eq. 1, by:

$$s = I_{x_2}/I_{x_1} \qquad s = I_{y_2}/I_{y_1}, \tag{5}$$

where I_{x_i} and I_{y_i} are the horizontal and vertical repetition intervals in the rectified image respectively. Eq. 5 is used to select sets of consistent interval values. Thus, when estimating the transformation between the two rectified images we are left with only two out of eight degrees of freedom of a general projective transformation H: the two coordinates of the relative translation t_x and t_y.

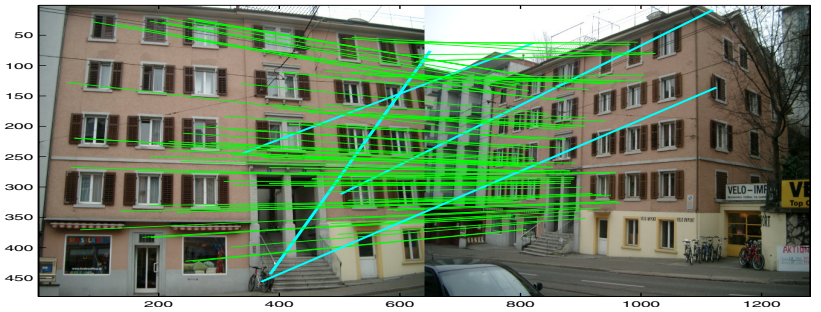
In order to detect H candidates, following a similar strategy as in the non-periodic case, it is required to check all the points from two images belonging to matched clusters and compute the translation between each pair. Every such point pair could yield a transformation H. As a result there would be many more candidates in this case than in the non-periodic case. Therefore, we change our approach slightly. As we have mentioned earlier, we assume that if the H is correct, it should be supported not only by repeating key-points, but also by corresponding locations of unique SIFT key-points. Thus, it is possible to build H candidate from each non-repeating key-point correspondence. Identically to the non-periodic case, for each candidate transformation, all feature point pairs from different clusters which satisfy the transformation relation are found and are considered point matches which support the homography. Based on them we compute H using LO-RANSAC [3]. The output of this step is a list of candidate homographies. The following steps of homography ranking and image registration are identical to those in the non-periodic case.

3 Experimental Results

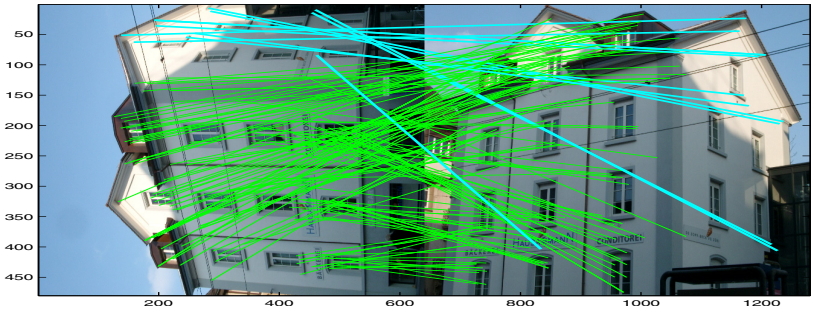
We will now present experimental results of our implementation of the algorithm. We ran experiments with the same settings on all the results included in this work. We used the publicly available ZubuD database [25] to test our method. The database contains 1005 color images of 201 buildings (5 images per building) of scenes in Zurich, taken from different viewpoints and illumination conditions.

As we were interested in the additional value that our method can contribute, we compared it to the state-of-the-art wide baseline registration algorithms BLOGS [7] and BEEM [10], which can estimate the epipolar geometry in many difficult cases. We first automatically selected all the image pairs, that at least one of them failed to find a correct fundamental matrix for. We successfully ran our algorithm on 20 such image pairs of different buildings. Due to the low number of unique feature points in those images BLOGS succeeded only for 4 image pairs, whereas BEEM succeeded for 3. We also verified that a SIFT matching step followed by a standard RANSAC implementation failed on all image pairs. The results of running our algorithm on all of them, as well a comparison to other registration algorithms are included in the supplementary material submitted with this paper. They include the image pairs and a table presenting for each run numerical results of the various steps of the algorithm.

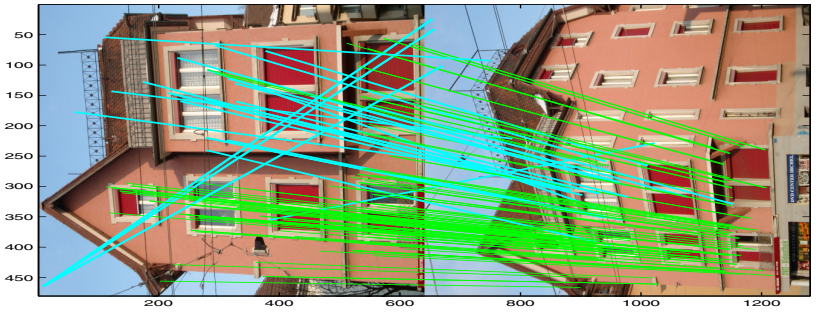
In Fig. 6 we present four representative results. For each image pair we present non-repeating key-points that are inliers of a fundamental matrix F. The key-points, that are also inliers of the homography H are connected with green lines and, those that were considered as F supporters are connected with the cyan lines.



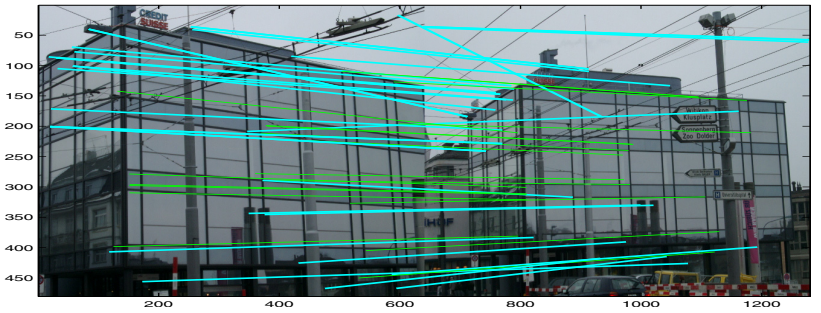
(a) object0010. Our method: H type.



(b) object0033. Our method: F type.



(c) object0066. Our method: F type.



(d) object0131. Our method: P-F type.

Fig. 6. Experimental results of our method on object0010, object0033, object0066 and object0131 from the ZuBuD database

In Fig. 6(a) we can see an example of a fully planar case, since there is only one building facade in the left image. As a result, there is an infinite number of fundamental matrices that could be chosen, one of which is shown here. In that case, as discussed earlier, the confidence in F is low due to a small number of its supporters (cyan lines) and we report only the recovered H with its inliers (green lines).

Fig. 6(b), on the contrary presents both images having two facades. As a consequence, we obtain a large number of key-points at different depths and report a fundamental matrix F along with its inliers. We can clearly see a color differentiation between on-plane and off-plane key-points. Key-points located on the plane are connected by green lines, whereas off-plane matches are in cyan.

A building with two parallel planes on the same facade is presented in Fig. 6(c). In this situation, the correct H, maps only one of the planes, whereas the key-points from the other have different depths and are colored in cyan. In this example the correct H maps key-points from the inner plane. The matches from the other plane are used to estimate the fundamental matrix correctly.

Finally, in Fig. 6(d) we demonstrate a periodic case. Here there are enough matches on the second facade to estimate the fundamental matrix correctly.

4 Conclusions and Future Work

In this paper we presented a wide baseline registration algorithm for scenes of building facades with repeating structures. The algorithm was implemented and tested successfully on a large number of image pairs for which general state-of-the-art algorithms usually fail.

Future research will be dedicated to developing an algorithm that can deal also with scenes of non-planar man-made objects and natural scenes with repeating objects.

Acknowledgment. This research was supported by the VULCAN consortium of the ministry of industry and commerce.

References

1. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)
2. Fischler, M., Bolles, R.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, 381–395 (1981)
3. Chum, O., Matas, J., Kittler, J.: Locally optimized random sample consensus. In: German Pattern Recognition Symposium, pp. 236–243 (2003)
4. Tordoff, B., Murray, D.W.: Guided Sampling and Consensus for Motion Estimation. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part I. LNCS, vol. 2350, pp. 82–96. Springer, Heidelberg (2002)
5. Chum, O., Matas, J.: Matching with PROSAC progressive sample consensus. In: CVPR, pp. 220–226 (2005)
6. Goshen, L., Shimshoni, I.: Guided sampling via weak motion models and outlier sample generation for epipolar geometry estimation. IJCV 80, 275–288 (2008)

7. Brahmachari, A., Sarkar, S.: BLOGS: Balanced local and global search for non-degenerate two view epipolar geometry. In: ICCV, pp. 1685–1692 (2009)
8. Schaffalitzky, F., Zisserman, A.: Multi-view Matching for Unordered Image Sets, or How Do I Organize My Holiday Snaps? In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part I. LNCS, vol. 2350, pp. 414–431. Springer, Heidelberg (2002)
9. Chum, O., Matas, J., Obdrzalek, S.: Enhancing RANSAC by generalized model optimization. In: ACCV, pp. II:812–II:817 (2004)
10. Goshen, L., Shimshoni, I.: Balanced exploration and exploitation model search for efficient epipolar geometry estimation. PAMI 30(7), 1230–1242 (2008)
11. Wu, C., Frahm, J.-M., Pollefeys, M.: Detecting Large Repetitive Structures with Salient Boundaries. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 142–155. Springer, Heidelberg (2010)
12. Wenzel, S., Drauschke, M., Forstner, W.: Detection of repeated structures in facade images. PRAI 18, 406–411 (2008)
13. Jiang, N., Tan, P., Cheong, L.: Multi-view repetitive structure detection. In: ICCV (2011)
14. Liu, Y., Collins, R., Tsin, Y.: A computational model for periodic pattern perception based on frieze and wallpaper groups. PAMI 26, 354–371 (2004)
15. Hays, J., Leordeanu, M., Efros, A.A., Liu, Y.: Discovering Texture Regularity as a Higher-Order Correspondence Problem. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 522–535. Springer, Heidelberg (2006)
16. Lee, J., Yow, K., Chia, A.S.: Robust matching of building facades under large viewpoint changes. In: ICCV, pp. 1258–1264 (2009)
17. Baatz, G., Köser, K., Chen, D., Grzeszczuk, R., Pollefeys, M.: Handling Urban Location Recognition as a 2D Homothetic Problem. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part VI. LNCS, vol. 6316, pp. 266–279. Springer, Heidelberg (2010)
18. Schindler, G., Krishnamurthy, P., Lublinerman, R., Liu, Y., Dellaert, F.: Detecting and matching repeated patterns for automatic geo-tagging in urban environments. In: CVPR, pp. 1–7 (2008)
19. Robertson, D., Cipolla, R.: An image-based system for urban navigation. In: BMVC, pp. 819–828 (2004)
20. Roberts, R., Sinha, S., Szeliski, R., Steedly, D.: Structure from Motion for Scenes with Large Duplicate Structures. In: CVPR, pp. 3137–3144 (2011)
21. Serradell, E., Özuysal, M., Lepetit, V., Fua, P., Moreno-Noguer, F.: Combining Geometric and Appearance Priors for Robust Homography Estimation. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part III. LNCS, vol. 6313, pp. 58–72. Springer, Heidelberg (2010)
22. Rabin, J., Delon, J., Gousseau, Y., Moisan, L.: MAC-RANSAC: a robust algorithm for the recognition of multiple objects. In: 3DPVT (2010)
23. Sur, F., Noury, N., Berger, M.O.: Image point correspondences and repeated patterns. Technical Report RR-7693, INRIA (2011)
24. Zhang, W., Kosecka, J.: Generalized RANSAC framework for relaxed correspondence problems. In: 3DPVT, pp. 854–860 (2006)
25. Shao, H., Svoboda, T., Gool, L.: ZuBuD Zurich Buildings Database for Image Based Recognition. In: Technical Report 260, CVL, ETH Zurich (2003)
26. Hartley, R., Zisserman, A.: Mltiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press (2004)
27. Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms (2008)

Non-rigid Self-calibration of a Projective Camera

Hanno Ackermann and Bodo Rosenhahn

Leibniz University Hannover

Abstract. Rigid structure-from-motion (SfM) usually consists of two steps: First, a projective reconstruction is computed which is then upgraded to Euclidean structure and motion in a subsequent step. Reliable algorithms exist for both problems. In the case of non-rigid SfM, on the other hand, especially the Euclidean upgrading has turned out to be difficult. A few algorithms have been proposed for upgrading an *affine* reconstruction, and are able to obtain successful 3D-reconstructions. For upgrading a non-rigid *projective* reconstruction, however, either simple sequences are used, or no 3D-reconstructions are shown at all.

In this article, an algorithm is proposed for estimating the self-calibration of a projectively reconstructed non-rigid scene. In contrast to other algorithms, neither prior knowledge of the non-rigid deformations is required, nor a subsequent step to align different motion bases. An evaluation with synthetic data reveals that the proposed algorithm is robust to noise and it is able to accurately estimate the 3D-reconstructions and the intrinsic calibration. Finally, reconstructions of a challenging real image with strong non-rigid deformation are presented.

1 Introduction

Approaches for *rigid* structure-from-motion (SfM) usually consist of two steps. Given 2D-feature correspondences between several images, a projective reconstruction is estimated which is identical to the true solution up to a projective transformation. In a second step, usually referred to as *self-calibration* or *auto-calibration*, this projective distortion is removed by imposing a certain structure on the motion matrices [1]. Assuming the basis model introduced by Bregler *et al.* in [2], we consider the problem of computing the self-calibration of a *projective* camera which observes a *non-rigidly* deforming body or scene. We assume that this camera has an unknown focal length which may vary or be constant, zero skew and principal point at the origin. Furthermore, the proposed algorithm is more general than other works as particular non-rigid deformations need not be known.

Self-calibrating a projective camera can be considered a mature field if the observed body is *rigid* [3,4,5,6].

In the case of a *non-rigid* body observed by an *affine* camera, Xiao *et al.* [7] proposed a linear solution. Brand [8] suggested an algorithm in which the motion constraints are first imposed for a particular, arbitrarily chosen deformation mode, and all other deformation modes are corrected with respect to the

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 177–190, 2013.

initially chosen one, an approach which is non-optimal as the error is concentrated in all deformation modes but the reference one. Olsen and Bartoli [9] used a smoothness prior on the camera motion to determine the self-calibration. Torresani *et al.* [10] imposed the prior knowledge that the coefficients of non-rigid deformation satisfy a Gaussian distribution. In a seminal work, Paladini *et al.* [11] introduced an iterative projection algorithm which alternates unconstrained optimization with projection of the motion matrices to the required structure.

To this day, only two algorithms consider the problem of self-calibrating a *projective* camera observing a body deforming non-rigidly. Xiao and Kanade [12] extended their work from [7] to a projective camera with constant focal length. Hartley and Vidal [13] proposed a method which requires that the intrinsic camera parameters are fixed and known. Similar to [8] they first correct a particular, arbitrarily chosen deformation mode. Remaining modes are subsequently estimated with respect to the previously corrected ones. While being an elegant, non-iterative solution, no 3D-reconstructions are shown in this article.

In this article, an algorithm is presented for self-calibration of a projective camera observing a non-rigidly deforming object. It is assumed that the skew is zero, the focal length unknown while varying or being constant throughout the sequence, and the principal point is at the origin. Though seemingly similar to the requirements in [12], the current work does not demand particular non-rigid deformation coefficients to be known. Furthermore, the proposed algorithm does not require a second step (*Orthogonal Procrustes Analysis*) to enforce identical rotations. The advantage is that the error should be more fairly distributed between the bases. To align the bases, additional constraints are necessary. We therefore generalize the equations introduced by Brand [14] to the projective camera model. It is proven that the solution is unique up to a global rotation and reflection of the world coordinate system and individual scalings of each basis. The accuracy of the proposed algorithm is evaluated with experiments on synthetic data. Furthermore, 3D-reconstructions are presented for a challenging real-image sequence showing a body with strong local and global non-rigid deformation.

This work is structured as follows: In Section 2, the problem of self-calibrating a projective camera observing a non-rigidly deforming body or scene is defined. Constraints by which the problem can be determined are derived in Section 3. It will be proven that these constraints are necessary and sufficient to obtain the required structure of the motion matrices. Synthetic and real image experiments are presented in Section 4 before a summary and conclusions in Section 5.

Capital letters denote matrices, bold capital letter scalar constants and bold lower-case letters vectors. Normal lower-case letters denote scalar variables or counters.

2 Problem Definition

Let there be $\mathbf{K}$ $4 \times n$ *basis shape matrices* X_k, $k = 1, \ldots, \mathbf{K}$, consisting of n homogeneous 3D-points X^j, $j = 1, \ldots, \mathbf{N}$, each, $\mathbf{M}$ images with the 3×4

projection matrices P^i, $i = 1, \ldots, \mathbf{M}$ and mixing coefficients α_k^i blending the $\mathbf{K}$ basis shapes

$$\lambda_{ij}\mathbf{x}_{ij} = P^i \left(\sum_{k=1}^{\mathbf{K}} \alpha_k^i X_k \right). \tag{1}$$

The linear mixing model was introduced by Bregler *et al.* [2] for an affine camera model. Here, the scalars λ_{ij} are the *projective depths* necessary for Eq. (1) to hold true under perspective projection. The projection matrices P^i consist of the orientations R^i, positions $\mathbf{t}^i$ and calibrations K^i of the cameras[1]

$$P^i = K^i \left[R^i | \mathbf{t}^i \right], \qquad K^i = \begin{bmatrix} f_i & 0 & 0 \\ 0 & f_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

with f_i being the unknown focal length of the ith camera.

It can be seen that the measurement matrix W consisting of all 2D-features $\mathbf{x}_{ij}$ rescaled with the correct projective depths λ_{ij} has rank $3\mathbf{K} + 1$ if the two matrices P and X each have rank $3\mathbf{K} + 1$

$$W = \begin{bmatrix} \lambda_{11}\mathbf{x}_{11} & \cdots & \lambda_{1n}\mathbf{x}_{1\mathbf{N}} \\ \vdots & \ddots & \vdots \\ \lambda_{\mathbf{M}1}\mathbf{x}_{\mathbf{M}1} & \cdots & \lambda_{\mathbf{M}\mathbf{x}\mathbf{M}\mathbf{N}} \end{bmatrix} = \underbrace{\begin{bmatrix} \alpha_1^1 K^1 R^1 & \cdots & \alpha_{\mathbf{K}}^1 K^i R^1 & K^i \mathbf{t}^1 \\ \vdots & & & \vdots \\ \alpha_1^m K^{\mathbf{M}} R^{\mathbf{M}} & \cdots & \alpha_{\mathbf{K}}^{\mathbf{M}} K^{\mathbf{M}} R^{\mathbf{M}} & K^{\mathbf{M}} \mathbf{t}^{\mathbf{M}} \end{bmatrix}}_{P} \cdot \underbrace{\begin{bmatrix} X_1 \\ \vdots \\ X_{\mathbf{K}} \\ \mathbf{1} \end{bmatrix}}_{X} \tag{3}$$

Given all projective depths λ_{ij}, for instance by the algorithms proposed in [15,16], the matrix W can be factorized by singular value decomposition by Eq. (1)

$$W = U \Sigma V^\top, \tag{4}$$

where $U \in \mathbb{R}^{3m \times (3\mathbf{K}+1)}$, $\Sigma \in \mathbb{R}^{(3\mathbf{K}+1) \times (3\mathbf{K}+1)}$, and $V \in \mathbb{R}^{(3\mathbf{K}+1) \times n}$. We may consider U as projectively distorted camera matrices P, and ΣV as structure matrix X perturbed by the inverse distortion.

The problem of non-rigid projective self-calibrating is to determine a $(3\mathbf{K} + 1) \times (3\mathbf{K})$ matrix A which transforms U such that UA satisfies the required structure of the first $3\mathbf{K}$ columns of P, *i.e.* each row triple of UA must consist of scaled instances of a rotation R^i distorted by some K^i.

[1] With some risk of confusion, we use the symbol K^i for the intrinsic camera calibration in the ith image whereas the bold letter $\mathbf{K}$ denotes the number of basis shapes.

3 Deriving Constraints on Non-rigid Self-calibration

Let U^i denote the ith row triple of U. Straightforwardly applying the derivation of the dual absolute quadric of rigid scenes to the non-rigid case, we arrive at

$$\omega_i = \mathbf{K} \begin{bmatrix} f_i^2 & 0 & 0 \\ 0 & f_i^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \frac{1}{\gamma_i^2 \beta_i} U^i A A^\top {U^i}^\top \tag{5}$$

where ω_i denotes the dual image of the absolute conic $\omega_i = K_i K_i^\top$ at image i, $\beta_i = \left((\alpha_1^i)^2 + \cdots + (\alpha_K^i)^2\right)$, and the scalars γ_i account for the perspective projection in image i. The positive-semidefinite $(3\mathbf{K}+1) \times (3\mathbf{K}+1)$ matrix $\Omega_\infty = AA^\top$ of rank $3\mathbf{K}$ is the extension of the dual absolute quadric to the non-rigid case.

It is obvious that Eq. (5) is ambiguous: any change in γ_i, for instance can be compensated by a scaling of β_i. Similarly, and scaling of all α_k^i, $i = 1, \ldots, \mathbf{M}$ requires an inverse scaling on the kth structure basis X_k.

Given ω as defined in Eq. (5), we can obtain four equations per image for determining $\Omega_\infty = AA^\top$

$$\mathbf{u}_a^{i\top} A A^\top \mathbf{u}_b^i = 0, \tag{6a}$$

$$\mathbf{u}_a^{i\top} A A^\top \mathbf{u}_a^i - \mathbf{u}_b^i A A^\top \mathbf{u}_b^i = 0 \tag{6b}$$

where $\mathbf{u}^{i\top}_{\{a,b\}}$, $a \neq b$, denotes the first, second, or third row of U^i. Equations (6) are the so-called *orthogonality constraints* derived by Xiao *et al.* for the problem of self-calibrating an affine [7] or projective camera [12].

While it seems straightforward to determine Ω_∞ by solving Eq. (6), it was shown that even the affine problem is indeterminate [7,17]. With a slight risk of confusion, denote by P^i the row triple corresponding to image i of matrix P in Eq. (3). In the case of a projective camera, we obtain for the ambiguity:

Lemma 1. *Let there be a* $3\mathbf{K} \times 3\mathbf{K}$ *matrix* D*,*

$$D = \begin{bmatrix} d_{11}O_1 & d_{12}O_2 & d_{13}O_3 & \\ d_{21}O_1 & d_{22}O_2 & d_{23}O_3 & \cdots \\ & \vdots & & \end{bmatrix}, \tag{7}$$

where d_{ab} *are scalar factors and the* 3×3 *matrices* O_c, $c = 1, \ldots, \mathbf{K}$*, are arbitrary elements of the orthogonal group,* i.e. $O_c O_c^\top = I$.

Then, Eqs. (6) *are always satisfied for* $\Omega_\infty = DD^\top$*, yet* P^i *and* $P^i D$ *are not invariant up to a similarity transformation.*

Proof. Assume a general deformation matrix

$$D = \begin{bmatrix} d_{11}D_{11} & \cdots & d_{1K}D_{1K} \\ \vdots & \ddots & \vdots \\ d_{K1}D_{K1} & \cdots & d_{KK}D_{KK} \end{bmatrix}, \tag{8}$$

where the 3×3 matrices D_{ab}, and the scalars d_{ab} are arbitrary. Letting $S = DD^\top$, S_{ab} 3×3 blocks of S, l_{ab} be sums of the d_{ab}, and

$$S' = \left[\left(l_1^i l_1^i S_{11} + \cdots + l_K^i l_1^i S_{K1}\right) + \cdots + \left(l_1^i l_K^i S_{1K} + \cdots + l_K^i l_K^i S_{KK}\right)\right], \quad (9)$$

we obtain the three equations

$$\gamma_i^2 \beta_i f_i^2 = {\mathbf{r}^i}_1^\top S' \mathbf{r}^i{}_1 \quad (10a)$$

$$\gamma_i^2 \beta_i f_i^2 = {\mathbf{r}^i}_2^\top S' \mathbf{r}^i{}_2 \quad (10b)$$

$$0 = {\mathbf{r}^i}_a^\top S' \mathbf{r}^i{}_b, \qquad a \neq b, \quad (10c)$$

where $\mathbf{r}^i_{\{1,2,3\}}$ denotes the first, second, or third row vector of R^i.

If we take the 3×3 matrices $D_{ak} = O_k$, $a = 1, \ldots, \mathbf{K}$, all the matrices S_{ab} are scaled identity matrices, $S_{ab} = s_{ab} I$, for arbitrary scalars s_{ab}, hence Equations (10) are always satisfied. □

Please notice that the third rows of the rotation matrices are only constrained by the orthogonality constraint (10c). Since ${\mathbf{r}^i}_3^\top S' \mathbf{r}^i{}_3 = \gamma_i^2 \beta_i$, the lengths of the third rows are arbitrary. As the equations including the focal length depend on depend on the third row (by γ_i^2 and β_i), the focal lengths are also arbitrary, therefore[2].

Furthermore, the Equations (10) do not define constraints *between* A_{k_1} and A_{k_2}, $k_1 \neq k_2$, $A = \begin{bmatrix} A_1 & \cdots & A_{\mathbf{K}} \end{bmatrix}$. Brand gave such constraints in [14] for an affine camera. Due to the affine model, they only define constraints on the first two rows, hence the ambiguity between focal lengths and projective depths as well as non-rigid mixing coefficients remains.

The problem is thus to define constraints between the different A_{k_1} and A_{k_2}, and on the third rows ${\mathbf{u}_3^i}^\top A_k$. We now arrive at the central contribution of this article, namely additional constraints for constraining the self-calibration matrix A of a projective camera.

Theorem 1. *Given projectively distorted* $3 \times (3\mathbf{K}+1)$ *matrices* U^i*, a matrix* $A = \begin{bmatrix} A_1 & \cdots & A_{\mathbf{K}} \end{bmatrix}$ *satisfying Eqs.* (6) *and*

$$\left({\mathbf{u}_a^i}^\top A_{k_1} A_{k_2}^\top \mathbf{u}_a^i\right)^2 - \left({\mathbf{u}_a^i}^\top A_{k_1} A_{k_1}^\top \mathbf{u}_a^i\right) \cdot \left({\mathbf{u}_a^i}^\top A_{k_2} A_{k_2}^\top \mathbf{u}_a^i\right) = 0 \quad (11a)$$

$$\left({\mathbf{u}^i}_1^\top A_{k_1} A_{k_1}^\top \mathbf{u}_1^i\right) \cdot \left({\mathbf{u}^i}_3^\top A_{k_2} A_{k_2}^\top \mathbf{u}_3^i\right) - \left({\mathbf{u}^i}_1^\top A_{k_2} A_{k_2}^\top \mathbf{u}_1^i\right) \cdot \left({\mathbf{u}^i}_3^\top A_{k_1} A_{k_1}^\top \mathbf{u}_3^i\right) = 0 \quad (11b)$$

for $a = \{1, 2, 3\}$ *and* $k_1 \neq k_2$ *in the unknown column triples* A_k *of* A *transforms a projectively distorted* U *to the structure required by Eq.* (3)*. Equations* (11) *are necessary and sufficient to transform matrices* $U^i A$ *such that the column*

[2] Such an indeterminacy could be attractive to fit a non-rigid model if some or all focal lengths are *a-priorily* known.

triples $U^i A_k$ constitute aligning orthogonal systems, and the lengths of the first two vectors $\mathbf{u}^{i\top}_{\{1,2\}} A_{k_1}$ of any basis k_1 and the lengths of the first two vectors of any other basis k_2 are related by multiplication with $(\alpha^i_{k_2})^2$ and $(\alpha^i_{k_1})^2$.

Proof. Necessity: By Eqs. (6), the six vectors $\mathbf{u}^{i\top}_{\{1,2,3\}} A_{k_1}$ and $\mathbf{u}^{i\top}_{\{1,2,3\}} A_{k_2}$ form two systems of orthogonal vectors. Provided sufficiently many images,

$$\frac{\mathbf{u}^{i\top}_a A_{k_1} A^\top_{k_2} \mathbf{u}^i_a}{\|\mathbf{u}^{i\top}_a A_{k_1}\| \cdot \|\mathbf{u}^{i\top}_a A_{k_2}\|} = 1 \tag{12}$$

holds true if and only if each pair of vectors $\mathbf{u}^{i\top}_a A_{k_1}$ and $\mathbf{u}^{i\top}_a A_{k_2}$ points into the same direction, thus Equation (11a) imposes that the two systems of orthogonal vectors align for $a = \{1, 2, 3\}$. Equations (3) and (5) further require that

$$\frac{\mathbf{u}^{i\top}_1 A_{k_1} A^\top_{k_1} \mathbf{u}^i_1}{\mathbf{u}^{i\top}_3 A_{k_1} A^\top_{k_1} \mathbf{u}^i_3} = \frac{\mathbf{u}^{i\top}_1 A_{k_2} A^\top_{k_2} \mathbf{u}^i_1}{\mathbf{u}^{i\top}_3 A_{k_2} A^\top_{k_2} \mathbf{u}^i_3} = (\phi^i)^2 \tag{13}$$

for some scalar variables ϕ^i from which we obtain Eq. (11b).
Sufficiency: If A satisfies the Eqs. (11), the matrix $U^i A$ has the following structure

$$U^i A = \begin{bmatrix} \phi^i & 0 & 0 \\ 0 & \phi^i & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sigma^i_1 R^i & \cdots & \sigma^i_K R^i \end{bmatrix} \tag{14}$$

for some scalars σ^i. □

Please notice that Eq. (11a) has to be imposed for all three vectors $\mathbf{u}^i_a$, $a = \{1, 2, 3\}$ in order to define a constraint on $\gamma_i^2 \beta_i$.

If the focal length is known to be constant yet unknown, we can impose that constraint by requiring that $\sigma^1 \phi^1 = \cdots = \sigma^{\mathbf{M}} \phi^{\mathbf{M}}$. In the following, denote by i_1 and i_2 two different image numbers.

Corollary 1. *The equation*

$$\left(\mathbf{u}^{i_1\top}_1 A_k A^\top_k \mathbf{u}^{i_1}_1\right) \cdot \left(\mathbf{u}^{i_2\top}_3 A_k A^\top_k \mathbf{u}^{i_2}_3\right) - \left(\mathbf{u}^{i_2\top}_1 A_k A^\top_k \mathbf{u}^{i_2}_1\right) \cdot \left(\mathbf{u}^{i_1\top}_3 A_k A^\top_k \mathbf{u}^{i_1}_3\right) = 0 \tag{15}$$

for $i_1 \neq i_2$ imposes constant focal length throughout the images.

Proof. We must require that any ϕ^{i_1} equals any other ϕ^{i_2} for $i_1 \neq i_2$, hence we obtain from Eq. (11)

$$\frac{\mathbf{u}^{i_1\top}_1 A_k A^\top_k \mathbf{u}^{i_1}_1}{\mathbf{u}^{i_1\top}_3 A_k A^\top_k \mathbf{u}^{i_1}_3} = \frac{\mathbf{u}^{i_2\top}_1 A_k A^\top_k \mathbf{u}^{i_2}_1}{\mathbf{u}^{i_2\top}_3 A_k A^\top_k \mathbf{u}^{i_2}_3} \tag{16}$$

from which Eq. (15) follows directly. □

The set of Eqs. (6) and (11) impose the required structure on the matrices U^i. The question is the remaining ambiguity.

Lemma 2. *Given a transformation A satisfying the sets of Eqs. (10) and (11) which brings each U^i to the required structure, scalars d_k, $k = 1, \ldots, K$, and an arbitrary 3×3 matrix O_g which is an element of the orthogonal group,* i.e. *$O_g O_g^\top = I$, then A is ambiguous up to multiplication with a matrix D*

$$D = \begin{bmatrix} d_{11}O_g & d_{12}O_g & \\ d_{21}O_g & d_{22}O_g & \cdots \\ & \vdots & \end{bmatrix}. \tag{17}$$

Proof. To satisfies Eqs. (10), we may assume that D has the structure as defined lemma (1). Let D_k denote the kth column triple of D, and let

$$S_{kk} = D_k D_k^\top = \begin{bmatrix} d_{11}^2 I & d_{1k}d_{2k}I & \cdots & d_{1k}d_{Kk}I \\ & \vdots & & \\ d_{Kk}d_{1k}I & d_{Kk}d_{2k}I & \cdots & d_{Kk}^2 I \end{bmatrix} \quad \text{and} \tag{18a}$$

$$S_{k_1k_2} = D_{k_1} D_{k_2}^\top = \begin{bmatrix} d_{1k_1}d_{1k_2}O_{k_1}O_{k_2}^\top & \cdots & d_{1k_1}d_{Kk_2}O_{k_1}O_{k_2}^\top \\ & \vdots & \\ d_{Kk_1}d_{1k_2}O_{k_1}O_{k_2}^\top & \cdots & d_{Kk_1}d_{Kk_2}O_{k_1}O_{k_2}^\top \end{bmatrix} \tag{18b}$$

where I denotes the 3×3 identity matrix, and O_{k_1} and O_{k_2}, $k_1 \neq k_2$, are 3×3 matrices of the orthogonal group.

Let P^i denote the row triple of P corresponding to the ith image. Then, we have

$$\begin{aligned} P^i S_{k_1k_2} {P^i}^\top = &\left((\alpha_1^i)^2 d_{1k_1}d_{1k_2} + \ldots \right. \\ &\left. + (\alpha_K^i)^2 d_{Kk_1}d_{Kk_2}\right) K^i R^i O_{k_1} O_{k_2}^\top {R^i}^\top {K^i}^\top \quad \text{and} \end{aligned} \tag{19a}$$

$$P^i S_{kk} {P^i}^\top = \left((\alpha_1^i)^2 d_{1k}^2 + \ldots + (\alpha_K^i)^2 d_{Kk}^2\right) K^i {K^i}^\top \tag{19b}$$

From Eq. (19a) and Eq. (11a), we can see that O_{k_1} and O_{k_2} must be identical if there are sufficiently many images. Equation (11b) imposes no further constraints on the structure of D. □

Lemma 2 implies that any matrix A satisfying Eqs. (6) and (11) is unique up to a global rotation and reflection of the world coordinate system. Furthermore, the bases are unique up an individual scaling of each basis.

Minimizing Eqs. (10) and (11) amounts to minimizing the Frobenius-norm

$$\left\| U^i A A^\top {U^i}^\top - K^i {K^i}^\top \right\|_F. \tag{20}$$

Since minimizing the Frobenius-norm of A is equivalent to minimizing its singular values[3], it is necessary to prevent a rank-degeneracy of A. We therefore impose

[3] Since $\|A\|_F = \sqrt{\sum_i \sigma(A)_i^2}$ where $\sigma(A)_i$ is the ith singular value of A.

the constraint that the smallest singular value of A is larger than 0.1. This constraint also prevents the trivial solution due to the scalar factors γ_i and β_i in Eq. (5).

4 Experiments

4.1 Synthetic Image Experiments

For synthetic evaluation we created a 25-image sequence consisting of 726 3D-points of an ellipsoid morphing into a sphere. Six images of this sequence are shown in Fig. 1(a). At each image the 3D-shape rotates by 7.2° around the y-axis while translating in direction of the x-axis.

To measure the influence of noise, we added normally distributed noise with standard deviation set to 0% to 3.0% in steps of 0.5% of the maximum variation in x, y and z-direction. For each noise level, we created 10 contaminated data sets to compute average errors. As error measure, we took the average of the

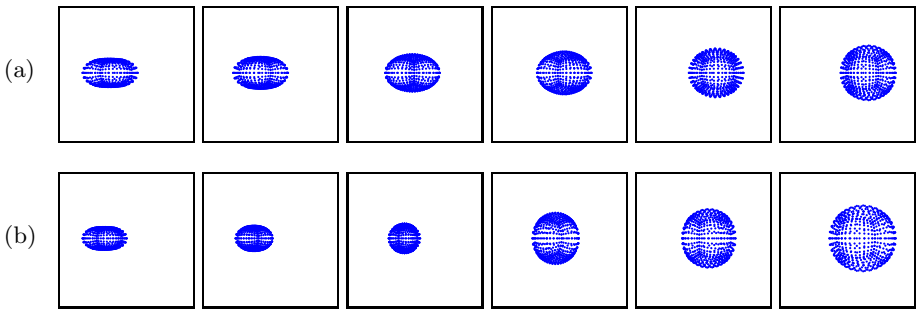


Fig. 1. (a) Six images of a sequence of 25 images showing an ellipsoid morphing into a sphere. At each image the 3D-shape rotates by 7.2° around the y-axis (upwards) and translates in direction of the x-axis. The focal length is constant throughout the sequence. (b) Same structure and motion while the focal length changes between images 1-12 and 13-25.

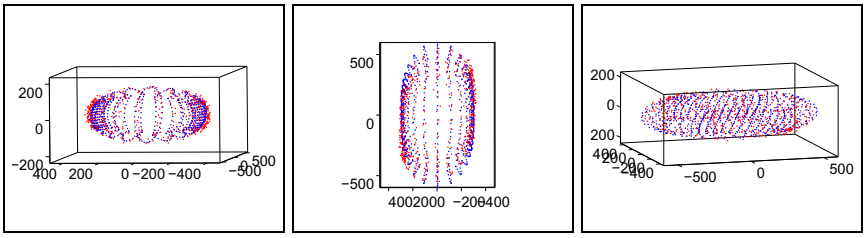


Fig. 2. Example of a 3D-reconstruction if the data is contaminated with normally distributed noise (blue: reconstructed shape; red: ground truth shape). The standard deviation was set to 1% of the maximum variation in x, y, and z-direction.

Euclidean distance between the 3D-points of the ground truth shape and the reconstruction after translating it so that the centroids of both point clouds coincide since the dual absolute quadric constraint ignores the $(3\mathbf{K}+1)$st column of matrix P in Eq. (3). We normalized this number by the Frobenius norm of the ground truth shape

$$\epsilon = \frac{1}{n} \frac{\|X^{gt} - X^{est}\|_F}{\|X^{gt}\|_F}. \tag{21}$$

Here, X^{gt} denotes the matrix consisting of the ground truth 3D-points (for simplicity we omitted an index denoting the image number), and X^{est} the matrix consisting of the the estimated 3D-points. The symbol $\|\cdot\|_F$ denotes the Frobenius norm.

For optimization, we use *semi quadratic programming*. Since the algorithm is susceptible to local minima, we randomly initialize it 40 times and take the best result.

We reconstructed 3D-shapes using two basis shapes ($K = 2$). Figure 3, left plot, shows a the average error as the noise increases. As can be seen, the proposed method is quite robust with respect to noise. In the right plot of Fig. 3, we show average errors *per image* for noise levels 0%, 1% and 2%. The error is not evenly distributed yet there are no exceptional spikes.

For a second experiment, we used the same structure and motion shown in Fig. 1(a) yet changed the focal length between images 1-12 and 13-24. This sequence is shown in Fig. 1(b). The left plot of Fig. 4 shows the reconstruction errors. The right plot of this figure shows the reconstruction errors per image.

To evaluate the estimated calibration matrices we computed the following error metric

$$\epsilon_i = \frac{1}{9} \left\| \frac{1}{\gamma_i^2 \beta_i} U^i A A^\top {U^i}^\top - K^i {K^i}^\top \right\|_F. \tag{22}$$

The left plot in Fig. 5 shows the calibration errors for constant focal length (corresponding to the sequence shown in Fig. 1(a)), the right plot for varying f (Fig.1(b)). Apparently, the proposed algorithm can handle constant and changing focal lengths well.

The average estimated focal lengths per image are shown in Fig. 6. The left plot shows the estimations for constant $f = 5$ whereas the right plot shows them for $f = 4$ in images 1 until 12 and $f = 6$ in images 13 until 25. It can be seen that under noise, the algorithm deviates more from the true values as each image induces its own estimate of the focal length.

Figure 2 shows an example of the reconstructed 3D-shape in the first if the data is perturbed with noise of standard deviation 1%. Blue points denote estimated 3D-points, red points the ground truth. Apparently, the estimated points and the ground truth points almost coincide.

4.2 Real Image Experiments

Figure 7 shows six images of a 25-image sequence. It shows a box whose sides and top paper deform non-rigidly. Please notice that the top paper exhibits strong

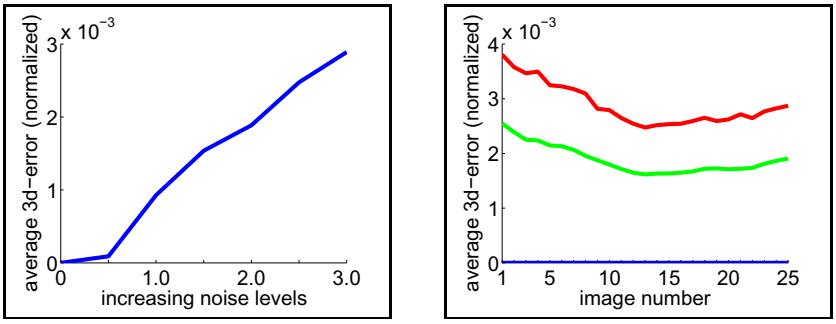


Fig. 3. Left: Average 3D-errors for increasing levels of noise with constant yet unknown focal length (corresponding to the sequence shown in Fig. 1(a)). Right: Average 3D-error per image for noise levels of 0% (solid blue line), 1% (dash-dotted green line) and 2% (dashed red line).

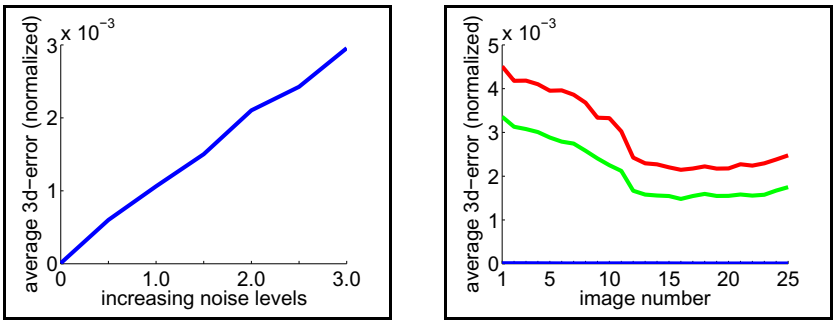


Fig. 4. Average 3D-errors for a changing focal length and the sequence shown in Fig. 1(b). Left: Average 3D-errors for increasing levels of noise. Right: Average 3D-error per image for noise levels of 0% (solid blue line), 1% (dash-dotted green line) and 2% (dashed red line).

deformations which cannot be explained by a multi-body or articulated chain model. A total of 375 points were tracked throughout the sequence.

For projective 3D-reconstruction we used the algorithms proposed in [15,16] which amounts to camera resectioning and intersectioning. We assumed two rigid basis shapes ($K = 2$) and thus optimized for a rank of 7 of the observation matrix.

3D-reconstructions of the shapes observed in every fifth image are shown in Fig. 8. From left to right are shown the image number, the 3D-reconstruction corresponds to, the image, a top view of the estimated shape, a side view (from left), a frontal view, and another side view from the right.

The planar sides of the box show a strong perspective distortion. This is due to the estimated projective depths. The configuration of the frontal and the

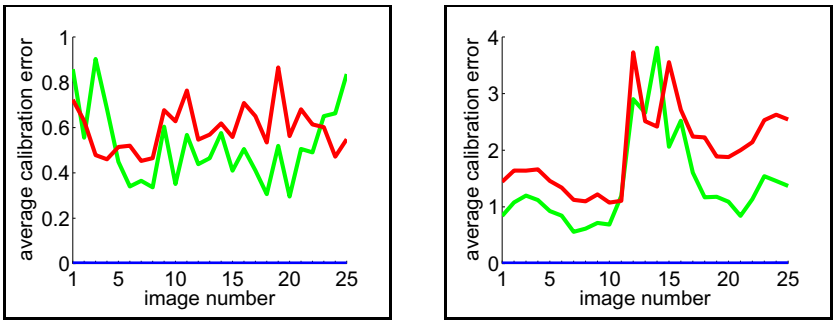


Fig. 5. Left: Average calibration errors for different levels of noise (blue: no noise, green: $\sigma = 1.0$, red: $\sigma = 2.0$) per image. Left: constant focal length; right: focal length varies between images 12 and 13.

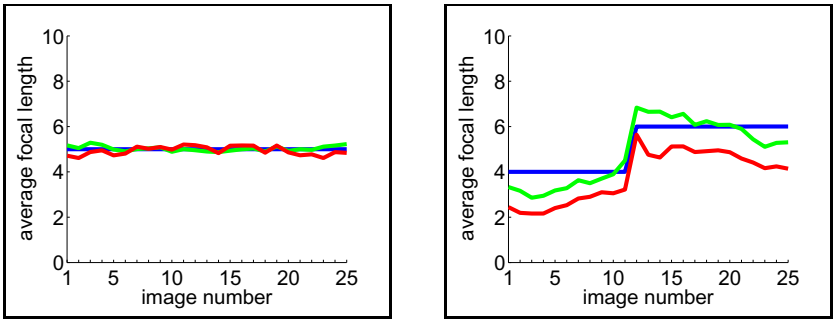


Fig. 6. Left: Average focal lengths for different levels of noise (blue: no noise, green: $\sigma = 1.0$, red: $\sigma = 2.0$) per image. Left: constant focal length $f = 5$; right: focal length varies: $f = 4$ in images 1 until 12 and $f = 6$ in images 13 until 25.

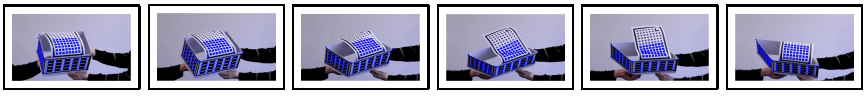


Fig. 7. Six images of a 25-image sequence with 375 trajectories showing a box deforming non-rigidly. The top paper deforms non-rigidly, so a multi-body model would not be satisfied.

left plane to each other closely reflect the shape of the box in the images. The non-rigid bending of the 3D-points on the top structure also closely resembles the shape of the top paper in the images. Overall, the reconstruction looks reasonable.

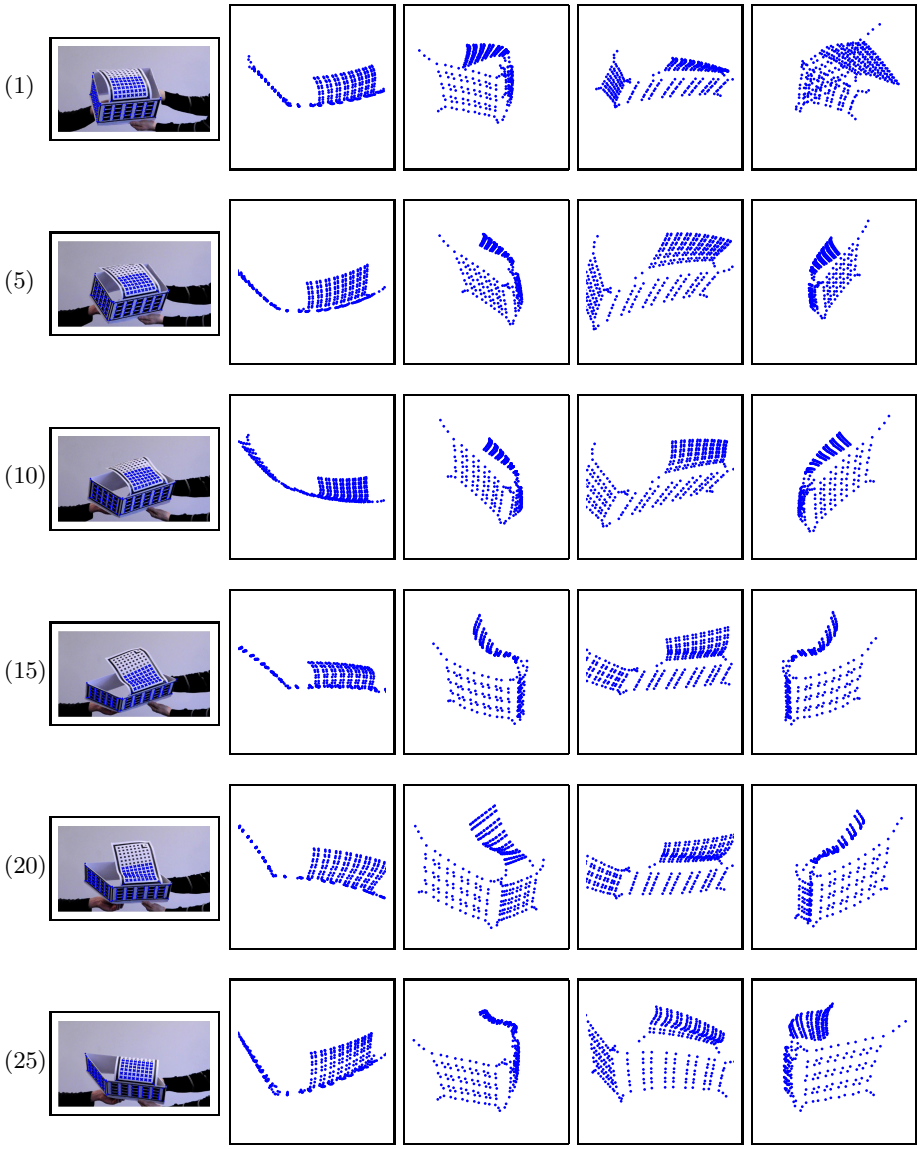


Fig. 8. Six images of a 25-image sequence with 375 trajectories showing a box deforming non-rigidly. The top paper deforms non-rigidly, so a multi-body model would not be satisfied. Shown from left to right are image number, image, top view, left side view, frontal view and right side view of the reconstructed 3D-shape corresponding to each image.

5 Summary and Conclusions

The contributions made in this article can be summarized as follows: Considering a pinhole camera with unknown focal length which may be varying or constant, the problem considered in this work was to determine the Euclidean upgrading if this camera observes a non-rigidly deforming object or scene. To align all motion bases simultaneously during optimization, *i.e.* enforce identical rotations, constraints were derived which allow joint estimation of all motion bases. In terms of error distribution such a joint estimation should be more fair with respect to the different bases.

It was proven that the upgrading transformation is unique up to rotation and reflection of the world coordinate system and individual scalings of each basis. By evaluation of synthetic data as well as a 3D-reconstruction of a difficult real image sequence in which the object exhibits highly non-rigid distortion, it was shown that the proposed algorithm is indeed quite robust to increasing noise and able to reconstruct accurate 3D-shapes.

In future works we will focus on generalizing the camera model to a fully projective model whose intrinsic parameters are all varying and unknown. Furthermore, means of global optimization will be investigated.

References

1. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press (2004) ISBN: 0521540518
2. Bregler, C., Hertzmann, A., Biermann, H.: Recovering non-rigid 3d shape from image streams. In: IEEE Computer Vision and Pattern Recognition (CVPR), Hilton Head, SC, USA, pp. 690–696 (2000)
3. Triggs, B.: Autocalibration and the absolute quadric. In: Conf. Comp. Vis. and Pat. Recog. (CVPR) (1997)
4. Pollefeys, M., Koch, R., van Gool, L.: Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. Int. J. Comp. Vis. (IJCV) 32, 7–25 (1999)
5. Seo, Y., Heyden, A.: Auto-calibration by linear iteration using the DAC equation. Img. Vis. Comp. 22, 919–926 (2004)
6. Chandraker, M., Agarwal, S., Kahl, F., Nistér, D., Kriegman, D.: Autocalibration via rank-constrained estimation of the absolute quadric. In: Conf. Comp. Vis. and Pat. Recog. (CVPR) (2007)
7. Xiao, J., Chai, J., Kanade, T.: A closed-form solution to non-rigid shape and motion recovery. International Journal of Computer Vision 67, 233–246 (2006)
8. Brand, M.: A direct method for 3D factorization of nonrigid motion observed in 2d. In: IEEE Computer Vision and Pattern Recognition (CVPR), Washington, DC, USA, pp. 122–128 (2005)
9. Olsen, S., Bartoli, A.: Implicit non-rigid structure-from-motion with priors. Journal of Mathematical Imaging and Vision 31, 233–244 (2008)
10. Torresani, L., Hertzmann, A., Bregler, C.: Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. IEEE Pattern Analysis and Machine Intelligence (PAMI) 30, 878–892 (2008)

11. Paladini, M., Del Bue, A., Stosic, M., Dodig, M., Xavier, J., Agapito, L.: Factorization for non-rigid and articulated structure using metric projections. In: IEEE Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, pp. 2898–2905 (2009)
12. Xiao, J., Kanade, T.: Uncalibrated perspective reconstruction of deformable structures. In: Proceedings of the 10th International Conference on Computer Vision (ICCV), vol. 2, pp. 1075–1082 (2005)
13. Hartley, R.I., Vidal, R.: Perspective Nonrigid Shape and Motion Recovery. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 276–289. Springer, Heidelberg (2008)
14. Brand, M.: Morphable 3D models from video. In: IEEE Computer Vision and Pattern Recognition (CVPR), pp. 456–463 (2001)
15. Heyden, A., Berthilsson, R., Sparr, G.: An iterative factorization method for projective structure and motion from image sequences. International Journal on Computer Vision 17, 981–991 (1999)
16. Mahamud, S., Hebert, M.: Iterative projective reconstruction from multiple views. In: The Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 430–437 (2000)
17. Akhter, I., Sheikh, Y., Khan, S.: In defense of orthonormality constraints for nonrigid structure from motion. In: IEEE Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, pp. 1534–1541 (2009)

Piecewise Planar Scene Reconstruction and Optimization for Multi-view Stereo

Hyojin Kim, Hong Xiao, and Nelson Max

University of California, Davis, One Shields Avenue Davis, CA 95616, USA
{mrhkim,hxiao,nlmax}@ucdavis.edu

Abstract. This paper presents a multi-view stereo algorithm for piecewise planar scene reconstruction and optimization. Our segmentation-based reconstruction algorithm is iterative to minimize our defined energy function, consisting of reconstruction, refinement and optimization steps. The first step is a plane initialization to allow each segment to have a set of initial plane candidates. Then a plane refinement based on non-linear optimization improves the accuracy of the segment planes. Finally a plane optimization with a segment-adjacency graph leads to optimal segment planes, each of which is chosen among possible plane candidates by evaluating its relationship with adjacent planes in 3D. This algorithm yields better accuracy and performance, compared to the previous algorithms described in this paper. The results show our method is suitable for outdoor or aerial urban scene reconstruction, especially in wide baselines and images with textureless regions.

1 Introduction

Our goal is to reconstruct a 3D geometry model from a multi-view image sequence of an urban scene from aerial or ground level viewpoints. Since these scenes consist mostly of a set of man-made planar structures, we use an image segmentation where each segment can be well approximated by a plane.

Most segmentation-based stereo algorithms begin scene recovery by performing pixel-by-pixel dense correspondence and triangulation to obtain a set of 3D points, giving matches for an image pair. Given a segment with its 3D points, a plane is assigned by plane fitting methods (least-squares or RANSAC). Then the depth- or disparity- based planes are propagated among neighboring segments to obtain an optimal plane set [1,2,3]. These segmentation-based approaches have been popular due to the reconstruction quality, compared to conventional dense stereo methods.

In the case of wide baselines or images with textureless regions that we aim to reconstruct, however, the plane-fitting-based approach becomes poor due to T-junctions or mismatches in the initial dense correspondence that the fitting relies on. These algorithms also use a smoothness term in their cost function to suppress incorrect planes and to give continuity (smoothness) among adjacent segments. However, an optimal solution to guarantee that many parts (segments)

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 191–204, 2013.

of the scene are well-connected with their neighbors is still needed for high quality piecewise planar scene recovery.

In this paper, we propose a robust piecewise planar scene reconstruction and optimization algorithm for outdoor or aerial imagery. Our iterative reconstruction consists of plane initialization, photo-consistency refinements, adjacency optimization, and outlier removal to effectively converge to a local minimum of the energy function. In particular, the optimization step chooses the best piecewise plane so far in terms of continuity among adjacent segments according to a segment-adjacency graph.

This paper is organized as follows: Section 2 discusses previous work, together with our contribution. Section 3 gives our algorithm in detail. Section 4 provides experimental results, followed by the conclusion in Section 5.

2 Related Work and Contribution

Robust Piecewise Planar Scene Reconstruction. Our first contribution is the robustness of our planar reconstruction, especially for wide baselines and images with textureless regions. Our iterative framework effectively minimizes the energy function to obtain an optimal plane set. One related work is the algorithm of Kim *et al.* [4]. We adopt their direct plane homography estimation in our plane initialization because their method is appropriate for wide baselines and textureless regions. However, their approach is computationally expensive, due to the large number of sample planes that must be evaluated. Their algorithm also does not include any plane-wise smoothness energy term. We overcome these problems, as addressed in the following sections.

Another related work is Manhattan World Stereo [5] in which planes are axis-aligned to a "Manhattan World" urban or indoor scene, given initial 3D points from PMVS [6]. But we want our algorithm to be more general so that planes that are not perpendicular to one of the three axes can also be recovered. Multi-View Superpixel Stereo [7] also recovers a set of planes by restricting the number of plane orientations to avoid plane ambiguity. Like Manhattan World Stereo, however, a slanted plane may be suppressed into parts of other planes.

Plane Sweeping [8] is a planar reconstruction algorithm using a set of feature matches with an intensity-based cost evaluation. Gallup *et al.* [9] present a RANSAC-based method to classify a set of initially reconstructed points into planar or non-planar regions. Iterative Plane Fitting [10] is a planar reconstruction that requires manual plane initialization. Again, all these methods rely heavily on initial pixel-wise matching, which may not be suitable for wide baselines.

Sinha *et al.* [11] incorporate a set of initial planes and line segments. Their smoothness term uses a lower label difference cost for pixels on opposite sides of a crease edge. In the case that two planes' normals are similar (but differ by more than 5 degrees), the intersection line may not give an accurate 2D edge due to the ambiguity of discretization so that some neighboring pixel pairs can be located on the same plane. Also, their recovered planes seem too simplified - multiple planes are merged into one plane. Zhang *et al.* [12] do bundle

optimization by non-linearly optimizing planes using approximate derivatives, similar to our refinement step using analytic derivatives.

Improved Plane Initialization. As discussed earlier, our plane initialization adopts the algorithm of [4] where a plane for each segment is obtained by taking equally spaced sample points along camera rays through three corners of the segment, and then selecting every plane through three such sample points, one on each ray. Compared to the exhaustive sampling in [4], our plane initialization is made much faster and more efficient by incorporating an adaptive sampling scheme with an image pyramid and an image descriptor. When a segment is relatively large, we use the same region of a lower resolution image because large segments are less sensitive to noise and the energy cost is more likely to converge to the minimum. Low resolution images also reduce the number of ray sample points for plane sampling, which significantly reduces computation time.

Another speedup is achieved by using the DAISY descriptor [13]. Instead of using an image descriptor for pixel-by-pixel matching, we utilize it as an assistant to skip the energy evaluation for a large number of unnecessary candidate planes. For instance, if none of the three sampled 3D corner points induces a descriptor vector distance lower than a threshold, we skip the evaluation of that plane.

Plane-Adjacency-Based Optimization. Many Markov Random Field (MRF) formulations have a smoothness term in the energy function to suppress reconstruction noise. However, the pixel-wise smoothness that many algorithms use is not appropriate for planar structure recovery. For instance, the algorithm of [2] regards all boundary pixels in each segment as adjacent pixels in the smoothness disparity constraints, which causes an incorrect (too smoothed out) surface in case of perpendicular segment planes. The algorithm of [14] uses a discontinuity cost to smooth out depths except at the segment boundary, given a segment-wise MRF. To find an optimal plane for each segment in the planar scene recovery, however, we need to know which pixels belong to the connecting region between an adjacent segment pair.

Our plane-wise smoothness has more sophisticated adjacency information to locate which pixels are adjacent to a certain segment, resulting in more accurate planar geometry. In each optimizing iteration, we rebuild the segment-adjacency graph by checking the likelihood of pixel adjacency and by finding a combination of two adjacent segment planes that maximizes the likelihood of adjacent pixels in order to give good connectivity among adjacent planes. A segment edge in the graph stores not only the connectivity between two adjacent segments, but also adjacency of each individual boundary pixel of the segment.

3 Algorithm

3.1 Reconstruction Overview

The algorithm starts with the Mean-Shift color segmentation [15] that many segmentation-based stereo algorithms use. Given a set of segments in the

reference image, we do a plane initialization to collect several good plane candidates for each segment, each of which has an energy cost lower than a threshold. Then we iterate a photo-consistency-based plane refinement and a segment-adjacency-based optimization so that the total cost converges to the minimum of the energy function. From our experiments, two or three iterations are sufficient. Afterwards, any plane outlier is filtered out. Fig. 1 and Fig. 2 show the overall process and an example of reconstruction according to the iterative process.

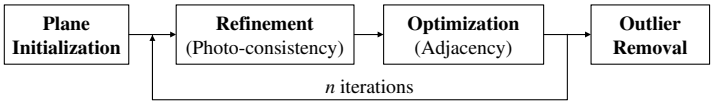


Fig. 1. Iterative reconstruction and optimization process

Fig. 2. Reconstruction results according to the iteration. From *left* to *right*, plane initialization, refinement, optimization, and after multiple refinement and optimization steps.

The goal is to find an optimal plane set X, one plane for each segment, that minimizes the following three terms in our energy function E. That is, each segment plane x has good photo-consistency E_{Photo} in target views in which the segment plane is visible, good visibility $E_{Visibility}$ in all target views, and good 3D adjacency $E_{Adjacency}$ to its neighboring segment planes. The energy function is defined as follows:

$$E = \sum_{x} (E_{Photo}(x) + \lambda_1 E_{Visibility}(x) + \lambda_2 E_{Adjacency}(x)) \quad (1)$$

where λ_1 and λ_2 are adjustable weights. (Defaults are 0.5 and 0.5, respectively.)

Photo Term. The photo term $E_{Photo}(x)$ measures photo-consistency between a reference segment plane x and its homography-remapped region in a target image i in which the segment plane is visible. This term is based on the summed squared color/intensity differences, which is widely used in other stereo algorithms. Since we do not apply the boundary matching energy term in [4], we instead use a dilated segment to force the boundaries to match, and to exclude very slanted

planes which might otherwise give good color matches in the segment interior. The photo term is defined as

$$E_{Photo}(x) = \left(\frac{\sum_i V_i(x) \sum_p C(H_i(p)) \{P(p) - P_i(H_i(p))\}^2}{\sum_i V_i(x) \sum_p C(H_i(p))} \right)^{1/2} \quad (2)$$

where p is a pixel position within the dilated reference segment region, $V_i(x)$ is 1 if x is visible in target image i and 0 otherwise (see next paragraph), $C(p)$ is 1 if p is within the target image region and 0 otherwise, $H_i(p)$ is the homography that the plane x and the camera geometry induce between the reference image and a target image i, and $P(p)$ and $P_i(p)$ return the pixel value of the pixel p from the reference image and target image i, respectively.

Visibility Term. The visibility term indicates whether or not a segment plane is visible by checking the plane normal and the boundary clipping. When a plane is too slanted in a target view (i.e., the angle between the plane normal and the ith camera direction is larger than a threshold), or if too large a proportion of the homography-mapped segment's pixels are outside the boundary of target image i, the plane is regarded as "not visible". The visibility term is defined as

$$E_{Visibility}(x) = \frac{N - \sum_i V_i(x)}{N} \quad (3)$$

where N is the total number of target images.

Adjacency Term. This adjacency term is our smoothness term that not only suppresses error planes, but also ensures connectivity among adjacent segment planes. As discussed earlier, many segmentation-based stereo algorithms use a pixel-wise smoothness term that adds a discontinuity penalty to the given energy function. However, we believe such a pixel-wise smoothness does not guarantee a well-reconstructed planar geometry. Therefore we do not measure the minimum distances of all adjacent point pairs between the current segment and its surrounding segments. Instead, we measure a ratio of the number of adjacent point pairs in the 3D domain to the number of adjacent pixel pairs in the 2D image domain for each segment s with a candidate plane x. An adjacent point pair is an adjacent pixel pair in the 2D image domain whose 3D points are within a distance threshold. Thus

$$E_{Adjacency}(s, x) = 1 - \frac{\sum_a A_x(a)}{M_s} \quad (4)$$

where a is an adjacent pixel pair between segment s and one of its 2D adjacent segments, $A_x(a)$ is 1 if the two 3D points of a are within the 3D distance threshold and 0 otherwise, and M_s is the number of 2D-adjacent pixel pairs of segment s. The segment-adjacency graph is also adaptively generated, along with measuring the adjacency term. More details on the adjacency-based optimization are described in Section 3.4.

3.2 Plane Initialization

In this step, we consider only the first two terms to be minimized since the adjacency term cannot be measured without a set of segment planes. We perform this plane initialization to obtain a set of initial segment planes by directly comparing each homography-remapped reference segment with target images in which the segment plane is visible.

However, the number of plane samples to be evaluated is extremely large. If each camera ray is discretized into m samples, m^3 plane samples need to be evaluated. The number m of samples on each camera ray varies (from 20 to 500 in our experiments), depending on the input images and the initial 3D bounding box. The hierarchical refinement in [4] is still computationally expensive because we need to evaluate in the first pass a large number of sample planes in what may turn out to be empty space. To overcome this inefficiency, we apply an image hierarchy starting with an optimal m, and use the DAISY image descriptor.

The idea of an image hierarchy is to reduce m by using a coarse-level image if a segment is relatively large. Initially, the number m_s of ray samples for segment s is proportional to the maximum pixel distance of the projected points on the first target view (i.e., a target view closest to the reference view, with the narrowest-baseline) between the starting and ending points of the segments of the three camera rays that are within the 3D bounding box of some initial sparse 3D matched points. Thus

$$m_s = \alpha \max_{i=1,2,3} |C_0 S_{s,i} - C_0 E_{s,i}| \tag{5}$$

where C_0 is the camera projection matrix of the first target view, $S_{s,i}$ and $E_{s,i} (i = 1, 2, 3)$ are starting and ending points along ray i, obtained from the 3D bounding box, and α is a parameter, depending on the scene (mostly 2 to 5 from our experiments). Optionally, we may pick a target view that gives the widest baseline for more robustness; however, this increases m_s and thus the computation time. Now we want to reduce m_s by using an image hierarchy. Given the number N_s of pixels in a segment s, we compute an initial image hierarchy level as

$$Level(s) = \begin{cases} \frac{1}{2} \log_2 \frac{N_s}{\delta} & \text{if } N_s > \delta \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where δ is a threshold ($\sim$500 in experiments). If this function returns 0, the original image set is used. Otherwise, we use a Gaussian down-sampled image set at level $Level(s)$, and reduce m_s by a factor of $1/2^{Level(s)}$. For segment dilation, we dilate the segment region by $2 + Level(s)/2$ pixels. Once we find the best plane in the first pass, we hierarchically search for a better plane by sampling the corner rays in smaller intervals near the corner points of the best plane so far. Unlike the algorithm of [4], our coarse-to-fine search is done with the pyramid image so that the next pass uses a finer image level. Also, since we consider the first two terms without the adjacency term in this initialization, we sometimes fails to converge to a global minimum. We keep all good plane

candidates ($E_{Photo}(x) + E_{Visibility}(x) < \gamma$) obtained in every pass, where γ is a threshold (0.5 in our experiments), instead of discarding all other planes except the one best plane so far in the previous pass.

The next enhancement to improve the initialization performance is to use the DAISY descriptor to skip bad plane samples. Prior to plane sampling and evaluation, given three corners of a segment and their corner samples along the rays, we compute the Euclidean distances of DAISY descriptor vectors between each corner pixel in the reference image and the projected corner samples onto the first target image (i.e., corner points along the epipolar line in the first target view), together with the minimum distance. Then every sample in each camera ray is tagged as 1 if the ratio of the distance to the minimum distance is less than a threshold (∼1.7 from our experiments), and 0 otherwise, as shown in Fig.3. When we do the plane sampling and evaluation, we examine the three tag values in a given plane sample to determine whether or not the current plane is likely. If at least two tags are 1, that plane sample is evaluated. Otherwise, that plane sample is regarded as "unlikely", and is not evaluated.

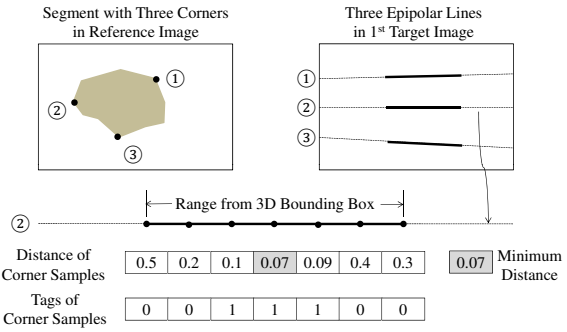


Fig. 3. Plane initialization with an image descriptor along three corner rays

3.3 Refinement

Given a set of initial planes, we refine the plane set by optimizing the four coefficients for each plane so that the homography minimizes the photo term in the given energy function, with the L-BFGS non-linear optimization algorithm, which uses analytic derivatives of a cost function, similar to the photo term, with respect to the plane coefficients. This derivative-based refinement is much faster than the sampling method discussed above, but it converges to what may be only a local minimum, so it must be preceded by a more exhaustive approximate search.

This refinement step and the following optimization step are iterative, that is, this refinement can also be done after the optimization. Each refinement iteration continues until the L-BFGS algorithm converges to the local minimum, using a sub-segment region in the reference image that excludes occluded pixels

or pixels out of the target image region. To improve the quality of segment planes or to quickly converge to a global minimum of the cost function, we do multiple refinements, each of which re-initializes segment visibility information (i.e., which pixels are visible in which view).

3.4 Optimization

In this step, we consider the adjacency term, together with the first two terms. For each segment, we choose the optimal plane that minimizes our energy function among a set of good plane candidates. The adjacency term is obtained from the number of adjacent 3D point pairs between the segment plane and the planes of its 2D adjacent segments. This plane optimization is also iterative, since choosing the best segment plane given its neighboring segment planes also affects the adjacency term of the neighboring segment planes.

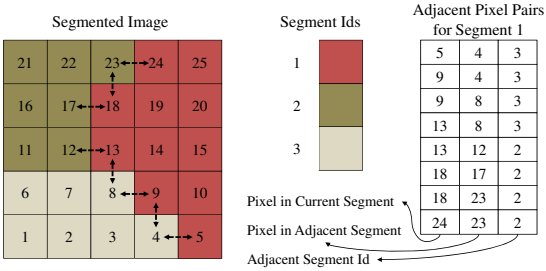


Fig. 4. Adjacent pixel pairs between segment 1 and its adjacent segments 2 and 3

Adjacent pixel pairs are a set of pixel pairs, each of which has one pixel in the segment and the adjacent pixel in another segment, as illustrated in Fig. 4. For each pixel pair, we compute the actual 3D distance between the 3D points found from intersecting the pixel pair's viewing rays with their respective candidate planes, to see if the distance is smaller than a threshold. Since the threshold depends on the scene, we compute the 3D adjacency threshold $\epsilon(s, x)$ separately for each segment s with its best plane so far, plane x.

$$\epsilon(s, x) = \beta |Pl(x, p_s) - Pl(x, p_s^{'})| \tag{7}$$

where β is a constant (3 from our experiments), $p_s = (px, py)$ is the center pixel in segment s, $p_s^{'} = (px+1, py+1)$, and $Pl(x, p)$ gives pixel p's 3D point on plane x with respect to the reference camera (*i.e.*, intersection point of the pixel's viewing ray and the plane x).

Once we choose the best plane among plane candidates, we also build a segment adjacency-graph where each segment is a node and a graph edge represents adjacency between two segments. This reflects actual adjacency in 3D between

neighboring segments. Initially, we add a graph edge between every pair of segments that are adjacent in the image domain. Then we update the segment-adjacency graph, as the optimization iteration goes on. That is, if few of the 2D adjacent point pairs between two segments are adjacent in 3D, we remove the graph edge between them. The final graph is used for the next step to get rid of plane outliers.

3.5 Outlier Removal

Any bad segment plane is filtered out in this step. First, we remove any segment plane whose total energy function is higher than a threshold, due mostly to occlusion. We also check if either the photo or adjacency term is higher than a threshold. We do an additional outlier removal by using the segment-adjacency graph generated in the plane optimization. If a segment plane has no adjacent segment plane in the graph, we discard it, similar to the filtering in [4]. If a segment plane has a number of adjacent segment planes less than a threshold (*e.g.*, 1 or 2) and is also too slanted with respect to its neighbors (*i.e.*, the angle difference between the two plane normals is large), then it is also filtered out.

There is a trade-off between the accuracy and the completeness, depending on the thresholds. Unlike dense stereo algorithms that interpolate disparities of occluded pixels, such thresholds are inevitable, due to the cases that a segment is not planar or is occluded from many target views. A discarded non-planar segment can be reconstructed as individual points using conventional dense stereo algorithms, as in [4].

4 Experiments and Discussion

To evaluate the effectiveness of our algorithm, we performed a synthetic scene reconstruction, together with quantitative evaluations such as accuracy and performance. For these experiments, we used a synthetic dataset with known camera poses and ground-truth data from [16]. We also used two outdoor scene datasets from [16], and two aerial urban scene datasets from [17,18]. For the camera poses of these real datasets, we used Bundler [19].

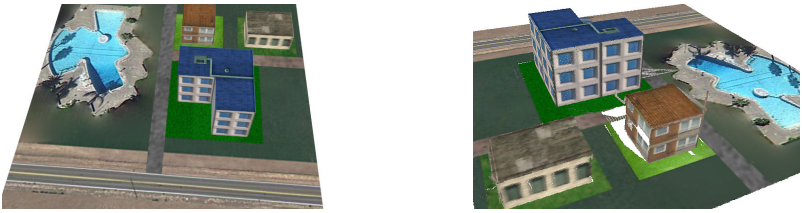


Fig. 5. Synthetic Scene Reconstruction. One of the input images (*left*), and a reconstructed scene (*right*).

Table 1. Quantitative evaluations of the synthetic scene reconstruction. *top*: accuracy according to reconstruction steps: (1) plane initialization; (2) refinement; (3) optimization; (4) refinement; (5) optimization; (6) outlier removal. *bottom*: accuracy vs. completeness measures according to the total energy cost threshold.

	(1)	(2)	(3)	(4)	(5)	(6)
RMS Error	0.00371	0.00291	0.00254	0.00241	0.00242	0.00097

Threshold	0.2	0.3	0.4	0.5	0.6
Completeness (%)	84	88	95	98	99
RMS Error	0.00085	0.00097	0.00110	0.00111	0.00242

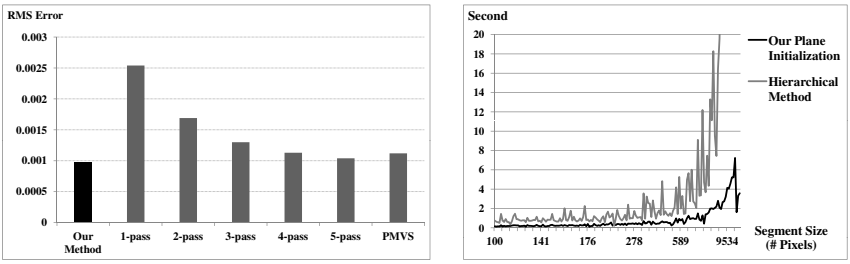


Fig. 6. Quantitative evaluations of the synthetic scene reconstruction. Accuracy of reconstruction by different methods (*left*), and initialization performance according to segment size (*right*).

4.1 Accuracy

Fig. 5 shows our reconstruction result for the synthetic scene dataset. Table 1 summarizes two quantitative evaluations of the synthetic scene reconstruction. The RMS error indicates the RMS distance between the reconstructed result and the ground-truth of the synthetic model. The first evaluation is to compare the accuracy according to reconstruction steps, as shown in Table 1 (*top*). As the refinement and optimization iteration went on, the RMS error decreased. Table 1 (*bottom*) shows the trade-off between the accuracy and the completeness when changing the total energy cost, one of the thresholds, in the outlier removal.

We did another quantitative comparison of the accuracy, together with other algorithms, as shown in Fig. 6 (*left*). We compared our algorithm with the hierarchical planar reconstruction of [4], up to five passes. We also include the accuracy of PMVS [6], one of the most popular multi-view reconstruction algorithms. This result shows our algorithm more effectively converges to a minimum, even compared to five passes of the brute-force hierarchical reconstruction.

4.2 Performance

The total running time is 400-550 seconds for one depth map (1 reference image with 6-8 target images, each of which has resolution 1593 x 705) and about

Fig. 7. Two outdoor scene reconstruction results. Each row shows one of the input images (*left*) and its reconstructed scene (*right*).

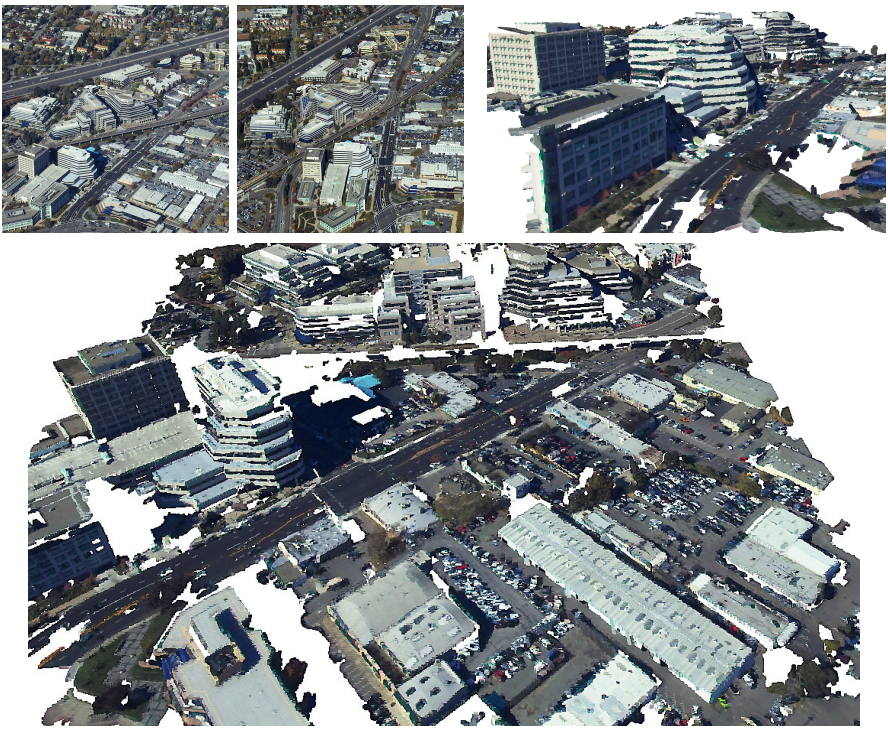


Fig. 8. Aerial urban scene reconstruction (Walnut Creek, CA, USA). Two consecutive input images(*left top, middle top*) and a reconstructed scene (*right top, bottom*).

Fig. 9. Aerial urban scene reconstruction (Stockton, CA, USA). One of the input images (*left top*) and a reconstructed scene (*right top, bottom*).

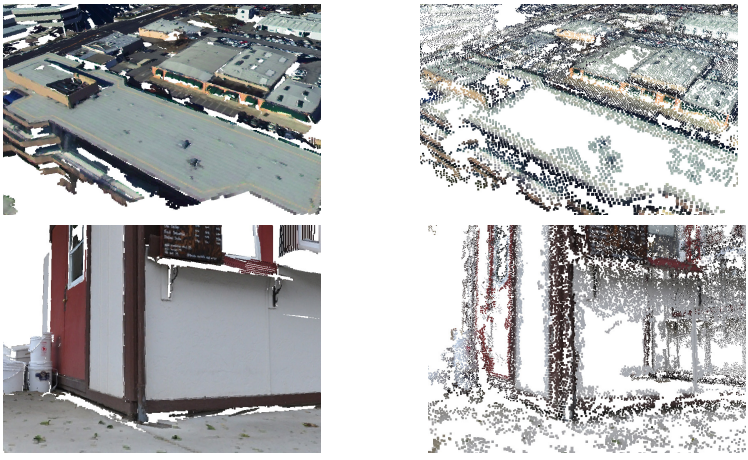


Fig. 10. Comparison with a popular multi-view stereo software. Each row shows a reconstructed scene by our algorithm (*left col.*) and by PMVS [6] (*right col.*).

3 depth maps are enough to complete the scene reconstruction. Fig. 6 (*right*) shows the difference in performance (computation time) between our plane initialization and the hierarchical reconstruction of [4], according to segment sizes. The speedup on average is approximately 16 times. Our plane initialization without GPU-parallelism gives even better performance than the GPU implementation of [4]. Since our plane initialization and optimization can also be parallelized using a GPU, due to the independence of each plane's computation, additional speedup could be easily achieved.

4.3 Real Scene Reconstruction

Fig. 7 shows our reconstruction results for two outdoor scene datasets. The results show that our reconstruction and optimization effectively reconstructs planar structures with few holes or gaps, regardless of the existence of texture on the structures. However, non-planar segments such as vegetation are also modeled as planes, which may not be appropriate for large non-planar segments. We may need to use the hybrid approach presented by [4].

We also performed tests on two aerial urban scene datasets, as shown in Fig. 8 and Fig. 9. These urban scenes, consisting of a number of man-made planar structures, are well-reconstructed. In particular, the result in Fig. 8 shows that our method is also suitable for wide-baseline images. The remaining incorrect segment planes or artifacts near the scene corners are mostly due to an improper image segmentation. Other segmentation-based reconstruction methods suffer from similar problems.

In addition, our algorithm provides more dense and well-connected planar geometry in the reconstructed scene, compared to a popular multi-view stereo software package (PMVS), as shown in Fig. 10.

5 Conclusion

We presented a new multi-view stereo approach for piecewise planar scene reconstruction. Our method iteratively recovers a scene to give an optimal planar geometry. The algorithm begins with plane initialization, photo term refinement, and adjacency optimization, followed by outlier removal. The plane initialization gives fast convergence, compared to brute-force hierarchical plane sampling and evaluation. The optimization with the segment-adjacency graph yields more accurate segment planes consistent with adjacent planes. Our method works effectively for scenes consisting of man-made structures such as outdoor or aerial urban images.

Acknowledgement. This research was supported in part by the United States Department of Energy, under grant number DE-FG52-08NA28777, and by the National Science Foundation, under grant number DMS-1016712.

References

1. Tao, H., Sawhney, H.S.: Global matching criterion and color segmentation based stereo. In: IEEE Workshop on Applications of Computer Vision, pp. 246–253 (2000)
2. Hong, L., Chen, G.: Segment-based stereo matching using graph cuts. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 74–81 (2004)
3. Bleyer, M., Gelautz, M.: A layered stereo matching algorithm using image segmentation and global visibility constraints. ISPRS Journal of Photogrammetry and Remote Sensing 59, 128–150 (2005)
4. Kim, H., Hunter, Q., Duchaineau, M., Joy, K., Max, N.: Gpu-friendly multi-view stereo for outdoor planar scene reconstruction. In: Eighth International Conference on Computer Vision Theory and Applications (VISAPP), part of VISIGRAPP, pp. 255–264 (2012)
5. Furukawa, Y., Curless, B., Seitz, S.M., Szeliski, R.: Manhattan-world stereo. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1422–1429 (2009)
6. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multi-view stereopsis. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 1362–1376 (2009)
7. Micusik, B., Kosecka, J.: Multi-view superpixel stereo in man-made environments. International Journal of Computer Vision 89, 106–119 (2010)
8. Gallup, D., Frahm, J.M., Mordohai, P., Yang, Q., Pollefeys, M.: Real-time plane-sweeping stereo with multiple sweeping directions. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
9. Gallup, D., Frahm, J.M., Pollefeys, M.: Piecewise planar and non-planar stereo for urban scene reconstruction. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1418–1425 (2010)
10. Habbecke, M., Kobbelt, L.: Iterative multi-view plane fitting. In: 11th International Fall Workshop, Vision, Modeling, and Visualization, pp. 73–80 (2006)
11. Sinha, S.N., Steedly, D., Szeliski, R.: Piecewise planar stereo for image-based rendering. In: IEEE International Conference on Computer Vision, pp. 1881–1888 (2009)
12. Zhang, G., Jia, J., Wong, T.T., Bao, H.: Consistent depth maps recovery from a video sequence. IEEE Transactions on Pattern Analysis and Machine Intelligence 31, 974–988 (2009)
13. Tola, E., Lepetit, V., Fua, P.: Daisy: An efficient dense descriptor applied to wide-baseline stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 815–830 (2010)
14. Taguchi, Y., Wilburn, B., Zitnick, C.L.: Stereo reconstruction with mixed pixels using adaptive over-segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
15. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 603–619 (2002)
16. Kim, H.: Outdoor, Aerial and Synthetic Datasets for Multi-View Stereo. Website (2011), `http://idav.ucdavis.edu/~hkim/mvs/dataset`
17. Wayne, B.: Aerial images of Walnut Creek, California (2007), `http://www.cognigraph.com/walnut_creek_Nov_2005`
18. Romero, C.K.: Aerial images of Stockton, California (2009), `http://www.cognigraph.com/kique_D80-Card1_101NIKON`
19. Snavely, N., Seitz, S.M., Szeliski, R.: Modeling the world from internet photo collections. International Journal of Computer Vision 80 (2008)

A Bayesian Approach to Uncertainty-Based Depth Map Super Resolution

Jing Li, Gang Zeng, Rui Gan, Hongbin Zha, and Long Wang

Key Laboratory of Machine Perception, Peking University, Beijing, 100871, China

Abstract. The objective of this paper is to increase both spacial resolution and depth precision of a depth map. Our work aims to produce a super resolution depth map with quality as well as precision. This paper is motivated by the fact that errors of depth measurements from the sensor are inherent. By combining prior geometry of the scene, we propose a Bayesian approach to the uncertainty-based depth map super resolution. In particular, uncertainty of depth measurements is modeled in terms of kernel estimation and is used to formulate the likelihood. In this paper, we incorporate a gauss kernel on depth direction as well as an anisotropic spatial-color kernel. We further utilize geometric assumptions of the scene, namely the piece-wise planar assumption, to model the prior. Experiments on different datasets demonstrate effectiveness and precision of our algorithm compared with the state-of-art.

1 Introduction

Depth map is the simplest and most convenient way of representing and storing the depth measurements taken from a scene. A depth map is like a gray scale image except the depth information replaces intensity, so as to express geometry of the scene. Nowadays, we can easily obtain a depth map via depth sensors, e.g. time-of-flight (TOF) camera and Microsoft Kinect camera. Unfortunately, resolution of depth map under current technique and reliability of measurements (especially on depth edges) are rather low, thus limits applications in fields like computer vision and robotics. Moreover, high accuracy of the depth map is also necessary in detailed reconstruction work.

Inspired by [1], where the video super resolution problem is solved via Baysian approach. In this paper, we propose a new method to depth map super resolution under Bayesian framework. We focus on how to achieve a high-res depth image of a static scene from a low-res depth map and its corresponding high-res color image. The resulting high-res depth map is of sufficient quality, reliability and accuracy. Our method produces promising results on real-world scenes, where depth boundaries are preserved and refined, and the smooth areas are recovered from depth bleeding. Furthermore, our approach also increases depth precision by adding more depth layers. Specifically, we incorporate uncertainty of the depth measurements and geometry prior of the scene to maximize a posterior probability. Uncertainty is modeled using kernel estimation. And the prior is

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 205–216, 2013.

based on piece-wise planar assumption of the scene, which is capable to represent rough geometry of most indoor scenes.

The main contributions of this paper are

(1). We use Bayesian approach to solve the problem of depth map super resolution.

(2). We take the intrinsic errors of the camera into consideration and model the high-res depth map under uncertainty theory.

(3). We utilize piece-wise planar assumption to regulate global geometry of the scene.

1.1 Related Work

From aspect of the input, previous work on depth map super resolution are classified as either depth-only techniques that use only low-res depth maps of the scene from slightly displaced viewpoints or depth-texture techniques that usually combine a low-res depth map with the corresponding high-res color image. Our approach falls into the second category. However, both techniques share the same objectives. First, preserving local maxima at depth boundaries; second, keeping the overall smoothness at continuous areas; last but not the least, improving accuracy of the reconstruction.

Depth-Only Techniques. The goal of depth-only techniques from multi-view input is to enhance resolution by combining depth recordings of the scene taken from slightly displaced viewpoints, and produce a satisfactory high-res depth map. Kil et al. [2] were almost among the first to explore the ideal using a laser scanner by oversampling the scene to achieve the upsampled geometry. Due to small error of laser scanner, their results were acceptable. Recent work of performing super resolution on depth camera [3,4] followed the way of designing an objective function that contains both fidelity term and a regularization term. Schuon et al. [3] applied traditional multi-frame super resolution method on color image into depth map via bilateral regularization, whereas the optimization framework developed by [4] made use of a geometry prior regularization term to guide the optimizer towards plausible 3D reconstructions and preserved both depth edge discontinuity and overall smoothness. Despite measurements from depth sensor, Li et al. [5] produced a high accuracy depth map by optimizing the 3D point cloud from multi-view stereo. Our work is more related to [6], where they used a multiple depth hypotheses approach and extracted the true depth under spatial consistency constraint. In this paper, depth hypotheses are incorporated into piece-wise planar geometry, thus we can find the optimal solution in continuous depth space.

Depth-Texture Techniques. The core idea to solve the depth-texture problems is to jointly use both depth and color information. Intuition behind this idea is that texture image can provide significant information to enhance the raw range image. In real-world scene, depth discontinuities often co-occur with

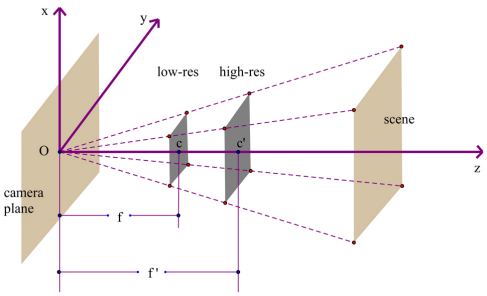


Fig. 1. Pin-hole camera model

intensity changes. Meanwhile, no-texture areas of the scene often appears to be continuous in depth changes. For example, the joint occurrence of depth and intensity edges, or smoothness of geometry in areas of largely uniform color. Diebel and Thrun [7] proposed a Markov Random Field (MRF) method for integrating low-res depth image with high-res color image, where the resulting MRF was defined through a depth measurement potential and a depth smoothness prior. A weighting factor on the smoothness prior was determined by color distances of adjacent image pixels. Based on this definition, Yang et al. [8] enforced smoothness in terms of both spatial resolution and depth precision. And Park et al. [9] further extended the smoothness term with an edge weighting scheme from high-res image features.

Joint bilateral filtering technique [10] is widely used to interpolate the high-res depth values, which incorporates spatiality of the depths with the high-res color image. Although bilateral filtering techniques perform fast [11], they suffer from over smoothing of fine details. As joint bilateral filtering also smoothes depth discontinuity boundaries, [8] used a sub-pixel estimation algorithm on quadratic polynomial interpolation. Zhu et al. [12] on the other hand combined both stereo and depth measurement for better discontinuity detection and a high accuracy depth map. Chan et al. [11] presented a noise-aware filter to address the specific requirements of depth super resolution and denoising from real-time 3D sensors. The noise-aware filter can prevent unwanted artifacts like edge blurring that the depth edges have similar color and texture copying that the smooth area has a distinct texture.

A number of recent work reveals that nonlocal means (NLM) filtering [13,14,9] can maintain fine detail and local structure. The NLM regularization follows similar idea as the bilateral filtering approach. However, it protects fine structures by allowing the pixels on the same nonlocal structure to reinforce each other within a larger neighborhood. In other words, the NLM filtering protects fine details at the cost of high computational complexity.

Remaining of this paper is arranged as follows. We first briefly describe the relationship of camera models between low and high-res images in Sec. 2. Sec. 3 describes our proposed uncertainty-based Bayesian approach to depth map

super resolution in detail. Experimental results and comparisons are presented in Sec. 4. And concluding remarks are drawn in the end.

2 Preprocessing

The input consists a low-res depth map and a high-res color image. Camera parameters are estimated using the calibration method introduced by Zhang [15].

The well-known pin-hole model describes the image information process using matrices A, R and t, describing the internal parameters, rotation and translation of the camera respectively. In the internal matrix

$$A = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

f_x and f_y are the focal length, expressed in pixels. s is the skew of the pixels. In CCDs skew is always zero. c_x and c_y are the coordinates of the principle point, which is the intersection of the optical axis with the image plane.

Assume the low-res depth map is captured using camera with internal matrix A and extrinsic parameters R and t. Then the corresponding high-res depth image is supposed to be captured with rotational matrix $R' = R$ and translational vector $t' = t$. In other words, they share the same extrinsic parameters. Difference between the two cameras lies in internal parameters. As shown in Fig. 1, $c = (c_x, c_y)$ and $f = (f_x, f_y)$ are principle point and focal length for the camera capturing low-res image, whereas c' and f' for the high-res camera. We assume the camera coordinate be accordance with the world coordinate, thus the depth axis z is coincident with the optical axis. Suppose η is the amplification factor, it is obvious that $c' = \eta c$ and $f' = \eta f$. Thus, the two cameras capture the same scene with different resolutions. Intrinsic parameter of the high-res camera is then

$$A = \begin{bmatrix} \eta f_x & s & \eta c_x \\ 0 & \eta f_y & \eta c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

3 A Bayesian Approach to Depth Map Super Resolution

Given the registered color and depth images $J = \{I_c, I_d\}$ and the corresponding internal parameters A of the low-res depth camera, our goal is to recover the target high-res depth map X.

We use Bayesian MAP to find the optimal solution

$$X^* = \arg\max_X p(X|J) \tag{3}$$

where the posterior is the product of likelihood and prior

$$p(X|J) \propto p(J|X)p(X) \tag{4}$$

Here we ignore the numerator $p(J)$ as it is constant.

3.1 Geometry Prior

We use the piece-wise planar hypotheses to represent geometry of the scene. This assumption fits most of the indoor and outdoor environment [16,17,18,19]. A depth measure is of high prior probability if it is close to the hypothetical geometry of the scene. The plane hypotheses for the input depth map is obtained using RANSAC. We first reconstruct point cloud of the specifical scene. Neighborhood of the point cloud follows the four-neighbor lattice, which is in accordance with their correspondences on the depth image. Then we seek to find multiple locally fit models. There are several important aspects for achieving a good set of planes.

Sampling. A plane model can be obtained from three point from the point cloud sampled at random. The first point is randomly selected. The second and third point are selected from an $M \times M$ neighborhood centered at the first point.

Inlier Selection. Each model is evaluated against all the points which are not assigned to an proper plane. Instead of scoring simply by the inlier count, where the number of points within a threshold distance to the plane is calculated, we score by the likelihood of each point fitting the plane, according to the MLE-sac method [20].

This procedure is repeated inside RANSAC loop. Our scheme supports the idea that large planes are more likely to be found at first than small planes. Thus, we can avoid fragments and produce robust models. In this way, we reconstruct multiple plan models from input low-res depth map. Suppose there are totally N planes that pixel q relates to, we denote $PLN_{q,1}, PLN_{q,2}, ...PLN_{q,N}$ the hypothetical depths from the prior models. And suppose X_q is the estimated depth of q, thus, and each plane is weighted by distance measurement to X_q. In this way, our prior is calculated using

$$p(X) = \prod_{q \in X} \exp(-\sum_{i=1}^{N} w_{q,i} \, |PLN_{q,i} - X_q|) \tag{5}$$

where $w_{q,i}$ is weight of each candidate plane and is expressed as

$$w_{q,i} = \frac{(PLN_{q,i} - X_q)^2}{\sum\limits_{i=1}^{N} (PLN_{q,i} - X_q)^2} \tag{6}$$

3.2 Uncertainty Modeling

Our objective is to reconstruct an accurate high-res depth map, where accuracy can be described in terms of uncertainty. A measurement would reach the requisite accuracy when its uncertainty value is small. From this perspective, uncertainty can be represented using accuracy rating. Suppose d_m is the measured depth and d_e is the estimate one. If d_m locates far away from d_e, measurement d_m becomes more unreliable. A 1D Gaussian is a suitable analogy to uncertainty, where small variance represents centralization at the mean, demonstrating determinacy of the data. As the input depth contains intrinsic noise from the sensor, we are reasonable to assign an uncertainty to each of the input measurements.

Each pixel q in the input low-res depth map contains location information in the x-y coordinate of the image plane as well as a depth value in z direction. As the input measurement may contain errors, we impose a noise on it. Imaging that each pixel of the input depth map is surrounded by an ellipsoid, where the estimated depth may be recovered within this ellipsoid. If the true depth of q is far away from the ellipsoid center, we say that the measurement of q is of high uncertainty. It is not surprising that in the ellipsoid, the major axis is along z coordinate because of greater uncertainty at the depth dimension. With this image in mind, we are now easy associate point uncertainty with precision rating, where smaller uncertainty of a point means larger probability of being accurate. Assume the depth dimension is independent from the spatial coordinate, then the accuracy probability of one pixel is $p = p_z \cdot p_{x,y}$. Here p_z and $p_{x,y}$ are probability on the depth direction and spatial image plane, respectively. Our goal is to maximize the total probability and obtain a high-res depth map with sub-pixel precision.

We use kernel functions to simulate uncertainty. We assume a Gaussian kernel along z axis, and a spatial-color anisotropic kernel on the image plane. Suppose $q = (q_x, q_y, q_z, q_c)$ is a pixel with (q_x, q_y) the coordinate on image plane, q_z the recovered depth, and q_c its intensity value. Thus, the kernel function of q centered at the corresponding measurement $\mu = (\mu_x, \mu_y, \mu_z, \mu_c)$ is

$$K(q, \mu) = K_z(q, \mu) K_{x,y}(q, \mu) \tag{7}$$

where the kernel along depth direction is

$$K_z(q, \mu) = K_G(q_z, \mu_z, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(q_z - \mu_z)^2}{2\sigma^2}} \tag{8}$$

and the spatial-color kernel on the image plane is

$$K_{x,y}(q, \mu) = \prod_{q' \in N(q)} e^{-\lambda \left\| q_c - q'_c \right\|_2} \tag{9}$$

In Eqn. 8, σ is the variance in the Gaussian kernel K_G; and Eqn. 9 states that, if q' is within q's neighborhood $N(q)$, we would calculate the spatial-color kernel by

measuring color distance between them. On calculating the spatial-color kernel, we can enforce that the image edges be coherent with depth changes within local neighborhood. In our formulation, the depth kernel denotes fidelity of the input, and the spatial-color anisotropic kernel works as a smoothness term to regulate continuous changes of the neighboring depths. Thus, the likelihood is

$$p(J|X) = K(q,\mu) = \prod_q \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(q_z-\mu_z)^2}{2\sigma^2}} \cdot \prod_q \prod_{q' \in N(q)} e^{-\lambda \left\| q_c - q'_c \right\|_2} \tag{10}$$

where q and μ are pixels from the unknown high-res depth image X and the low-res input depth map I_d respectively. So the uncertainty of a pixel is formulated as joint distribution of both spatial and depth kernels centered at the input measurement.

3.3 Optimization

Given geometry prior, we estimate the high-res depth image X by solving

$$\begin{aligned} X^* = & \arg\min_X \sum_q \sum_{q' \in N(q)} \lambda \|q_c - q'_c\|_2 \\ & + \sum_q \tfrac{(X_q-\mu_z)^2}{2\sigma^2} + \log(\sqrt{2\pi}\sigma) + \sum_{i=1}^{N} w_{q,i} \, |PLN_{q,i} - X_q| \end{aligned} \tag{11}$$

where $X_q \in X$ is the variable to be optimized, which is equivalent to q_z in Eqn. 8.

We optimize this objective function using α-expansion algorithm from the Middlebury computer vision pages. Computational time for a 640×480 image is about 4.5 minute on an Inter(R) Core (TM) 2 Duo workstation with 3 GB available RAMs.

4 Experimental Results

In order to verify capability and accuracy of our algorithm, we test it on real indoor scenes. Suppose resolution of the input low-res depth map is $m \times n$, with 128 depth layers, and the input high-res color image size is $2m \times 2n$. Our final goal is the optimized high-res depth map be of the same size as the input color image, and with a 2 times depth precision incensement.

We use two datasets, the first is the public RGB-D Object Dataset [21], and the other is the Middlebury stereo dataset. In both datasets, the input depth and color images are already registered. We down sample the ground truth depth map using nearest interpolation to create the low-res depth input. The original color image is used as the high-res RGB image.

For the prior, each candidate plane under the piece-wise assumption is modeled using RANSAC and points are assigned to its nearest plane. Each RANSAC

loop contains 200 iterations to find the proper plane model. During each iteration, errors of the unassigned points fits the plane is calculated and scored. In the end, the plane fits most of the points is selected. A point belongs to a plane model if its residual error is less-than a threshold. In our experiment, the number of hypothesised planes for each point N is set to be 10. In most scenes, we found that 10 plane models are sufficiently precise to represent the prior.

For the likelihood, variance of the Gaussian kernel on depth dimension is various due to different sources of the input error. However, this difference is not significant. To show robustness and clarity of our algorithm, we fix the variance for each pixel. In our experiment, we set $\sigma = 1$. In fact, we have tested different values of σ and found no big influence on the final result. The parameter λ on spatial-color kernel is set to be 10.

Table 1. Quantitative evaluation

	White Board				Gifts			
	MSE	NMSE	SNR	PSNR	MSE	NMSE	SNR	PSNR
Bilateral	0.0484	0.0054	9.8544	26.2945	0.0376	0.0050	12.4615	28.4973
Guided [22]	0.0499	0.0057	9.5967	26.0369	0.0347	0.0042	13.1524	29.1883
Method [9]	0.0378	0.0033	12.0200	28.4601	0.0356	0.0044	12.9473	28.9832
Ours	**0.0264**	**0.0016**	**15.1252**	**31.5653**	**0.0266**	**0.0025**	**15.4800**	**31.5159**
	Box				Dolls			
	MSE	NMSE	SNR	PSNR	MSE	NMSE	SNR	PSNR
Bilateral	0.0470	0.0143	12.3934	26.5648	0.0333	0.0026	15.3318	29.5425
Guided [22]	0.0471	0.0144	12.3665	26.5378	0.0312	0.0023	15.9106	30.1213
Method [9]	0.0408	0.0108	13.6112	27.7825	0.0333	0.0026	15.3388	29.5495
Ours	**0.0302**	**0.0059**	**16.2242**	**30.3956**	**0.0248**	**0.0015**	**17.9024**	**32.1131**
	Shelf				Rocks			
	MSE	NMSE	SNR	PSNR	MSE	NMSE	SNR	PSNR
Bilateral	0.0339	0.0032	14.6643	29.4083	0.0642	0.0088	9.2053	23.8487
Guided [22]	0.0290	0.0024	16.0201	30.7640	0.0559	0.0067	10.4009	25.0443
Method [9]	0.0319	0.0029	15.1704	29.9144	0.0581	0.0072	10.0746	24.7179
Ours	**0.0258**	**0.0019**	**17.0275**	**31.7715**	**0.0399**	**0.0034**	**13.3401**	**27.9835**
	Couch				Bowling			
	MSE	NMSE	SNR	PSNR	MSE	NMSE	SNR	PSNR
Bilateral	0.0445	0.0051	9.1421	27.0331	0.0336	0.0034	17.4447	29.4715
Guided [22]	0.0457	0.0054	8.9150	26.8061	0.0296	0.0026	18.5607	30.5876
Method [9]	0.0364	0.0034	10.8909	28.7819	0.0302	0.0027	18.3812	30.4081
Ours	**0.0355**	**0.0033**	**11.1157**	**29.0067**	**0.0268**	**0.0022**	**19.4237**	**31.4506**

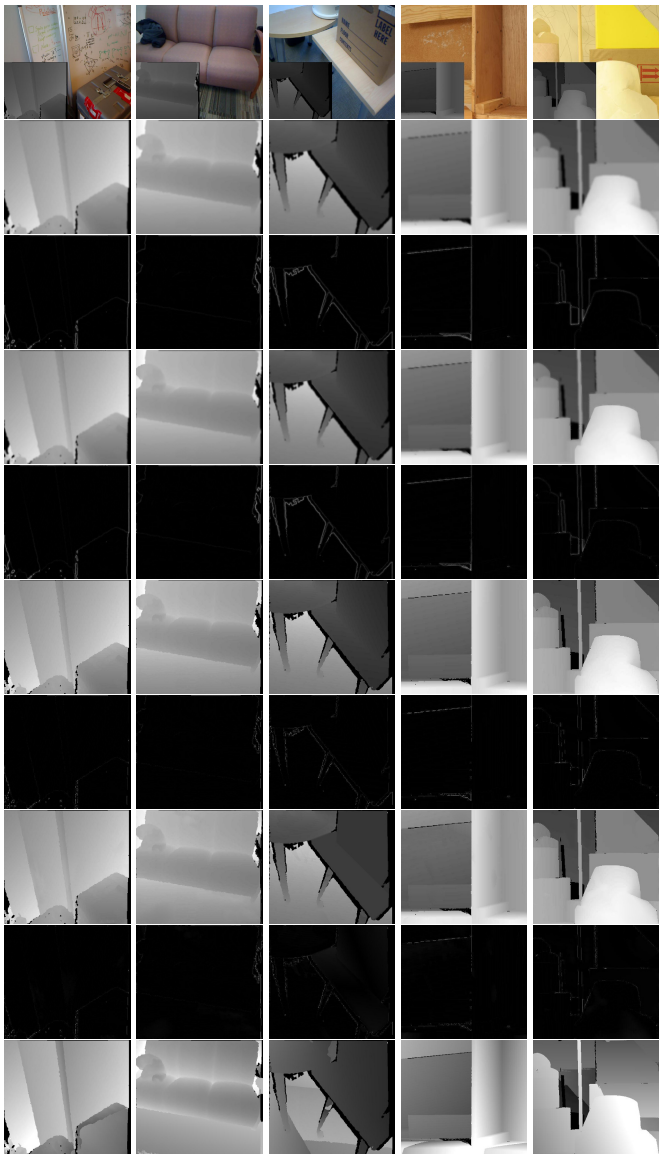


Fig. 2. Results on scenes with strong planar geometry. Each column shows one scene, from left to right, is white board, couch, box, cupboard, and lamp. The first row shows the input high-res color image and low-res depth image. The following 6 rows represent, result of bilateral filter, difference between bilateral filter estimation and the ground truth, result of guided image filter [22], difference between guided image filter estimation and the ground truth, result from [9] and difference between [9] and the ground truth. The 8th row shows our result, and the following shows difference between our method and the ground truth. And the last row shows one randomly chosen hypothesised planar model.

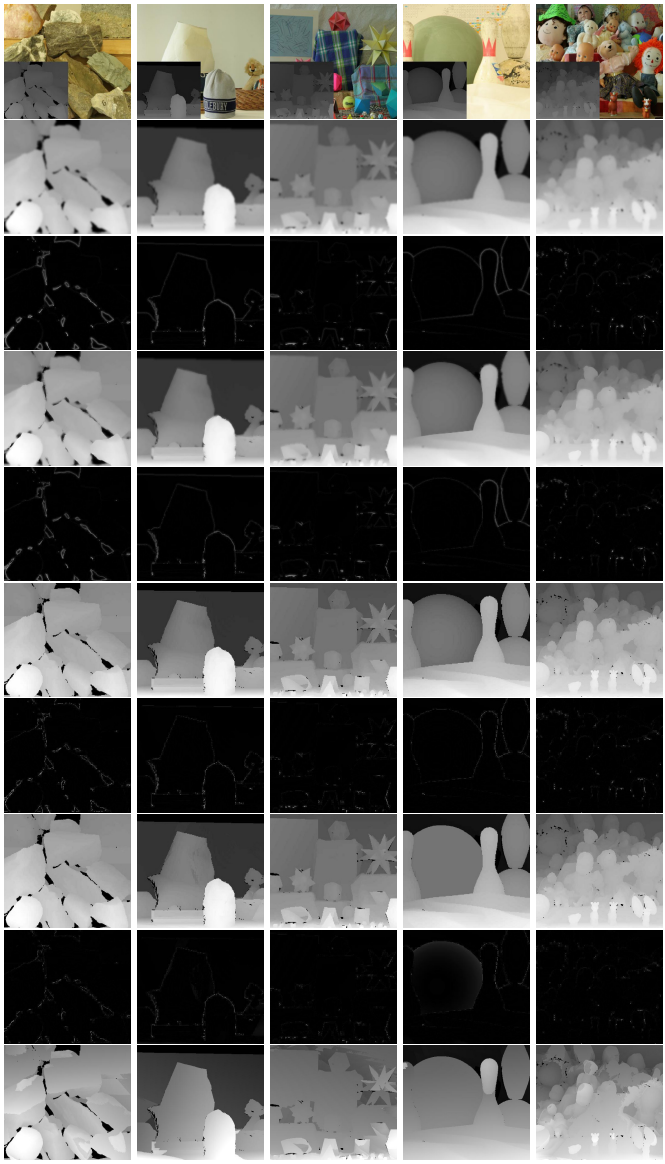


Fig. 3. Results on scenes without planar geometry. Each column shows one scene, from left to right, is rocks, middlebury, gifts, bowling, and dolls. The first row shows the input high-res color image and low-res depth image. The following 6 rows represent, result of bilateral filter, difference between bilateral filter estimation and the ground truth, result of guided image filter [22], difference between guided image filter estimation and the ground truth, result from [9] and difference between [9] and the ground truth. The 8th row shows our result, and the following shows difference between our method and the ground truth. And the last row shows one randomly chosen hypothesised planar model.

We compare our results with joint bilateral filtering, guided image filtering [22], and a recent work on high quality depth map upsampling [9]. Fig. 2 shows all the results from scenes that follow strong planar geometry. In terms of depth map quality, we find that joint bilateral filter smoothes not only flat areas, but also depth boundaries. The results from [9] rely heavily on the high-res color image, which is more likely to suffer from texture copying on textured areas, while our approach is the most capable to smooth flat areas and preserve details at depth boundaries. Fig. 3 shows experimental results on scenes that defy planar geometry. For these scenes, our algorithm also out-performs the stat-of-art. These are two explanations behind this phenomenon. First, small planes can approximately model the non-planar scene. Second, as the prior contains various weighted planar models, the final estimation is independent from any of the fixed planes.

For quantitative evaluation, we compared measures of Mean Square Error (MSE), Normalized Mean Square Error (NMSE), Signal to Noise Ratio (SNR), and Peak Signal to Noise Ratio (PSNR) among different methods on the RGB-D Object Dataset and the Middlebury stereo dataset. Tab 1 shows the quantitative results on both planar and non-planar scenes that our approach would get the lowest error measure and the highest signal ratio, demonstrating effectiveness and precision of our algorithm.

In most experiments our algorithm outperforms the prior arts, since it encodes piecewise planarity to increase both spatial and depth resolution. However, we also encounter some extreme failure cases as in Tab. 2. The reason behind is that wrong estimation of the hypothesised planes may not match the scene properly, and results in inferior estimation of the final depth.

Table 2. Some failure cases on PSNR measure

	Bilateral	Guided [22]	Method [9]	Ours
Art	26.8417	**28.5874**	28.0959	28.3605
Head	27.9672	27.8009	**29.3099**	27.0704

5 Conclusion

In this paper, we present a new method of depth map super resolution base on Bayesian framework. We make use of internal error of the depth sensor, and model these errors via kernel estimation. We suppose uncertainty on the depth dimension is independent from that of the spatial one. We also suppose the indoor geometry to be composed of piece-wise planes. We have validated our approach on several real datasets both with and without strong planar geometry. Comparisons to the stat-of-art demonstrate that our approach gives clear improvements. In the near future, we are to explore uncertainty-based depth map super resolution under multiple-view settings.

Acknowledgment. This work was supported by NSFC (61005037, 90920304, 61020106005, 10972002) and 973 Program (2011CB302202, 2012CB821203).

References

1. Liu, C., Sun, D.: A bayesian approach to adaptive video super resolution. In: CVPR, pp. 209–216 (2011)
2. Kil, Y., Mederos, B., Amenta, N.: Laser scanner super-resolution. In: PBG, pp. 9–16 (2006)
3. Schuon, S., Theobalt, C., Davis, J., Thrun, S.: High-quality scanning using time-of-flight depth superresolution. In: CVPRW (2008)
4. Schuon, S., Theobalt, C., Davis, J., Thrun, S.: Lidarboost: Depth superresolution for tof 3d shape scanning. In: CVPR (2009)
5. Li, J., Li, E., Chen, Y., Xu, L., Zhang, Y.: Bundled depth-map merging for multi-view stereo. In: CVPR, pp. 2769–2776. IEEE (2010)
6. Campbell, N.D.F., Vogiatzis, G., Hernández, C., Cipolla, R.: Using Multiple Hypotheses to Improve Depth-Maps for Multi-View Stereo. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 766–779. Springer, Heidelberg (2008)
7. Diebel, J., Thrun, S.: An application of markov random fields to range sensing. NIPS 18, 291 (2006)
8. Yang, Q., Yang, R., Davis, J., Nister, D.: Spatial-depth super resolution for range images. In: CVPR (2007)
9. Park, J., Kim, H., Tai, Y., Brown, M., Kweon, I.: High quality depth map upsampling for 3d-tof cameras. In: ICCV (2011)
10. Kopf, J., Cohen, M., Lischinski, D., Uyttendaele, M.: Joint bilateral upsampling. TOG 26, 96 (2007)
11. Chan, D., Buisman, H., Theobalt, C., Thrun, S.: A noise-aware filter for real-time depth upsampling. In: ECCVW (2008)
12. Zhu, J., Wang, L., Yang, R., Davis, J.: Fusion of time-of-flight depth and stereo for high accuracy depth maps. In: CVPR (2008)
13. Huhle, B., Schairer, T., Jenke, P., Straßer, W.: Fusion of range and color images for denoising and resolution enhancement with a non-local filter. CVIU 114, 1336–1345 (2010)
14. Favaro, P.: Recovering thin structures via nonlocal-means regularization with application to depth from defocus. In: CVPR (2010)
15. Zhang, Z.: A flexible new technique for camera calibration. PAMI 22, 1330–1334 (2000)
16. Sinha, S., Steedly, D., Szeliski, R.: Piecewise planar stereo for image-based rendering. In: ICCV, pp. 1881–1888 (2009)
17. Furukawa, Y., Curless, B., Seitz, S., Szeliski, R.: Reconstructing building interiors from images. In: ICCV, pp. 80–87. IEEE (2009)
18. Micusik, B., Kosecka, J.: Piecewise planar city 3d modeling from street view panoramic sequences. In: CVPR, pp. 2906–2912. IEEE (2009)
19. Gallup, D., Frahm, J., Pollefeys, M.: Piecewise planar and non-planar stereo for urban scene reconstruction. In: CVPR, pp. 1418–1425. IEEE (2010)
20. Torr, P., Zisserman, A.: Mlesac: A new robust estimator with application to estimating image geometry. CVIU 78, 138–156 (2000)
21. Lai, K., Bo, L., Ren, X., Fox, D.: A large-scale hierarchical multi-view rgb-d object dataset. In: ICRA, pp. 1817–1824 (2011)
22. He, K., Sun, J., Tang, X.: Guided Image Filtering. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 1–14. Springer, Heidelberg (2010)

Cross Image Inference Scheme for Stereo Matching

Xiao Tan[1,2], Changming Sun[2], Xavier Sirault[3], Robert Furbank[3], and Tuan D. Pham[4]

[1] SEIT of UNSW Canberra, Canberra, ACT 2610, Australia
xiao.tan@csiro.au
[2] CSIRO Mathematics, Informatics and Statistics, Locked Bag 17, North Ryde, NSW 1670, Australia
changming.sun@csiro.au
[3] CSIRO Plant Industry, Clunies Ross Street, Canberra, ACT 2601, Australia
{xavier.sirault,robert.furbank}@csiro.au
[4] Aizu Research Cluster for Medical Engineering and Informatics, The University of Aizu, Fukushima 965-8580, Japan
tdpham@u-aizu.ac.jp

Abstract. In this paper, we propose a new interconnected Markov Random Field (MRF) or iMRF model for the stereo matching problem. Comparing with the standard MRF, our model takes into account the consistency between the label of a pixel in one image and the labels of its possible matching points in the other image. Inspired by the turbo decoding scheme, we formulate this consistency by a cross image reference term which is iteratively updated in our matching framework. The proposed iMRF model represents the matching problem better than the standard MRF and gives better results even without using any other information from segmentation prior or occlusion detection. We incorporate segmentation information and the coarse-to-fine scheme into our model to further improve the matching performance.

1 Introduction

Researches have been carried out on stereo matching for many years. To formulate the stereo matching problem, most of the well performed algorithms use the Markov Random Field (MRF) formulation which is based on the assumption that the scene is piecewise smooth. Employing some other information or constraints such as color similarity [1], plane or curved surface hypotheses [2–5], and object recognition [6], stereo matching problem can be solved by minimizing an energy function. In models of all these algorithms, the source image is only used for calculating the correlation or for occlusion detection for the reference image. However, we find that the use of the source image in the stereo matching problem can go further.

The main contribution of this paper lies in the modification of the standard MRF model for stereo matching. Our new model is based on the idea that the

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 217–230, 2013.

labels of matching pixels should be consistent in most parts of both images. Therefore, given the disparity map of the reference image, we can infer the disparity map of the source image and vice versa. Thus, a pixel in the network of our interconnected MRF (iMRF) model is adjacent not only to its neighboring pixels in the same image but also to its potential matching pixels in the other image. As a result, two images are treated as a whole in this model and the labels in both images are updated simultaneously. Quantitative evaluations with ground truths show that by considering the consistency of the potential matching pixels, our new model improves the result from the standard MRF which only considers the consistency of pixels in the neighborhood within one image.

1.1 Previous Work

A survey of stereo matching problems and the quantitative evaluation of disparity estimation algorithms is reported by Scharstein and Szeliski [7]. Stereo matching algorithms can be roughly categorized into local and global algorithms. Local algorithms give acceptable results in the smooth and textured areas with relatively cheaper computation; however, any inappropriate selection of the shape or size of the support windows may cause the incidence of wrong estimation. To solve this problem, many techniques have been proposed using adaptive windows [8, 9], multiple-windows [10], or support weighted windows [11–13].

The global method is characterized by using an MRF stereo formulation which is further converted to the problem of optimization for a specific energy function. The design of an energy function has become the hottest research area in recent years. Employing different constraints such as the uniqueness constraint [14], ordering constraint [15], Ground Control Point constraint [16, 17], and segment constraint [18], these methods regularize the labeling under the Bayes rule. Finding the maximum solution for a specific energy function is usually a NP-hard problem; generally an approximate solution is desired. Several methods such as Mean-Field Annealing [19], Dynamic Programming [20], Graph Cut [21], and Belief Propagation [22], have been proposed to provide the approximate solution to the problem. In the models of most of the approaches mentioned above, only the consistency of labels of neighboring pixels is considered, and the consistency of labels of their matching pixels is ignored.

In previous researches, the labels for the source image are mainly used for occlusion detection. The visibility constraint detects the occluded pixels in the reference image by checking whether there exists at least one matching pixel from the source image [23]. In [24], the labels for the source image are used to define the possible disparity range for a given pixel under the visibility constraint. The unique configuration is used in [25] to enforce each pixel to participate only in one assignment to a pixel in the other image. The cross check requires the labels of two matching pixels to be equal based on the uniqueness constraint [26, 4]. None of these approaches use the labels of the source image when estimating the labels of the unoccluded pixels in the reference image. If one pixel in the reference image matches a pixel in the source image, it is intuitive that the latter pixel has a very high probability to match back to the former one. The

common shortcoming of the above mentioned researches lies on the difficulty of incorporating the hard constraint into the probability inference framework. The method in [27] gives a predefined penalty to the labels which break the label consistency between two views; it may give an over penalty to the horizontally slanted object, as stated in [23].

2 Interconnected MRF Model and Turbo BP Optimization

2.1 Two Properties of Stereo Matching

As disparities of pixels in both images are required in our framework, we consider an image as the reference image when its disparity map is being updated and consider the other image as the source image. Let P and P' be the point sets in the reference and source images, respectively. The set of possible matching pixels of a certain pixel $p\,(p \in P)$ is defined as

$$\psi\,(p) = \left\{p' \in P' | p'_y = p_y, B_l \leq p'_x - p_x \leq B_u\right\} \tag{1}$$

where x and y are the horizontal and vertical coordinates of a pixel. B_l and B_u are the lower and upper bounds of disparity search range. To unify the signs of disparities in both images, we define the disparity between p and p' as $d\,(p, p') = p^l_x - p^r_x$, where p^l is the pixel in the left image of the pixel pair and p^r is the one in the right image.

The first property is called the *equality constraint*: assuming p in the reference image matches p' in the source image with disparity d, if p' also matches p, the disparity of p' is strictly equal to d:

$$d\,(p, p') = d = d(p', p) \tag{2}$$

Another property of the stereo matching problem is that a certain pixel p in the reference image has a one-to-one interconnection to p' in the source image through a given disparity d. That is, there exists only one p' satisfying:

$$p' \in \psi(p) : d(p, p') = d \tag{3}$$

We call this the *interconnection constraint*. These two properties are self-evident considering the definition of disparity. We now describe our iMRF model that applies these two properties into our cross image inference scheme to improve the performance of stereo matching.

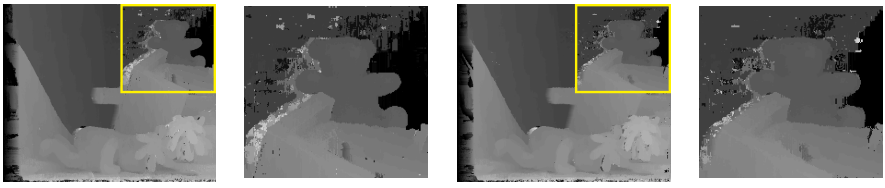
2.2 Interconnected MRF Model

The standard MRF model is used to formulate the local smoothness property in the neighborhood of pixels. However, the dependency between matching pairs is not formulated in this model. In other words, the probability of labeling $d(p, p')$ to p indicates the matching probability between p and p'. On the other hand,

this matching probability also influences the labeling $d(p', p)$ to p'. Denoting by f_p the label of p, we formulate this cross image label dependency as:

$$\Pr\{f_p = d(p, p')\} \propto \Pr\{f_{p'} = d(p', p)\} \tag{4}$$

Hence, given the matching probability of all pixels in one image of the stereo pair, we can obtain the inference for the matching probability of pixels in the other image. As a result, each pixel in the stereo image pair has two matching labels. One corresponds to the MRF model where the pixel is located; the other corresponds to the inference from its possible matching pixels in the other image. Fig. 1 shows a sample of two disparity maps (one is obtained from the data cost of the left image, the other is obtained from the data cost of the right image applying the cross image inference scheme that we proposed). As the possible matching pixels are also in an MRF model, the two MRF models are interconnected. We call this model the iMRF model. In this model, we consider the two labels to be equal for the reason that they are corresponding to the same pixel. The relation between the pixel and its two labels is similar to the relation of the data bit and its interpretations of two code sequences in the turbo coding scheme [28]. Inspired by the implementation of BP to the turbo decoding scheme [29], we give an maximum posterior probability (MAP) estimation to our proposed model using the Max-product BP.



(a) (b) (c) (d)

Fig. 1. (a) and (b) are the results after applying Winner-Take-All (WTA) matching to the data cost term of the left image. (c) and (d) are the results after applying WTA matching to the cross inference term from the data cost of the right image.

2.3 Turbo BP Optimization

In this section, we first present the network of our model in Fig. 2(a) and then show the message updating rule under the BP framework.

In Fig. 2(a), a solid line between pixels encodes the pair-wise smoothness constraint by a potential function V which can be a Potts model, a linear model or a quadratic model as discussed in [30]. A linear model used in our scheme is:

$$V(f_p, f_q) = \min(\rho_V |f_p - f_q|, T_V) \tag{5}$$

where ρ_V is a parameter which discourages disparity jump between neighboring pixels, and T_V is a truncation threshold which limits the penalty on the disparity jump at the disparity edge of a disparity map. The blue dash lines in Fig. 2(a)

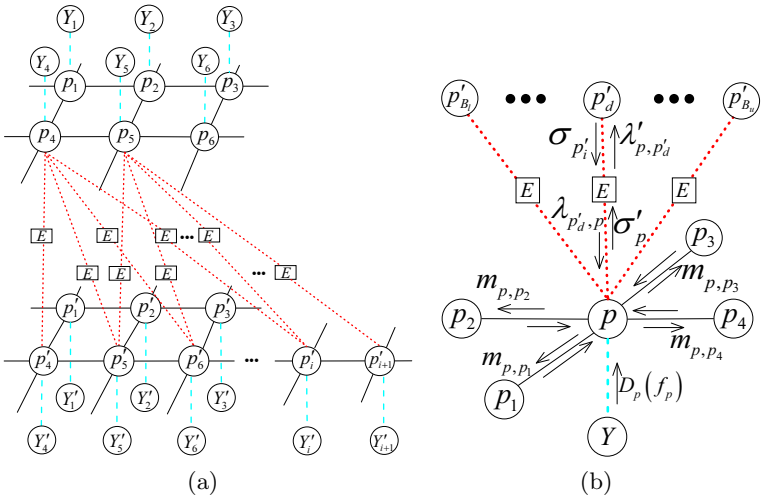


Fig. 2. (a) The network of our iMRF model. (b) A portion of the network showing message exchange.

represent the data costs Y which can be obtained from any correlation calculation algorithm. The red dash lines encode the cross image inference based on the assumption given in Eq. (4). Under this assumption, the labeling probability of a sender node is sent to its receiver node. As discussed in Section 2.1, the labels of two matching pixels should be equal to their disparity under the equality constraint which is denoted by the box with an "E" as shown in Fig. 2(a).

For clarity, a small portion of the full network is shown in Fig. 2(b). Here, we take one image as the reference image. As the two images are treated equally in our model, all discussions in the rest of this section can be similarly applied to the network when the other image is taken as the reference image. Let $N(p)$ be the set of neighboring pixels of p in the same image, p'_d be the matching points of p in the other image with disparity d. We denote a specific neighboring pixel of p in $N(p)$ by q. When the negative log model is used to formulate the probability, the message that p sends to q at iteration t under the Max-product rule is:

$$m^t_{p,q}(f_q) = \min_{f_p} \left(D_p(f_p) + V(f_p, f_q) + \sum_{p'_d \in \psi(p)} \lambda^{t-1}_{p'_d,p}(f_p) + \sum_{p_s \in N(p) \setminus q} m^{t-1}_{p_s,p}(f_p) \right) - \kappa_{p,q} \quad (6)$$

where f_i is the label of node i. D_p is the message sent by the data cost. $\lambda_{p'_d,p}$ is the message sent to pixel p by its possible matching pixel p'_d. $\kappa_{p,q}$ is a normalization factor for preventing overflow, which is constant for f_q but variable for pixel pairs. According to the interconnection constraint, for a given label f_p, only one possible matching pixel whose disparity with p is equal to f_p corresponding with this label. As a result, only one edge in the set of $\lambda_{p'_d,p}$ is activated for a given f_p. We let d be equal to f_p in Eq. (6) and then obtain,

$$m_{p,q}^{t}(f_q) = \min_{f_p} \left(D_p\left(f_p\right) + V(f_p, f_q) + \lambda_{p'_{f_p},p}^{t-1}\left(f_p\right) + \sum_{p_s \in N(p)\setminus q} m_{p_s,p}^{t-1}(f_p) \right) - \kappa_{p,q} \quad (7)$$

The message sent by p to p'_d at iteration t after applying the equality constraint following the Max-product rule is,

$$\lambda_{p,p'_d}^{\prime t}\left(f_{p'_d}\right) = \min_{f_p} \left(\sigma_p^{\prime t}\left(f_p\right) - \log\left(\delta(f_{p'_d} - f_p)\right) \right) = \sigma_p^{\prime t}\left(f_{p'_d}\right) \quad (8)$$

where $\delta()$ is the Dirac function and $\sigma_p^{\prime t}$ is the message sent out by p to p'_d before applying the equality constraint. According to the BP framework and the assumption given in Eq. (4), $\exp\left(-\sigma_p^{\prime t}(f_{p'_d})\right)$ should be proportional to the posterior probability of the label to p, given the labels of its possible matching pixels. The posterior probability is based on the summation over all incoming messages to the node p except the ones from edges between p and its possible matching pixels. We denote the summation of incoming messages for posterior probability calculation as J:

$$J\left(f_p\right) = \sum_{p_s \in N(p)} m_{p_s,p}^{t-1}(f_p) + D_p\left(f_p\right) \quad (9)$$

For different labels of p, the message of cross image inference is sent by different pixels in its possible matching pixel set. In order to make the message proportional to the posteriori probability, a normalization to J over all its possible matching pixels is necessary. For simplicity, we denote componentwise exponentiation and logarithmic on the message x by $x_{\exp}$ and $x_{\log}$. Furthermore, we introduce Pearl's α notation to define an operation on the message, which is similar to the operation on the vector described in [31]. $y = \alpha x$ means that $y\left(i\right) = x\left(i\right)\left(\sum_{k=1}^{n} x\left(k\right)\right)^{-1}$, for $1 \le i \le n$, where n is the dimension of the message. In other words, α converts a message to its probability vector whose elements are proportional to the values in the message. After defining these operations, $\sigma_p^{\prime t}\left(f_{p'_d}\right)$ is given by:

$$\sigma_p^{\prime t}\left(f_{p'_d}\right) = -\left(\alpha(-J)_{\exp}\right)_{\log}\left(f_{p'_d}\right) \quad (10)$$

As discussed in [32, 33], the labeling converges with the increase of the numbers of iterations and the message sent by the cross image inference in the first few iterations is not reliable. So the confidence of $\sigma_p^{\prime t}\left(f_{p'_d}\right)$ should be controlled by the number of iterations:

$$\sigma_p^{\prime t}\left(f_{p'_d}\right) = -w_\sigma\left(\alpha(-J)_{\exp}\right)_{\log}\left(f_{p'_d}\right) \quad (11)$$

where w_σ is a weighting factor which increases with the number of iterations given by $w_\sigma = i/i_{\max}$, where i is the number of iterations that has been performed, $i_{\max}$ is the total number of iterations needed which is a stopping criterion given by users. Since more reliable estimation from the cross image inference will

be obtained in the last few iterations compared with the estimation from the data cost, the effect of data cost on the message passing should be diminished as the number of iterations increases. Therefore, we weight the message from the data cost term by $w_D = 1 - w_\sigma$.

3 iMRF for Stereo Matching

In this section, we introduce our iMRF model into stereo matching via integrating segmentation information and the coarse-to-fine scheme.

3.1 Segmentation Prior

In our iMRF model, we use the segmentation prior which is formulated by 3D planes as a soft constraint. In order to avoid missing the extraction of the correct plane, we extract several possible planes for each segment using current disparity map and weight them accordingly.

We perform sequential RANSAC [34] on the obtained disparity map to calculate plane parameters for N_{R} times of the sequence or until no outliers are left. N_{R} is a parameter controlling the number of planes to be extracted for each segment, which is set to 5 in our implementation. The weight of each possible plane for a segment is given by its average cost. This is based on the fact that a correct plane has a low average cost. Given the cost volume, we define the average cost of an extracted plane as:

$$C^{(j)} = \frac{\sum\limits_{p \in S} D_p\big(f^{(j)}\big)}{\mathrm{card}\,(S)} \tag{12}$$

where j is the index of the plane, S is the set of pixels in a segment, $f^{(j)}$ is the plane-fitted label given by the jth plane. The cost in a stereo matching problem is a discrete function but $f^{(j)}$ is a continuous label. The subpixel estimation is obtained by linear interpolation between two nearest integer labels: $f_-^{(j)}\big(f_-^{(j)} \leq f^{(j)}\big)$ and $f_+^{(j)}\big(f_+^{(j)} \geq f^{(j)}\big)$:

$$D_p\big(f^{(i)}\big) = \big(f_+^{(i)} - f^{(i)}\big) D_p\big(f_-^{(i)}\big) + \big(f^{(i)} - f_-^{(i)}\big) D_p\big(f_+^{(i)}\big) \tag{13}$$

Then we weight the plane by a normalized negative exponent function based on the average cost of the plane:

$$w^{(j)} = \frac{\exp\big(-C^{(j)}\big)}{\sum\limits_{r} \exp\big(C^{(r)}\big)} \tag{14}$$

Given the possible planes and their weights, we use the truncated Total Variance model [22, 35] as our potential function:

$$\rho^{(j)}(f) = -\ln\Big((1 - T_s)\exp\Big(\frac{-\big|f - f^{(j)}\big|}{\eta}\Big) + T_s\Big) \tag{15}$$

where T_s controls the truncation and η controls the penalty of deviation from a fitted result. In this paper, T_s is set to $\exp\left(-\frac{S_r}{10}\right)$ and η is set to 1, where S_r is the disparity search range. Given the potential function and the weightings of planes, we define the plane fitting term as:

$$S\left(f\right) = w_S \sum_j w^{(j)} \rho^{(j)}\left(f\right) \tag{16}$$

where w_S is the weighting of the plane fitting term. Under this constraint, the update rule in Eqs. (7) and (9) should be changed accordingly:

$$m_{p,q}^t(f_q) = \min_{f_p} \Big(D_p\left(f_p\right) + V(f_p, f_q) + \lambda_{p'_{f_p},p}^{t-1}\left(f_p\right) + \sum_{p_s \in N(p)\setminus q} m_{p_s,p}^{t-1}(f_p) + S_p\left(f_p\right) \Big) - \kappa_{p,q} \tag{17}$$

$$J\left(f_p\right) = \sum_{p_s \in N(p)} m_{p_s,p}^{t-1}\left(f_p\right) + D_p\left(f_p\right) + S_p\left(f_p\right) \tag{18}$$

where S_p is the plane fitting term which is defined in Eq. (16) for pixel p.

3.2 Coarse-to-Fine Scheme

There are three types of messages in our algorithm: the smoothness message, the cross image inference message, and the plane fitting message. In our implementation, we use an amended version of Multi-Grid BP [30] to initialize the smoothness message. The initialization of the cross image inference message at a higher level in a pyramidal scheme is obtained by the summation of all messages of pixels in the corresponding block at the finest level, which is calculated using Eq. (18) on the estimation of message m at the finest level and data cost D_p. The plane fitting message is obtained from the plane fitting result, which does not need to be initialized.

We build the cost pyramid from the finest level to the coarsest level as described in [30]. Because a linear cost function is used for the smoothness term, the discontinuity cost is constant. The smoothness message in the next level can be estimated directly from the smoothness message at the corresponding block in the current level. Assuming the finest level is level 1, the cross image inference message and the plane fitting message used at level l is the summation within a block of 2^{l-1} by 2^{l-1} region in the finest level. The estimation is computed using Eqs. (16) and (18). The message m needed in the computation for the cross image inference message and plane fitting is approximated by the message m at the corresponding place in the current level at the latest iteration.

3.3 Procedure for Stereo Matching Using iMRF

The steps of our stereo matching algorithm are:

1. Compute the correlation volume as the data cost; build a cost volume pyramid from the finest level to the coarsest level L. Set current level to L and initialize all messages in the current level to 1.

2. Use the plane fitting scheme described in Section 3.1 on current disparity map for both images and obtain S message as given in Eq. (16).
3. Initialize $i = 1$ and start to update the message.
   (a) Use Eq. (17) to update the message m in both images at level l.
   (b) Use the message m at the current level (l) to approximate the corresponding message m at the finest level. Calculate the estimation of cross image inference message λ at the finest level using Eq. (18) for both images.
   (c) Obtain the cross image inference message λ at the current level (l) for both images by the summation of λ at the finest level.
   (d) $i = i + 1$; if $i = i_{\max}$ go to step (e); otherwise go back to step (a).
   (e) Compute the current disparity map by

$$f_p = \arg\min_f \left(D_p(f) + \sum_{q \in N(p)} m_{q,p}(f) + \lambda_{p'_f,p}(f) + S_p(f) \right) \tag{19}$$

   if $l = 1$ go to step (4); otherwise initialize m at the next level using the corresponding m at the current level (l); initialize λ for next level by the summation of λ at the finest level obtained in step (b); set $l = l - 1$ and then go to step (2).
4. Obtain the disparity map.

In our application, we update the plane fitting term once in each level of the pyramid as computing the plane parameters is relatively expensive.

4 Experimental Results

We implement the algorithm using Visual C++ 2008 and test images from the Middlebury website [36] and our own images. In the first experiment, we do not use the segmentation information to compare our proposed iMRF model with the standard MRF model. Being a generic stereo matching model, our model does not require any specific data cost acquisition scheme. The data cost can be obtained using any different algorithms such as adaptive support-weight approach [13], cross-based approach [37], or 3D-support windows [38]. In our experiments, we use the cross-based approach [37] to calculate the pixel correlation.

Parameters which affect the performance of the two models are ρ_d, T_V, and ρ_V. ρ_d is a value related to the pixel correlation as the data cost. T_V is the penalty to large depth jumps. ρ_V and ρ_d control the smoothness of the result. For simplicity, we set ρ_V to 1 and change ρ_d and T_V in our experiments.

In our experiments, ρ_d is in the range from 0.2 to 0.9 and T_V is set to L_d/N, where L_d is the number of disparity levels, N varies from 5 to 10. Fig. 3 shows the performance of the two models under different parameter settings.

The result shows that the parameter ρ_d has much more influence to the bad pixel percentage than the parameter T_V, which can be seen from Fig. 3. We then test our proposed model on different datasets with different ρ_d and fixed T_V which is set to $L_d/10$. The results and comparisons are shown in Table 1.

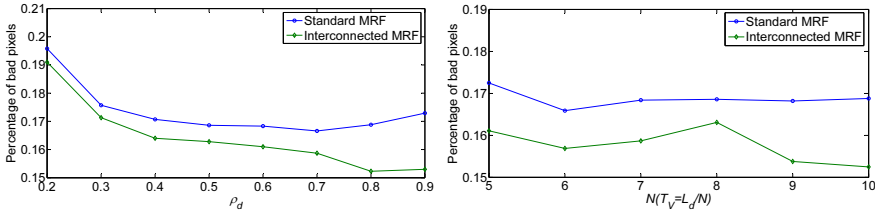


Fig. 3. Left graph: The percentage of bad pixels with respect to ρ_d for the Teddy dataset, by setting $T_V = L_d/10$. Right graph: The percentage of bad pixels with respect to T_V, by setting $\rho_d = 0.8$.

Table 1. Comparison results between the standard MRF and our iMRF according to percentage of bad pixels

ρ_d	0.3		0.5		0.7	
	Standard MRF	iMRF	Standard MRF	iMRF	Standard MRF	iMRF
Teddy	0.176	0.171	0.169	0.163	0.167	0.159
Tsukuba	0.032	0.038	0.028	0.029	0.028	0.027
Venus	0.019	0.018	0.027	0.022	0.026	0.013
Cones	0.111	0.131	0.115	0.114	0.124	0.114

Note: red score represents our iMRF having a better performance.

In our experiments, the iMRF model with our turbo BP algorithm provides a much better performance than the standard MRF with BP optimization. A sample of disparity results corresponding to the last two columns of Table 1 are shown in Fig. 4. Note that, we do not use any other information or constraints such as "segment", "texture", or "occlusion detecting" in our iMRF model. In our experiments, we use the Max-product BP inference scheme for both models with the same parameters. We believe the reason that our model provides better results is due to the fact that the matching problem is better formulated by using the probability inference between two images; however, the standard MRF only considers the consistency between the label of p and the labels of its neighboring pixels within one image and ignores the information from its potential matching points in the other image. In our model, we use the cross images inference term to encourage cross image consistency, which is much closer to the reality of the matching problem.

4.1 Results Using Segmentation

In our next experiment, we test our turbo BP algorithm by incorporating the segmentation information and the occlusion handling scheme as described in Section 3. The parameters used for the rest of the paper are: $\rho_d = 0.7$, $L = 5$, $i_{\max} = 10$, $\rho_V = 1$, $T_V L_d/10$, $w_S = 2 + 0.5\,(L - l)$. L is the number of the levels of the image pyramid, $i_{\max}$ is the number of the iterations within each level.

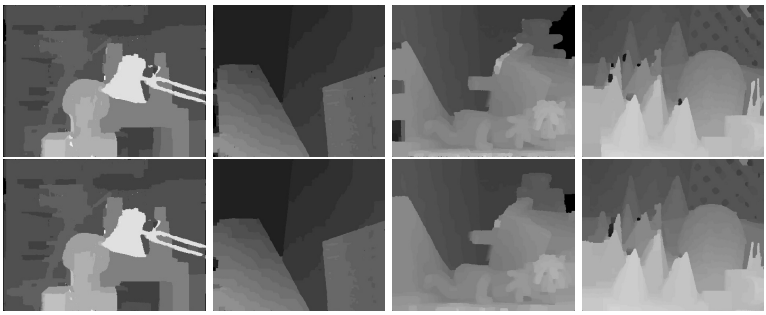


Fig. 4. Top row: Results from the standard MRF model; Bottom row: Results from our iMRF model

We set w_S to $2 + 0.5\,(L - l)$ for two reasons. First, the detailed information may not be available in the disparity map when using coarser level messages to approximate finer level messages; therefore, the plane fitting result is not very reliable for iterations in a coarser level. Second, as one pixel is associated with its four neighboring pixels in the smoothness term, we set the maximum value of w_S to be 4, the maximum discontinuity penalty, with the purpose that a plane fitting result will not break the smoothness constraint.

Our method has the top performance in the algorithms based on the symmetrical model. The comparison is summarized in Table 2. The final and intermediate results together with the ground truths are shown in Fig. 5.

The foreground and background with a slight color difference may be mistakenly regarded as one segment. As a result, some errors may occur at region boundaries due to these possible false segmentations. For example, there is an error at the right part of the paper box in the Tsukuba image pair. The runtime of our algorithm on the Tsukuba dataset without using segmentation information is 45 seconds, and it is 280 seconds when using segmentation information.

The results for some other image pairs in the Middlebury website [36] and our own image pairs are given in Fig. 6. The first two test images in Fig. 6 are from the Middlebury 2006 datasets. The third is a ground view of a tower with textureless sky as the background. The last image is the close-view of a rock. The results show that our algorithm performs well on many different types of images.

Table 2. The results of our algorithm with the Middlebury stereo data and comparisons with other methods which are based on the symmetrical model

Algorithm	Tsukuba			Venus			Teddy			Cones		
	unocc	all	disc	unocc	all	disc	unocc	all	disc	unocc	all	disc
OurMethod	1.14	1.51	5.98	0.17	0.38	2.04	5.72	9.97	15.0	3.14	8.95	8.86
SymBP+occ[23]	0.97	1.75	5.09	0.16	0.33	2.19	6.47	10.7	17.0	4.79	10.7	10.9
Segm+visib[39]	1.30	1.57	6.92	0.79	1.06	6.76	5.00	6.54	12.3	3.72	8.62	10.2
MultiCam GC[24]	1.27	1.99	6.48	2.79	3.13	3.60	12.0	17.6	22.00	4.89	11.8	12.10

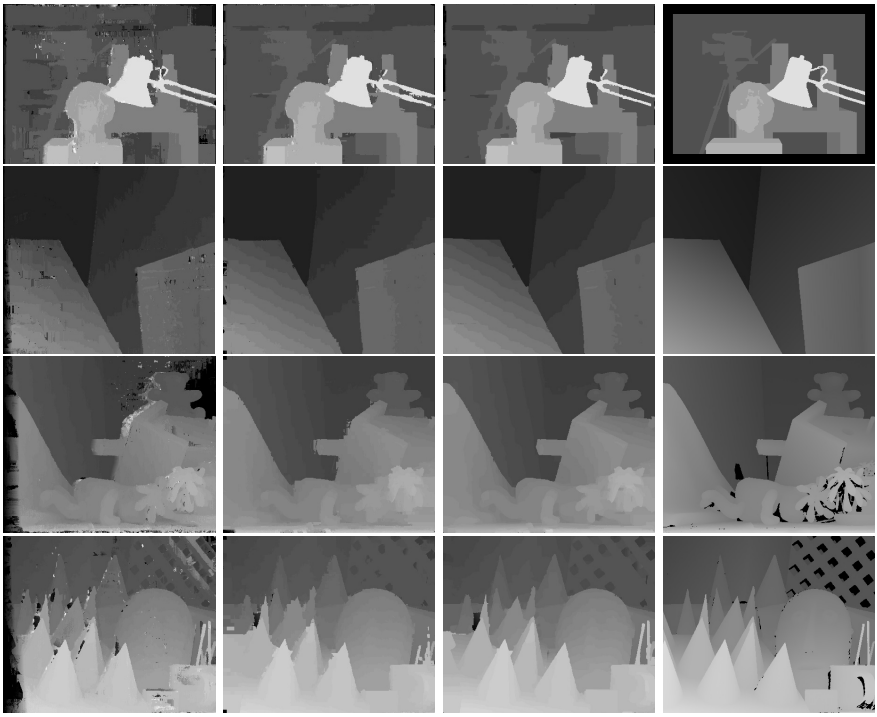


Fig. 5. Fist column: Data costs of datasets; Second column: Intermediate disparity maps from the second level of image pyramid; Third column: Final results of our iMRF based method; Last column: Ground truths of each dataset

Fig. 6. Top row: Original left images; Bottom row: Disparity maps obtained using our iMRF based method

5 Conclusions

A new iMRF model is proposed for stereo matching. In this model, the smoothness term is used for formulating the consistency of the labels of neighboring

pixels. The consistency of matching in two images is formulated by the cross image inference term which is iteratively updated cross both images. We use the Max-product belief propagation on the network of our iMRF model together with the segmentation information and the coarse-to-fine scheme to give an MAP estimation to the disparity problem. Experimental results show that our iMRF model gives a much better estimation to the stereo matching problem than the standard MRF model and the algorithm based on this model provides very good matching results.

Acknowledgements. We thank Chao Zhang for his comments. Tan was partially supported by the China Scholarship Council. Sun was partially supported by the CSIRO's Transformational Biology Capability Platform.

References

1. Larsen, E.S., Mordohai, P., Pollefeys, M., Fuchs, H.: Temporally consistent reconstruction from multiple video streams using enhanced belief propagation. In: ICCV, pp. 1–8 (2007)
2. Birchfield, S., Tomasi, C.: Multiway cut for stereo and motion with slanted surfaces. In: ICCV, vol. 1, pp. 489–495. IEEE (1999)
3. Tao, H., Sawhney, H., Kumar, R.: A global matching framework for stereo computation. In: ICCV 2001, vol. 1, pp. 532–539. IEEE (2001)
4. Yang, Q., Wang, L., Yang, R., Stewénius, H., Nistér, D.: Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. PAMI 31, 492–504 (2008)
5. Bleyer, M., Rother, C., Kohli, P.: Surface stereo with soft segmentation. In: CVPR, pp. 1570–1577 (2010)
6. Bleyer, M., Rother, C., Kohli, P., Scharstein, D., Sinha, S.: Object stereo - joint stereo matching and object segmentation. In: CVPR, pp. 3081–3088 (2011)
7. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV 47, 7–42 (2002)
8. Boykov, Y., Veksler, O., Zabih, R.: A variable window approach to early vision. PAMI 20, 1283–1294 (1998)
9. Veksler, O.: Stereo correspondence with compact windows via minimum ratio cycle. PAMI 24, 1654–1660 (2002)
10. Kang, S.B., Szeliski, R., Chai, J.: Handling occlusions in dense multi-view stereo. In: CVPR, vol. 1, pp. 103–110 (2001)
11. Darrell, T.: A radial cumulative similarity transform for robust image correspondence. In: CVPR, pp. 656–662 (1998)
12. Xu, Y., Wang, D., Feng, T., Shum, H.Y.: Stereo computation using radial adaptive windows. In: ICPR, vol. 3, pp. 595–598 (2002)
13. Yoon, K.J., Kweon, I.S.: Adaptive support-weight approach for correspondence search. PAMI 28, 650–656 (2006)
14. Zitnick, C.L., Kanade, T.: A cooperative algorithm for stereo matching and occlusion detection. PAMI 22, 675–684 (2000)
15. Ishikawa, H., Geiger, D.: Occlusions, Discontinuities, and Epipolar Lines in Stereo. In: Burkhardt, H.-J., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1406, p. 232. Springer, Heidelberg (1998)
16. Bobick, A.F., Intille, S.S.: Large occlusion stereo. IJCV 33, 181–200 (1999)

17. Wang, L., Yang, R.: Global stereo matching leveraged by sparse ground control points. In: CVPR, pp. 3033–3040 (2011)
18. Xu, L., Jia, J.: Stereo Matching: An Outlier Confidence Approach. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 775–787. Springer, Heidelberg (2008)
19. Geiger, D., Girosi, F.: Parallel and deterministic algorithms from mrfs: Surface reconstruction. PAMI, 401–412 (1991)
20. Sun, C.: Fast stereo matching using rectangular subregioning and 3D maximum-surface techniques. IJCV 47, 99–117 (2002)
21. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. PAMI 23, 1222–1239 (2001)
22. Sun, J., Zheng, N.N., Shum, H.Y.: Stereo matching using belief propagation. PAMI 25, 787–800 (2003)
23. Sun, J., Li, Y., Kang, S.B., Shum, H.Y.: Symmetric stereo matching for occlusion handling. In: CVPR, vol. 2, pp. 399–407 (2005)
24. Kolmogorov, V., Zabih, R.: Multi-camera Scene Reconstruction via Graph Cuts. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part III. LNCS, vol. 2352, pp. 82–96. Springer, Heidelberg (2002)
25. Kolmogorov, V., Zabih, R.: Computing visual correspondence with occlusions using graph cuts. In: ICCV 2001, vol. 2, pp. 508–515. IEEE (2001)
26. Egnal, G., Wildes, R.P.: Detecting binocular half-occlusions: Empirical comparisons of five approaches. PAMI 24, 1127–1133 (2002)
27. Wu, C., Frahm, J., Pollefeys, M.: Repetition-based dense single-view reconstruction. In: CVPR 2011, pp. 3113–3120. IEEE (2011)
28. Berrou, C., Glavieux, A., Thitimajshima, P.: Near Shannon limit error-correcting coding and decoding: Turbo-codes (1). In: IEEE International Conference on Communications, vol. 2, pp. 1064–1070 (1993)
29. McEliece, R.J., MacKay, D.J.C., Cheng, J.F.: Turbo decoding as an instance of Pearl's belief propagation algorithm. IEEE Journal on Selected Areas in Communications 16, 140–152 (1998)
30. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient belief propagation for early vision. IJCV 70, 41–54 (2006)
31. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann (1988)
32. Pyndiah, R.M.: Near-optimum decoding of product codes: Block turbo codes. IEEE Transactions on Communications 46, 1003–1010 (1998)
33. Lehmann, F.: Turbo segmentation of textured images. PAMI 33, 16–29 (2010)
34. 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, 381–395 (1981)
35. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena 60, 259–268 (1992)
36. Scharstein, D., Szeliski, R. (2011), `http://www.vision.middlebury.edu/stereo/`
37. Zhang, K., Lu, J., Lafruit, G.: Cross-based local stereo matching using orthogonal integral images. CSVT 19, 1073–1079 (2009)
38. Richardt, C., Orr, D., Davies, I., Criminisi, A., Dodgson, N.A.: Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part III. LNCS, vol. 6313, pp. 510–523. Springer, Heidelberg (2010)
39. Bleyer, M., Gelautz, M.: A layered stereo algorithm using image segmentation and global visibility constraints. ICIP 5, 2997–3000 (2004)

Bayesian Epipolar Geometry Estimation from Tomographic Projections

Sami S. Brandt, Katrine Hommelhoff Jensen, and François Lauze

Department of Computer Science, University of Copenhagen, Universitetsparken 1, 2100 Copenhagen, Denmark

Abstract. In this paper, we first show that the affine epipolar geometry can be estimated by identifying the common 1D projection from a pair of tomographic parallel projection images and the 1D affine transform between the common 1D projections. To our knowledge, the link between the common 1D projections and the affine epipolar geometry has been unknown previously; and in contrast to the traditional methods of estimating the epipolar geometry, no point correspondences are required. Using these properties, we then propose a Bayesian method for estimating the affine epipolar geometry, where we apply a Gaussian model for the noise and non-informative priors for the nuisance parameters. We derive an analytic form for the marginal posterior distribution, where the nuisance parameters are integrated out. The marginal posterior is sampled by a hybrid Gibbs–Metropolis–Hastings sampler and the conditional mean and the covariance over the posterior are evaluated on the homogeneous manifold of affine fundamental matrices. We obtained promising results with synthetic 3D Shepp–Logan phantom as well as with real cryo-electron microscope projections.

1 Introduction

The common line theorem states that there is a common 1D projection shared by a pair of 2D tomographic parallel projections of a 3D object [1]. The common 1D projection follows from the Fourier slice theorem or the fact that the 2D Fourier transform of a tomographic parallel projection forms a central slice of the 3D Fourier transform of the object [2]. Two central slices generally intersect on a 3D line, whose image interpretation, after using the Fourier slice theorem once again on the image plane, is a 1D projection that is identical in the two projection views. Geometrically, the common 1D projection can be identified as the projection of the rotation axis onto the calibrated views [3].

We will show that the common lines and the affine epipolar geometry [4] are related. The relation comes from the fact that the projection over the corresponding epipolar lines is identical in the normalised calibrated views since in this case it is the integral over the corresponding epipolar plane. Moreover, the common 1D projection is the corresponding line orthogonal to the epipolar lines as the rotation axis of the object is orthogonal to the epipolar planes. In the more general uncalibrated case, one has to find the 1D affine transform to link

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 231–242, 2013.

the points on the common 1D projections, in addition to the direction of the common 1D projection, in order to recover the affine epipolar geometry.

After proofing this geometric relationship, we propose a Bayesian method to estimate the affine epipolar geometry for a pair of views where no pointwise correspondences [5] nor explicit 3D reconstruction [6] are required. The marginal posterior distribution of the geometry is derived in closed-form using certain non-informative priors for nuisance parameters. To sample the posterior, we propose a hybrid Gibbs–Metropolis–Hastings sampler, where the proposal distributions are generated by spline interpolated approximations of the 1D conditional distributions. The posterior samples allow us to estimate the conditional mean and covariance over the homogeneous manifold of affine fundamental matrices.

The results of this paper are central in applications that rely on the geometric relationship between parallel tomographic projections such as x-ray or transmission electron microscope images. Due to the statistical nature of the proposed method, this is especially a very promising tool to geometrically register single particle images in cryo transmission electron microscopy [7] aiming at 3D reconstruction of the particle. In cryo-EM, conventional computer vision methods, based on point correspondences [3,8], are not applicable due to the very high level of noise.

The organisation of this paper is as follows. In Section 2, we show the formal relationship between the common lines and affine epipolar geometry. In Section 3, we derive the marginal posterior distribution for the affine epipolar geometry for a pair of tomographic parallel projections. In Section 4, we describe how we sample and summarise the posterior. The experiments are in Section 5, and Section 6 concludes the paper.

2 Epipolar Geometry and Common Lines

We start with showing the geometric relationship between the common 1D projections with uncalibrated parallel projection geometry. Without a loss of generality, we may identify the common 1D projection as the line through the image origin, where the line has the direction angles θ_1, θ_2 in the views one and two, respectively. The common 1D projections p_1 and p_2 onto the identified line are the integrals over the parallel lines orthogonal to the line (see Fig. 1). With uncalibrated projection geometry, the common 1D projections also have an unknown affine ambiguity after the directions have been identified. More formally, we state as follows.

Theorem 1. *Given two uncalibrated tomographic projection images I_1 and I_2 of a bounded object, there are corresponding 1D projections $p_1(x)$ and $p_2(x)$ that are related by $p_1(x) = \gamma p_2(\alpha x + \beta)$ where α, β and γ are unknown scalar constants.*

Proof. Without a loss of generality, the projection of the inhomogeneous 3-vector $\mathbf{X} \in \mathbb{R}^3$ onto the normalised, calibrated coordinates onto two 2D images can be described by the geometric projection equation

$$\mathbf{u} = \mathbf{P}\mathbf{R}\mathbf{X}, \quad \mathbf{u}' = \mathbf{P}\mathbf{R}'\mathbf{X} \tag{1}$$

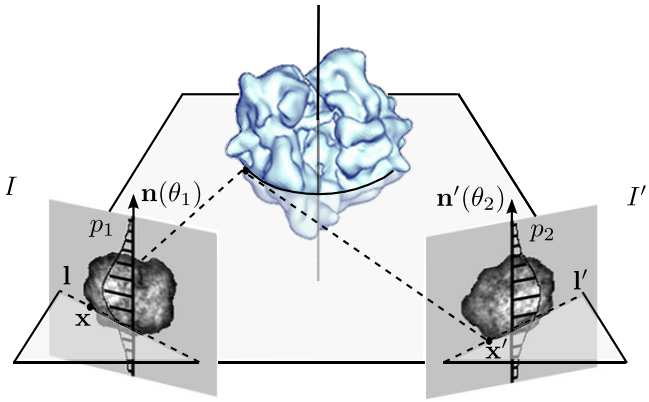


Fig. 1. An illustration of the connection between the common 1D projection and affine epipolar geometry. Two 2D parallel projection images I and I' of the 3D object are illustrated with the common 1D projections p_1 and p_2 as well as matching 2D points $\mathbf{x}$ and $\mathbf{x}'$ along the corresponding epipolar lines $\mathbf{l}$ and $\mathbf{l}'$. The value on the common 1D projection represents the 3D object density integrated over the epipolar plane.

where $\mathbf{u}, \mathbf{u}' \in \mathbb{R}^2$ are the corresponding projected 2D coordinates vector in the normalised views $\tilde{I}_1$, $\tilde{I}_2$, $\mathbf{P}$ is the 2×3 orthographic projection matrix with the unity diagonal, and $\mathbf{R}$, $\mathbf{R}'$ are 3D rotations. The inhomogeneous image coordinates relate to the normalised coordinates by the affine transform

$$\mathbf{x} = \mathbf{K}\mathbf{u} + \mathbf{t}, \quad \mathbf{x}' = \mathbf{K}'\mathbf{u}' + \mathbf{t}', \tag{2}$$

where $\mathbf{K}$, $\mathbf{K}'$ are non-singular, 2×2 upper triangular matrices containing the affine parameters of the cameras, and $\mathbf{t}$, $\mathbf{t}'$ are 2×1 translation vectors.

From the well known common line theorem it follows that there is a common 1D projection between the normalised projection images. In other words, the integral over an epipolar plane is equal to the integral over the corresponding epipolar lines in the normalised images (Fig. 1). The common 1D projection is

$$\tilde{p}(s) = \int_{\tilde{\mathbf{n}}^{\mathrm{T}}\mathbf{u}=s} \tilde{I}_1(\mathbf{u})\mathrm{d}L = \int_{\tilde{\mathbf{n}}'^{\mathrm{T}}\mathbf{u}'=s} \tilde{I}_2(\mathbf{u}')\mathrm{d}L \tag{3}$$

where the normalised images $\tilde{I}_1(\mathbf{u}) = I_1(\mathbf{K}\mathbf{u} + \mathbf{t})$ and $\tilde{I}_2(\mathbf{u}') = I_2(\mathbf{K}'\mathbf{u}' + \mathbf{t}')$, and the unit vectors $\tilde{\mathbf{n}}$, $\tilde{\mathbf{n}}'$ are orthogonal to the epipolar lines in the normalised images, respectively. For simplicity, we define the sign of the unit vectors so that they point towards the positive direction of the 3D rotation axis defined by the 3D rotation from the view 1 to the view 2.

Let us first consider the first view. We may make the coordinate transform for the normalised images $\tilde{I}_1$ by making the substitution $\mathbf{u} = \mathbf{K}^{-1}(\mathbf{x} - \mathbf{t})$. The Jacobian of the substitution mapping is $\mathbf{K}^{-1}$, hence,

$$\tilde{p}(s) = \int_{\tilde{\mathbf{n}}^{\mathrm{T}}\mathbf{u}=s} \tilde{I}_1(\mathbf{u})\mathrm{d}L = c\int_{\mathbf{n}^{\mathrm{T}}\mathbf{u}=t} I_1(\mathbf{x})\mathrm{d}L = cp_1(t), \tag{4}$$

where $c = |\det(\mathbf{K})|^{-1}$, $\mathbf{n}$ is the unit vector orthogonal to the epipolar lines in the unnormalised coordinate frame, where the sign of $\mathbf{n}$ set as for $\tilde{\mathbf{n}}$, $p_1(t)$ is the 1D projection of the image into perpendicular direction of $\mathbf{n}$, and t is the signed orthogonal distance of the epipolar line from $\mathbf{t}$. Making the corresponding analysis to the second view yields $\tilde{p}(s) = c'p_2(t')$, where $c' = |\det(\mathbf{K}')|^{-1}$. Since $c, c' \neq 0$, it follows that $p_1(t) = \gamma p_2(t')$, where $\gamma = c'/c$.

It remains to be shown that the 1D coordinates t and t' on the common 1D projections, respectively, are related by an 1D affine transform on the unnormalised image planes. For the unnormalised first view, the line corresponding to the common 1D projection in the normalised frame is $s\mathbf{K}\tilde{\mathbf{n}}+\mathbf{t}$, where s is the real parameter. On the other hand, we may equivalently parameterise the common 1D projection by the line $t\mathbf{n}$ which meets the origin, where $t = s\mathbf{n}^{\mathrm{T}}\mathbf{K}\tilde{\mathbf{n}} + \mathbf{n}^{\mathrm{T}}\mathbf{t}$. Similarly for the second view, $t' = s\mathbf{n}'^{\mathrm{T}}\mathbf{K}'\tilde{\mathbf{n}}' + \mathbf{n}'^{\mathrm{T}}\mathbf{t}'$, which implies that $t' = \alpha t + \beta$, where $\alpha = \frac{\mathbf{n}'^{\mathrm{T}}\mathbf{K}'\tilde{\mathbf{n}}'}{\mathbf{n}^{\mathrm{T}}\mathbf{K}\tilde{\mathbf{n}}}$ and $\beta = \mathbf{n}'^{\mathrm{T}}\mathbf{t}' - \frac{\mathbf{n}'^{\mathrm{T}}\mathbf{K}'\tilde{\mathbf{n}}'}{\mathbf{n}^{\mathrm{T}}\mathbf{K}\tilde{\mathbf{n}}}\mathbf{n}^{\mathrm{T}}\mathbf{t}$.

□

Corollary 1. *The scaling parameter γ between the common uncalibrated 1D projections of two views defines the scalar constraint $\gamma = |\det(\mathbf{K})/\det(\mathbf{K}')|$ for the affine camera parameters.*

The geometry of common 1D projections for a view pair is defined by the four parameters $(\theta_1, \theta_2, \alpha, \beta)$. On the other hand, the affine epipolar geometry is defined by the linear epipolar equation

$$f_{13}x' + f_{23}y' + f_{31}x + f_{32}y + f_{33} = 0, \tag{5}$$

where $f_{..}$ represents the elements of the affine fundamental matrix, (x, y) and (x', y') are the inhomogeneous coordinates of the points $\mathbf{x}$ and $\mathbf{x}'$ in the first and second view, respectively. The following theorem states the equivalence between the common line geometry and the affine epipolar geometry.

Theorem 2. *Given two uncalibrated tomographic parallel projection images I and I', identifying the corresponding 1D projections together with the 1D affine transform between the 1D coordinates is equivalent to identifying the affine epipolar geometry between the views.*

Proof. Assume first that the affine epipolar geometry is known, i.e., there is an affine fundamental matrix $\mathbf{F}$ [4] available. The epipolar line corresponding to a point $\mathbf{x} = (x, y, 1)$ in the first view is $\mathbf{l}' = \mathbf{F}\mathbf{x} = (f_{13}, f_{23}, f_{31}x + f_{32}y + f_{33})$ in the second view. Let the unit direction vector orthogonal to the epipolar line, defining the orientation of the 1D projection be $\mathbf{n}' = -a'(f_{13}, f_{23})/s'$ where $s' = \|(f_{13}, f_{23})\| > 0$, $a' = \pm 1$ and correspondingly let $\mathbf{n} = a(f_{31}, f_{32})/s$ where $s = \|(f_{31}, f_{32})\| > 0$, $a = \pm 1$ in the first view.[1] The cases $s = 0$ or $s' = 0$ are not possible since these cases do not represent valid affine epipolar geometries between the views. Hence, we may write $\mathbf{l}'=(-a's'\mathbf{n}', as\mathbf{n}^{\mathrm{T}}(x, y) + f_{33})$.

[1] The signs of a and a' can be chosen arbitrarily, but the selection $a' = a$ corresponds to the case where the normals $\mathbf{n}$ and $\mathbf{n}'$ point to the same side of the epipolar plane.

Now, using the standard formula for the distance between a point and a line, the signed orthogonal distance between the origin O' and the epipolar line $\mathbf{l}'$ is $d' = \zeta|as\mathbf{n}^{\mathrm{T}}(x,y) + f_{33}|/s' = \zeta|asd + f_{33}|/s'$ where the sign is set to $\zeta = a'\mathrm{sign}(as\mathbf{n}^{\mathrm{T}}(x,y) + f_{33})$. This corresponds to the choice that the signed distance $d = \mathbf{n}^{\mathrm{T}}(x,y)$ in the first view has the equal sign to the corresponding distance d' in the case where the normals $\mathbf{n}$ and $\mathbf{n}'$ point towards the same side of the epipolar plane, and the opposite sign otherwise.[2] Hence, $d' = a'(asd + f_{33})/s' = \alpha d + \beta$, where $\alpha = (as)/(a's')$ and $\beta = f_{33}/(a's')$. We have thus shown that the 1D projection directions and the parameters α and β of the affine transform between the identified 1D projections can be computed from a known affine fundamental matrix between the views.

Conversely, assume that we have identified the common 1D projection parameterised by the respective unit direction vectors $\mathbf{n}, \mathbf{n}' \in \mathbb{R}^2$, defining the positive directions of the 1D coordinate systems; and the constants $\alpha \neq 0$ and β defining the affine transform between the common 1D projections. Without a loss of generality, we may assume that the origins of the 1D projections have been set to the projections of the respective 2D image origins.

The unit normal of epipolar line $\mathbf{l}'$ is $\mathbf{n}'$, hence, $(f_{13}, f_{23}) = t'\mathbf{n}'$, where $t' \neq 0$ is a scalar. Similarly, $(f_{31}, f_{32}) = t\mathbf{n}$, $t \neq 0$. The corresponding points on the 1D projections $\mathbf{x} = (d\mathbf{n}, 1)$ and $\mathbf{x}' = ((\alpha d + \beta)\mathbf{n}', 1)$ must satisfy the epipolar equation $\mathbf{x}'^{\mathrm{T}}\mathbf{F}\mathbf{x} = 0$ for all d and d' that yields

$$\begin{aligned} 0 &= t'(\alpha d + \beta)\mathbf{n}'^{\mathrm{T}}\mathbf{n}' + td\mathbf{n}^{\mathrm{T}}\mathbf{n} + f_{33} \\ &= (t + t'\alpha)d + (f_{33} + t'\beta), \end{aligned} \tag{6}$$

thus, $t' = -t/\alpha$ and $f_{33} = t\beta/\alpha$.

Collecting the constraint equations into a matrix from yields

$$\begin{pmatrix} -\alpha/t & 0 & 0 & 0 & 0 \\ 0 & -\alpha/t & 0 & 0 & 0 \\ 0 & 0 & 1/t & 0 & 0 \\ 0 & 0 & 0 & 1/t & 0 \\ 0 & 0 & 0 & 0 & \alpha/t \end{pmatrix} \begin{pmatrix} f_{13} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{pmatrix} = \begin{pmatrix} \mathbf{n}' \\ \mathbf{n} \\ \beta \end{pmatrix} \tag{7}$$

where there is a unique solution for all $\alpha, t \neq 0$. The fundamental matrix is a homogeneous quantity so, it is defined only up to scale, thus the scale parameter t can be set to an arbitrary non-zero value. We have thus shown that identifying the corresponding 1D projections and the 1D affine transform between the 1D projections imply that the affine epipolar geometry is identified between the views.

□

[2] This can be seen from the fact that $\lim_{d\to\infty} \mathrm{sign}(asd + f_{33}) = a$, and $a = a'$ when the normals point towards the same side of the epipolar plane and $a = -a'$ otherwise.

3 Posterior Distribution for Epipolar Geometry

After showing the relationship between the common-lines and the affine epipolar geometry, we now derive the posterior distribution for the affine epipolar geometry. Let D denote the measured data or the pixel intensities in the images, and $\mathbf{p}_1$ and $\mathbf{p}_2$ the vectors representing the discretised 1D projections of the 2D planes, respectively. Assuming i.i.d. zero mean Gaussian noise on the 1D projections, we should consider the likelihood, associated with the reprojection error,

$$L(D|\theta) = \sigma^{-2N} \exp\left(-\frac{1}{2\sigma^2}\left(\|\mathbf{p}_1 - \gamma\hat{\mathbf{p}}_2\|^2 + \|\mathbf{p}_2 - \hat{\mathbf{p}}_2\|^2\right)\right) \tag{8}$$

where $\theta = (\hat{\mathbf{p}}_2, \theta_1, \theta_2, \alpha, \beta, \gamma, \sigma)$, N is the number of discrete points on the projection $\mathbf{p}_i = \mathbf{p}_i(\theta_i)$, $i = 1, 2$; and the noise free projections $\hat{\mathbf{p}}_1$ and $\hat{\mathbf{p}}_2$ are related by $\hat{\mathbf{p}}_1 = \gamma\hat{\mathbf{p}}_2$ according to Theorem 1.

The noise-free projections, the scaling parameter γ, and the noise variance σ^2 are nuisance parameters and should be integrated out. First, using a uniform prior for $\hat{\mathbf{p}}_2$, after some algebra the marginalisation yields

$$\begin{aligned}\int_{-\infty}^{\infty} L(D|\theta)\mathrm{d}\hat{\mathbf{p}}_2 &\propto \frac{\sigma^{-N}}{(1+\gamma^2)^{N/2}} \exp\left(-\frac{1}{2\sigma^2(1+\gamma^2)}\|\mathbf{p}_1 - \gamma\mathbf{p}_2\|^2\right) \\ &\propto \sigma_r^{-N} \exp\left(-\frac{1}{2\sigma_r^2}\|\mathbf{p}_1 - \gamma\mathbf{p}_2\|^2\right) \\ &= L(D|\theta'),\end{aligned} \tag{9}$$

which can still be interpreted as another likelihood function, since $\sigma_r^2 = \sigma^2(1+\gamma^2)$ is the variance of the residual $r = p_1 - \gamma p_2$, and $\theta' = (\theta_1, \theta_2, \alpha, \beta, \gamma, \sigma_r)$.

We assume a uniform prior for γ and the non-informative, Jeffrey's prior $p(\sigma_r) \propto \frac{1}{\sigma_r}$ for the residual deviation [9,10]. The marginal posterior finally is

$$\begin{aligned}p(\theta_1, \theta_2, \alpha, \beta|D) &\propto \iint L(D|\theta')p(\sigma_r)\mathrm{d}\gamma\mathrm{d}\sigma_r \\ &\propto \int_0^{\infty}\int_{-\infty}^{\infty} \sigma_r^{-N-1} \exp\left(-\frac{1}{2\sigma_r^2}\|\mathbf{p}_1 - \gamma\mathbf{p}_2\|^2\right)\mathrm{d}\gamma\mathrm{d}\sigma_r \\ &\propto \frac{1}{\|\mathbf{p}_2\|}\int_0^{\infty} \frac{1}{\sigma_r^N} \exp\left(-\frac{N}{2\sigma_r^2}\left(\bar{p_1^2} - \frac{(\mathbf{u}_2^{\mathrm{T}}\mathbf{p}_1)^2}{N}\right)\right)\mathrm{d}\sigma_r \\ &\propto \frac{\left(\bar{p_1^2} - \frac{1}{N}(\mathbf{u}_2^{\mathrm{T}}\mathbf{p}_1)^2\right)^{\frac{1-N}{2}}}{\|\mathbf{p}_2\|},\end{aligned} \tag{10}$$

where $\mathbf{u}_2$ is the vector $\mathbf{p}_2$ normalised to the unit length.

4 Estimation Algorithm

4.1 Sampling the Posterior

We propose to sample the posterior (10) by a hybrid method that combines Gibbs sampling with the Metropolis–Hastings method [11]. We have an outer

Algorithm 1. Gibbs–Metropolis–Hastings Sampler

Input: The projection images I and I', initial guess $\vartheta^{(0)} = (\theta_1^{(0)}, \theta_2^{(0)}, \alpha^{(0)}, \beta^{(0)})$ for the parameters.
Output: Effectively independent samples from the posterior $p(\theta_1, \theta_2, \alpha, \beta|D)$.

1. Set $n = 1$, $d = 1$.
2. Draw a sample ϑ'_d from $p_{\text{approx}}(\vartheta_d|\vartheta_1^{(n)}, \ldots, \vartheta_{d-1}^{(n)}, \vartheta_{d+1}^{(n-1)}, \ldots, \vartheta_4^{(n-1)})$.
3. Accept or reject ϑ'_d to be a sample from $p(\vartheta_d|\vartheta_1^{(n)}, \ldots, \vartheta_{d-1}^{(n)}, \vartheta_{d+1}^{(n)}, \ldots, \vartheta_4^{(n)})$ with the Metropolis–Hastings criterion (12). If accepted, set $\vartheta_d^{(n)} = \vartheta'_d$, otherwise go to Step 2.
4. Increment d by one. Repeat from Step 2 until all the four dimensions have been processed.
5. Increment n by one, set $d = 1$. Repeat from Step 2 until enough samples have been obtained.

Gibbs sampler that sequentially samples the one-dimensional conditionals while the conditional densities are sampled the Metropolis–Hastings method. Assuming bounded parameter space, the proposal distribution can be generated by discretising one-dimensional conditional density and generating a continuous approximation by spline interpolation.

Let $\vartheta = (\theta_1, \theta_2, \alpha, \beta)$. Looking at the d^{th} variable ϑ_d, the conditional is approximated as

$$\begin{aligned} &p_{\text{approx}}(\vartheta_d|\vartheta_1, \ldots, \vartheta_{d-1}, \vartheta_{d+1}, \ldots, \vartheta_4) \\ &= \frac{\sum_m g(\vartheta_d - \vartheta_d^m) p(\vartheta_d^m|\vartheta_1, \ldots, \vartheta_{d-1}, \vartheta_{d+1}, \ldots, \vartheta_4)}{\sum_m p(\vartheta_d^m|\vartheta_1, \ldots, \vartheta_{d-1}, \vartheta_{d+1}, \ldots, \vartheta_4)}, \end{aligned} \tag{11}$$

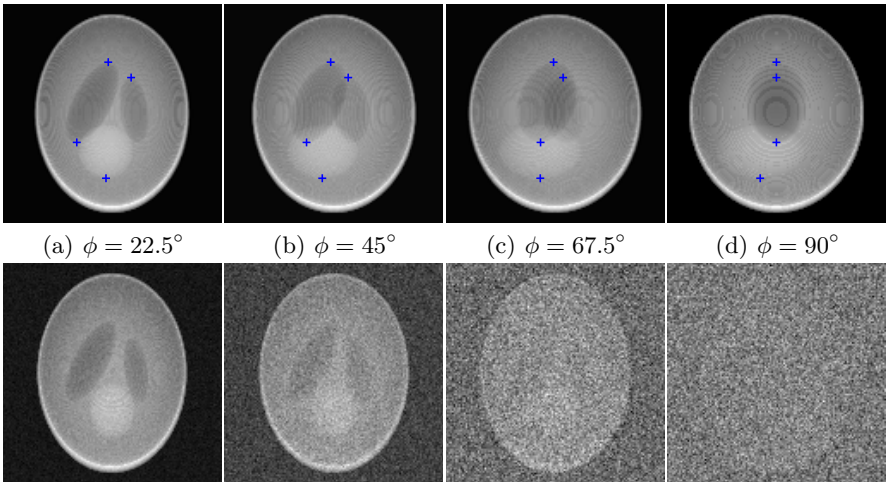
where $\vartheta_d^m = \vartheta_d^1 + (m-1)\Delta\vartheta_d$, $m = 1, \ldots, M$ are equidistant samples of the d^{th} variable, g is the interpolation kernel, where we use the zeroth order spline.

Sampling from the conditional distribution is now straightforward. First a sample is generated from the proposal density by drawing a sample $s \sim \text{Uniform}(0, 1)$, and finding the inverse of the conditional probability distribution $v'_d = P^{(-1)}_{\text{approx}}(s|\vartheta_1, \ldots, \vartheta_{d-1}, \vartheta_{d+1}, \ldots, \vartheta_4)$, which can be implemented by linearly interpolating the cumulative sum of the discretised conditional density. The proposed sample v'_d is accepted by the Metropolis–Hastings criterion or it is accepted with the probability

$$\begin{aligned} \min\Bigg(1, &\frac{p(\vartheta'_d|\vartheta_1, \ldots, \vartheta_{d-1}, \vartheta_{d+1}, \ldots, \vartheta_4)}{p(\vartheta_d|\vartheta_1, \ldots, \vartheta_{d-1}, \vartheta_{d+1}, \ldots, \vartheta_4)} \times \\ &\times \frac{p_{\text{approx}}(\vartheta_d|\vartheta_1, \ldots, \vartheta_{d-1}, \vartheta_{d+1}, \ldots, \vartheta_4)}{p_{\text{approx}}(\vartheta'_d|\vartheta_1, \ldots, \vartheta_{d-1}, \vartheta_{d+1}, \ldots, \vartheta_4)}\Bigg), \end{aligned} \tag{12}$$

otherwise it is rejected.

The method is summarised in Algorithm 1.



(a) $\phi = 22.5^\circ$ (b) $\phi = 45^\circ$ (c) $\phi = 67.5^\circ$ (d) $\phi = 90^\circ$

(e) $\sigma/\mu = 0.1$ (f) $\sigma/\mu = 0.4$ (g) $\sigma/\mu = 1.6$ (h) $\sigma/\mu = 6.5$

Fig. 2. (top) Shepp–Logan phantom rotated over the y-axis with 3D reference points (blue) superimposed; (bottom) Different levels of noise illustrated ($\phi = 0$)

4.2 Fundamental Matrix and Its Covariance

Having the samples from the posterior $(\theta_1^{(n)}, \theta_2^{(n)}, \alpha^{(n)}, \beta^{(n)})$, $n = 1, \ldots, N_{\text{samples}}$ we also have the affine fundamental matrix estimates, according to Theorem 2. We summarise the samples by estimating the *conditional mean* of the affine fundamental matrices over the posterior and the *posterior covariance* of the fundamental matrices. However, due to the homogeneous scaling ambiguity, we need to define in which sense the mean is computed.

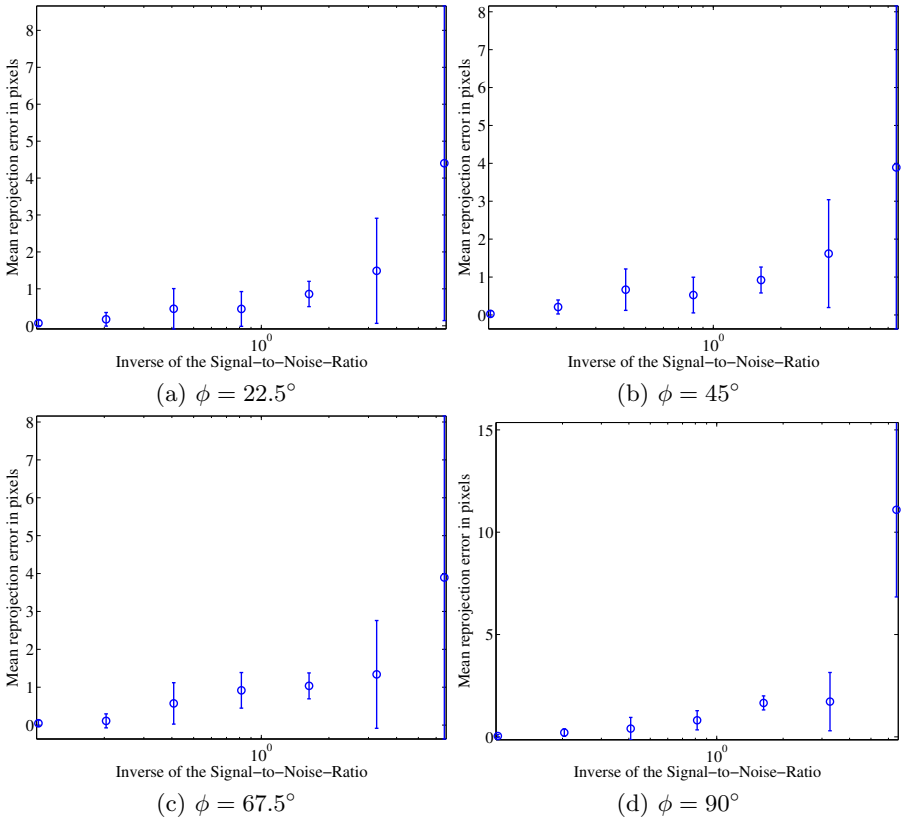
Without a loss of generality, the affine fundamental matrix estimate can be parameterised to be a point on the four dimensional unit sphere $\mathbb{S}^4$. Due to the construction, we can moreover assume that all samples are in the same half-sphere. Assuming uniform weighting, we use the standard Riemannian distance on $\mathbb{S}^4$. The mean affine fundamental matrix is defined as the estimate that minimises the intrinsic variance for the distance d on the sphere [12] or,

$$\hat{\mathbf{f}} = \operatorname*{argmin}_{\mathbf{f} \in \mathbb{S}^4} \sum_n d(\mathbf{f}^{(n)}, \mathbf{f})^2. \tag{13}$$

The posterior covariance is naturally estimated in the tangent space of the mean, so that

$$\hat{\mathbf{C}}_{\mathbf{f}} = \frac{1}{N_{\text{samples}}} \sum_n \log_{\hat{\mathbf{f}}} \mathbf{f}^{(n)} \left(\log_{\hat{\mathbf{f}}} \mathbf{f}^{(n)} \right)^{\mathrm{T}}, \tag{14}$$

where $\log_{\hat{\mathbf{f}}}$ is the Riemannian Log map [12] computed at the mean $\hat{\mathbf{f}}$.



(a) $\phi = 22.5°$ (b) $\phi = 45°$

(c) $\phi = 67.5°$ (d) $\phi = 90°$

Fig. 3. Reprojection error on the Shepp–Logan phantom in the function of the inverse SNR (σ/μ)

5 Experiments

We first experimented our method with the classic 3D Shepp–Logan ($128 \times 128 \times 128$), see Fig. 2. The left view was taken at the rotation $\phi = 0$ over the y-axis and the right view at $\phi = 22.5°, 45°, 67.5°, 90°$. For all the projections, i.i.d. Gaussian noise were added with inverse signal-to-noise-ratio (σ/μ, where μ is the mean intensity) from 0.1 to 6.5. The parameter bounds were set so that $\theta_1 \in [0, \pi]$, $\theta_2 \in [0, 2\pi)$, $\alpha \in [-1.5, -0.5]$, $\beta \in [-N/5, N/5]$. For each image pair and noise level, our method was initialised to the MAP estimate for θ_1 and θ_2 (no burn-in), conditioned on $\alpha = \alpha^{(0)}$ and $\beta = \beta^{(0)}$, whereas α and β were initialised to minus one and zero, respectively. We drew 100 samples and computed the mean affine fundamental matrix from (13) and evaluated the standard reprojection error [4] of the reference points superimposed in Fig. 2. The computations were repeated 7 times on each noise and angle combination, and the mean of the mean estimates with the standard deviations of the means are shown in Fig. 3. It can be seen that the method is very robust to noise, as the reprojection error

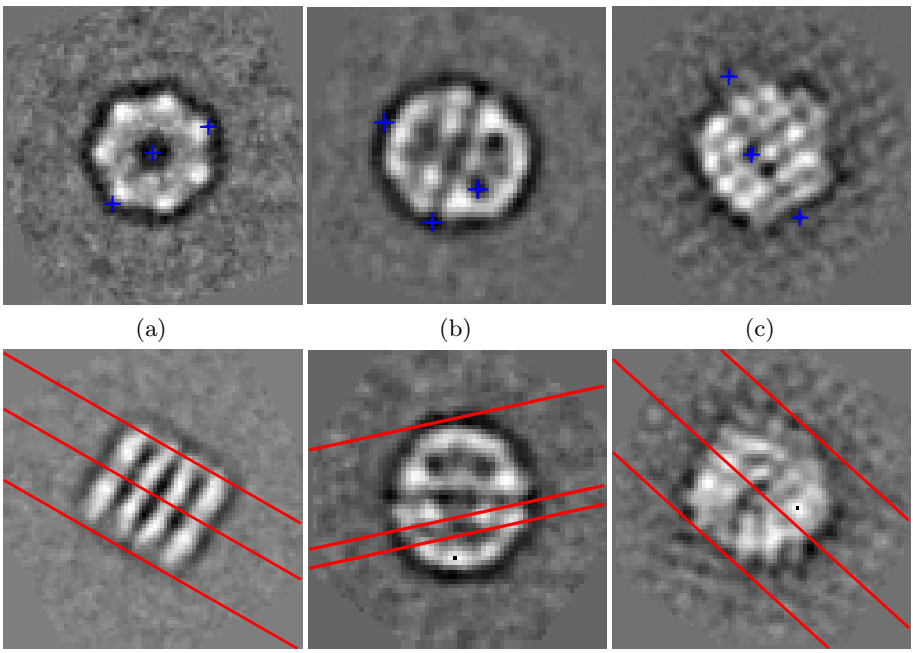


Fig. 4. Estimated Epipolar Geometry between the real class average images of cryo TEM projections: (a,d) GroEL, (b,e) Mm-Cpn particle, (c,f) Ribosome. The features pointed by the three points meet the corresponding points on the epipolar lines. Original images are available from http://blake.bcm.edu/emanwiki/Ws2011/Eman2.

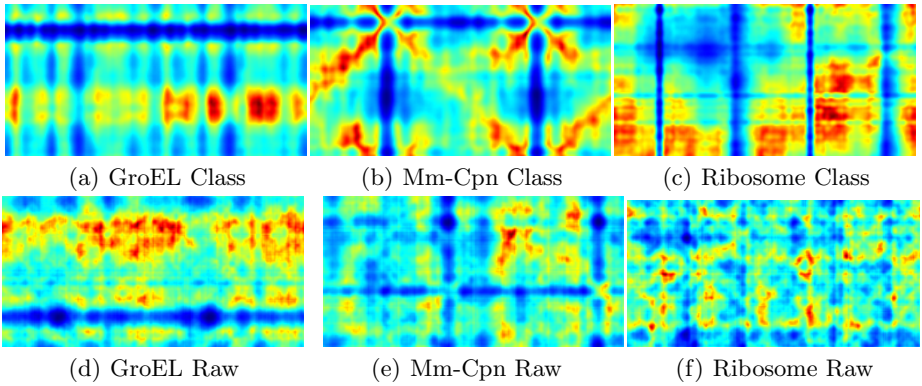


Fig. 5. Log posterior maps of (θ_1, θ_2) conditioned on the $\alpha = \alpha^{(0)}$, $\beta = \beta^{(0)}$, using (a-c) the class average image pairs, and (d-e) individual particle image pairs. The red colour indicates the highest probability, blue the lowest

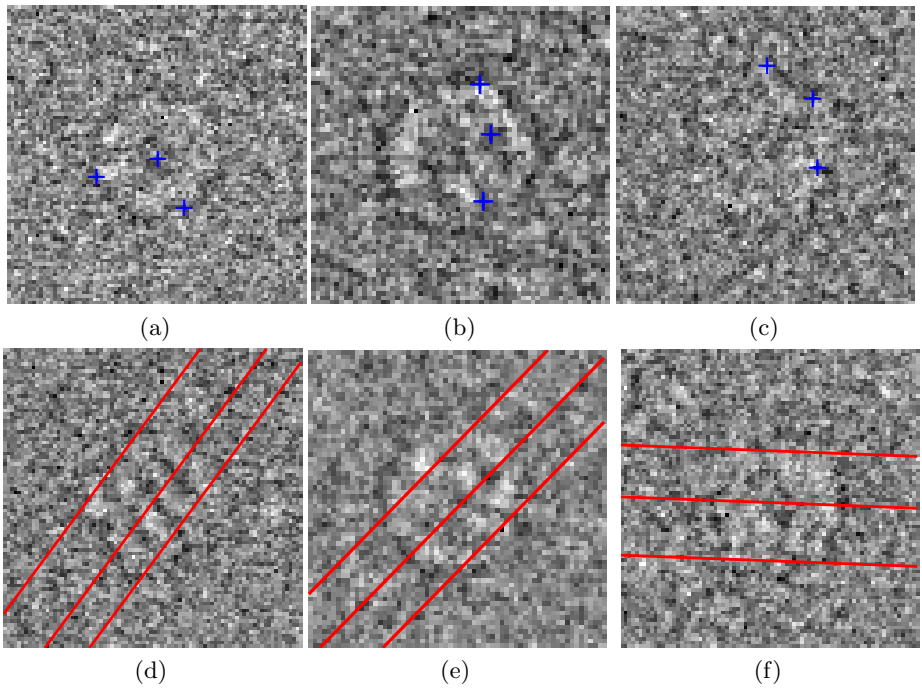


Fig. 6. Real particle projections by a cryo TEM with three points and the estimated epipolar lines; (a,d) GroEL; (b,e) Mm-Cpn particle; (c,f) Ribosome; original images available from http://blake.bcm.edu/emanwiki/Ws2011/Eman2

was mostly below 2 pixels, and it does not break down until the level of noise is about six times higher than the level of the signal.

In the the real data experiment, we experimented the CroEl, Mm-Cpn, and Ribosome particles taken by a cryo transmission electron microscope. The first experiments were made with class average images, i.e., the mean images over noisy sets of 53 to 418 projections that have been classified to indicate a similar projection view. The second set of real experiments were performed on the raw particle images. In the experiments, the parameter bounds and the initial guess were set similarly as above. Fig. 5 illustrates the log posterior of the parameters conditioned on the initial guess for α and β in all the six cases experimented.

We drew samples for the parameters, converted them to affine fundamental matrices, and computed their conditional mean as the result. Fig. 4 shows the results on class averages, where we have shown three points on one image and show the corresponding, estimated mean epipolar lines in the other image. In all the three cases the correct epipolar geometry has been found, which can be concluded from the known 3D structure and symmetry of the particles. Fig. 6 displays the results on the raw particle images. It can be seen that a reasonable estimate was found in all these cases, but the high noise level slightly deviates the direction of the epipolar lines with the GroEl particle from the ground truth.

6 Conclusions

In this paper we have shown the equivalence between uncalibrated common line geometry and the affine epipolar geometry. In contrast to the traditional methods, no point correspondences are required to estimate the epipolar geometry. Using the common-line geometry and Gaussian noise model, we then derived the marginal posterior distribution for the affine epipolar geometry using certain non-informative priors for the nuisance parameters. We additionally proposed a hybrid Gibbs-Metropolis-Hastings sampler to sample the posterior distribution. We summarised the samples by the conditional mean on the manifold of the affine fundamental matrices. Our experiments on both synthetic and real data show that our approach is successful in recovering the affine geometry of tomographic parallel projections with a very low signal to noise ratio.

References

1. Crowther, R., Amos, L., Finch, J., De Rosier, D., Klug, A.: Three dimensional reconstructions of spherical viruses by fourier synthesis from electron micrographs. Nature 226, 421–425 (1970)
2. Kak, A.C., Slaney, M.: Principles of Computerized Tomographic Imaging. IEEE Press (1988)
3. Brandt, S.S.: Markerless alignment in electron tomography. In: Frank, J. (ed.) Electron Tomography. Springer (2006)
4. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge (2000)
5. Brandt, S., Heikkonen, J.: Optimal method for the affine F-matrix and its uncertainty estimation in the sense of both noise and outliers. In: Proc. ICCV, vol. 2, pp. 166–173 (2001)
6. Brandt, S.S., Kolehmainen, V.: Structure-from-motion without correspondence from tomographic projections by bayesian inversion theory. IEEE Trans. Med. Imaging 26, 238–248 (2007)
7. Frank, J.: Three-Dimensional Electron Microscopy of Macromolecular Assemblies. Oxford (2006)
8. Brandt, S.S., Ziese, U.: Automatic TEM image alignment by trifocal geometry. Journal of Microscopy 222, 1–14 (2006)
9. Bretthorst, G.L.: Bayesian Spectrum Analysis and Parameter Estimation. Springer (1988)
10. Brandt, S.S., Palander, K.: A Bayesian Approach for Affine Auto-calibration. In: Kalviainen, H., Parkkinen, J., Kaarna, A. (eds.) SCIA 2005. LNCS, vol. 3540, pp. 577–587. Springer, Heidelberg (2005)
11. MacKay, D.: Information Theory, Inference and Learning Algorithms. Cambridge (2003)
12. Pennec, X.: Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements. JMIV 25, 127–154 (2006)

On the Global Self-calibration of Central Cameras Using Two Infinitesimal Rotations

Ferran Espuny

Université Paris Descartes, Laboratoire MAP5 (CNRS UMR 8145)
Ferran.Espuny@parisdescartes.fr

Abstract. The calibration of a generic central camera can be described non-parametrically by a map assigning to each image pixel a 3D projection ray. We address the determination of this map and the motion of a camera that performs two infinitesimal rotations about linearly independent axes. A complex closed-form solution exists, which in practice allows to visually identify the geometry of a range of sensors, but it only works at the center of the image domain and not accurately.

We present a new two-step method to solve the stated self-calibration problem that overcomes these drawbacks. Firstly, the Gram matrix of the camera rotation velocities is estimated jointly with the Lie bracket of the two rotational flows computed from the data images. Secondly, the knowledge that such Lie bracket is also a rotational flow is exploited to provide a solution for the calibration map which is defined on the whole image domain. Both steps are essentially linear, being robust to the noise inherent to the computation of optical flow from images.

The accuracy of the proposed method is quantitatively demonstrated for different noise levels, rotation pairs, and imaging geometries. Several applications are exemplified, and possible extensions and improvements are also considered.

1 Introduction

There exists a wide variety of central cameras, not limited to systems with multiple lenses (dioptric), but also comprising combinations of lenses and mirrors (catadioptric) [1]. Rather than using a specific parametric model for each central system, we consider the non-parametric generic camera model, which calibration map associates to each image pixel a projection ray [2,3,4].

Concisely, we address in this paper the self-calibration of this model (estimation of both camera motion and calibration map) using only images acquired with two infinitesimal rotations of the camera. The considered problem is closely related to that of recovering a mirror's shape from rotational specular flows [5]. However, rotational specular flows offer additional cues for self-calibration [6].

In the case of central cameras, the generic calibration problem (finding a projection direction for each pixel) is equivalent to non-parametric distortion correction for those cameras with a field of view smaller than 180 degrees; for this equivalence, it suffices to place an auxiliary plane in front of the camera. Other

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 243–256, 2013.

Fig. 1. Left: example image acquired by a central camera with high radial distortion, and superimposed the optical flows computed using two small camera rotations. Centre: undistortion obtained with the state-of-the-art method in [18]. Right: global undistortion obtained with our non-parametric linear method; its average global error is 2 pixels, from which only 0.5 pixels correspond to non-perspective distortion.

applications of camera calibration include motion estimation and mosaicing/3D reconstruction from images. Examples are shown through the paper.

Several non-parametric solutions exist using a calibration pattern for (generic) camera calibration [2,4] and distortion correction [7,8]. The self-calibration of non-parametric distortion models, limited to be radial, has been addressed in [7,9,10,11]. In contrast, we do not assume any structure on the scene geometry, and only require the generic calibration map to be smooth.

The metric self-calibration (i.e. up to rotation) of a generic central camera has been solved, at least at a proof-of-concept level, for particular (non-infinitesimal) motions [12,13,14]. However, these methods estimate *motion flows* from image matches and thus the camera relative rotations can not be general. In contrast, we use infinitesimal camera rotations with axes being allowed to be general.

Another generic self-calibration method exists that requires a large image dataset to (inaccurately) estimate a discrete generic calibration map, assumed to be correlated with dissimilarity measures in the input images [15]. In contrast, we only require two rotational image sequences to solve a problem equivalent to theirs in the case of smooth generic central cameras.

In [16], the optical flows produced by three infinitesimal camera rotations are used for the self-calibration, up to projective transformation, of a smooth generic central camera. In [17], two rotations are used for the metric self-calibration of said camera model, the motion estimation depending on second order derivatives of the data flows, and the calibration map estimation depending rationally on the flow coordinates. A re-formulation of [17] using the Lie bracket of the flows is given in [18]; results corresponding to "real" data flows are shown, which are neither accurate nor defined near the image borders.

The computation of rotational flows compatible with the existence of a common camera is outlined in [18] as a potential improvement of [17,18]; however, it is cast as "highly non-linear optimization problems", which remain unsolved. It is also shown in [18] that three different rotations may be used to achieve a global result, more regular, but still inaccurate, even if using exact motions.

We present a new step-wise linear method for the global self-calibration of a smooth generic central camera from two rotational flows (see Fig. 1). We show that the camera motion can be accurately determined by its joint estimation with the Lie bracket of the two data flows, which in fact corresponds to a third unobserved rotational flow. This first step represents a workaround solution to the computation of compatible rotational flows pursued in [18]; moreover, it requires the use of derivatives of the data flows only up to first order.

We also show that the Lie bracket flow obtained during the motion estimation step allows an accurate global estimation of the calibration map. This second computation overcomes the division by the data flow coordinates associated to the closed-form formulae in [17,18]: the calibration map can be linearly determined thanks to the previous estimation of the Lie bracket flow.

Next, we introduce the necessary notation and background, concisely stating the problem; in Section 3, we overview and analize the existing closed-form solution. We present our two-step method in Section 4. An experimental evaluation and analyisis of our proposal is performed in Section 5 using simulated image sequences, and an example with real images is also shown before the Conclusion.

Notation. In order to allow an easy comparison with [18], we adopt their notation conventions. The symbol ∂ will be used for differentiation, a cross symbol $\times$ will denote the cross product operator, and Id_2 the identity matrix of size 2.

2 A Generic Self-calibration Problem

2.1 Preliminaries

Following [2,3,4] we consider a *generic camera* to be a set of image points in (possibly non-parametric) correspondence with a set of 3D projection rays. We say that a generic camera is *central* if all its projection rays intersect in a single point, called the *camera centre* [2,3,4]. In this case, we can use the unit sphere S^2 to describe the possible projection rays. Accordingly, we define the *calibration map* of a generic central camera as a map from $\mathcal{U}$, an open connected subset of $\mathbb{R}^2$ (image pixels), on the unit sphere S^2 (oriented projection directions):

$$\begin{array}{rcl} f : \mathcal{U} \subset \mathbb{R}^2 \rightarrow & & S^2 \\ (u, v) & \mapsto & f(u, v) \; . \end{array} \tag{1}$$

This map sends a planar image to its undistorted version on the sphere. In order to use optical flow, we assume that the calibration map f is smooth [16,17,18].

In the following, we will use f for theoretical demonstration purposes, with no further constraint on it that having norm one. In contrast, we will only show results corresponding to calibration maps with $f_3 > 0$, meaning in practice that the camera angular field of view is smaller than 180 degrees (for omnidirectional cameras, the visualization of a calibration map is not straightforward). For this purpose, we introduce the *undistortion map*

$$g = (g_1,\ g_2)^T := (f_1/f_3,\ f_2/f_3)^T \; . \tag{2}$$

In Fig. 1 we show an example of image with high radial distortion, together with its undistorted version using the map g. By the smoothness assumption on the calibration map f, the undistortion map g is also smooth, which we will impose using b-splines (see the Appendix).

We consider the *optical flow* in a sequence of images to be the velocity field in the image domain tangent to the image transformation that takes one image into the next one (not the transformation itself).

2.2 Problem Statement

Assume that we know two point-wise linearly independent optical flows V_1, V_2 observed on an open subset $\mathcal{U}$ of the image, corresponding to rotations of a generic central camera about two linearly independent axes passing through the camera centre. The self-calibration problem consists in determining the camera rotational velocities ω_1, ω_2 and the calibration map f that are compatible with these flows, i.e. satisfying the following equations [3,18]:

$$\mathrm{D}f(u,v) \cdot V_i(u,v) = -\omega_i \times f(u,v) \; . \tag{3}$$

It is proven in [17] that this problem can be solved up to an orthogonal transformation. Accordingly, we assume as given an orthonormal basis $\{u_1, u_2\}$ so that

$$(\omega_1, \omega_2) = (u_1, u_2) \begin{pmatrix} a & b \\ 0 & c \end{pmatrix} \; , \tag{4}$$

for some unknown scalars a, b, c satisfying $a, c > 0$. Observe that this introduces an asymmetry in the problem, since the direction of one of the axis of rotation is known, whereas the other axis is only constrained to lie on a known two-dimensional semi-space. A symmetric but less general approach, not followed here, would be to consider as given the two directions of rotation.

3 Existing Closed-Form Solution and Analysis

The generic self-calibration of a central camera from two rotational flows is solved in closed-form in [17,18], using two auxiliary functions given by:

$$\Delta_1 = -\mathrm{tr}(\mathrm{D}V_2) + \frac{1}{\det V} \mathrm{D} \det V \cdot V_2 \; , \tag{5}$$

$$\Delta_2 = \quad \mathrm{tr}(\mathrm{D}V_1) - \frac{1}{\det V} \mathrm{D} \det V \cdot V_1 \; . \tag{6}$$

Concisely, the Gram matrix

$$G_\omega := (\omega_1, \omega_2)^T \cdot (\omega_1, \omega_2) \tag{7}$$

is determined by averaging the following pixel-wise estimators, taking into account the expected positive definiteness of G_ω:

$$G_\omega = \begin{pmatrix} \Delta_2 \\ -\Delta_1 \end{pmatrix} \cdot (-\Delta_2, \Delta_1) + \begin{pmatrix} \mathrm{D}\Delta_2 \\ -\mathrm{D}\Delta_1 \end{pmatrix} \cdot V \; . \tag{8}$$

Then, the camera motion is extracted from this Gram matrix using (4) and (7), and finally the calibration map f is computed as the norm-one map such that

$$f \propto \left(\omega_1, \omega_2, \omega_1 \times \omega_2\right) \cdot \left(\Delta_1, \Delta_2, 1\right)^T \ . \tag{9}$$

Remark 1 (Analysis). By (9), the relation between the undistortion map g, defined by (2), and the functions Δ_1, Δ_2, defined by (5–6), is rational except when ω_1 and ω_2 are orthogonal to the Z axis. For instance, for $\omega_1 = (0, 1, 0)^T$ and $\omega_2 = (0, 0, 1)^T$, we have that $g_1 = 1/\Delta_2$, $g_2 = \Delta_1/\Delta_2$; therefore, any error in the estimation of Δ_2 affects severely the estimation of the undistortion map (g_1, g_2). In addition, the formulae (5–6) require the first order derivatives of the data flows, which are later differentiated in (8) to estimate the camera motion. As a result, the method in [17,18] has trouble to determine the solution close to the image borders in presence of noise or real flows.

It is shown in [18] that the functions (Δ_1, Δ_2) defined by (5–6) are in fact (with reversed sign) the coordinates in the point-wise vector basis V_1, V_2 of the Lie bracket vector field:

$$[V_1, V_2] := \mathrm{D}V_2 \cdot V_1 - \mathrm{D}V_1 \cdot V_2 \ . \tag{10}$$

The geometric interpretation of this Lie bracket is also given: it is the optical flow of a (non-performed) rotation with angular velocity $\omega_1 \times \omega_2$. As a consequence, Eq. (8) can be written as:

$$G_\omega = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \cdot (V_1, V_2)^{-1} \cdot \left([V_1, [V_1, V_2]] \ \ [V_2, [V_1, V_2]]\right) \ . \tag{11}$$

In the next section, we exploit these theoretical formulae from [18] to propose a new global self-calibration method.

4 A Two-Step (Linear) Method

Assume that we are given two optical flows V_1, V_2 as described in Section 2.2. These flows can be computed linearly from two initial image sequences by minimizing the Linearized Brightness Constancy Constraint [20], which we do in practice by following [18] with a b-spline model. We denote the coefficients of the Gram matrix G_ω in (7) as $G_{i,j} := \omega_i^T \cdot \omega_j$.

4.1 Estimation of the Camera Angular Velocities and the Lie Bracket Flow

Due to noise in V_1, V_2, a direct computation using (10) of the Lie bracket flow

$$V_3 = [V_1, V_2] \tag{12}$$

is not likely to be compatible with the desired existence of a constant positive definite 2×2 matrix G_ω satisfying (11). Since the optical flows V_1, V_2 are assumed to be known, such constraint can be written as a differential linear combination of V_3 and $G_{i,j}$:

$$[V_1, V_3] + G_{1,1}V_2 - G_{1,2}V_1 = 0\,, \tag{13}$$
$$[V_2, V_3] + G_{1,2}V_2 - G_{2,2}V_1 = 0\,. \tag{14}$$

In summary, denoting by $V_i^T \cdot \nabla$ the operator $V_{i,1}\partial_u + V_{i,2}\partial_v$, the flow V_3 and the coefficients $G_{i,j}$ of the Gram matrix satisfy the following 3 vector equations (i.e. 6 scalar equations) per pixel:

$$\begin{pmatrix} \mathrm{Id}_2 & 0 & 0 & 0 \\ \mathrm{Id}_2\left(V_1^T \cdot \nabla\right) - \mathrm{D}V_1 & V_2 & -V_1 & 0 \\ \mathrm{Id}_2\left(V_2^T \cdot \nabla\right) - \mathrm{D}V_2 & 0 & V_2 & -V_1 \end{pmatrix} \cdot \begin{pmatrix} V_3 \\ G_{1,1} \\ G_{1,2} \\ G_{2,2} \end{pmatrix} = \begin{pmatrix} [V_1, V_2] \\ 0 \\ 0 \end{pmatrix}\,. \tag{15}$$

In practice, we model the smooth optical flow V_3 using b-splines, as explained in the Appendix. The resulting sparse linear system has $2N + 3$ unknowns, being N the size of the b-spline coefficient vector. If we do not impose the positive definiteness on G_ω, it can be solved either using least squares or a more robust procedure, as detailed in Appendix. Otherwise, the corresponding constrained problems can be solved using Second Order Cone Programming (SOCP) [19]. Given that the order of the Lie bracket coefficients can be quite dissimilar from that of the Gram matrix coefficients, we use an initial estimation given by the L_2 optimization to normalize the equations and then re-estimate the parameters.

Remark 2 (Potential Use for Parametric Camera Rotational Flows). Equation (15) does only involve the optical flows and the camera motion: it is independent of the model used to describe a central camera. Therefore, it can be used in a (possibly uncalibrated) parametric context as a constraint on rotational flows and/or on the motion of a rotating camera using those flows for computation and/or evaluation purposes.

Remark 3 (Computing Optical Flows Compatible with the Rotation of a Common Camera). A further improvement of the previous motion estimation method could be achieved by the joint estimation, directly from the rotational image sequences, of the two rotational flows, their Lie bracket and the Gram matrix of the camera motions. The difficulty in such problem lies in the need for using a robust penalization for the non-linear terms arising from (13),(14) when considered as a function of V_1, V_2, V_3, $G_{i,j}$, and shall be a topic of future research (experiments using L_2 penalization were not successful).

4.2 Linear Estimation of the Global Undistortion Map

After the previous step, we may assume as known both the rotational flows V_1, V_2, their Lie bracket V_3, and the corresponding camera angular velocities ω_1,

ω_2, $\omega_1 \times \omega_2$, which can be determined with the Cholesky decomposition of the Gram matrix G_ω using (7).

Observe that now the joint matrix of the three available rotational flows, (V_1, V_2, V_3), has point-wise rank two (except at most in two isolated image pixels). Therefore, we can compute the undistortion map g linearly by adapting the 3-flow methods in [16,18]. Concisely, using that, by (4), we have

$$(\omega_1, \omega_2, \omega_1 \times \omega_2)^{-1} = \frac{1}{ac} \begin{pmatrix} cu_1^T - bu_2^T \\ au_2^T \\ {u_1 \times u_2}^T \end{pmatrix} , \tag{16}$$

the undistortion map (g_1, g_2) satisfies the following linear constraint:

$$\big(V_1, V_2, V_3\big) \cdot \begin{pmatrix} cu_1^T - bu_2^T \\ au_2^T \\ u_1 \times u_2^T \end{pmatrix} \cdot \begin{pmatrix} g_1 \\ g_2 \\ 1 \end{pmatrix} \;=\; 0 \,. \tag{17}$$

In practice, we model the undistortion map g using b-splines, as explained in the Appendix, and we solve the resulting sparse linear system with either L_2 or robust $L_{1\varepsilon}$ penalization, as detailed in that section. Observe that the result (17) still holds true if we take a general calibration map f instead of $(g_1, g_2, 1)^T$, and therefore it may be used for omnidirectional cameras.

4.3 Summary

We conclude this section by summarizing the proposed self-calibration method, leaving clear the used parameters and its computational cost. In contrast with the existing closed-form method [17,18], not only Step 1 but all the computations required by the algorithm are step-wise linear, excepting a Cholesky decomposition of the Gram matrix G_ω in Step 3, and a positive constraint on G_ω and a quadratic formula for $[V_1, V_2]$ in Step 2. Moreover, with respect to that early method, the use of flow derivatives has been reduced to first order, and these are only needed for computing the Lie bracket flow $V_3 = [V_1, V_2]$ in Step 2.

Algorithm 1. Self-calibration from Two Infinitesimal Rotations

Input. Two sequences of rotational images; two orthonormal 3-vectors u_1, u_2
1. Compute the generic rotational flows V_1, V_2 from the images
2. Compute $V_3 = [V_1, V_2]$ and G_ω by solving (15)
3. Extract the rotation angular velocites ω_i from G_ω and the u_i using (4)
4. Compute g by solving (17)

return ω_1, ω_2, g

Remark 4 (Number of Parameters). The proposed algorithm only requires as parameters two orthonormal vectors to avoid the orthogonal ambiguity in the self-calibration problem (Section 2.2). In practice, we use 2D b-splines in several estimations: the optical flows V_1, V_2 in Step 1, their Lie bracket V_3 in Step 2,

and the undistortion map g in Step 4. Since we assume equi-distributed knots (Appendix), we need three parameters for each b-spline: the number of knots in each direction, n_1, n_2, and a smoothing factor λ. We only used the latter when computing the optical flows in Step 1, where an extra parameter σ is also needed for pre-smoothing the images.

Remark 5 (Computational Cost). The algorithm requires the resolution of linear systems of different sizes: $n \times 2N$ in Step 1, $6n \times (2N+3)$ in Step 2, and $2n \times 2N$ in Step 4, being n the number of image pixels and N the length of each b-spline coefficient vector. We solve both the L_2 minimization (18) and the iteratively re-weighted $L_{1\varepsilon}$ minimization (19) through the computation of the corresponding normal equations and a Cholesky-based resolution of those equations. When solving (15), we only impose the positive definiteness on the Gram matrix G_ω and solve the constrained least squares problem if the unconstrained linear methods fail to find a positive definite solution; in practice, this will likely happen with either high noise or with optical flows (close to) being linearly dependent.

5 Experimental Results

5.1 Error Measures

For simulated image sequences, we will measure the errors in optical flow (using angular and relative norm errors) and the camera angular velocities (using relative metric errors) as in [17,18]. In addition, we introduce two measures for the evaluation of an estimated undistortion map g when its groundtruth $\hat{g}$ is known. First, the absolute global error GE (in pixels), defined pixel-wise by $GE = \text{dist}(g, \hat{g})$. This serves as an overall error, since the calibration results are coupled with the motion estimation errors, as it follows from formula (17).

We use as second error measure of an estimated g the error after correcting it with the "best" homography H approaching g to $\hat{g}$, H being computed with the DLT algorithm. We refer to this as the absolute non-perspective error, NPE (in pixels), defined at each pixel as $NPE = \text{dist}(H \cdot g, \hat{g})$. The motivation for this choice is that the correctly undistorted images should be perspective-like images, for which the infinitesimal camera rotations induce image homographies.

5.2 An Example with High Radial Distortion

We generated sequences of 500×500 images corresponding to rotations about the Y and Z axes, with angular velocities of norm equal to 0.003, of a camera with high radial distortion (Fig. 1 left contains an example image). To avoid "maquillaging" the solution, we fixed beforehand the b-spline parameters $n_1 = n_2 = 5$ and $\lambda = 0$ in (19) for the computation of the Lie bracket V_3 or the undistortion map g.

We computed the optical flows V_1, V_2 corresponding to only the first two images of each sequence by minimizing the $L_{1\varepsilon}$ penalization of the Linearized Brightness Constancy Constraint [20] with $\sigma = 3.0$ for image pre-smoothing, and

respective thin-plate regularity weights $\lambda_{V_1} = 10^6$, $\lambda_{V_2} = 10^{10}$ (we refer to the Appendix for this parameter). The groundtruth flows are depicted in Fig. 1; it can be observed that the flow V_1 (horizontal green arrows) has a very changing scale according to the different resolutions in the image. Due to this fact, in general the selection of a constant smoothing parameter σ is not optimal, and we expect bigger errors, specially in the derivatives, for high scale changing flows. As we show later, image smoothing can be improved after calibration information is available. The results are summarized in Table 1 (left). We include the errors for the flow derivatives because they are used for the next motion estimation step, and they turn out to be much higher than the flow errors for V_1.

Table 1. Left: Average and standard deviation of the angular error AE (in degrees) and relative norm error RNE (%) of the data optical flows and derivatives. Right: relative metric errors (%) in the estimated camera angular velocities, and errors in the estimated Lie bracket flow, both being used in the second self-calibration step.

Flow	μ_{AE}	σ_{AE}	μ_{RNE}	σ_{RNE}
V_1	0.18	0.19	0.90	0.59
$\partial_u V_1$	3.19	3.49	3.29	3.47
$\partial_v V_1$	2.62	2.57	1.50	1.50
V_2	0.13	0.21	0.25	0.31
$\partial_u V_2$	0.08	0.04	0.04	0.03
$\partial_v V_2$	0.06	0.03	0.08	0.06

	Motion Errors			**Lie Bracket Errors**			
Method	$\|\omega_1\|$	$\|\omega_2\|$	$\widehat{\omega_1,\omega_2}$	μ_{AE}	σ_{AE}	μ_{RNE}	σ_{RNE}
Esp07 [17]	4.13	0.14	1.86	1.28	1.91	2.11	2.88
E_{L_2}	2.30	0.39	1.82	0.53	0.54	1.78	2.49
$E_{L_{1\varepsilon}}$	**0.39**	**0.02**	**0.54**	**0.21**	**0.20**	**0.83**	**0.46**

Given the two rotational flows, we computed their Lie bracket and the two camera angular velocities using the three available methods. First, the closed-form formulae and the averaging process in [17,18]; second (resp. third), their joint optimization as described in Section 4.1 with a L_2 (resp. $L_{1\varepsilon}$) penalization for the resulting linear system (15). We see from the results, summarized in Table 1 (right), that the $L_{1\varepsilon}$ method overperforms the other ones: it estimates the motion with far below a 1% of relative error.

Fig. 2. Left: groundtruth sensor (red lines) and the estimated undistortion map (dashed blue lines). Centre: original image smoothed using the metric induced by this map. Right: mosaic performed with the Z rotational sequence of calibrated images.

We finally computed the calibration map using the $L_{1\varepsilon}$ penalization, and obtained results as observed in Fig. 1: the global error had mean and standard deviation values $\mu_{GE} = 2.548, \sigma_{GE} = 0.758$ (in pixels), whereas the non-perspective correction error was $\mu_{NPE} = 0.568, \sigma_{NPE} = 0.595$ (in pixels), the error measures being computed as explained at the beginning of this section. It can be observed (better zooming in) at Fig. 2 left that, as typical in methods for the correction of geometric distortion, the calibration errors are mostly concentrated at the image corners. In comparison with the state of the art in [18], ours is a more accurate and globally defined solution.

The achieved accuracy allows us to perform typical operations with calibrated cameras. A first example of application consists in smoothing the original images taking into account the sensor geometry, i.e. non-uniformly (Fig. 2, centre). This is done by solving the heat-diffusion equation [21], which we perform via an iterative evaluation of the Laplace-Beltrami operator and an update of the smoothed image, as for the parametric spherical camera model in [22].

A second example application, using the obtained undistorted images and motion estimation, consists in creating a mosaic by reversing the effect of the estimated rotations to place all such images on a common frame. In Fig. 2 right we show a 560×560 mosaic generated in this way using a sequence of 50 images corresponding to a camera rotation about the Z axis, the first two images being the ones used for camera calibration.

5.3 Further Evaluation

It seems natural to ask what would have been the ouput in the previous experiment if the noise in the flows had been different from the obtained with the given images. Moreover, we may wonder whether the two measured flow errors, namely the angular and relative norm error, affect the results in equal measure or not; we study these two factors as possible main error sources, although, as already pointed out, our results depend not only on the optical flow components but also on their derivatives.

In Fig. 3 we show the average result errors after performing 50 simulations of noise in the two data flows for each combination of angular and relative norm errors between 0.2 and 1. We fitted b-splines to the noisy data, again without tuning the involved parameters, being the actual average noise in the fitted b-spline flows possibly higher than reported. This simulation must be considered carefully, taking into account that two rotational flows in general do not have the same angular and norm noise levels; moreover, it is mostly tentative, since other factors affect the calibration. In all the tested cases, we outperformed [18] and obtained the worst results for bigger norm errors in the data flows, being the motion estimation also affected, in smaller measure, by the angular errors.

We performed a last evaluation experiment by fixing the Y axis and varying the second rotation axis (Fig. 4). We obtained the best results for the axes orthogonal to the Y axis. The results are again sensor-dependent (for the used radial distortion sensor, the Z axis optical flow is quite easy to model), and could be improved by using the smoothness parameters that we set to zero.

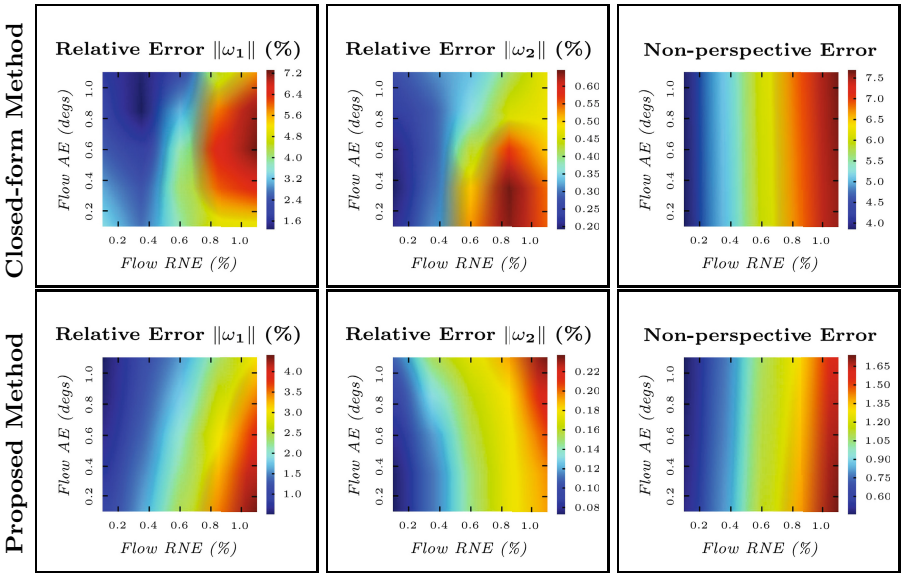


Fig. 3. Errors corresponding to the method in [18] (top row) and ours (bottom row) for the Y and Z rotations with different levels of relative norm and angular error in the two data optical flows. We omitted: the errors in $\widehat{\omega_1, \omega_2}$, which were similar to those for $\|\omega_2\|$, and the global errors, which were mostly affected by the bigger errors in $\|\omega_1\|$.

5.4 A Highly Non-linear Sensor

In Fig. 5, we consider a sinusoidal sensor inspired by [16,18], which is maybe not very realistic, but it shows the power of our non-parametric method to self-calibrate highly non-linear sensors. Taking only as input two sequences of 10 images (Y and Z axis rotations), our method estimates the camera motion with relative metric errors below 0.5%. The errors, explained at the beginning of this section, without imposing regularity are quite good: $\mu GE = 1.24$, $\mu NPE = 0.82$.

5.5 Real Images

We finally consider the fish-eye image sequences from Section 8.2 in [18].[1]The image sequences are particularly cumbersome for optical flow computation, due to the lack of smoothness or texture in certain image regions; our results, shown in Fig. 6, are clearly worst in those regions. The mean error between the corners of the undistorted checkerboard pattern and the projection of an ideal pattern under a best least-squares fitting homography is the 17.1% of the average length of an undistorted square side (standard deviation equal to 9.8). A further bundle adjustment process is currently under study. The interested reader may compare our results with those in Fig. 10 in [18] to assess the improvement in the area of image that we calibrate (improvement already outlined in Fig. 1).

[1] Images available at
`http://atlas.mat.ub.es/personals/fespuny/Research.html#CGRF`

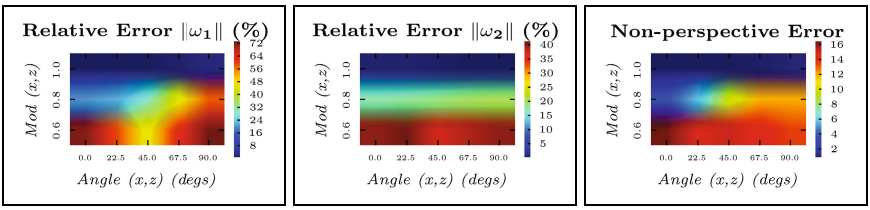


Fig. 4. Average errors obtained with AE=0.15 degs, RNE=0.5% in two flows corresponding to the Y axis and different linearly independent second axes; we use polar representation in the (x,z) coordinates of the second axis, e.g. angle 0 and norm 1 means the X axis, and angle 90 and norm 0.6 represents the axis of direction $(0, 0.8, 0.6)$.

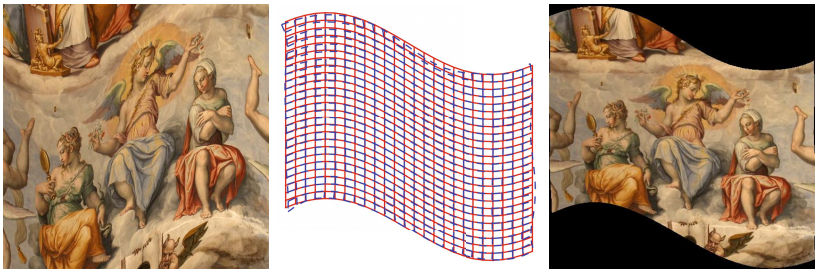


Fig. 5. Left: example image with sinusoidal distortion. Centre: estimated undistortion map (dashed blue lines). Right: undistorted image using b-spline regularity ($\lambda = 10^3$).

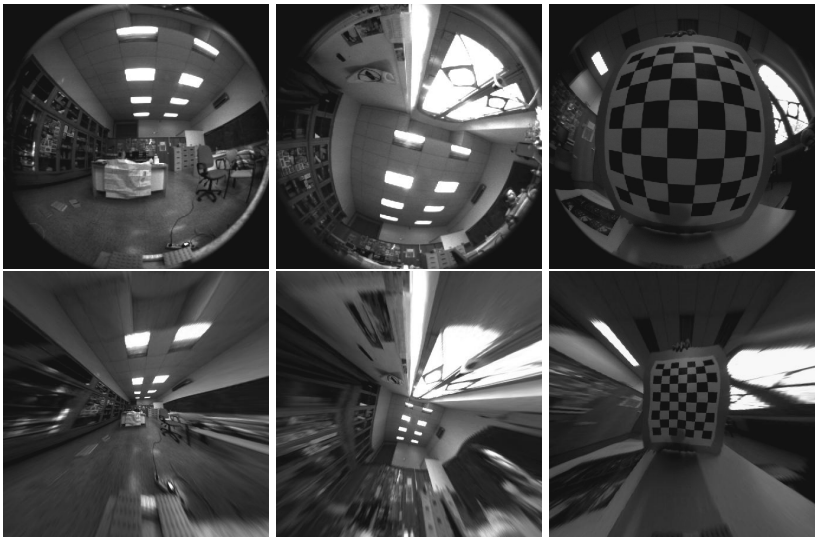


Fig. 6. Top: original images from [18]. Bottom: global undistortion results.

6 Conclusion

We have presented a new method for the global self-calibration of central cameras using the optical flows produced by two infinitesimal camera rotations. Despite a positive definiteness constraint on the Gram matrix of the rotation angular velocities, all the involved steps are linear, being the method quite robust and accurate, specially for orthogonal rotation axes. As discussed in the paper, the results can be applied both in parametric and non-parametric settings.

In fact, we have shown that any two optical flows covering an image region give enough information for self-calibrating that region. Therefore, when having more than two rotations, the resulting optical flows are no longer required to be dense for self-calibrating the camera, as far as they locally overlap pair-wise on the whole image. In conclusion, the simplicity of our proposal may conduct to a highly dynamical and accurate non-parametric method for the self-calibration of central cameras with multiple infinitesimal rotations.

Acknowledgement. The author is grateful to José I. Burgos Gil, Lionel Moisan, and Pascal Monasse for valuable discussions. This work was partly supported by MAP5 (CNRS UMR 8145) and the projects MTM2009-14163-C02-01 (Spain), Callisto ANR-09-CORD-003 (France), and IMAGINE-ENPC (France).

Appendix. Using B-Splines for Smooth 2D Functions

We are interested in the estimation of several two-dimensional smooth functions $F = F(u, v) = (F_1, F_2)^T$ defined on the image, e.g. the undistortion map in (2). We will model each component F_i as a tensor b-spline surface with a grid of $n_1 \times n_2$ equi-distributed knots (see e.g. [23]): $F_i(u, v) = w(u, v)^T \cdot k^i$, where $w(u, v)$ is an N-dimensional vector of weights, $N = (n_1+3)(n_2+3)$, and k^i is an N-dimensional vector of coefficients corresponding to F_i, $i = 1, 2$. The b-spline regularity can be imposed with a discrete version of its thin-plate energy E_{TP}.

The linear systems proposed in Section 4 are over-determined, their first $2N$ unknowns being the coefficients of the two coordinates of the B-spline approximation of a smooth 2D function. They can be expressed in matrix form as $A \cdot X = B$. As a first option, we may take as solution the least squares minimum of:

$$E_{L_2}(X) = \|A \cdot X - B\|^2 \; + \; \lambda \, E_{TP}(k^1, k^2) \; , \tag{18}$$

for some constant $\lambda \geq 0$ (in practice only used for optical flow computation). In order to make the estimation less sensitive to outliers, and to possibly big (non-Gaussian) errors, we can use a smoothed L^1 penalization for the errors:

$$E_{L_{1\varepsilon}}(X) = \sum_{i=1}^{n} \sqrt{(A^i \cdot X - B_i)^2 + \varepsilon^2} \; + \; \lambda \, E_{TP}(k^1, k^2) \; , \tag{19}$$

where A^i is the i-th row of A, B_i is the i-th component of B, and $\varepsilon > 0$ is a small constant (in practice, $\varepsilon = 10^{-3}$). We can minimize $E_{L_{1\varepsilon}}$ via iteratively reweighted least squares. The SOCP problems arising when imposing the positive definiteness on G_ω in (15) can be solved using branch and bound [19].

References

1. 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 6, 1–183 (2010)
2. Grossberg, M., Nayar, S.: A general imaging model and a method for finding its parameters. In: Proc. ICCV. IEEE (2001)
3. Pless, R.: Using many cameras as one. In: Proc. CVPR. IEEE (2003)
4. Sturm, P., Ramalingam, S.: A generic concept for camera calibration. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3022, pp. 1–13. Springer, Heidelberg (2004)
5. Diez-Cañas, G., Vasilyev, Y., Adato, Y., Zickler, T., Gortler, S., Ben-Shahar, O.: A linear formulation of shape from specular flow. In: Proc. ICCV. IEEE (2009)
6. Vasilyev, Y., Zickler, T., Gortler, S., Ben-Shahar, O.: Shape from specular flow: is one flow enough? In: Proc. CVPR. IEEE (2011)
7. Hartley, R., Kang, S.: Parameter-free radial distortion correction with centre of distortion estimation. In: Proc. ICCV. IEEE (2005)
8. Grompone von Gioi, R., Monasse, P., Morel, J.M., Tang, Z.: Towards high-precission lens distortion correction. In: Proc. ICIP. IEEE (2010)
9. Tardif, J.-P., Sturm, P., Roy, S.: Self-calibration of a general radially symmetric distortion model. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 186–199. Springer, Heidelberg (2006)
10. Ramalingam, S., Sturm, P., Boyer, E.: A factorization based self-calibration for radially symmetric cameras. In: Proc. 3DPVT (2006)
11. Thirthala, S., Pollefeys, M.: Radial multi-focal tensors. Int. J. Comput. Vis. 96, 195–211 (2011)
12. Ramalingam, S., Sturm, P., Lodha, S.: Towards generic self-calibration of central cameras. In: Proc. OMNIVIS. INRIA (2005)
13. Ramalingam, S., Sturm, P., Lodha, S.K.: Generic self-calibration of central cameras. Comput. Vis. Image Understand. 114, 210–219 (2010)
14. Taddei, P., Espuny, F., Caglioti, V.: Planar motion estimation and linear ground plane rectification using an uncalibrated generic camera. Int. J. Comput. Vis. 96, 162–174 (2012)
15. Grossman, E., Gaspar, J., Orabona, F.: Discrete camera calibration from pixel streams. Comput. Vis. Image Understand. 114, 198–209 (2010)
16. Nistér, D., Stewénius, H., Grossmann, E.: Non-parametric self-calibration. In: Proc. ICCV. IEEE (2005)
17. Espuny, F.: A closed-form solution for the generic self-calibration of central cameras from two rotational flows. In: Proc. VISAPP (2007)
18. Espuny, F., Burgos Gil, J.: Generic self-calibration of central cameras from two rotational flows. Int. J. Comput. Vis. 91, 131–145 (2011)
19. Sturm, J.: Using SEDUMI 1.02, a Matlab toolbox for optimization over symmetric cones (Updated for version 1.05). Tilburg Univ. (2001)
20. Horn, B., Schunck, B.: Determining optical flow. Artif. Intell. 17, 185–203 (1981)
21. Bogdanova, I., Bresson, X., Thiran, J.P., Vandergheynst, P.: Scale space analysis and active contours for omnidirectional images. IEEE Trans. Pattern Anal. Mach. Intell. 16, 1888–1901 (2007)
22. Puig, L., Guerrero, J.J.: Scale space for central catadioptric systems: Towards a generic camera feature extractor. In: Proc. ICCV. IEEE (2011)
23. Dierckx, P.: Curve and Surface Fitting with Splines. Oxford Univ. Press (1993)

Adaptive Structure from Motion with *a Contrario* Model Estimation

Pierre Moulon[1,2], Pascal Monasse[1], and Renaud Marlet[1]

[1] Université Paris-Est, LIGM (UMR CNRS), Center for Visual Computing, ENPC, 6-8 av. Blaise Pascal, 77455 Marne-la-Vallée, France
[2] Mikros Image. 120 rue Danton, 92300 Levallois-Perret, France
firstname.lastname@enpc.fr, pmo@mikrosimage.eu

Abstract. Structure from Motion (SfM) algorithms take as input multi-view stereo images (along with internal calibration information) and yield a 3D point cloud and camera orientations/poses in a common 3D coordinate system. In the case of an incremental SfM pipeline, the process requires repeated model estimations based on detected feature points: homography, fundamental and essential matrices, as well as camera poses. These estimations have a crucial impact on the quality of 3D reconstruction. We propose to improve these estimations using the *a contrario* methodology. While SfM pipelines usually have globally-fixed thresholds for model estimation, the *a contrario* principle adapts thresholds to the input data and for each model estimation. Our experiments show that adaptive thresholds reach a significantly better precision. Additionally, the user is free from having to guess thresholds or to optimistically rely on default values. There are also cases where a globally-fixed threshold policy, whatever the threshold value is, cannot provide the best accuracy, contrary to an adaptive threshold policy.

1 Introduction

There are numerous approaches to estimate the structure from motion (scene structure and camera motion) from multiple images. Thanks to recent progress in image matching and optimization, it is now possible to compute large scale 3D reconstruction from millions of internet images on reasonable sized cluster [1] or even on a single high-end computer [2]. All these methods aim at working with large datasets of images, but few consider the accuracy of the reconstruction.

Most current Structure from Motion (SfM) pipelines are sequential: they start from a minimal reconstruction and incrementally add new views using pose estimation and 3D point triangulation algorithms. There is no guarantee that the reconstruction converges to the global optimum solution. The implementation often relies on many bundle adjustment steps to optimize the solution and uses hard thresholds for robust model estimation. Recently the L_∞ framework [3] has been shown to solve multi-view geometry problems, minimizing directly the maximal reprojection error rather than the sum of squared error. Although the global minimum is found using convex or linear programming, it becomes computationally expensive when dealing with outliers and large problems [4].

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 257–270, 2013.

This paper makes use of the *a contrario* theory to study the adaptation of model estimation thresholds to input data. We show how to automatically compute these thresholds and illustrate the advantages: besides user-friendly parameterless procedures, we can also reach optimization levels that would be unattainable with globally-fixed thresholds. Our adaptive thresholds are implemented in a SfM pipeline that targets high precision. Examples output of our SfM pipeline applied to 128 (resp. 119) images are shown in Fig. 1. Note the wide variation of the automatic threshold for pose estimation. Our SfM produces a sparse 3D point cloud based on image feature points, not a dense 3D reconstruction; a subsequent multiple-view stereovision pipeline has to be used for that, such as PMVS [5] or the pipeline described in [6]. The dense reconstruction quality critically depends on the calibration computed from SfM.

The paper is organized as follows. Section 2 recalls the principles of Structure from Motion. Section 3 briefly reviews robust model estimation. Section 4 describes the *a contrario* methodology and its general application to model estimation. Section 5 describes the particular stages of the classical incremental SfM that we replace with a specific *a contrario* model estimation. Section 6 details evaluation results on real and synthetic datasets, and Section 7 concludes.

2 Structure from Motion — The Classical Pipeline

Structure from Motion (Fig. 2) computes an external camera pose per image (the motion) and a 3D point cloud (the structure) representing the pictured scene. Inputs are images and internal camera calibration information. Feature points are detected in each image (e.g., SIFT [9] or SURF [10]) and matched between image pairs. There are two main approaches to correlate detected features and solve the SfM problem: the *incremental* pipeline and the *global* method.

The incremental pipeline is a growing reconstruction process. It starts from an initial two-view reconstruction (the seed) that is iteratively extended by adding new views and 3D points, using pose estimation and triangulation. Due to the incremental nature of the process, successive steps of non-linear refinement, like bundle adjustment and Levenberg-Marquardt steps, are performed to minimize the accumulated error (drift) [11,12].

The general feature correspondence and SfM processes are described in algorithms 1 and 2. The first algorithm outputs pairwise correspondences that are consistent with the estimated fundamental matrix. Homography estimation is used to choose an initial image pair with numerous correspondences while keeping a wide enough baseline. The second algorithm takes these correspondences as input and yields a 3D point cloud as well as the camera poses. Steps marked with a star ($*$) are those we redefine within the *a contrario* framework. This allows critical thresholds to be automatically adapted to the input images, which yields more accurate SfM as we shall see.

State of the art systems and methods for SfM include Bundler [13], Samantha [14], image triplets based approaches [15,7] and Visual Odometry systems [16,17]. All these systems and methods rely on RANSAC-based model

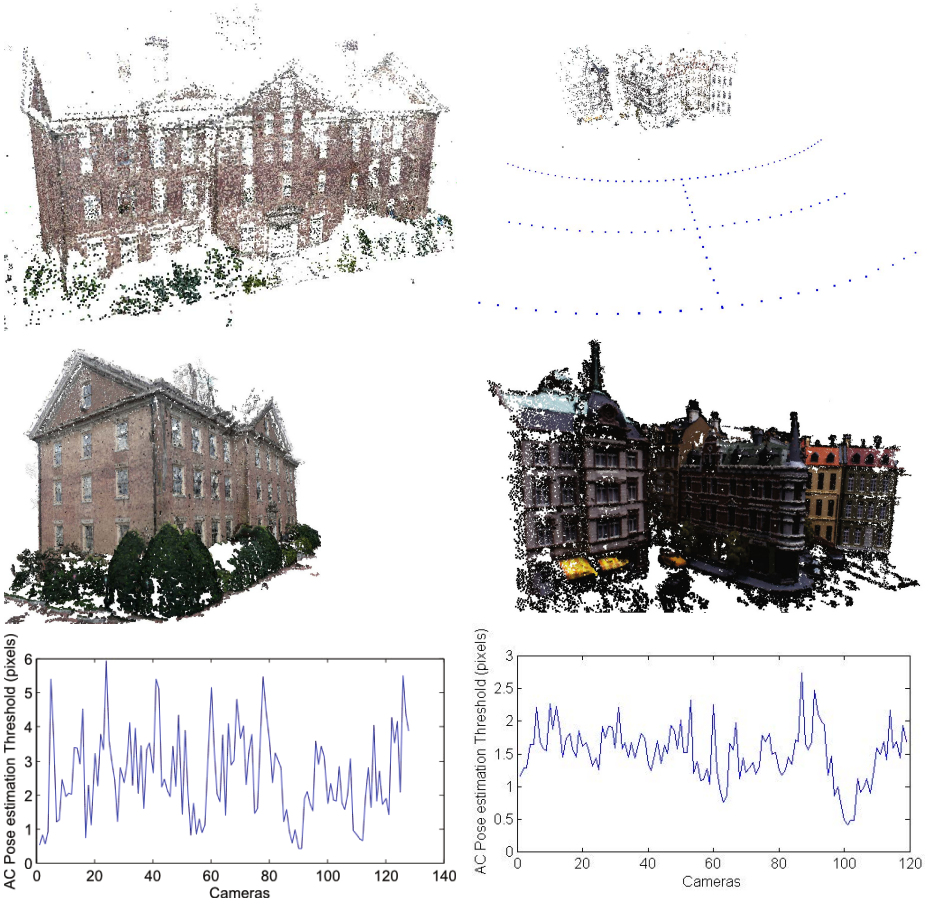


Fig. 1. From top to bottom: sparse 3D reconstruction from our SfM pipeline, PMVS densification [5] and variation of the automatic threshold of pose estimation. Left: the 128 images dataset from [7] source code. Right: the 119 images of 004 scene from [8] dataset. Estimated camera positions represented as blue dots.

estimation to be robust to noise/false data. However, it introduces static thresholds, which have to be set empirically.

The global methods compute essential matrix for all possible input pairs and perform the reconstruction in a two-step process. First globally consistent rotations are computed from the relative pairwise rotations (see Martinec and Pajdla [18] and Govindu [19,20]), then structure and translation equations are solved via the L_∞ constraint [3], or L_1 penalization [4] to deal with outliers. As in the incremental pipeline, the basis of the method is a robust estimation of a model that is controlled by a static empirical threshold.

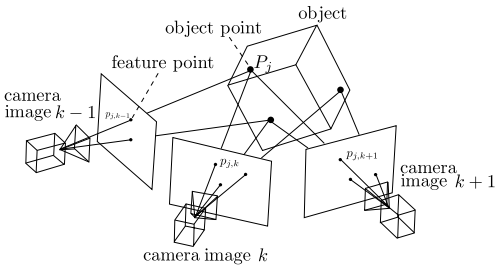


Fig. 2. Structure from Motion

Algorithm 1. Computation of geometry-consistent pairwise correspondences

Require: image set
Ensure: pairwise point correspondences that are geometrically consistent
  Compute putative matches:
    detect features in each image and build their descriptor
    match descriptors (brute force or approximate nearest neighbor)
  Filter geometric-consistent matches:
∗ estimate fundamental matrix F
∗ estimate homography matrix H

3 Parameterizing Robust Model Estimation

Robust model estimation from noisy data that are corrupted by outliers is often performed with the RANSAC (RANdom SAmple Consensus) algorithm [21], or one of its variants. This is the case for the above-mentioned SfM systems.

RANSAC is a randomized procedure due to complexity considerations. It repeatedly selects random sample sets S from the data, whose minimal size is sufficient to estimate the parameters of a model. At each trial, inliers are defined as the data that fits the model within an acceptable error threshold T. After a given number of iterations, the model parameters that maximize the number of corresponding inliers are returned.

The RANSAC algorithm depends on a critical parameter: the choice of threshold T. If T is too small, then little data is selected as inliers, which leads to model imprecision and even, sometimes, to the impossibility to estimate a model because the number of inliers is too small. If T is too large, then outliers (false positives) contaminate inliers, which also leads to inaccurate or wrong models.

But the user is generally clueless about the choice of a threshold value. This is very much the case for SfM. Even though SfM thresholds are generally expressed in pixels, which could make sense to the user, they actually refer indirectly to complex operations concerning feature points, and it is practically impossible to adjust or guess any sensible threshold by just looking at the pictures of a dataset. Threshold selection is exemplified in Fig. 3 for image registration.

Algorithm 2. Incremental Structure from Motion

Require: internal camera calibration (matrix K, possibly from EXIF data)
Require: pairwise geometry consistent point correspondences
Ensure: 3D point cloud
Ensure: camera poses
  compute correspondence tracks t
  compute connectivity graph G (1 node per view, 1 edge when enough matches)
  pick an edge e in G with sufficient baseline (compare F and H)
∗ robustly estimate essential matrix from images of e
  triangulate $t \cap e$, which provides an initial reconstruction
  contract edge e
  while G contains an edge **do**
    pick edge e in G that maximizes track(e) $\cap$ {3D points}
∗   robustly estimate pose (external orientation/resection)
    triangulate new tracks
    contract edge e
    perform bundle adjustment
  end while

Fig. 3. Robust homography estimation. From left to right: RANSAC threshold (transfer error through homography) $T = 0.5$ pixels yields 6 points correspondences, threshold $T = 2$ pixels (i.e., default Bundler threshold) yields 19 points, and threshold $T = 6.8$ pixels yields 50 points as well as a better estimated homography. This last value was actually automatically computed with an *a contrario* technique, that statistically determines a confidence threshold (cf. Section 4).

RANSAC thus faces the user with a dilemma: setting a low threshold and possibly underestimating inliers, which may reduce model accuracy and jeopardize model existence, or setting a high threshold and possibly corrupt data with outliers, which may also decrease precision. In practice, the user relies on default threshold values, that work reasonably well although they might be sub-optimal.

Another issue relates to the globality of parameterization. In practical settings, many instances of a model estimation problem have to be solved independently, for different elements of a dataset. For instance, in SfM, pose has to be estimated many times for a number of different image pairs. The fact is that each problem instance calls for a specific threshold value, adapted to the corresponding data noise. However, most systems only accept a global threshold value for treating a whole dataset. Such a global threshold is naturally too low for some data, and too high for others. There are thus cases where even a perfect oracle can only provide a sub-optimal global parameterization.

4 *A contrario* Model Estimation

Our approach to address the issues listed in Section 3 is to use a methodology for finding a model that best fits the data with a confidence threshold that adapts automatically to noise. For this, we use an *a contrario* model estimation.

In this framework, the computed thresholds are such that they have a good chance of correctly telling apart inliers from outliers. As a result, the accuracy of model estimation tends to be as good as possible (given the sampling strategy), and there are less risks of inadvertently selecting too few inliers for a model to be estimated. Moreover, as thresholds adapt to data, they can vary depending on each image, which allows estimations that would otherwise be impossible with a globally-fixed threshold. Last, the user is free from having to set opaque values or to optimistically rely on default values. Automatic and specific *a contrario* threshold values are illustrated in 3.

4.1 The *a contrario* Methodology

The *a contrario (AC)* methodology relies on the Helmholtz principle: "an observed strong deviation from the background model is relevant information". In other words, a configuration that is unlikely to be explained by chance is conspicuous. This theory has been first introduced by Desolneux *et al.* in [22] and applied to detection in images.

Applied to model estimation, the *a contrario* approach answers the question "Does this model arise by chance?" and thus decides the meaningfulness of a model. The corresponding statistical criterion is data-specific and avoids empirically setting thresholds for inlier/outlier discrimination. It thus provides a parameter-free evolution of RANSAC, called AC-RANSAC [23]. Additionally, once a meaningful model is found, the convergence of AC-RANSAC can be accelerated by reducing the number of random samples and drawing further samples among the inliers of this model. *A contrario* model estimation has first been introduced to estimate the fundamental matrix under the name of ORSA (Optimized-RANSAC) [24], later renamed as AC-RANSAC and extended to multiple model estimation under the name MAC-RANSAC [25].

AC-RANSAC looks for a consensus set that includes a controlled Number of False Alarms (NFA), as described below. A false alarm in this context is a model that is actually due to chance. This requires the definition of a background model $\mathcal{H}_0$ and of a rigidity measure. $\mathcal{H}_0$, called the null hypothesis, is a model of random correspondence: a pair of independent points that are uniformly distributed in their respective image. The rigidity measure is the residual error (of inliers) with respect to an estimated model.

The generic NFA for a rigid model M, which is a generalization of Moisan and Stival's NFA [24], also mentioned in [25], is:

$$NFA(M,k) = N_{\text{out}}(n - N_{\text{sample}}) \binom{n}{k} \binom{k}{N_{\text{sample}}} (e_k(M)^d \alpha_0)^{k - N_{\text{sample}}} \qquad (1)$$

where

- k is the number of hypothesized inlier correspondences,
- n is the total number of correspondences,
- N_{sample} is the cardinal of a RANSAC sample,
- N_{out} is the number of models that can be estimated from a RANSAC sample of N_{sample} correspondences (N_{sample} is often such that $N_{\text{out}} = 1$),
- $e_k(M)$ is the k-th lowest error to the model M among all n correspondences,
- α_0 is the probability of a random correspondence having error 1 pixel,
- d is the error dimension: 1 for point-to-line distance, 2 for point-to-point.

α_0 is independent on the tested model M, being the probability of a random correspondence under *background model* distribution having error 1 pixel: e.g., ratio of area of band of radius 1 and of the area of image for point-to-line distance. The term $e_k(M)^d\alpha_0$ is the probability of a random correspondence having error at most $e_k(M)$. The last factor in the formula is thus the probability of $k - N_{\text{sample}}$ correspondences having error at most $e_k(M)$. The other factors represent a number of tests. In other words, this is an expectation of false alarms for model M having k inliers under the null hypothesis. Model M is considered as valid if

$$NFA(M) = \min_{k=N_{\text{sample}}+1\ldots n} NFA(M,k) \leq \epsilon\,. \tag{2}$$

The only parameter is ϵ. It is usually set to 1, and the inlier/outlier error threshold for model M is e_k, with k minimizing (2).

AC model estimation requires finding $\arg\min_M NFA(M)$ among all models M computed from all possible N_{sample} correspondences. For a given M, the complexity of computing $NFA(M)$ is $O(n \log n)$ since it requires sorting the errors $e_k(M)$ of all n correspondences. However, the number of possible models is $N_{\text{out}} \binom{n}{N_{\text{sample}}}$, which becomes exceedingly large as soon as $N_{\text{sample}} > 2$, hence the random model sampling tests of RANSAC.

Minimizing the NFA instead of maximizing the inlier count (if an inlier/outlier threshold T is given) or minimizing the median of errors (in the least median of squares variant) is the task of AC-RANSAC. The definite advantage over standard RANSAC is that the precision $e_k(M)$, that replaces T, adapts to the data. In our experiments, we let AC-RANSAC [23] set the threshold without any additional constraint. More precisely, we only impose that the returned model provides at least $2N_{\text{sample}}$ inliers.

4.2 Rigidity Measures for Robust Structure from Motion Models

The robust model estimations that are required to define an incremental 3D reconstruction pipeline are the fundamental matrix, homography, essential matrix and pose estimations (see Section 2). Each kind of model has its own definition of rigidity. To devise the *a contrario* rigid model estimation algorithm for these cases, we need to determine the values of α_0, d, N_{sample} and N_{out} assuming a uniform distribution of correspondences. Two main groups of measures are needed: "point to point" and "point to line" distances.

Table 1. Number of samples and number of models for the model estimators

Model	Fundamental		Homography	Essential		Pose estimation	
N_{sample}	7	8	4	5 (see [26])	8	4 + K (see [27])	6
N_{out}	3	1	1	10	1	1	1

Point to point distance. For homography and camera pose estimation:
- $\alpha_0 = \frac{\pi}{A}$: it is the ratio of the radius 1 disk area to image area A.
- $d = 2$: the disk area grows quadratically with its radius.

Point to line distance. For essential and fundamental matrix estimation:
- $\alpha_0 = \frac{2D}{A}$: considering a band of "radius" 1 around an image line, whose length cannot exceed the image diameter D, α_0 is the upper bound of the ratio of areas of such a band to area of the image. Notice this is only an upper bound used for faster computation, which may be more selective than strictly necessary. The actual α_0 should depend on the considered line.
- $d = 1$: the band area grows linearly with the distance to the line.

N_{out} is the maximum number of models that can be computed for a set of N_{sample} correspondences. It depends on the estimation procedure. The values of N_{out} are listed in Table 1. Note that N_{out} may also depend on the actual sample: e.g., computing a fundamental matrix with the 7-point algorithm requires finding roots of a third degree polynomial, which can have 1 or 3 solutions; similarly, the 5-point algorithm for the essential matrix solver involves finding real roots of a 10-degree polynomial. In such a case, we consider the maximum possible number of algorithm outcomes to get an upper bound of the NFA.

AC estimation of a fundamental matrix and of a homography have been described before [24,25]. In the case of homography estimation, we additionally pick inliers among those that where previously selected for the fundamental matrix estimation. Our AC estimation of the essential matrix and of the pose is original. Note that our pose estimation involves a single image domain instead of two in the other formulations.

5 An *a contrario,* Incremental Structure from Motion

Robust model estimation in incremental SfM is traditionally implemented using RANSAC and controlled via globally-fixed thresholds, which has the above-mentioned drawbacks (cf. Section 3). Bundler, for instance, uses as default parameters a 9-pixel reprojection threshold for the fundamental matrix estimation, 6 pixels for homography and 4 pixels for pose. These are heuristic choices that yield decent results in many datasets but cannot adapt to all situations.

Using the *a contrario* approach, we have adaptive thresholds for all components of a SfM pipeline that require a robust model estimation (cf. Section 4). Our *a contrario* 3D reconstruction pipeline is separated in two AC blocks: first, the computation of feature correspondences, and second, the SfM process itself.

***A contrario* correspondence.** For the computation of fundamental matrices and homographies, we replace estimations by RANSAC with AC-RANSAC. For homography, we additionally pick inliers among those that where previously selected for the fundamental matrix estimation, which reduces the search space. This yields the statistically most consistent set of matches between feature sets as well as a computed threshold for the model found. As we shall see, it selects more stable matches for the camera pose estimation.

***A contrario* camera pose estimation.** For pose estimation, which may need the matrix of instrinsics, we also replace RANSAC with AC-RANSAC. The computed threshold of the resection is particularly valuable because it provides a confidence estimate on the current view that is used as a threshold for outlier rejection of new possible triangulated tracks. Each newly triangulated point yielding a larger reprojection error is discarded.

Our reconstruction pipeline is not bound by the usage of a static threshold T_m per kind of model m. It provides adaptive thresholds $T_{m,i}$ for each computed model, i.e., for each kind of model and for each model of a given kind to estimate, given corresponding data (typically, a pair of images).

6 Experiments

We have implemented an *a contrario,* incremental SfM system, as described in Section 5. Our reconstruction pipeline is entirely written in high level C++, with flexible template modules. In particular, we use a generic AC-RANSAC implementation [23] and new model solvers only need to be warped into a given structure. We plan to open source our system to make available an easy to read/use/modify platform for SfM. Unit tests have been designed for each computer vision building block, that also illustrate how to use the various modules.

In the following, we mainly compare our system with Bundler [13], a popular and efficient system that is open source and fairly easy to use. For comparison, our code has 8,000 lines of code while Bundler has 20,000. AC-RANSAC results for specific kinds of models are also illustrated by comparison to RANSAC only.

To evaluate our approach, we have experimented with datasets where ground truth is available. We have used the datasets of Strecha *et al.* [28], with laser ground truth, dataset 004 of [8] with calibration ground truth and 2 additional synthetic generated dataset.

6.1 Threshold Variation for Fundamental Matrix Estimation

To assess the interest of adaptive thresholds for feature correspondence estimation, we have estimated fundamental matrices on [28] and measured the average baseline error of the SfM reconstruction, over all views of the dataset, for various values of the corresponding threshold T_F. Results are shown in Table 2.

Note that the rank-1 threshold value varies depending on the dataset, meaning there is no ideal static threshold that leads to the best results in the Bundler

Table 2. Fundamental matrix threshold consequence over reconstruction: average error (in meters) w.r.t. ground truth (over all views of the dataset). For Bundler: average baseline error and corresponding rank, depending on threshold values. Best in **bold**. X denotes a failed calibration, one of the views being rejected by Bundler. For AC-SfM: average baseline error and distribution of the computed threshold values.

Scene		Bundler T_F threshold					AC-SfM T_F threshold			
		1	3	6	9	12	auto	min	med	max
FountainP11	error	0.002	0.003	0.003	0.004	0.005	**0.001**			
	ranking	**1**	3	2	4	5		0.57	1.00	10.5
HerzJesusP8	error	0.004	0.003	0.003	0.007	0.003	**0.001**			
	ranking	4	**1**	3	5	2		0.63	1.88	5.26
HerzJesusP25	error	0.004	0.010	0.005	**0.004**	0.004	0.005			
	ranking	3	5	4	**1**	2		0.23	1.53	82.8
CastleP19	error	8.22	0.029	0.032	0.039	X	**0.015**			
	ranking	4	**1**	2	3	X		0.69	0.91	15.7
CastleP30	error	0.055	0.057	0.043	0.042	0.045	**0.011**			
	ranking	4	5	2	**1**	3		0.55	0.92	284

chain. This is confirmed by the distribution of the computed AC threshold (over all views of the dataset): there is no stable median value and extreme values (min and max) greatly vary. The average AC baseline error is significantly lower than with the best static threshold in most cases. The error is however almost equal for HerzJesusP25. The reason is that some false matches and bad estimates can still occur in the AC-RANSAC case.

6.2 Camera Pose Estimation

To evaluate camera pose estimation, two views are first used for building a 3D point cloud, then the other images are compared to that point cloud to estimate their pose. The results are displayed in Fig. 4.

The default RANSAC threshold $T = 4$ of Bundler fails for images 0, 2, 10-13 and 23-24, because not enough correspondences with that precision are found. For the other images, the error with respect to the ground truth is worse in baseline and in angle than for the much larger threshold $T = 12$. The AC-RANSAC adaptive thresholds provides errors that are similar to $T = 12$ RANSAC. A closer study shows that the $T = 4$ selection incorporates outliers and thus yields a less accurate result. Naturally, these outliers are also present for $T = 12$, but an averaging effect happens to produce a good accuracy. Still, the *a contrario* pose estimation is more discriminative, strongly adjusting to the context, and yields an accuracy comparable to $T = 12$ with slightly fewer correspondences. No system could register camera 13 though, because of a lack of overlap with the initial pair.

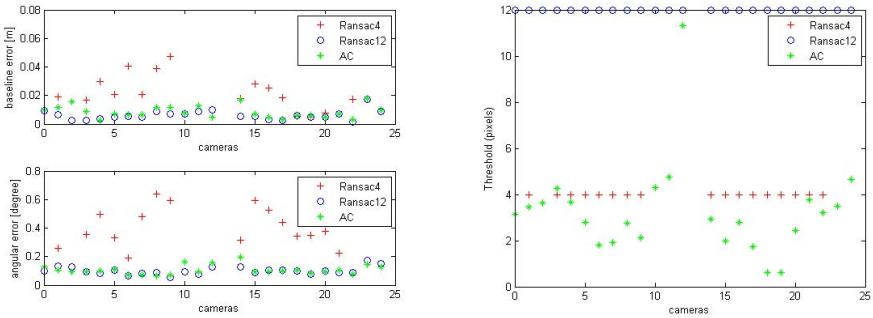


Fig. 4. Evaluation of camera pose estimation with RANSAC and AC-RANSAC on HerzJesusP25. Two views are first registered with a fundamental matrix, yielding a 3D point cloud. Then the pose of all other images is computed relatively to that point cloud. Thresholds 4 (Bundler's default) and 12 are tested. On the left is displayed the pose error relative to ground truth (baseline, angle), on the right the used thresholds.

6.3 Structure from Motion Accuracy Comparison

Finally, we have evaluated the whole AC-SfM pipeline, comparing it to Bundler. The same inputs are considered, i.e., SIFT keypoints and a maximum ratio of 0.6 for the best to second best descriptor matches. For this evaluation, we used both real [28,8] and synthetic datasets. The quality of the reconstructions is

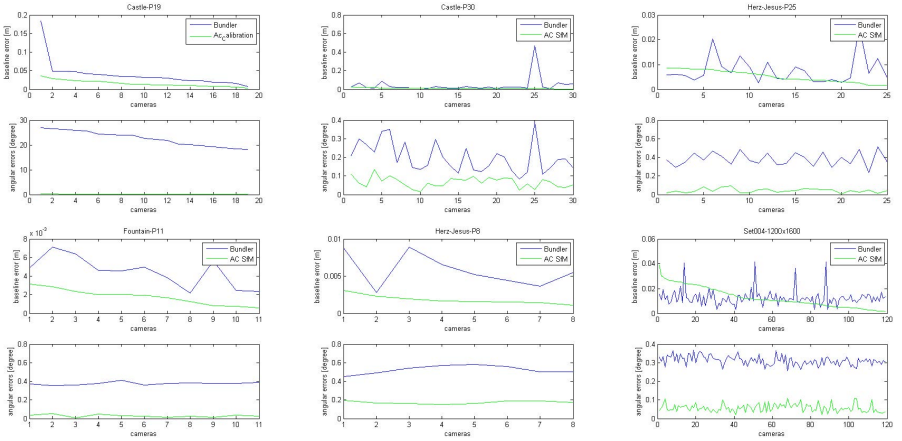


Fig. 5. Evaluation of camera calibration with Bundler and *a contrario* SfM on [28], and scene 004 from [8]. To facilitate the graph reading, the error measures (baseline and angle) for the different views are sorted in decreasing order of *a contrario* SfM baseline error. The angular error for Castle-P19, averaging 25° for Bundler, could not be explained and should not be considered in this comparison. On the whole AC SfM is significantly more accurate than Bundler, and has a more equal distribution of errors.

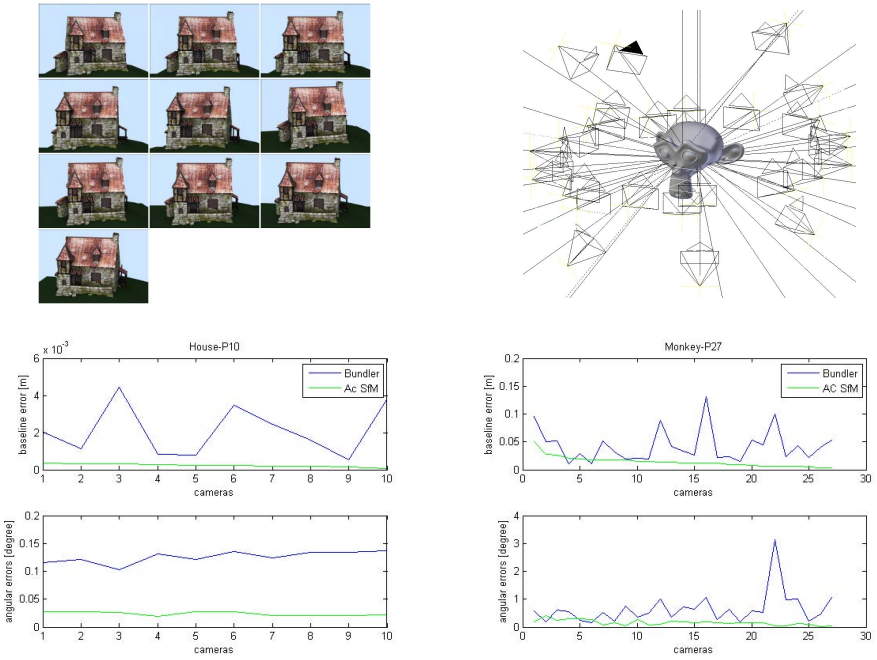


Fig. 6. Camera calibration with Bundler and *a contrario* SfM on synthetic datasets

evaluated on camera external position and rotation with a 7-degree of freedom rigid transform registration (scale, translation, rotation) [29]. The rigid transform is used to preserve angles and distance ratios. Results are shown in Fig. 5.

The angular error is significantly lower in our pipeline for all datasets while the baseline error is comparable to that of Bundler, although most of the time better and more equally distributed among the views. The exception is Set0004-1200x1600, for which 60% of the baseline errors of the camera are more precise, but all the angular error are better. Figure 6 presents a comparative evaluation on synthetic datasets. *A contrario* SfM gives again significantly more accurate results, thanks to its adaptive thresholds.

It can be noted that Bundler, being considered state of the art, already performs quite well, rotation errors being below a fraction of degree. But there is still room for improvement for applications requiring high-precision, which is what we aim at. In fact, these experiments show that AC consistently yields a better precision (up to factor 10). This provides substantial benefits for 3D reconstruction: a $0.2°$ difference in a ray direction at $10\,m$ distance (typical in most experiments of Figure 5) yields an arc length of $(0.2/180 \times \pi) * 10m = 3.5\,cm$, whereas for such scenes we would like to achieve a $1\,cm$ precision.

7 Conclusion

We have argued the interest of model estimators with fine-grain, adaptive thresholds and described how to automatically perform such model estimations for SfM

within the framework of the *a contrario* theory. We have presented a practical 3D reconstruction pipeline that implements these AC estimators and we have shown that our threshold-free system can select inliers with a better discrimination than classical RANSAC, yielding better reconstructions and poses.

Our original contribution includes the *a contrario* threshold definition for the estimation of the essential matrix and for resection. It can be noted that pose estimation involves here a single image, contrary to symmetrical errors used in other parameterizations. Also original is the use, when estimating a homography, of inliers that were selected for the fundamental matrix estimation. Finally, we have systematized AC estimation in a concrete SfM pipeline and showed that it often outperforms state-of-the-art systems. Prior work had indicated feasibility for some components, not efficiency for a complete system and large-scale data.

A few fixed parameters remain, but we believe some of them can be removed too. In particular, we think that the commonly used SIFT descriptor distance ratio for feature matching can be replaced by an *a contrario* descriptor matching [30]. There is also encouraging work on *a contrario* disparity map estimation [31]. This opens the way to a robust, parameter-free, multiple-view stereovision process computing dense point clouds and 3D meshes.

Acknowledgments. This work was carried out in IMAGINE, a joint research project between École des Ponts ParisTech (ENPC) and the Scientific and Technical Centre for Building (CSTB) and supported by Mikros Image and Agence Nationale de la Recherche ANR-09-CORD-003 (Callisto project).

References

1. Agarwal, S., Snavely, N., Simon, I., Seitz, S., Szeliski, R.: Building Rome in a day. In: 12th IEEE International Conference on Computer Vision (ICCV), pp. 72–79 (2009)
2. Frahm, J.-M., Fite-Georgel, P., Gallup, D., Johnson, T., Raguram, R., Wu, C., Jen, Y.-H., Dunn, E., Clipp, B., Lazebnik, S., Pollefeys, M.: Building Rome on a Cloudless Day. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 368–381. Springer, Heidelberg (2010)
3. Kahl, F.: Multiple view geometry and the L_∞-norm. In: ICCV, pp. 1002–1009 (2005)
4. Dalalyan, A., Keriven, R.: L_1-penalized robust estimation for a class of inverse problems arising in multiview geometry. In: NIPS, pp. 441–449 (2009)
5. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 1362–1376 (2010)
6. Hiep, V., Keriven, R., Labatut, P., Pons, J.: Towards high-resolution large-scale multi-view stereo. In: CVPR, pp. 1430–1437 (2009)
7. Zach, C., Klopschitz, M., Pollefeys, M.: Disambiguating visual relations using loop constraints. In: CVPR, pp. 1426–1433 (2010)
8. Aanæs, H., Dahl, A., Steenstrup Pedersen, K.: Interesting interest points. International Journal of Computer Vision 97, 18–35 (2012)
9. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (IJCV) 60, 91–110 (2004)
10. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)

11. Lourakis, M.I.A., Argyros, A.A.: SBA: A software package for generic sparse bundle adjustment. ACM Transactions on Mathematical Software (TOMS) 36 (2009)
12. Wu, C., Agarwal, S., Curless, B., Seitz, S.M.: Multicore bundle adjustment. In: CVPR, pp. 3057–3064 (2011)
13. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3D. ACM Transactions on Graphics (TOG) 25, 835–846 (2006)
14. Gherardi, R., Farenzena, M., Fusiello, A.: Improving the efficiency of hierarchical structure-and-motion. In: CVPR, pp. 1594–1600 (2010)
15. Havlena, M., Torii, A., Knopp, J., Pajdla, T.: Randomized structure from motion based on atomic 3D models from camera triplets. In: CVPR, pp. 2874–2881 (2009)
16. Scaramuzza, D., Fraundorfer, F.: Visual odometry: Part I - the first 30 years and fundamentals. IEEE Robot. Automat. Mag. 18 (2011)
17. Fraundorfer, F., Scaramuzza, D.: Visual odometry: Part II - matching, robustness, and applications. IEEE Robot. Automat. Mag. 19 (2012)
18. Martinec, D., Pajdla, T.: Robust rotation and translation estimation in multiview reconstruction. In: CVPR (2007)
19. Govindu, V.M.: Combining two-view constraints for motion estimation. In: CVPR, vol. 2, pp. II.218–II.225 (2001)
20. Govindu, V.M.: Robustness in Motion Averaging. In: Narayanan, P.J., Nayar, S.K., Shum, H.-Y. (eds.) ACCV 2006. LNCS, vol. 3852, pp. 457–466. Springer, Heidelberg (2006)
21. 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 (CACM) 24, 381–395 (1981)
22. Desolneux, A., Moisan, L., Morel, J.M.: From Gestalt theory to image analysis: a probabilistic approach, 1st edn. Springer (2007)
23. Moisan, L., Moulon, P., Monasse, P.: Automatic homographic registration of a pair of images, with a contrario elimination of outliers. Image Processing On Line (2012), `http://dx.doi.org/10.5201/ipol.2012.mmm-oh`
24. Moisan, L., Stival, B.: A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix. Int. J. of Computer Vision (IJCV) 57, 201–218 (2004)
25. Rabin, J., Delon, J., Gousseau, Y., Moisan, L.: MAC-RANSAC: a robust algorithm for the recognition of multiple objects. In: Proc. of 3DPTV 2010, Paris (2010)
26. Nistér, D.: An efficient solution to the five-point relative pose problem. In: CVPR, vol. 2, pp. II.195–II.202 (2003)
27. Lepetit, V., Moreno-Noguer, F., Fua, P.: EPnP: an accurate $O(n)$ solution to the PnP problem. International Journal of Computer Vision (IJCV) 81, 155–166 (2009)
28. Strecha, C., von Hansen, W., Van Gool, L.J., Fua, P., Thoennessen, U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. In: CVPR, pp. 1–8 (2008)
29. Haralick, R.M., Shapiro, L.G.: Computer and Robot Vision, 1st edn. Addison-Wesley Longman Publishing Co., Inc., Boston (1992)
30. Rabin, J., Delon, J., Gousseau, Y.: A statistical approach to the matching of local features. SIAM J. Imaging Sciences 2, 931–958 (2009)
31. Sabater, N., Almansa, A., Morel, J.M.: Meaningful matches in stereovision. IEEE Transactions on Pattern Analysis and Machine Intelligence 99 (2011)

Precise 3D Reconstruction from a Single Image

Changqing Zou[1,2,3], Jianbo Liu[1], and Jianzhuang Liu[1,4,5]

[1] Shenzhen Key Lab for CVPR, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
[2] Graduate University of Chinese Academy of Sciences, Beijing, China
[3] Department of Physics and Electronic Information Science, Hengyang Normal University, Hengyang, China
[4] Department of Information Engineering, The Chinese University of Hong Kong
[5] Media Lab, Huawei Technologies Co. Ltd., China
{cq.zou,jb.liu}@siat.ac.cn, liu.jianzhuang@huawei.com

Abstract. 3D object reconstruction from single images has extensive applications in multimedia. Most of existing related methods only recover rough 3D objects and the objects are often required to be interconnected. In this paper, we propose a novel method which uses a set of auxiliary reference grids to precisely reconstruct 3D objects from a single uncalibrated image. In our system, the user first draws the line drawings of the objects. Then, the initial focal length f of the camera is computed with a calibration method, and then the initial focal length is refined by a reference grid. With the refined f, a 3D position measurement environment is constructed, and a world coordinate system is defined by the user. After that, a set of reference grids are used to find the precise 3D locations of the object points and the wireframes of the objects are recovered automatically. Finally, the system generates the surfaces and renders the complete 3D objects. Besides the precise 3D modeling, our reconstruction method does not require the objects in a scene are interconnected. A set of examples are provided to demonstrate the ability of handling complex polyhedral objects and curved surfaces within one framework.

1 Introduction

3D object reconstruction from a single 2D view of a scene has attracted considerable attention. It has many applications such as 3D environment modeling, 3D object design, and 3D object retrieval. This research is also a challenging problem because it is ill-posed with less information available from a single view.

Many methods have been proposed for 3D reconstruction from single images, such as [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15]. Debevec et al. [4] recover 3D buildings with parametric primitives from one or several photos taken by a calibrated camera. This system requires a large amount of user interaction involved [12]. Besides, it requires the camera to be pre-calibrated with known intrinsic parameters. Liebowitz et al. [6] create architectural models by geometric relationships from architectural scenes. Their method needs to

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 271–282, 2013.

compute the vanishing lines of all object faces with auxiliary planes and the reconstruction errors are easy to accumulate with the face by face propagation. Sturm and Maybank's method [7] first does camera calibration, then recovers part of the points and planes, and finally obtains the 3D positions of other faces by propagation. The limitation of this method is that the parts of objects have to be sufficiently interconnected and the reconstruction errors may also be accumulated. Guillou et al. [5] carry out 3D reconstruction with a rectangular 3D box that fits at best with the potential objects within the scene. It can only recover simple planar objects. Jelinek and Taylor [8] propose a method of polyhedron reconstruction from single images using several camera models, which requires that the polyhedra have to be linearly parameterized. Zhang et al. [2] try to reconstruct free-form 3D models from single images. The method costs the user a lot of time to specify the constraints from an image.Shimodaira [11] uses the shading information, one horizontal or vertical face, and convex and concave edges to recover the shape of polyhedron in single images. This method handles simple polyhedral only. Li et al. [13] reconstruct objects from single images by obtaining a closed-form solution for the shape vector, using connectivity and perspective symmetry properties. This method only considers planar objects. Liu et al. [14], [15] try to reconstruct complex 3D planar objects from single images. These methods suppose the imaging is parallel or weak-perspective projection and they only obtain rough 3D models.

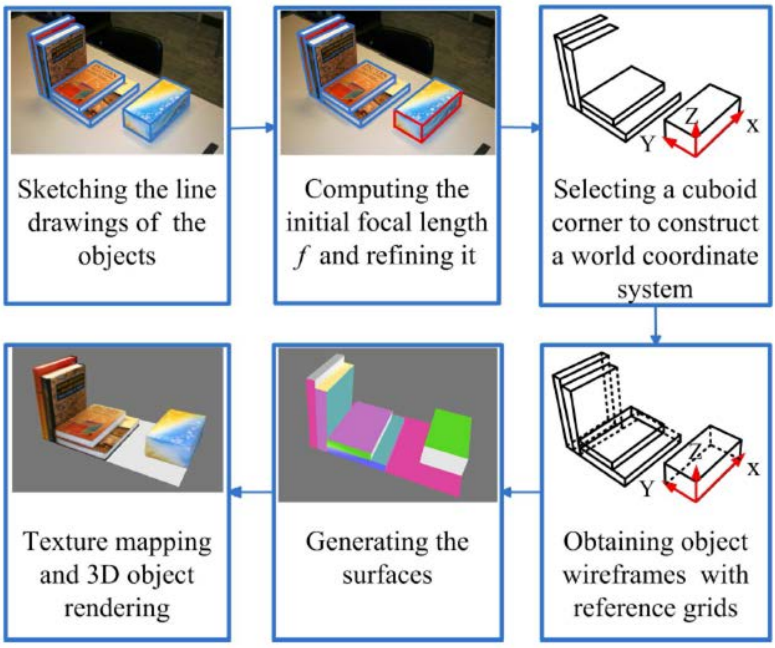


Fig. 1. Flow chart of our approach

In this paper, we propose a novel method which uses a set of auxiliary reference grids to precisely reconstruct both polyhedral objects and curved-surfaced objects from a single image with unknown camera parameters. In our system, the user first draws the edges and vertices of the objects. Then, the initial focal length f of the camera is computed with a calibration method. After that, we use a reference grid to obtain an accurate focal length. With the accurate f, we construct a 3D measurement environment and a world coordinate system. The 3D wireframes of the objects of interest are created by the reference grids. Finally, we generate the surfaces and render the complete 3D objects. The pipeline of our system is shown in Fig. 1.

Similar to previous works, we also focus on man-made objects. However, previous methods mainly consider plausible object reconstruction, while ours concentrates on precise reconstruction. Man-made objects usually have regular shapes such as rectangular faces, parallel lines, orthogonal corners,symmetrical structures, and lines perpendicular to the ground. These properties are easy to be identified by humans. Compared with previous works, ours has the following advantages: (1) Because of the refined camera focal length, the reference grids used to obtain accurate measurement, and a little user interaction, our method can precisely recover complex up-to-a-scale 3D objects. (2) Compared to other approaches it may handle a wider class of objects and demonstrates the ability of handling polyhedral objects and curved surfaces within one framework. (3) It may obtain accurate relative positions between objects in a scene which are not interconnected.

2 Calibration and Measurement Environment

In this work, we assume that there is at least one visible rectangular cuboid corner that can be perceived by the user. Actually, this is common in man-made objects. This corner is used to estimate the focal length and build a world coordinate system.

We consider the viewing camera coordinate system as an orthogonal Cartesian system with x, y and z axes, associated with a pin-hole camera. The origin of the system coincides with the view point, and the z direction is the optical axis orthogonal to the screen (image plane). The focal length determines the appearance of the object image in the camera. Therefore, to recover the objects, it is necessary to determine the focal length. We assume that the principle point (the intersection point between the z axis and the image plane) is located at the image center and a camera model is given with the following intrinsic matrix

$$\mathbf{k} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{1}$$

where f is the focal length which needs to be found. The most widely used approach to find f is to use vanishing points and vanishing lines. However, obtaining the accurate f by the geometric constraints is not easy since computing

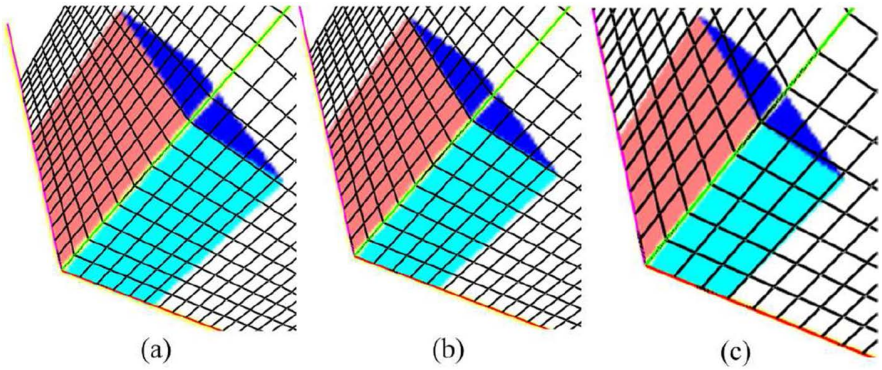


Fig. 2. Refining the focal length. Fig. 2a and 2c show the cases of $f_i \neq f_r$. Fig. 2b shows the case of $f_r = f_i$.

vanishing points and lines is numerically unstable. In our system, based on a known cuboid corner, we use the method in [16] to obtain an initial focal length.

After the initial focal length is obtained, the user employs an orthogonal reference grid (e.g., the one in Fig. 2) to refine the focal length, and then reconstructs the measurement environment. We take Fig. 2 as an example to illustrate this process, where the known cuboid corner is identified by the user.

The user first loads an image on the screen. OpenGL is used to render the image and the 3D reference grid. The initial focus length is used in the rendering . The reference grid, which can be rotated, translated, and zoomed by the user, is adjusted to match the lower cuboid corner of the block, as shown in Fig. 2 (In this process the cuboid corner point of reference grid is placed to a initialized position with a fixed Z coordinate). If the initial focal length f_i is not equal to the real focal length f_r, the upper edge of the block's faces will not overlap with two lines of the reference grid, as shown in Fig. 2a ($f_r > f_i$) and Fig. 2c ($f_i > f_r$). The accurate focal length can be found by adjusting it in an OpenGL rendering function such that the edges of the rectangular faces of the block coincide with the lines of the reference grid, as shown in Fig. 2b. Thanks to the initial focal length, this adjusting is done easily and quickly by the user.

Once the refined focal length has been obtained, the measurement environment is also constructed. It is the combination of the camera coordinate system and a world coordinate system. The origin of the world coordinate system is set at the cuboid corner and its axes coincide with the three edges of the cuboid (see Fig. 2b). In this environment, a set of 3D reference grids can be used to facilitate the 3D reconstruction of the objects.

3 Generating Object Wireframes

A 3D object wireframe consists of the 3D vertices and edges of the object. These 3D vertices and edges are derived with the help of a set of 3D reference grids put

on the image by the user. In this section, we first introduce the reference grids (Fig. 3), and then take the reference grids in Figs. 3a and b as an example to explain how to obtain 3D coordinates of an object's vertices in an image. Finally we discuss how to recover curves.

3.1 Reference Grids

Fig. 3 shows a set of 3D reference girds, which are used to conveniently find the 3D coordinates of the points of man-made objects in a scene. Fig. 3a is a rectangular reference plane used to measure coplanar object points. The reference grid in Fig. 3b has two orthogonal planes, which are used to refine the focal length and obtain the 3D point positions on a wedge with two orthogonal planes. Fig. 3c consists of three perpendicular planes used to recover cuboid objects. Sometimes the reference grid in Fig. 3c is more efficient than the combination of the reference grids in Fig. 3a and Fig. 3b. Fig. 3d shows a concentric disc, which can be moved along its axis. It is used to measure the height of a cylinder or cone and the radius of a circle. Fig. 3e is a combination of Fig. 3a and Fig. 3d, both on

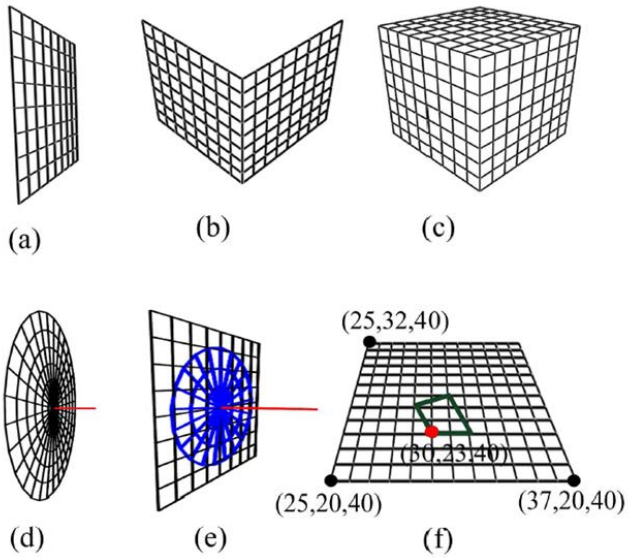


Fig. 3. 3D reference grids for measurement. (a) is a rectangular reference plane used to measure coplanar object points. (b) has two orthogonal planes, which are used to refine the focal length and obtain the 3D point positions on a wedge with two orthogonal planes. (c) is the combination of the reference grids in Fig. 3a and Fig. 3b. (d) is a concentric disc, which can be moved along its axis. It can be used to measure a surface of revolution (SOR). (e) is a combination of (a) and (d), both on the same plane. The concentric disc in (e) can be moved on the plane. It can be used to obtain the radii and center locations of circles. (f) shows the position of an arbitrary point on a plane can be obtained.

the same plane. The concentric disc can be moved on the plane. Fig. 3e can be used to obtain the radii and center locations of circles.

All these grids can be rotated, translated, and zoomed by the user, and the density and size of the grids can also be adjusted. To facilitate the measurement, we also develop an auxiliary measurement point which can be moved with a variable step size along the surface of a reference grid in 3D space by the user (as shown in Fig. 3f). The current 3D position of the point can be updated immediately when the user moves it, with the help of the known 3D location of the reference grid. Once the point is moved to coincide with an object vertex, the system obtains the precise 3D position of the vertex.

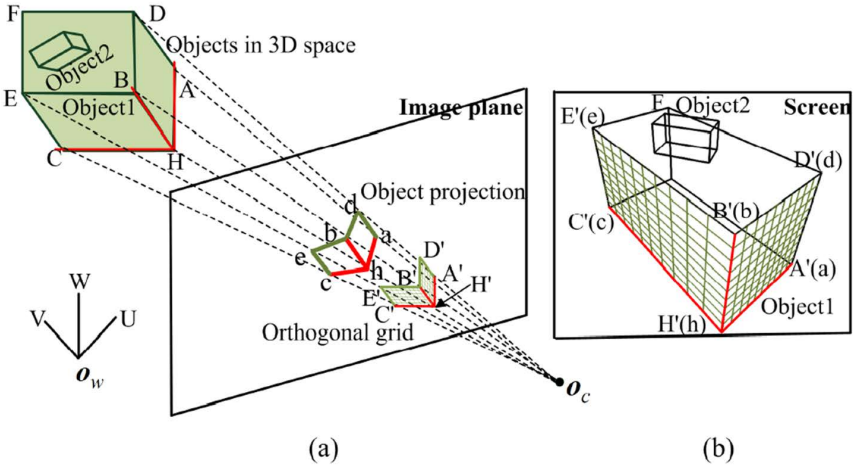


Fig. 4. 3D measurement with reference grids

3.2 3D Measurement

We take the reference grids in Figs. 3a and b as an example to explain how to obtain 3D coordinates of the object's vertices in Fig. 4. In Section 2, we have described how to establish the measurement environment and the 3D world coordinate system. Let the origin and three axes of this system be O_w, U, V, and W, respectively. Suppose that a rectangular block (Object1 in Fig. 4) is located on the plane spanned by U and V, i.e., $\overline{O_wU}$, $\overline{O_wV}$, $\overline{HA}$, and $\overline{HC}$ are on the same plane in 3D space. Then the 3D coordinates of H, A, and C can be obtained easily with the 3D reference grid in Fig. 3a and the auxiliary measurement point (as shown in Fig. 3f) when the grid is put on the same plane. (Actually, all the points on this plane can be measured.) It should be mentioned again that all the measurements are up to a scale.

Next we use the 3D reference grid in Fig. 3b to locate the 3D coordinates of D, B, and E of Object1. As shown in Fig. 4a, we first move the corner H' to

make it coincide with h (the projection of H), and rotate this grid such that its three axes $\overline{H'A'}$, $\overline{H'C'}$ and $\overline{H'B'}$ coincide with the three projected edges $\overline{ha}, \overline{hc}$ and $\overline{hb}$, respectively. Then, we can conveniently find the 3D coordinates of D, B and E by adjusting the density of the grid lines and the size of the grid. For example, the 3D coordinate of D is the same as that of D' when D' and d coincide.

If there is another object (Object2) located on Object1 as shown in Fig. 4, we can use the same method to obtain the 3D coordinates of Object2's vertices, because all the 3D points on the top of Object1 can be measured when the 3D locations of $\overline{BD}$ and $\overline{BE}$ are known.

3.3 Recovering Curves

Our system can also reconstruct objects with curved surfaces. As shown in Fig. 5a, $S1$ is a curved face, the wireframe of which consists of two curves l_1 and l_2, and two vertical lines. Using the reference grid in Fig. 3a and the auxiliary measurement point, we can obtain the 3D coordinates of several points on l_1 and l_2. Based on these 3D coordinates, we use Bezier curves to approximate l_1 and l_2 to obtain two smooth curves. Similarly, we can also find the 3D curves of the four circles in Fig. 5b.

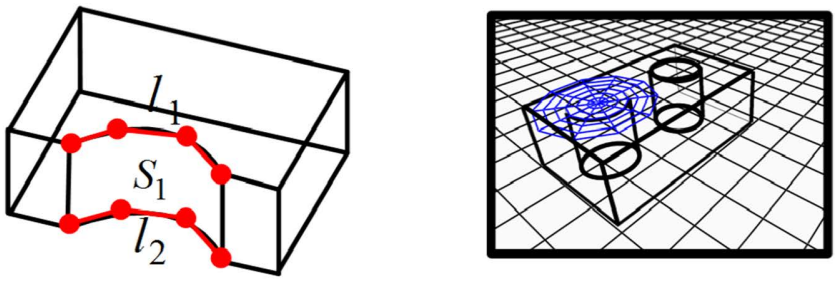


Fig. 5. 3D measurement with reference grids

4 Face Identification and Surface Generation

After the system obtains the wireframes of the 3D objects, the next step is to identify the circuits that represent the object faces and fill in them with surface patches. Face identification is not a trivial problem, and many methods have been proposed to find faces from a line drawing [17], [18], [19]. In this paper the algorithm proposed in [17] is used to detect the faces. In our interactive system, the user can also identify a face by selecting the edges of the face, which can correct possible wrongly detected faces by the algorithm in [17].

Generating a planar surface with its given 3D vertices is an easy task. For a curved face, after the system fits its curved edges with Bezier curves, the face is filled in with bilinearly blended Coons patches [20].

5 Experiments and Discussion

In this section, we show some experimental results to demonstrate the performance of our system. The most important step of our method is obtaining the 3D wireframes of objects, which depends on accurate 3D point localization.

Table 1 shows the precision of the reconstructed objects in Fig. 6a. We give the comparison between the measured result and the ground truth. The ground truth data are obtained manually when the scene is constructed. $L_g : W_g : H_g$ and $L_m : W_m : H_m$ are respectively the ratios of the length, width, and height of the ground truth and the measured result for an object. The error is obtained by setting

$$\alpha = L_g/L_m, \quad W_m^{'} = \alpha W_m, \quad H_m^{'} = \alpha H_m, \tag{2}$$

and computing

$$error = \frac{\sqrt{(W_m^{'} - W_g)^2 + (H_m^{'} - H_g)^2}}{\sqrt{W_g^2 + H_g^2}}. \tag{3}$$

From Table 1, we find that the measured ratios of the objects are very close to the ground truth. Even though there are small errors, they are still within an acceptable range. If the density of the grid lines is increased and the moving step size of the auxiliary measurement point is decreased, the errors can be reduced. It should be mentioned that most previous methods usually give only rough 3D reconstructed objects without precision shown.

We have conducted a number of experiments on real images to verify the effectiveness and precision of our system. Due to the space limitation, only some of them are given here. In Fig. 6, the first column shows the original images, and the other two columns show the reconstructed 3D objects with some with texture mapped, each in two views. Figs. 6a and b have objects with only planar faces and all the hidden edges are also drawn, the objects in Figs. 6c, d and e consist of both planar and curved faces and only the visible edges are drawn. These results show that, compared to previous methods like [1], [4], [5], [13],

Table 1. Comparison between the measured result and the ground truth for the objects in Fig. 6a

	$L_g : W_g : H_g$	$L_m : W_m : H_m$	Error
Book1	26.2 : 18.0 : 3.2	25 : 17 : 3	1.1%
Book2	24.2 : 18.0 : 4.7	24 : 18 : 5	0.8%
Book3	26.1 : 20.3 : 3.3	26 : 19 : 3	6.5%
Book4	23.5 : 16.3 : 3.2	24 : 17 : 3	2.5%
Box	23.0 : 10.5 : 7.8	22 : 10 : 7	3.7%

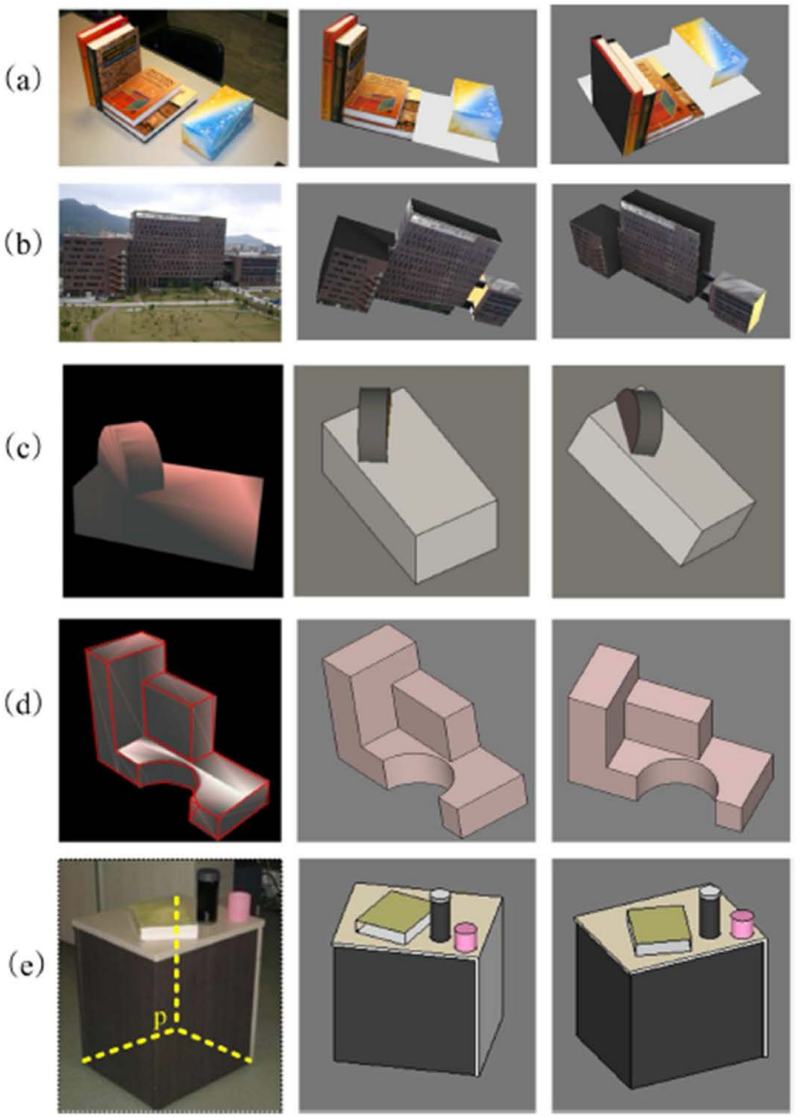


Fig. 6. Some experimental results

[6], [7], and [8], our system can precisely reconstruct complex objects with both planar and curved faces in one framework and the reconstruction time is within an acceptable range (e. g. Fig. 6c and d can be recovered within five minutes and the reconstruction time is less than the similar method in [4] and [5]).

Compared to previous approaches, which focuses on precise 3D reconstruction (like [13] which recovers planar objects with a closed form solution), our method can also recover the hidden parts precisely. As Fig. 6e shows, the

Table 2. Comparison between the measured results and the ground truth of internal diameters of the vases in Fig. 7

	Ground truth	Measured value	Error
vase1 (Fig. 7a)	$r_1 : r_2 : r_3 =$ $10.9 : 22.1 : 11.1$	$r_1' : r_2' : r_3' =$ $5 : 11 : 5$	7%
vase2 (Fig. 7b)	$L_1 : L_2 : L_3 : L_4 : L_5 =$ $10.2 : 4.4 : 10.2 : 9.1 : 11.1$	$L_1' : L_2' : L_3' : L_4' : L_5' =$ $20 : 9 : 20 : 18 : 22$	1.3%

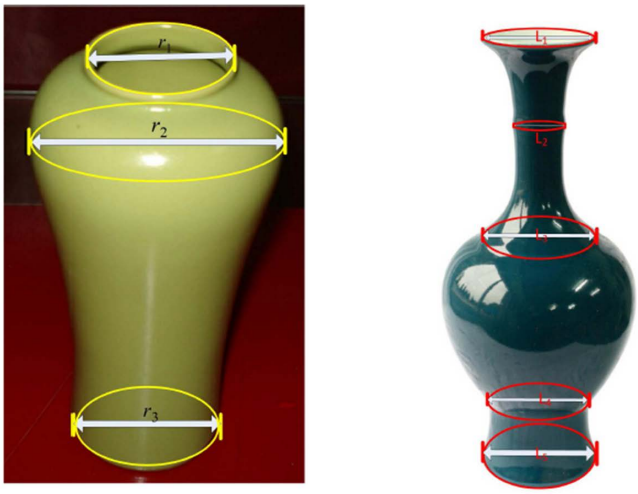


Fig. 7. Another application: measuring the geometric parameters with reference girds

hidden point P can be located accurately and sufficiently with the reference grid in Fig. 3a. Fig. 7a shows another example in which invisible parts of the cross section circles of a surface of revolution (SOR) can be recovered with the reference grid presented in this paper. Besides, Our approach overcomes the problem of accumulated reconstruction errors which exists in [6] and [7].

It should be mentioned that our method may not obtain the precise positions of the points on an irregular object. One example is the tip of a non-symmetric pyramid. Its position cannot be determined uniquely from a single view even if the bottom of the pyramid is located correctly. However, the user is still able to guess the location of the tip with a reference grid according to his/her perception (In Fig. 6e, the points of curved part are located by user's estimation).

Besides the 3D reconstruction, our method also provide another application: measuring the geometric parameters for perspective objects. Figs. 7a-b and Table 2 show a successful application in shape measurement with our method. From Table 2, we can see the internal diameters of typical parts of the two vases

can be measured precisely with the reference grid shown in Fig. 3d (Here we only demonstrate the measurement precision of our method and do not show the reconstruction results of Figs. 7a-b. Our method can also be used to recover perspective SORs since that the cross sections of a SOR can be obtained conveniently with reference grids in this paper and the focal length can be obtained by the algorithm in [21] and [22]).

6 Conclusion and Future Work

Most of existing methods of 3D reconstruction from a single image only recover rough 3D objects and the objects are often required to be interconnected. To address these problems, we have presented an efficient method using a set of reference grids to precisely locate both planar and curved 3D objects in a scene. Besides the precise 3D reconstruction, we also extended our approach to measuring the geometric parameters of perspective objects. The user interaction may be optimized and more complex 3D reference girds can be designed to facilitate the reconstruction in the future.

Acknowledgement. This work was supported by grants from Natural Science Foundation of China (60975029, 61070148), Science, Industry, Trade, Information Technology Commission of Shenzhen Municipality, China (JC200903180635A, JC201005270378A,ZYC201006130313A), Guangdong Innovative Research Team Program (No. 201001D0104648280), and the Construct Program of the Key Discipline in Hunan Province.

References

1. Criminisi, A., Reid, I., Zisserman, A.: Single view metrology. Int'l J. Computer Vision 40, 123–148 (2000)
2. Zhang, L., Dugas-Phocion, G., Samson, J., Seitz, S.: Single-view modelling of free-form scenes. The Journal of Visualization and Computer Animation 13, 225–235 (2002)
3. Prasad, M., Zisserman, A., Fitzgibbon, A.: Fast and controllable 3d modelling from silhouettes. In: Proc. Annual Conference of the European Association for Graphics (2005)
4. Debevec, P., Taylor, C., Malik, J.: Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach. In: Proc. SIGGRAPH (1996)
5. Guillou, E., Meneveaux, D., Maisel, E., Bouatouch, K.: Using vanishing points for camera calibration and coarse 3d reconstruction from a single image. The Visual Computer 16, 396–410 (2000)
6. Liebowitz, D., Criminisi, A., Zisserman, A.: Creating architectural models from images. Computer Graphics Forum 18, 39–50 (1999)
7. Sturm, P., Maybank, S., et al.: A method for interactive 3d reconstruction of piecewise planar objects from single images. In: Proc. BMVC (1999)

8. Jelinek, D., Taylor, C.: Reconstruction of linearly parameterized models from single images with a camera of unknown focal length. IEEE Trans. PAMI 23, 767–773 (2001)
9. Hong, W., Ma, Y., Yu, Y.: Reconstruction of 3-D Symmetric Curves from Perspective Images without Discrete Features. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3023, pp. 533–545. Springer, Heidelberg (2004)
10. Pavić, D., Schönefeld, V., Kobbelt, L.: Interactive image completion with perspective correction. The Visual Computer 22, 671–681 (2006)
11. Shimodaira, H.: A shape-from-shading method of polyhedral objects using prior information. IEEE Trans. PAMI 28, 612–624 (2006)
12. Jiang, N., Tan, P., Cheong, L.: Symmetric architecture modeling with a single image. ACM TOG 28, 113 (2009)
13. Li, Z., Liu, J., Tang, X.: A closed-form solution to 3d reconstruction of piecewise planar objects from single images. In: Proc. CVPR (2007)
14. Liu, J., Cao, L., Li, Z., Tang, X.: Plane-based optimization for 3d object reconstruction from single line drawings. IEEE Trans. PAMI 30, 315–327 (2008)
15. Liu, J., Chen, Y., Tang, X.: Decomposition of complex line drawings with hidden lines for 3d planar-faced manifold object reconstruction. IEEE Trans. PAMI 33, 3–15 (2011)
16. Svedberg, D., Carlsson, S.: Calibration, pose and novel views from single images of constrained scenes. Pattern Recognition Letters 21, 1125–1133 (2000)
17. Liu, J., Lee, Y., Cham, W.: Identifying faces in a 2d line drawing representing a manifold object. IEEE Trans. PAMI 24, 1579–1593 (2002)
18. Liu, J., Tang, X.: Evolutionary search for faces from line drawings. IEEE Trans. PAMI 27, 861–872 (2005)
19. Liu, J., Lee, Y.: Graph-based method for face identification from a single 2d line drawing. IEEE Trans. PAMI 23, 1106–1119 (2001)
20. Farin, G.: Curves and Surfaces for Computer-Aided Geometric Design: A Practical Code. Academic Press (1996)
21. Colombo, C., Del Bimbo, A., Pernici, F.: Metric 3d reconstruction and texture acquisition of surfaces of revolution from a single uncalibrated view. IEEE Trans. PAMI 27, 99–114 (2005)
22. Colombo, C., Comanducci, D., Del Bimbo, A.: Camera Calibration with Two Arbitrary Coaxial Circles. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 265–276. Springer, Heidelberg (2006)

An Efficient Image Matching Method for Multi-View Stereo

Shuji Sakai[1], Koichi Ito[1], Takafumi Aoki[1],
Tomohito Masuda[2], and Hiroki Unten[2]

[1] Graduate School of Information Sciences, Tohoku University,
Sendai, Miyagi, 980–8579, Japan
sakai@aoki.ecei.tohoku.ac.jp
[2] Toppan Printing Co., Ltd., Bunkyo-ku, Tokyo, 112–8531, Japan

Abstract. Most existing Multi-View Stereo (MVS) algorithms employ the image matching method using Normalized Cross-Correlation (NCC) to estimate the depth of an object. The accuracy of the estimated depth depends on the step size of the depth in NCC-based window matching. The step size of the depth must be small for accurate 3D reconstruction, while the small step significantly increases computational cost. To improve the accuracy of depth estimation and reduce the computational cost, this paper proposes an efficient image matching method for MVS. The proposed method is based on Phase-Only Correlation (POC), which is a high-accuracy image matching technique using the phase components in Fourier transforms. The advantages of using POC are (i) the correlation function is obtained only by one window matching and (ii) the accurate sub-pixel displacement between two matching windows can be estimated by fitting the analytical correlation peak model of the POC function. Thus, using POC-based window matching for MVS makes it possible to estimate depth accurately from the correlation function obtained only by one window matching. Through a set of experiments using the public MVS datasets, we demonstrate that the proposed method performs better in terms of accuracy and computational cost than the conventional method.

1 Introduction

In recent years, the topic of Multi-View Stereo (MVS) has attracted much attention in the field of computer vision [1,2,3,4,5,6,7,8,9,10]. MVS aims to reconstruct a complete 3D model from a set of images taken from different viewpoints. The major MVS algorithm consists of two steps: (i) estimating the 3D points on the basis of a photo-consistency measure and visibility model using a local image matching method and (ii) reconstructing a 3D model from estimated 3D point clouds. The accuracy, robustness and computational cost of MVS algorithms depend on the performance of the image matching method, which is the most important factor in MVS algorithms.

Most MVS algorithms employ Normalized Cross-Correlation (NCC)-based image matching to estimate 3D points [1,5,6,8,9,10]. Goesele et al. [5] have applied

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 283–296, 2013.

NCC-based image matching to the plane-sweeping approach to estimate a reliable depth map by cumulating the correlation values calculated from multiple image pairs with changing the depth. Campbell et al. [8] estimated a depth map more accurately than Goesele et al. [5] by using the matching results obtained from neighboring pixels to reduce outliers. Bradley et al. [9] and Furukawa et al. [10] achieved robust image matching by transforming the matching window in accordance with not only the depth but also the normal of the 3D points.

In the MVS algorithms mentioned in the above, an NCC value between matching windows is used as the reliability of a 3D point. The optimal 3D point is estimated by iteratively computing NCC values between matching windows with changing the parameter of 3D point, i.e., depth or normal. For example, the plane-sweeping approach such as that of Goesele et al. [5] computes NCC values between matching windows with discretely changing the depth and selects the depth that has the highest NCC value as the optimal one. To estimate the accurate depth, a sufficiently small step of the depth must be employed, which significantly increases computational cost. If the step of the depth is small, the translational displacement of a 3D point is a sub-pixel on the multi-view images. Most existing methods assume that the sub-pixel resolution of a matching window is represented by linear interpolation. This assumption, however, is not always true.

In this paper, we propose an efficient image matching method for MVS using Phase-Only Correlation (POC) (or simply "phase correlarion"). POC is a kind of correlation function calculated only from the phase components in Fourier transform. The translational displacement and similarity between two images can be estimated from the position and height of the correlation peak of the POC function, respectively. Kuglin et al. [11] proposed a fundamental image matching technique using POC, and Takita et al. [12] proposed a sub-pixel image registration technique using POC. The major advantages of using POC-based instead of NCC-based image matching are the following two points: (i) the correlation function is obtained only by one window matching and (ii) the accurate sub-pixel translational displacement between two windows can be estimated by fitting the analytical correlation peak model of the POC function. By applying POC-based image matching to depth estimation, the peak position of the POC function indicates the displacement between the assumed and true depth. Hence, we can directly estimate the true depth from the results of only one POC-based window matching. By introducing POC-based image matching to the plane-sweeping approach, we need little window matching to estimate the true depth from multi-view images. In addition, the accuracy of depth estimation can be improved by integrating the POC functions calculated from multiple stereo image pairs. Thus, using POC-based window matching for MVS makes it possible to estimate depth accurately from the correlation function obtained only by one window matching. Through a set of experiments using the public multi-view stereo datasets [13], we demonstrate that the proposed method performs better in terms of the accuracy and the computational cost than the method proposed by Goesele et al. [5].

2 Phase-Only Correlation

This section describes the fundamentals of POC-based image matching. Most existing POC-based image matching methods are for 2D images. The image matching between stereo images can be reduced to a 1D image matching through stereo rectification. In this paper, we employ 1D POC function to estimate the depth from multi-view images.

POC is an image matching technique using the phase components in Discrete Fourier Transforms (DFTs) of given images. Consider two N-length 1D image signals $f(n)$ and $g(n)$, where the index range is $-M, \cdots, M$ $(M > 0)$ and hence $N = 2M + 1$. Let $F(k)$ and $G(k)$ denote the 1D DFTs of the two signals. $F(k)$ and $G(k)$ are given by

$$F(k) = \sum_{n=-M}^{M} f(n) W_N^{kn} = A_F(k) e^{j\theta_F(k)}, \tag{1}$$

$$G(k) = \sum_{n=-M}^{M} g(n) W_N^{kn} = A_G(k) e^{j\theta_G(k)}, \tag{2}$$

where $k = -M, \cdots, M$, $W_N = e^{-j\frac{2\pi}{N}}$, $A_F(k)$ and $A_G(k)$ are amplitude, and $\theta_F(k)$ and $\theta_G(k)$ are phase. The normalized cross-power spectrum $R(k)$ is given by

$$R(k) = \frac{F(k)\overline{G(k)}}{\left|F(k)\overline{G(k)}\right|} = e^{j(\theta_F(k) - \theta_G(k))}, \tag{3}$$

where $\overline{G(k)}$ is the complex conjugate of $G(k)$, and $\theta_F(k) - \theta_G(k)$ denotes the phase difference. The POC function $r(n)$ is defined by Inverse DFT (IDFT) of $R(k)$ and is given by

$$r(n) = \frac{1}{N} \sum_{k=-M}^{M} R(k) W_N^{-kn}. \tag{4}$$

Shibahara et al. [14] derived the analytical peak model of 1D POC function. Let us assume that $f(n)$ and $g(n)$ are minutely displaced with each other. The analytical peak model of 1D POC function can be defined by

$$r(n) \simeq \frac{\alpha}{N} \frac{\sin\left(\pi(n+\delta)\right)}{\sin\left(\frac{\pi}{N}(n+\delta)\right)}, \tag{5}$$

where δ is a sub-pixel peak position and α is a peak value. The peak position $n = \delta$ indicates the translational displacement between the two 1D image signals and the peak value α indicates the similarity between the two 1D image signals. The translational displacement with sub-pixel accuracy can be estimated by fitting the model of Eq. (5) to the calculated data array around the correlation

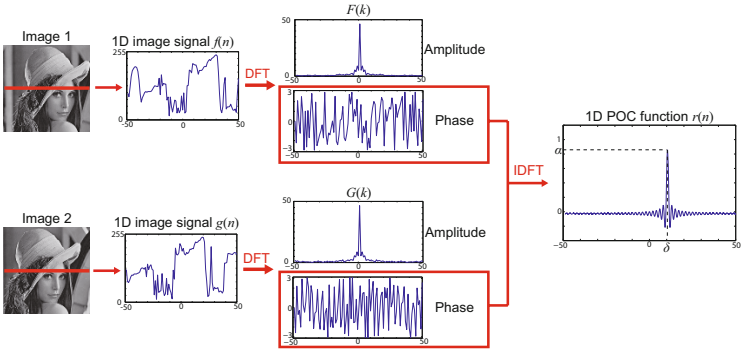


Fig. 1. Example of 1D POC-based image matching

peak, where α and δ are fitting parameters. In addition, we employ the following techniques to improve the accuracy of 1D image matching: (i) windowing to reduce boundary effects, (ii) spectral weighting for reducing aliasing and noise effects, and (iii) averaging 1D POC functions to improve peak-to-noise ratio [12,14]. Fig. 1 shows an example of 1D POC-based image matching.

3 POC-Based Image Matching for Multi-View Stereo

In this section, we describe a POC-based image matching method for MVS. The existing algorithms using NCC-based image matching need to do many NCC computations with changing the assumed depth to estimate the accurate depth of a 3D point. On the other hand, the proposed method estimates the accurate depth only with one window matching by approximating the depth change on a 3D point by the translational displacement on the stereo image and estimating the translational displacement using POC. The proposed method also enhances the estimation accuracy by integrating the POC functions calculated from multiple stereo image pairs.

The POC functions calculated from stereo images with different view-points indicate the different peak positions due to the difference in camera positions. To integrate the POC functions, the proposed method normalizes the disparity of each stereo image and integrates the POC functions on the same coordinate system. So far, Okutomi et al. [15] have proposed the disparity normalization technique to integrate correlation functions calculated from stereo images with different viewpoints. This technique, however, assumes that all cameras are located on the same line. This assumption is not suitable in a practical situation. The disparity normalization technique used in the proposed method, which is a generalized version of the technique proposed by Okutomi et al. [15], can integrate the correlation functions calculated from stereo images with different viewpoints even if the cameras are not located on the same line.

Let $\mathbf{V} = \{V_0, \cdots, V_{H-1}\}$ be the multi-view images with known camera parameters. We consider a reference view $V_R \in \mathbf{V}$ and neighboring views $\mathbf{C} = \{C_0, \cdots, C_{K-1}\} \subset \mathbf{V} - \{V_R\}$ as input images, where H and K are the number of the multi-view images and the number of the neighboring views, respectively. The proposed method generates K pairs of a rectified stereo image and estimates the depth of each point in V_R from the peak position of the correlation function obtained by integrating the POC functions with normalized disparity. We use a stereo rectification method employed in the Camera Calibration Toolbox for Matlab [16].

Next, we describe the key techniques of the proposed method: (i) normalizing the disparity and (ii) integrating the POC functions. Then, we describe the proposed depth estimation method using POC-based image matching.

3.1 Normalization of Disparity

We consider that the camera coordinate of the reference view V_R corresponds to the world coordinate. Let $V_{R,i}^{\mathrm{rect}}$-C_i^{rect} be the rectified stereo image pair, where $V_{R,i}^{\mathrm{rect}}$ is the rectified image of V_R so as to correspond to the view angle of C_i. The relationship among the 3D point $\mathbf{M} = [X, Y, Z]^T$ in the camera coordinate of V_R, the rectified stereo image $V_{R,i}^{\mathrm{rect}}$-C_i^{rect} ($C_i \in \mathbf{C}$) with disparity d_i, and the rectified stereo image $V_{R,j}^{\mathrm{rect}}$-C_j^{rect} ($C_j \in \mathbf{C} - \{C_i\}$) with disparity d_j is defined by

$$\mathbf{M} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \mathbf{R}_i \begin{bmatrix} (u_i - u_{0i})B_i/d_i \\ (v_i - v_{0i})B_i/d_i \\ \beta_i B_i/d_i \end{bmatrix} = \mathbf{R}_j \begin{bmatrix} (u_j - u_{0j})B_j/d_j \\ (v_j - v_{0j})B_j/d_j \\ \beta_j B_j/d_j \end{bmatrix}, \tag{6}$$

where (u_l, v_l) is the corresponding point of $\mathbf{M}$ in $V_{R,l}^{\mathrm{rect}}$, (u_{0l}, v_{0l}) is the optical center of $V_{R,l}^{\mathrm{rect}}$, β_l is focal length and B_l is baseline length between $V_{R,l}^{\mathrm{rect}}$-C_l^{rect} ($l = i, j$). $\mathbf{R}_l$ denotes a rotation matrix from the reference view V_R to the rectified reference view $V_{R,l}^{\mathrm{rect}}$ used in stereo rectification for $V_{R,l}^{\mathrm{rect}}$-C_l^{rect}, and is given by

$$\mathbf{R}_l = \begin{bmatrix} R_{l11} & R_{l12} & R_{l13} \\ R_{l21} & R_{l22} & R_{l23} \\ R_{l31} & R_{l32} & R_{l33} \end{bmatrix}. \tag{7}$$

From Eq. (6), we derive the relationship between d_i and d_j as follows

$$d_i = \frac{R_{i31}(u_i - u_{0i}) + R_{i32}(v_i - v_{0i}) + R_{i33}\beta_i}{R_{j31}(u_j - u_{0j}) + R_{j32}(v_j - v_{0j}) + R_{j33}\beta_j} \frac{B_i}{B_j} d_j. \tag{8}$$

From Eq. (8), the relationship between d_i and d_j is represented by the scaling factor that depends on the camera parameters and the coordinates of the corresponding points in V_R^{rect}. We define the normalized disparity d to take into account the scale factor for each disparity. If we consider the rectified stereo

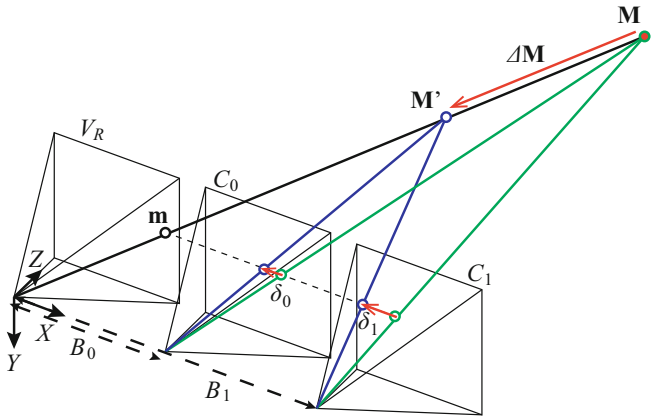


Fig. 2. Geometric relationship between the location of 3D point and the disparity on the images

image pair $V_{R,i}^{\text{rect}}$-C_i^{rect} $(i = 0, \cdots, K-1)$, the relationship between d_i in each rectified stereo pair and the normalized disparity d can be written as

$$d_i = s_i d, \tag{9}$$

where s_i denotes the scale factor for the disparity d_i and is given by

$$s_i = \frac{(R_{i31}(u_i - u_{0i}) + R_{i32}(v_i - v_{0i}) + R_{i33}\beta_i)B_i}{\frac{1}{K}\sum_{l=0}^{K-1}(R_{l31}(u_l - u_{0l}) + R_{l32}(v_l - v_{0l}) + R_{l33}\beta_l)B_l}. \tag{10}$$

In this case, the 3D point $\mathbf{M}$ can be defined by

$$\mathbf{M} = \mathbf{R}_i \begin{bmatrix} (u_i - u_{0i})B_i/(s_i d) \\ (v_i - v_{0i})B_i/(s_i d) \\ \beta_i B_i/(s_i d) \end{bmatrix}. \tag{11}$$

3.2 Integration of POC Function

We consider the 3D point $\mathbf{M}$ and its minutely displaced 3D point $\mathbf{M}' = \mathbf{M} + \Delta\mathbf{M}$, where $\Delta\mathbf{M} = [\Delta X, \Delta Y, \Delta Z]^T$ denotes the minute displacement, as shown in Fig. 2. Let d and d' be the normalized disparities of $\mathbf{M}$ and $\mathbf{M}'$, respectively. Assuming that $\mathbf{M}$ is the true 3D point, the relationship between d and d' is given by

$$d' = d + \delta, \tag{12}$$

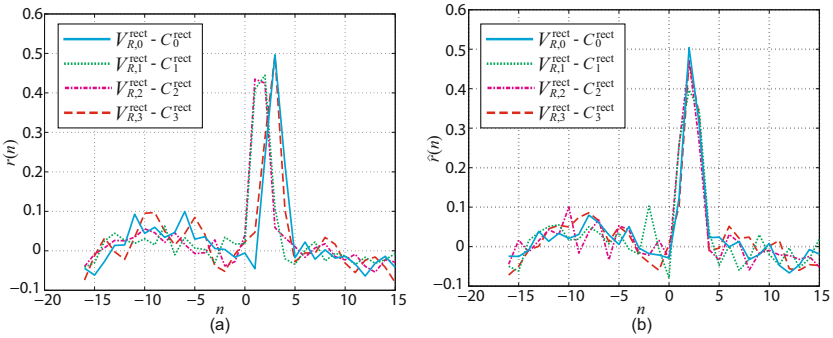


Fig. 3. Integration of the POC functions calculated from stereo image pairs with different viewpoints: (a) POC functions before disparity normalization and (b) POC functions after disparity normalization

where δ denotes the error between the normalized disparities d and d'. For the rectified stereo image pair $V_{R,i}^{\mathrm{rect}}$-C_i^{rect} $(i \in \{0, \cdots, K-1\})$, the relationship between the 3D point $\mathbf{M}'$ and the normalized disparity d is

$$\mathbf{M}' = \mathbf{R}_i \begin{bmatrix} (u_i - u_{0i})B_i/(s_i(d+\delta)) \\ (v_i - v_{0i})B_i/(s_i(d+\delta)) \\ \beta_i B_i/(s_i(d+\delta)) \end{bmatrix}. \tag{13}$$

Let f_i and g_i be the matching windows extracted from $V_{R,i}^{\mathrm{rect}}$ and C_i^{rect} centered on the corresponding point of $\mathbf{M}'$, respectively. Approximating the local image transformation by translational displacement, the translational displacement between f_i and g_i is $\delta_i = s_i\delta$. The displacement δ_i can be estimated from the correlation peak position of the POC function r_i between f_i and g_i as mentioned in Sect. 2. The different rectified stereo image pairs, however, have different translational displacements. For example, δ_i in $V_{R,i}^{\mathrm{rect}}$-C_i^{rect} and δ_j in $V_{R,j}^{\mathrm{rect}}$-C_j^{rect} $(j \in \{0, \cdots, K-1\} - \{i\})$ are not always equal. In other words, the POC functions r_i and r_j have different correlation peak positions.

Addressing this problem, we convert the coordinate system of the POC functions r_i and r_j into the same coordinate system by scaling the matching windows in accordance with each normalized disparity. Let w be the unified size of the matching window. The size of the matching windows of f_i and g_i is defined by $s_i w$. Scaling the image signals f_i and g_i by $1/s_i$, the size of the matching windows is normalized to w, where we denote $\hat{f}_i$ and $\hat{g}_i$ as the scaled version of the matching windows f_i and g_i, respectively. Hence, the correlation peak of the POC function $\hat{r}_i$ between $\hat{f}_i$ and $\hat{g}_i$ is located at δ. Similarly, for the rectified stereo image pair $V_{R,j}^{\mathrm{rect}}$-C_j^{rect}, the correlation peak of the POC function $\hat{r}_j$ between $\hat{f}_j$ and $\hat{g}_j$ is located at the same position δ, although the size of the matching window, i.e., $s_j w$, is different from that for $V_{R,i}^{\mathrm{rect}}$-C_i^{rect}, i.e., $s_i w$.

Fig. 3 (a) shows the POC functions before disparity normalization. In this case, the translational displacement δ_i between matching windows is different

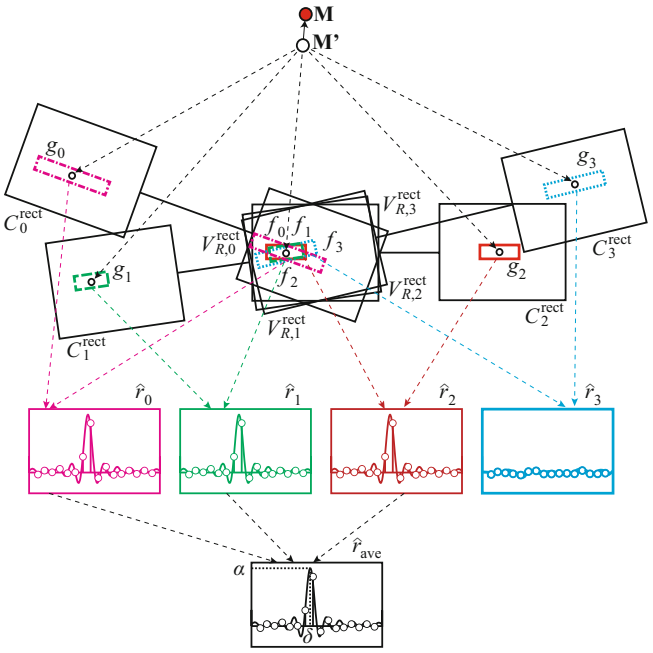


Fig. 4. Depth estimation using POC-based image matching

for each view-point. Thus, the positions of the correlation peaks are also different. On the other hand, Fig. 3 (b) shows the POC functions after disparity normalization. In this case, the translational displacement δ is the same for all the viewpoints. Therefore, all the POC functions overlap at the same position.

Using disparity normalization makes it possible to integrate the POC functions calculated from rectified stereo image pairs with different viewpoints. In this paper, we employ the POC function $\hat{r}_{\text{ave}}$, which is the average of the POC functions $\hat{r}_i$ $(i = 0, \cdots, K-1)$, as the integrated POC functions.

3.3 Depth Estimation Using POC-Based Image Matching

We describe the depth estimation method using POC-based image matching with two important techniques as described above. Fig. 4 shows the flow of the proposed method. First, the initial position of the 3D point $\mathbf{M}'$ is projected onto the rectified stereo image pair $V_{R,i}^{\text{rect}}$-C_i^{rect}, and the coordinates on $V_{R,i}^{\text{rect}}$ and C_i^{rect} are denoted by $\mathbf{m}_i = [u_i, v_i]$ and $\mathbf{m}_i^C = [u_i^C, v_i^C]$, respectively, where $i = 0, \cdots, K-1$. Next, the matching windows f_i and g_i extracted from $V_{R,i}^{\text{rect}}$ centered at $\mathbf{m}_i$ with the size $s_i w \times L$ and C_i^{rect} centered at $\mathbf{m}_i^C$ with the size $s_i w \times L$, respectively. Note that we extract L lines of the matching window to employ the technique averaging 1D POC functions to improve the peak-to-noise ratio as described in Sect. 2. Then, we apply the disparity normalization to the

matching windows f_i and g_i and calculate the 1D POC function $\hat{r}_i$ between $\hat{f}_i$ and $\hat{g}_i$. The correlation peak position of the 1D POC function $\hat{r}_i$ may include a significant error if 3D point $\mathbf{M}'$ is not visible from the neighboring view $C_i \in \mathbf{C}$ or the matching window is extracted from the boundary region of an object that has multiple disparities. In this case, we observe that the correlation peak value α_i drops, since the local image transformation between the matching windows cannot be approximated by translational displacement. To improve the accuracy of depth estimation, the average POC function $\hat{r}_{\text{ave}}$ is calculated from the POC functions $\hat{r}_i$ with $\alpha_i > th_{corr}$, where th_{corr} is a threshold. Finally, the correlation peak position δ with sub-pixel accuracy is estimated by fitting the analytical peak model of the POC function to $\hat{r}_{\text{ave}}$. From Eq. (11), Eq. (12), and δ, the true position of the 3D point $\mathbf{M}$ is estimated by

$$\mathbf{M} = \mathbf{R}_i \begin{bmatrix} (u_i - u_{0i})B_i/(s_i(d' - \delta)) \\ (v_i - v_{0i})B_i/(s_i(d' - \delta)) \\ f_i B_i/(s_i(d' - \delta)) \end{bmatrix}. \tag{14}$$

To generate a depth map, we apply the POC-base image matching to a plane-sweeping approach, and search the depth of each pixel in V_R. Since the POC-based image matching can estimate the depth corresponding to $\pm w/4$ pixel in the neighboring-view image, we search on the ray within the bounding box with changing the depth of $\mathbf{M}'$ in stpdf of $s_i w/4$ pixel in the stereo images. We also apply the the coarse-to-fine strategy using image pyramids to the proposed method described in the above. We first esimate the approximate depth in the coarsest image layer, and then refine the depth in the subsequent image layers.

4 Experiments and Discussion

We evaluate the reconstruction accuracy and the computational cost of the conventional method and the proposed method using the public multi-view stereo image datasets [13]. In the experiments, we employ the famous method using the plane-sweeping approach proposed by Goesele et al. [5] as the conventional method.

4.1 Implementation

We describe the implementation notes for Goesele's method and the proposed methods.

Goesele's Method [5]
The reconstruction accuracy and the computational cost of Goesele's method significantly depends on the step size ΔZ of the depth. In the experiments, we employ four variations of ΔZ such that the resolution of the disparity on the widest-baseline stereo image is $1, 1/2, 1/5$, and $1/10$ pixels. The size of NCC-based window matching is 17×17 pixels. The threshold value for averaging the NCC values calculated from stereo image pairs is 0.3.

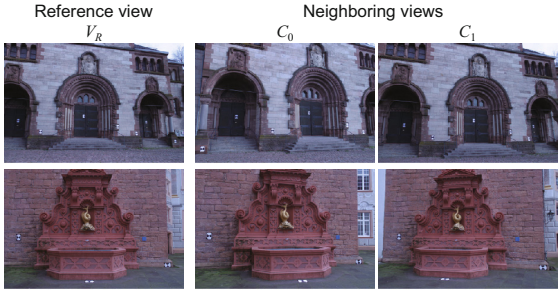


Fig. 5. Examples of reference-view image V_R and neighboring-view images $\mathbf{C}$ used in the experiments (upper: Herz-Jesu-P8, lower: Fountain-P11)

Proposed Method

The parameters for the proposed method used in the experiments are as follows. The threshold th_{corr} is 0.3, the matching window size w is 32 pixel and the number of POC functions L is 17. Note that the effective information of POC function with 32 pixels×17 lines is limited to 17 pixels×17 line, since we apply a Hanning widow with $w/2$-half width to the POC function to reduce the boundary effect as described in Sect. 2. We also employ the coarse-to-fine strategy using image pyramids. The numbers of layers are 2, 3, and 4 for 768×512, $1,536\times1,024$, and $3,072 \times 2,048$ pixels, respectively.

4.2 Evaluation of 3D Reconstruction Accuracy

We evaluate the 3D reconstruction accuracy using Herz-Jesu-P8 (8 images) and Fountain-P11 (11 images), which are available in [13]. The datasets Herz-Jesu-P8 and Fountain-P11 include the multi-view images with $3,072 \times 2,048$ pixels, camera parameters, bounding boxes, and the mesh model of the target object that can be used as the ground truth. For each dataset, we generate depth maps of all the view points using Goesele's method and the proposed method. We use two neighboring-view images $\mathbf{C}$ for one reference-view image V_R. Fig. 5 shows examples of V_R and $\mathbf{C}$ used in the experiments. The performance is evaluated for the three different image sizes : 768×512, $1,536 \times 1,024$, and $3,072 \times 2,048$ pixels.

We evaluate the accuracy of 3D reconstruction by the error rate e defined by

$$e = \frac{|Z_{\text{calculated}} - Z_{\text{ground truth}}|}{Z_{\text{ground truth}}} \times 100 \ [\%], \tag{15}$$

where $Z_{\text{calculated}}$ and $Z_{\text{ground truth}}$ denote the estimated depth and the true depth obtained from the ground truth, respectively. Fig. 6 shows the reconstructed 3D point clouds of Goesele's method and the proposed method for $1,536 \times 1,024$-pixel images. Fig. 7 shows the inlier rates for changing threshold of the error rates for each dataset. Fig. 8 shows the average error rates of inliers, where the inlier is defined by a 3D point whose error rate is less than 1.0%.

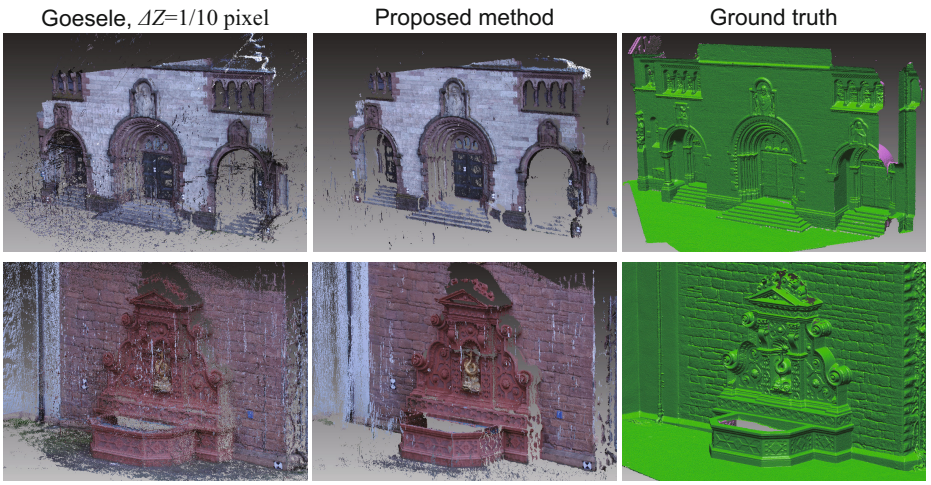


Fig. 6. Reconstruction results of $1,536 \times 1,024$-pixel images for each dataset (upper: Herz-Jesu-P8, lower: Fountain-P11)

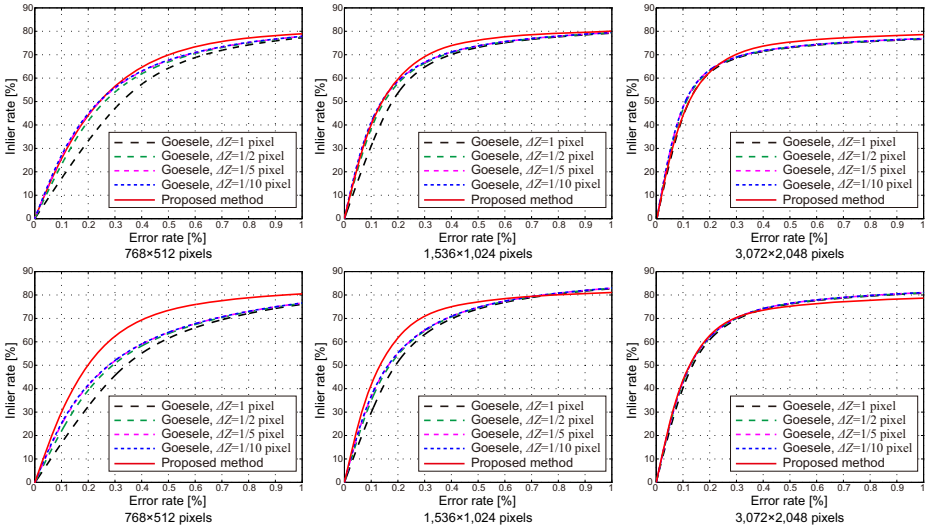


Fig. 7. Inlier rate for each dataset (upper: Herz-Jesu-P8, lower: Fountain-P11)

For Goesele's method, the error rates of the 3D point clouds are small when the step size ΔZ is sufficiently small. For the proposed method, we observe that the reconstructed 3D points are concentrated on smaller error rates than in Goesele's method with $\Delta Z = 1/10$ pixel. We also confirm this result from the average error rates in Fig. 8. For Fountain-P11, the proposed method can estimate more accurate depth than Goesele's method for all the image sizes. In Goesele's

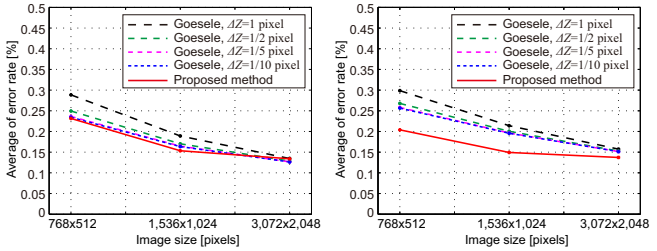


Fig. 8. Average error rates for each dataset (left: Herz-Jesu-P8, right: Fountain-P11)

method, to estimate the accurate depth, the sub-pixel displacement between the matching windows is represented by image interpolation. On the other hand, the proposed method employs the POC-based image matching, which can estimate the accurate sub-pixel displacement between the matching windows by fitting the analytical correlation peak model of the POC function.

As is observed in the above experiments, the proposed method exhibits higher reconstruction accuracy than Goesele's method.

4.3 Evaluation of Computational Cost

We evaluate the computational cost to estimate the depth of one point on the reference-view image for Goesele's method and the proposed method. When using the w-pixel matching window, the proposed method can estimate the displacement within $\pm w/4$ pixels for one window matching. In Goesele's method, we also estimate the displacement within $\pm w/4$ pixels using NCC-based image matching. Table 1 shows the computational cost for each method. Goesele's method with the small step size ΔZ requires high computational cost. On the other hand, the proposed method requires low computational cost that is comparable to that for Goesele's method with $\Delta Z = 1$ pixel or $\Delta Z = 1/2$ pixel. As described in Sect. 4.2, the reconstruction accuracy of the proposed method is higher than that of Goesele's method with $\Delta Z = 1/10$ pixel. Although the computational cost for Goesele's method can be reduced when ΔZ is large, the reconstruction accuracy drops significantly. Compared with Goesele's method,

Table 1. Computational cost to estimate the depth of one point on the reference-view image for each method

	Additions	Multiplications	Divisions	Square roots
Goesele, $\Delta Z = 1$ pixel	75,140	31,246	578	578
Goesele, $\Delta Z = 1/2$ pixel	150,280	62,492	1,156	1,156
Goesele, $\Delta Z = 1/5$ pixel	357,700	156,230	2,890	2,890
Goesele, $\Delta Z = 1/10$ pixel	751,400	312,460	5,780	5,780
Proposed method	40,000	34,496	2,176	1,088

the proposed method exhibits efficient 3D reconstruction from multi-view images in terms of the reconstruction accuracy and the computational cost.

5 Conclusion

This paper has proposed an efficient image matching method for Multi-View Stereo (MVS) using Phase-Only Correlation (POC). The proposed method with normalizing disparity and integrating POC functions can estimate the depth from the correlation function obtained only by one window matching. Also, the reconstruction accuracy of the proposed method is higher than that of NCC-based image matching, since POC-based image matching can estimate the accurate sub-pixel translational displacement between two windows by fitting the analytical correlation peak model of the POC function. Through a set of experiments using the public multi-view stereo datasets, we have demonstrated that the proposed method performs better in terms of accuracy and computational cost than Goesele's method. In future work, we will improve the accuracy of the proposed method to consider the normal vectors of 3D point and develop an MVS algorithm using the proposed method.

References

1. Szeliski, R.: Computer Vision: Algorithms and Applications. Springer-Verlag New York Inc. (2010)
2. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-views stereo reconstruction algorithms. In: Proc. Int'l Conf. Computer Vision and Pattern Recognition, pp. 519–528 (2006)
3. Strecha, C., Fransens, R., Gool, L.V.: Wide-baseline stereo from multiple views: A probabilistic account. In: Proc. Int'l Conf. Computer Vision and Pattern Recognition, pp. 552–559 (2004)
4. Strecha, C., Fransens, R., Gool, L.V.: Combined depth and outlier estimation in multi-view stereo. In: Proc. Int'l Conf. Computer Vision and Pattern Recognition, pp. 2394–2401 (2006)
5. Goesele, M., Curless, B., Seitz, S.M.: Multi-view stereo revisited. In: Proc. Int'l Conf. Computer Vision and Pattern Recognition, pp. 2402–2409 (2006)
6. Goesele, M., Snavely, N., Curless, B., Hoppe, H., Seitz, S.M.: Multi-view stereo for community photo collections. In: Proc. Int'l Conf. Computer Vision, pp. 1–8 (2007)
7. Strecha, C., von Hansen, W., Gool, L.V., Fua, P., Thoennessen, U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. In: Proc. Int'l Conf. Computer Vision and Pattern Recognition, pp. 1–8 (2008)
8. Campbell, N.D.F., Vogiatzis, G., Hernández, C., Cipolla, R.: Using Multiple Hypotheses to Improve Depth-Maps for Multi-View Stereo. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 766–779. Springer, Heidelberg (2008)
9. Bradley, D., Boubekeur, T., Heidrich, W.: Accurate multi-view reconstruction using robust binocular stereo and surface meshing. In: Proc. Int'l Conf. Computer Vision and Pattern Recognition, pp. 1–8 (2008)

10. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. IEEE Trans. Pattern Analysis and Machine Intelligence 32, 1362–1376 (2010)
11. Kuglin, C.D., Hines, D.C.: The phase correlation image alignment method. In: Proc. Int'l Conf. Cybernetics and Society, pp. 163–165 (1975)
12. Takita, K., Aoki, T., Sasaki, Y., Higuchi, T., Kobayashi, K.: High-accuracy subpixel image registration based on phase-only correlation. IEICE Trans. Fundamentals E86-A, 1925–1934 (2003)
13. Strecha, C.: (Multi-view evaluation), http://cvlab.epfl.ch/data/
14. Shibahara, T., Aoki, T., Nakajima, H., Kobayashi, K.: A sub-pixel stereo correspondence technique based on 1D phase-only correlation. In: Proc. Int'l Conf. Image Processing, V-221–V-224 (2007)
15. Okutomi, M., Kanade, T.: A multiple-baseline stereo. IEEE Trans. Pattern Analysis and Machine Intelligence 15, 353–363 (1993)
16. Bouguet, J.Y.: (Camera calibration toolbox for matlab), http://www.vision.caltech.edu/bouguetj/calib_doc/

Self-calibration of a PTZ Camera Using New LMI Constraints

François Rameau, Adlane Habed, Cédric Demonceaux,
Désiré Sidibé, and David Fofi

Université de Bourgogne, Le2i UMR 6306 CNRS, 12 rue de la fonderie,
71200 Le Creusot, France

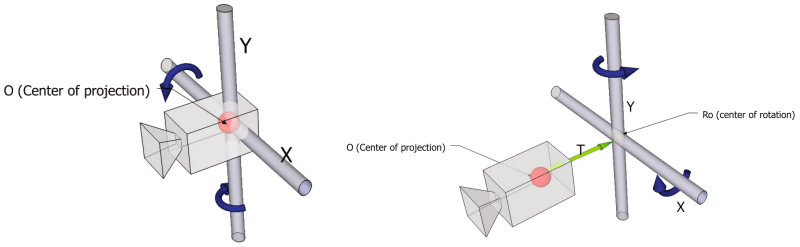
Abstract. In this paper, we propose a very reliable and flexible method for self-calibrating rotating and zooming cameras - generally referred to as PTZ (Pan-Tilt-Zoom) cameras. The proposed method employs a Linear Matrix Inequality (LMI) resolution approach and allows extra tunable constraints on the intrinsic parameters to be taken into account during the process of estimating these parameters. Furthermore, the considered constraints are simultaneously enforced in all views rather than in a single reference view. The results of our experiments show that the proposed approach allows for significant improvement in terms of accuracy and robustness when compared against state of the art methods.

1 Introduction

Pan-Tilt-Zoom cameras are commonly referred to as active cameras because of their ability to be mechanically oriented, typically with rotations around the X-axis for tilting and the Y-axis for panning. Through their zooming capabilities, such cameras make it possible to obtain high resolution images on a particular region of interest (ROI). Wide area coverage and high accuracy on ROIs make PTZ cameras well-suited for surveillance purposes [1], as well as particularly useful in such applications like robotics, panorama creation, video conferencing,... etc. [2,3]. Yet, the flexibility given by the motion and the zoom of the camera may also be a drawback in the sense that any zoom leads to a change in the internal geometry of the camera while any rotation affects its pose. Considering that many of the above mentioned applications require a good estimate of the camera's parameters, whether for controlling the camera or registering a set of images, calibration is a crucial and determinant step for their success.

Calibrating a camera with fixed parameters is a well-known problem which can be solved off-line by means of a calibration pattern as [4]. The case of varying camera parameters is, however, a more challenging task for which the traditional off-line calibration does not provide a viable solution. In this respect, Sturm has proposed in [5] a method that relies on repeatedly pre-calibrating the camera for various settings of its zoom lens as to establish an interdependence model between the internal parameters. Self-calibration, i.e. retrieving the camera's parameters solely from point correspondences across images, is a much more

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 297–308, 2013.



(a) (b)

Fig. 1. Location of the optical center and rotation axis (a) ideal case (b) real case

flexible approach in which resorting to a physical calibration pattern is at no time required in the process. The foundations of the self-calibration of moving cameras with constant parameters were laid by Faugeras *et al.* two decades ago in [6]. The case of purely rotating cameras (PT/PTZ camera) requires however a different formulation [7] to be properly processed.

In this paper we present a method that improves both the accuracy and robustness of the self-calibration of PTZ cameras. Our method employs Linear Matrix Inequalities (LMIs) derived from constraints on certain parameters of the camera and relying on prior knowledge about the aspect ratio and the position of the principal point. These constraints are adjustable and require only rough knowledge about the camera's parameters. Experiments with synthetic and real data show a significant improvement in term of accuracy and stability when compared against others methods.

This paper is organized as follows. Section 2 is dedicated to an overview of the necessary background and the introduction of our notations. We review in Section 3 previous methods on PTZ camera self-calibration. In Section 4, we describe our LMI-based self-calibration method. Section 5 summarizes the results of our experiments conducted using both simulated data and real images. Section 6 concludes our work.

2 Background

2.1 Stationary Cameras

In this paper we consider that the rotation axis of the camera passes through its optical center. This hypothesis hardly holds in real PTZ camera setups (see Fig. 1). However, although the deviation of the optical center from the rotation axis may be significant, it is usually considered insignificant with respect to the distance of the camera to the scene.

Neglecting the translational components of the motion of the camera allows to establish that the projections of any world point with coordinates X visible in two images taken by a rotating camera can be written as $x = KRX$ and $x' = K'R'X$. This leads to the following relationship

$$x' = K'R'R^{-1}K^{-1}x \tag{1}$$

where K and K' are the intrinsic parameters matrices respectively of the first and second image while R and R' are the rotation matrices. We recall that the intrinsic parameters matrix K is of the form:

$$K = \begin{bmatrix} f & s & u_0 \\ 0 & \lambda f & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

with f the focal length, λ the pixel's aspect ratio, s the skew and (u_0, v_0) the pixel coordinates of the principal point in the image.

From Equation (1), one can deduce that the matrix H representing the homography induced by the plane at infinity and given by

$$H = K'R'R^{-1}K^{-1} = K'RK^{-1} \tag{3}$$

can be calculated from point correspondences across the two considered images. Moreover, because $R^{-1} = R^T$, the following relationships can be deduced:

$$\omega'^* = H\omega^* H^T \tag{4}$$

$$\omega' = H^{-T}\omega H^{-1}. \tag{5}$$

Note that the relationships (4) and (5) are only dependent upon the homography of the plane at infinity and either the Image of the Absolute Conic (IAC) $\omega = K^{-T}K^{-1}$ or its dual $\omega^* = KK^T$ (DIAC). The scale factors appearing in (4) and (5) - due to the homogeneous nature of H - can be solved by normalizing H such that $det(H) = 1$.

2.2 Linear Matrix Inequalities

A Linear matrix inequality is an expression formulated as follows:

$$F(X) = F_0 + \sum_{i=1}^{n} x_i F_i \succ 0 \tag{6}$$

where, $X = (x_1, \cdots, x_n)$ is a vector of real scalars called the decision variables. $F_0 \cdots F_n$ are real symmetric matrices, and the symbol "$\succ$" stands for positive definiteness (all eigenvalues of $F(X)$ are positive). In this work we use LMI constraints to solve a convex optimization problem

$$\min_X c^T X \quad \text{s.t.} \quad F(X) \succ 0 \tag{7}$$

where c is a vector modeling the problem. The unfamiliar reader may refer to [8] for further details on LMI problems.

3 Related Work

Due to the popularity of stationary rotating cameras in practical visual systems, the problem of self-calibrating such cameras has attracted significant attention in the past two decades. For instance, in [9] and [10], the authors have proposed practical solutions to the self-calibration of Pan-Tilt (PT) cameras. These methods are however very sensitive to noise and can only be used to obtain some initial estimate to be refined through iterative schemes. In [11], Hartley proposed to use the homography between consecutive images to linearly calculate the unknown entries of the DIAC in the case in which the parameters of the camera remain unchanged. Once the DIAC is known, the camera parameters K can be easily obtained using a Cholesky decomposition. Note that it is also possible to solve the self-calibration problem linearly using additional knowledge about the rotation [12]. Moreover, Ji *et al.* [13] developed a strategy to remove the effect of the unknown but constant translational offset depicted in Fig. 1. However, all the above cited methods have in common that they do not take into account the case in which the internal parameters of the camera actually vary.

With a PTZ camera the intrinsic parameters change with each new image. In [7], Agapito *et al.* provide an elegant and effective reformulation of the problem in which the possible variation of internal parameters of the camera is taken into account. The authors have done so by expressing the problem in terms of the Image of the Absolute Conic (IAC) rather then on the DIAC. This has allowed to possibly express linear constraints on different parameters of the camera (these constraints are summarized in Table 1). In practice, the zero-skew constraint, i.e. $s = 0$, has provided poor results when used alone. In general, additionally constraining the aspect ratio $\lambda = 1$ is mostly used and allows to find the intrinsic parameters of the camera with only 3 images. The main drawback of this method lies in its sensitivity to noise often yielding a poor estimate of the IAC. If the IAC is not strictly positive definite, the Cholesky decomposition fails and the intrinsic parameters cannot be retrieved. This problem often occurs in presence of noise as demonstrated by the experiments reported in [14]. When the linear solution is successfully obtained, the optimal solution can be refined through bundle-adjustment [2] or by minimizing a cost function derived from (5) [7].

More recently, Agrawal *et al.* have proposed a LMI-based optimization approach for solving the calibration problem. For instance, in [15], a method of self-calibration of camera with fixed internal parameters using spheres is proposed. This was followed by a more general self-calibration approach based on

Table 1. Enforceable constraints on ω

Condition	Constraint	Type	Nb of Images
$s = 0$	$\omega_{12} = 0$	linear	5
$u_0 = v_0 = 0$	$\omega_{12} = \omega_{33} = 0$	linear	3
r=f_x/f_y	$\omega_{11} = r^2\omega_{22}$	linear	2
λ constant	$\omega^i_{11}/\omega^i_{22} = \omega^j_{11}/\omega^j_{22}$	quadratic	2

semi-definite programming [16]. The use of LMIs was later applied to the self-calibration of rotating cameras by Li *et al.* [14]. The use of LMIs in the context of camera calibration has many advantages. For instance, it allows to take into account the positive definiteness of the DIAC or IAC matrix representations and hence overcome the problem of Cholesky decomposition issues occurring with conventional algorithms. Furthermore, contrary to using local non-linear optimization methods, solving LMIs guarantees the convergence to a global solution. However, all linear equations need to be reformulated under matrix inequality forme. The work in [14] provides a LMI formulation of the PTZ camera self-calibration equations proposed by Agapito *et al.* in [7].

4 The Proposed Method

The present paper is an extension to Li *et al*'s LMI-based camera self-calibration method [14]. In their paper, the authors have reformulated the PTZ camera self-calibration problem given in [11] and [7] as an LMI optimization problem hence additionally enforcing the positive definiteness of the IAC ω. While the stability of the initial algorithm is highly improved, the results remain similar. The work in [14] does not use the full potential of the semi-definite programming. It is indeed possible to impose additional constraints to increase the performance of the algorithm. Furthermore, the majority of self-calibration approaches impose a hard constraint only on the first IAC: we can consequently talk about soft constraints for the others conics. In our method, we use LMIs in such a way to consider all images similarly. This results in a better description of the problem at hand and allows to outperform existing methods. The conditions we consider are as follows.

Condition 1: zero-skew $-\varepsilon < s < \varepsilon$

The absence of skew is commonly used in many applications because it is usually agreed that modern cameras have proper orthogonal pixels arrangements. The zero-skew can be expressed as the follows:

$$C_1^i = \begin{bmatrix} \varepsilon & \omega_{12}^i \\ \omega_{12}^i & \varepsilon \end{bmatrix} \succ 0.$$

Since we are dealing with the IAC, it is not difficult to enforce it to be close to zero by fixing ε very small (or by minimizing ε). In all the constraints, i is the image number.

Conditions 2 and 3: pixel aspect ratio $\delta_1 < \lambda < \delta_2$

When the camera rotates only around a single axis, information relative to the focal length on the other axis is lost. In this paper, we refer to those kind of motions as degenerate. To overcome this problem, we proposed a set of constraints directly embedded in our LMI system to enforce λ.

In the case of PT cameras, we offer the possibility to fix the pixel aspect ratio between two adjustable bounds, for instance $0.75 < \lambda < 1.25$, which is reasonable and allows to increase the robustness.

For PTZ cameras, our approach allows to enforce λ to be a chosen value. This constraint allows to deal with any kind of rotating motion (even degenerate). Then, like most papers about self-calibration of zooming cameras, we can consider the unit aspect ratio constraint ($\lambda = 1$) if no a priori knowledge is available. However, because λ is constant regardless of how the other parameters may vary, it is possible to calibrate the camera once at any level of zoom to recover its true value. The two bounds can be enforced separately using LMI constraints:

upper bound	lower bound
$\frac{f_y}{f_x} < \delta_2$	$\frac{f_y}{f_x} > \delta_1$
$C_2^i = \omega_{11}^i - \omega_{22}^i \delta_2^2 \succ 0$	$C_3^i = \omega_{22}^i \delta_1^2 - \omega_{11}^i \succ 0$

Conditions 4 and 5 : Principal point close to the center of the image

As shown in [7], the principal point is not strongly constrained so it is very sensitive to noise. According to [17] we can admit that the principal point (u_0, v_0) is close to the center of the image (x_c, y_c) even with a zooming camera. In [18], Hartley *et al.* have demonstrated that a relaxation of the principal point leads to a better estimation of the focal length. A range of variation can be imposed:

$$(u_0 - x_c)^2 < d^2 \rightarrow d^2 - (u_0 - x_c)^2 > 0 \tag{8}$$

$$(v_0 - y_c)^2 < d^2 \rightarrow d^2 - (v_0 - y_c)^2 > 0 \tag{9}$$

where d is the maximum distance (in pixels) between the principal point and the center of the image. The previous equations can be reformulated using only the entries of ω:

$$d^2\omega_{11} - (\omega_{13} - x_c\omega_{11}) > 0 \rightarrow \frac{d^2}{f_x^2} - \frac{(u_0 - x_c)^2}{f_x^2} > 0 \tag{10}$$

$$d^2\omega_{22} - (\omega_{23} - y_c\omega_{22}) > 0 \rightarrow \frac{d^2}{f_y^2} - \frac{(v_0 - y_c)^2}{f_y^2} > 0. \tag{11}$$

We hence obtain two additional terms in our LMI system:

$$C_4^i = d^2\omega_{11}^i - (\omega_{13}^i - x_c\omega_{11}^i) \succ 0$$
$$C_5^i = d^2\omega_{22}^i - (\omega_{23}^i - y_c\omega_{22}^i) \succ 0$$

This constraint needs to be used carefully. If the fixed bounds are too restrictive and the principal point is in fact out of this range, then the estimation of the others parameters will be affected. Taking bounds within $\pm 10\%$ from the center of the image remains a reasonable constraint for many cameras, but fixing the principal point within the image bounds generally suffices.

Condition 6: positive definiteness of the IAC $C_6^i = \omega^i \succ 0$

Conditions 1 to 6 have to be enforced for every single IAC (so for every new image). This forms a LMI system where all ω are affected by hard constraints. Finally, we minimize the error in (5) (for every homography) using the spectral norm:

Condition 7: $\omega^{i-1} = H^{-T}\omega^i H^{-1}$

$$C_7^i = \begin{bmatrix} t_i I & \omega^{i-1} - H^{-T}\omega^i H^{-1} \\ (\omega^{i-1} - H^{-T}\omega^i H^{-1})^T & t_i I \end{bmatrix} \succ 0$$

with I the 3×3 identity matrix and t_i a scalar to minimize. This last condition allows to link all conics together, they are then all inter-dependant. The only difference when we wish to self-calibrate a PT camera is that $\omega^i = \omega^1$ so the same IAC for all views needs to be considered.

Finally, the full optimization problem can be summarized as follows:

$$\min_{\omega, t_1 \cdots t_n} \sum_{i=0}^{n} t_i \tag{12}$$

$$\text{subject to} \quad \begin{bmatrix} C_1^1 & 0 & \cdots & 0 \\ 0 & C_2^1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C_7^n \end{bmatrix} \succ 0 \tag{13}$$

5 Experiments

5.1 Evaluation with Synthetic Data

To assess the performances of the proposed algorithm, a series of tests has been done by randomly generating 5000 3D points inside a cube of dimensions 1000x1000x1000. Although our 3D scene consists of 5000 points, due to a limited field of view, less than one hundred are visible in pairs of views and hence used to recover the homographies. The synthetic camera is located at the center of the point cloud where it will performs random rotations between $-30°$ and $+30°$. 3D points are then projected onto the image plane (of 640x480 pixels). The inter-image homography is linearly computed without any refinement. In order to test the robustness of the different algorithms, a random noise is added on the pixels positions. Different noise levels are used and 1000 trials are performed for each noise level. All presented results are computed using a set of 3 homographies for PTZ cameras and 2 for PT cameras.

Fixed Parameters. In the case of a PT camera, the evaluation is done using the following arbitrary intrinsic parameters: $f = 900$, $\lambda = 0.8889$, $u_0 = 325$ and $v_0 = 240$. Here 3 algorithms are compared: the method described by Hartley in [11], the LMI approach of Li *et al.* [14] and our approach.

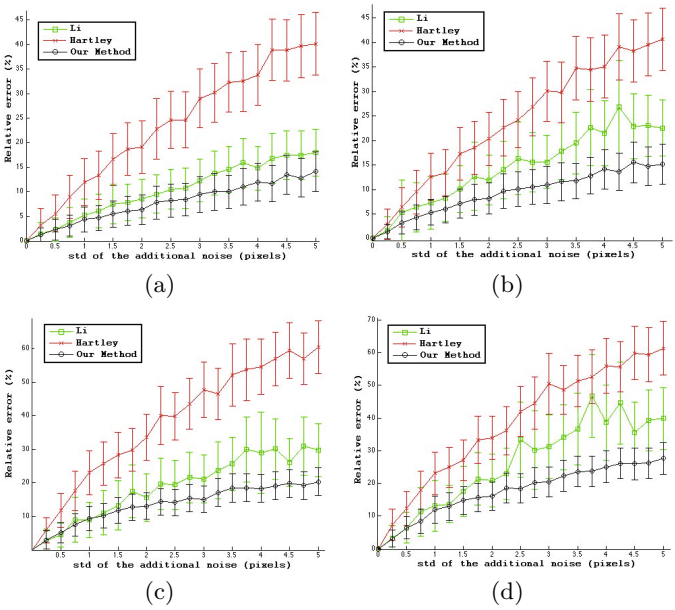


Fig. 2. Mean and standard deviation error for synthetic evaluation of the algorithms with fixed intrinsic parameters (a)f_x (b)f_y (c)u_0 (d)v_0

The results shown on figure 2 are obtained with 3 rotations which is the minimum needed for the others methods (yet our method required only 2). Here our algorithm is configured as follows : $0.75 < \lambda < 1.25$, the principal point inside the image and $s = 0$.

In this series of tests our strategy significantly outperforms the others methods due to multiple reasons. First, we are using hard constraints on every single conic, furthermore the central point is restricted inside the image and the pixel aspect ratio is fixed between two bounds. All these elements leads to a better robustness of the system even without knowing strong a priori on the camera.

Varying Parameters. In the following tests the camera performs a zoom sequence, then the focal length as well as the principal point undergoes big changes. In the case of a PTZ camera we compare our method with the techniques described in [7] and [14]. For convenience, we set the aspect ratio to one: $\lambda = 1$. Our configuration just considers the unit pixel aspect ratio, zero skew and a principal point inside the image. Without adding extra a priori knowledge, we obtained better results than existing methods (see figure 3). The strong inter-dependence between all constrained IACs justified this strong improvement. Moreover, taking more restrictive bounds can lead to even more robust results. However, the minimum number of images required is the same as in [7,14].

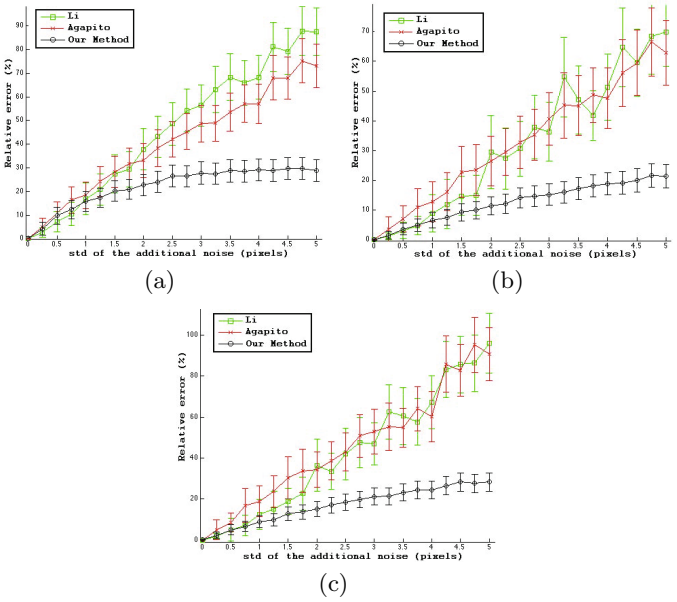


Fig. 3. Mean and standard deviation error for synthetic evaluation of the algorithms with varying intrinsic parameters and $\lambda = 1$ (a)f (b)u_0 (c)v_0

Influence of the Constraint on the Central Point. In the case of PTZ camera it is preferable to keep λ fixed for a better stability, so only the restricted position of the principal point will be important in the estimation of internal parameters. The only constraint on the principal point proposed in the literature is to fix it to a known value. We suggest a more flexible approach by limiting its location to a range of variations. The results presented in figure 4 show the influence of the constraints applied on the position of the principal point on the estimation of the focal length.

Even if the pixel aspect ratio and the principal point are known, their true values will drift in presence of noise. It means that imposing a strict value will lead to an accumulation of error affecting unknown parameters. Contrariwise, leaving the principal point totally free could make the system converging to a wrong minima. So forcing the position of the principal point inside a bound is a very good compromise in order to overcome bad converge and to balance the error. The curves in figure 4 validate those assumptions, in presence of noise the principal point between bounds provides a better accuracy than a free principal point inside the image and a fixed one.

5.2 Tests Using Real Data

Using a PT Camera. The following tests have been done using a simple webcam *Logitech Quickcam Sphere AF Web camera - pan / tilt*. This camera provides

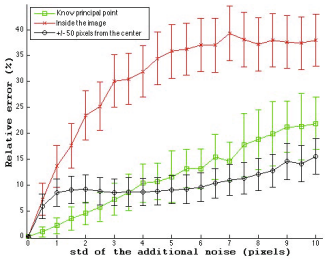


Fig. 4. Influence of the constraint on the principal point on f

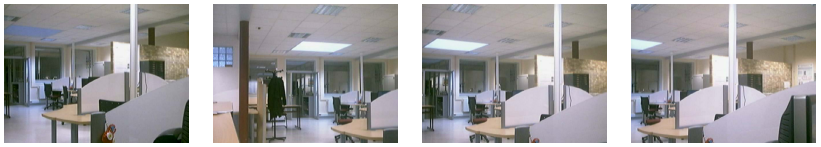
Table 2. Result obtained with real images (from PT camera) using different methods

		fx		U_0		V_0		λ	
		pixels	Error (%)	pixels	Error (%)	Pixels	Error (%)		Error (%)
GT : Bouguet		904.6	0	327.6	0	277.5	0	1.0007	0
Hartley	4 images	959	5.6726	365	11.4164	265.2	4.4324	0.9265	7.4148
Our method	2 images	919.3	1.6250	325.7	0.5800	243.48	12.2595	1.0511	5.0365
	4 images	913.4	0.9728	321.22	1.9475	232.66	16.1586	1.0103	0.9593

a noisy image of size 640x480 pixels. An initial calibration using Bouguet's toolbox [19], provides an estimation of intrinsic parameters of the camera, that is used as ground truth.

We auto-calibrate the camera successively with two and four images (see the results on table 2). In this test we do not consider any a priori about the camera: so the pixel aspect ratio is $0.75 < \lambda < 1.25$ and the principal point search inside the image. The comparison between our method and a conventional approach clearly shows a strong improvement in the estimation of almost all camera's parameters.

Using a PTZ Camera. Evaluating the performance of an auto-calibration algorithm for this particular type of camera is difficult because no trustful ground truth is available. This is the reason why we project the images onto the spherical unified model [20] to compare the quality of the mosaic obtained from a set of images. In fact, this modelling allows to represent images taken with any single view point camera onto a unitary sphere. However, this projection needs



(a) (b) (c) (d)

Fig. 5. Images obtained from a PT camera

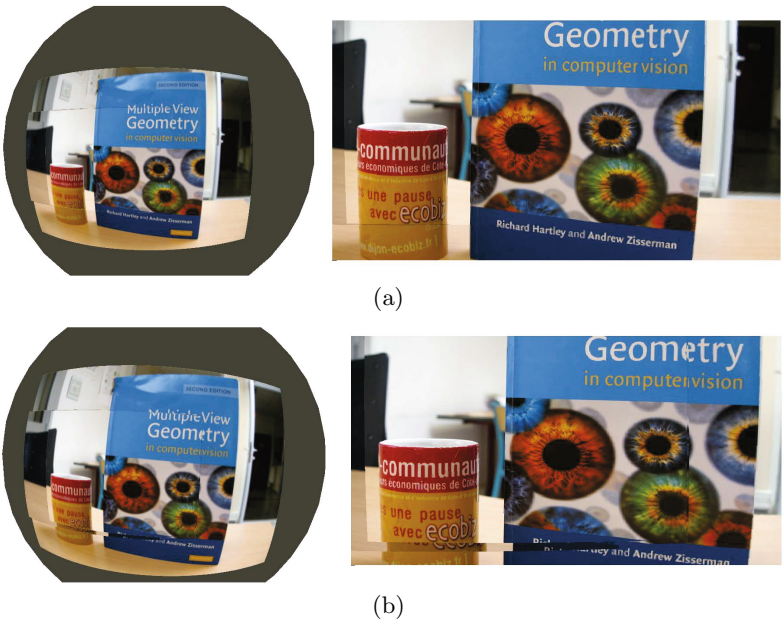


Fig. 6. Spherical multi-resolution mosaic of 5 images obtained with (a) Our method (b) [14]. The left column is the full spherical view of the panorama while the right one is a detailed view on a particular ROI.

a very good estimation of internal and external parameters. So, the quality of the resulting spherical image will be related to the accuracy of the PTZ camera calibration. The panoramas obtained are presented on figure 6, those computed using our method are much more accurate and contain less artefacts.

6 Conclusion

In this paper, a new approach using LMI is presented, where strong tunable constraints are applied on every conic. We proved that the LMI can be a very reliable tool for convex optimization problems using a priori knowledges. The semi definite programming is a not commonly used strategy in the field of computer vision, where it could be very useful for the resolution of many problems. Experimental results show that our constraints (especially those on the principal point) provide a better robustness and allow an accurate estimation of all the parameters of a PTZ camera.

Acknowledgements. This work was supported by DGA (Direction Générale de l'Armement). The first author would like to thank D. P. Paudel for all helpful discussions.

References

1. Lalonde, M., Foucher, S., Gagnon, L., Pronovost, E., Derenne, M., Janelle, A.: A system to automatically track humans and vehicles with a ptz camera. In: Proc. SPIE, vol. 6575, p. 657502 (2007)
2. Sinha, S.N., Pollefeys, M.: Pan–tilt–zoom camera calibration and high-resolution mosaic generation. Computer Vision and Image Understanding 103, 170–183 (2006)
3. Tanawongsuwan, R., Stoytchev, A., Essa, I.A.: Robust tracking of people by a mobile robotic agent (1999)
4. Zhang, Z.: A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 1330–1334 (2000)
5. Sturm, P.: Self-calibration of a moving zoom-lens camera by pre-calibration. Image and Vision Computing 15, 583–589 (1997)
6. Faugeras, O., Luong, Q.T., Maybank, S.: Camera Self-Calibration: Theory and Experiments. In: Sandini, G. (ed.) ECCV 1992. LNCS, vol. 588, pp. 321–334. Springer, Heidelberg (1992)
7. Agapito, L., Hayman, E., Reid, I.: Self-calibration of rotating and zooming cameras. International Journal of Computer Vision 45, 107–127 (2001)
8. Boyd, S., El Ghaoui, L., Feron, E., Balakrishnan, V.: Linear Matrix Inequalities in System and Control Theory. Studies in Applied Mathematics, vol. 15. SIAM, Philadelphia (1994)
9. Du, F., Brady, M.: Self-calibration of the intrinsic parameters of cameras for active vision systems. In: Proceedings of the IEEE Computer Society Conference on CVPR 1993, pp. 477–482. IEEE (1993)
10. Basu, A.: Active calibration: Alternative strategy and analysis. In: Proceedings CVPR 1993, pp. 495–500. IEEE (1993)
11. Hartley, R.I.: Self-calibration of stationary cameras. International Journal of Computer Vision 22, 5–23 (1997)
12. Stein, G.P.: Accurate internal camera calibration using rotation, with analysis of sources of error. In: ICCV, pp. 230–236. IEEE (1995)
13. Ji, Q., Dai, S.: Self-calibration of a rotating camera with a translational offset. IEEE Transactions on Robotics and Automation 20, 1–14 (2004)
14. Li, H., Shen, C.: An lmi approach for reliable ptz camera self-calibration. In: Proceedings of the IEEE International Conference on Video and Signal Based Surveillance, AVSS 2006. IEEE Computer Society, Washington, DC (2006)
15. Agrawal, M., Davis, L.S.: Camera calibration using spheres: A semi-definite programming approach. In: IEEE International Conference on Computer Vision, vol. 2 (2003)
16. Agrawal, M.: On automatic determination of varying focal lengths using semidefinite programming. In: Proceedings IEEE International Conference on Image Processing, Singapore (2004)
17. Willson, R., Shafer, S.: What is the center of the image? Journal of the Optical Society of America A 11, 2946–2955 (1994)
18. Hartley, R.I., Kaucic, R.: Sensitivity of Calibration to Principal Point Position. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part II. LNCS, vol. 2351, pp. 433–446. Springer, Heidelberg (2002)
19. Bouguet, J.Y.: Camera calibration toolbox for Matlab (2008)
20. John, B., Henry, A.: Issues on the geometry of central catadioptric image formation. In: CVPR, pp. 422–427 (2001)

Fast 3D Surface Reconstruction from Point Clouds Using Graph-Based Fronts Propagation

Abdallah El Chakik, Xavier Desquesnes, and Abderrahim Elmoataz

UCBN, GREYC - UMR CNRS 6972, 6. Bvd Marechal Juin, 14050 Caen, France

Abstract. This paper proposes a surface reconstruction approach that is based on fronts propagation over weighted graphs of arbitrary structure. The problem of surface reconstruction from a set of points has been extensively studied in the literature so far. The novelty of this approach resides in the use of the eikonal equation using Partial difference Equation on weighted graph. It produces a fast algorithm, which is the main contribution of this study. It also presents several examples that illustrate this approach.

1 Introduction

The main goal of this work is to propose a fast surface reconstruction method from point clouds, using a graph-based representation. This reconstruction is performed using a Partial difference Equation (PdEs) fronts propagation algorithm based on weighted graph.

Brief Literature Overview. Surface reconstruction from point clouds is an important problem in geometric modeling. Given a set of points $X = \{x_1, x_2, .., x_n\} \subset \mathbf{R}^n$ sampled from some unknown surface S, the surface reconstruction problem is to construct a surface $\hat{S}$ from the observed data X such as $\hat{S}$ approximates S.

Most surface representation techniques for point clouds reconstruction methods are classified into two categories, namely explicit and implicit methods. Explicit surface representations prescribe the surface location and geometry in an explicit manner and are mainly based on Delaunay triangulations or dual Voronoï diagrams. A popular technique is to construct a polyhedral surface from the input set of points using the Voronoï diagram [8]. Implicit surface representations embed surfaces as a co-dimension one level set of a scalar-valued function. In [5], a variational level set method was proposed, it introduced a distance-based energy functional, solved by level set method. Recently, this work was extended in [6,7]. In this paper, we focus on the level set transcription on weighted graph.

Level Set Method. The level set formulation to describe a curve evolution has been introduced by Osher-Sethian [1], and is used in many works for 3D surface reconstruction based on front propagation. In [4], Claisse and Frey solved the surface reconstruction problem using the following equation:

$$\frac{\partial_\phi}{\partial_t}(t,x) = |\nabla_\phi(t,x)|(\beta k(\phi)(t,x) + (\alpha d)(x)), \tag{1}$$

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 309–320, 2013.

where $k(\phi)(t,x)=\nabla.n(\phi)(t,x)=\left(\nabla.(|\frac{\nabla_u}{|\nabla_u|})t(tx)\right)$ is the local mean curvature of the surface considered, α and $\beta \in \mathbf{R}$ and $d(x)$ is here the distance between x and the initial point set in $\mathbf{R}^3$. An important drawback of the level set approach stems from the expanse by embedding the front in $\mathbf{R}^d$ as the level set of $d+1$ dimensional function. Considerable computational labor is required per time step.

Contribution. We propose in this work a different and efficient approach that reduces the computational time without loss of precision. Our contribution is the transcription of the problem from $\mathbf{R}^3$ to a weighted graph. Indeed, any set of discrete data can be modeled as a weighted graph $G = (V, E, w)$ where V is the set of vertices that represents the data, E is the set of weighted edges and w is a weight function that represents the interactions between the data (see Section 2 for more details). We demonstrate that the graph representation allows to significantly reduce the amount of space points to be treated by the different surface reconstruction algorithms, thus increasing their performance. Then we extend the previously introduced PdE based fronts propagation method on weighted graphs in [3] to the 3D surface reconstruction problem. This method is based on the resolution of the following equation : $\mathcal{F}(u)\|(\nabla_w^- T)(u)\|_p = 1$, where ∇_w^- is an upwind discrete weighted gradient on a graph, T is the arrival time function of the fronts and $\mathcal{F}$ is the propagation speed function.

Paper Organization. The rest of this paper is organized as follows. Section 2 presents a general definition of Partial difference Equations on weighted graph. It also describes our fronts propagation method on weighted graphs. Section 3 presents our graph-based surface reconstruction method. Section 4 presents some experiments. Finally, Section 5 concludes this paper.

2 Partial Difference Equations on Weighted Graphs

We begin briefly by reviewing some basic definitions and operators on weighted graphs.

Notions and Definitions. We assume that any discrete domain can be modeled by a weighted graph. Let $G = (V, E, w)$ be a weighted graph composed of two finite sets : $V = \{u_1, ..., u_n\}$ of n vertices and $E \subset V \times V$ a set of weighted edges. An edge $(u, v) \in$ E connects two adjacent vertices u and v. The weight w_{uv} of an edge (u, v) can be defined by a function $w : V \times V \to \mathbf{R}^+$ if $(u, v) \in E$, and $w_{uv} = 0$ otherwise. We denote by $N(u)$ the neighbor of a vertex u, i.e. the subset of vertices that share an edge with u.

Let $f : V \to \mathbf{R}$ be a discrete real-valued function that assigns a real value $f(u)$ to each vertex $\mathbf{u} \in \mathbf{V}$. We denote by $\mathscr{H}(\mathbf{V})$ the Hilbert space of such functions.

Operators on Weighted Graphs. For a better comprehension of the next Section, we now quickly recall some operators on weighted graphs as they are defined in [3,14]. Considering a weighted graph $G = (V, E, w)$ and a function $f \in \mathscr{H}(\mathbf{V})$, the *weighted discrete partial derivative operator* of $\mathbf{f}$ is:

$$(\partial_v f)(u) = \sqrt{w_{uv}}(f(v) - f(u)) \tag{2}$$

Based on this definition, two weighted directional difference operators are defined. The weighted directional external and internal difference operators are respectively :

$$\begin{aligned}(\partial_v^+ f)(u) &= \sqrt{w_{uv}}(f(v) - f(u))^+ and \\ (\partial_v^- f)(u) &= \sqrt{w_{uv}}(f(v) - f(u))^-\end{aligned} \tag{3}$$

with $(x)^+ = max(0, x)$ and $(x)^- = -min(0, x)$.
The weighted gradient of a function $f \in \mathscr{H}(\mathbf{V})$ at vertex u is the vector of all edge directional derivatives:

$$(\nabla_w f)(u) = (\partial_v f(u))_{v \in V}^T \tag{4}$$

And the weighted morphological external and internal gradient $(\nabla_w^+ f)(u)$ and $(\nabla_w^- f)(u)$ are:

$$(\nabla_w^\pm f)(u) = \left(\left(\partial_v^\pm f\right)(u)\right)_{v \in V}^T. \tag{5}$$

2.1 Front Propagation on Weighted Graphs

In this section we will present the fronts propagation approach on weighted graphs.

Let $G = (V, E, w)$ be a weighted graph. A front evolving on G is defined at initial time as a subset $\Omega_0 \subset V$, and is implicitly represented by a level set function ϕ_0 such that ϕ_0 equals 1 in Ω_0 and -1 on its complementary.
Then, the front propagation is described by the following equation

$$\begin{cases} \frac{\partial \phi}{\partial t}(u) &= (\mathcal{F}(u) \| (\nabla_w \phi)(u) \| \\ \phi_0(u) &= \phi_0 \end{cases} \tag{6}$$

with $\mathcal{F} \in \mathscr{H}(\mathbf{V})$, and $w : V \times V \to \mathbf{R}^+$ is the weighted function. Only considering the case $\mathcal{F} \geq 0$, and with $\phi(u, t) = t - T(u)$, The authors have shown in [3] that previous equation can be rewritten as

$$\begin{aligned}\frac{\partial_\phi(u, t)}{\partial t} &= \mathcal{F}(u) \| (\nabla_w^+ (t - T))(u) \|_p \\ &= \mathcal{F}(u) \| (\nabla_w^- T)(u) \|_p = 1, \\ &\| (\nabla_w^- T)(u) \|_p = P(u)\end{aligned} \tag{7}$$

where $P(u) = 1/\mathcal{F}(u)$.

This equation is the stationary version of the level set equation (6) that corresponds to the well-known eikonal equation and $T : V \to \mathbf{R}$ is the arrival-time function that associates the arrival time of Γ to each vertices of V. This function can also be considered as a distance function that provides the distance between each vertex u and Ω_0.

In [3], the authors have proposed several numerical schemes to solve such equations for different $\mathcal{L}_p$ norms. They also presented an efficient algorithm inspired from Fast Marching that allows to compute the propagation and the arrival time of many fronts evolving on a graph. In this case, equation (7) is associated with a label function that mark each vertex u with the label of the first front that reach u. The label function $L : V \to [0, K]$ is initialized as

$$L(u) = \begin{cases} i \in [1, K] & \text{if u belongs to front } i \text{ at initial time} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where K is the number of fronts. Interested readers should refers to [2,3] for more details. Such algorithm has been successfully used for geodesic distance computation on weighted graphs, image segmentation and data clustering. In the next section, we will propose a new surface reconstruction method based on this algorithm.

3 Method

Our reconstruction method consists of completing the initial point clouds with a very dense set of points that will be part of the resulting reconstructed surface. The completion is performed by selecting new points from a very dense set of candidate points, generated in the neighborhood of initial points, and including initial points. Our approach considers the set of candidate points as a weighted graph (constructed from the set) and the surface to be reconstructed as a subset of the vertices of this graph. Working on this graph, and by analogy with the level set method, the surface to be reconstructed is considered as the interface between two inner and outer fronts evolving on the whole graph. These two fronts are driven according to a potential field that controls their propagation speed, and defined on the graph vertices such that the two fronts collapse on the object boundary.

Graph Construction. The first step extends the initial point clouds in order to generate candidate points and constructs the associated weighted graph. In order to precisely fill in the holes of the object, we need a very dense additional point clouds in the neighborhood of the initial points. But high density point clouds penalizes the computational efficiency due to the high number of candidates to be treated. For this reason, we propose to use an adaptive approach that allows to add a very dense additional points only where it is necessary (near the initial points), and very sparse additional points elsewhere.

We consider the initial point clouds to be represented by a set of points $X = \{x_1, x_2, ..., x_n\} \subset \mathbf{R}^n$. This initial point clouds X is extended with new points as follows : Let C be the candidate points that are regularly added to the neighborhood of the initial points, such that the density of the new points is very high near the initial points and decreases as we move away. This is performed by the adaptive triangulation method proposed by [4] that produces a triangulated

adapted mesh which vertices include initial and candidates points. Let X' be the joint set of initial and candidate points ($X' = X \cup C$).

We denote $G(V, E, w)$ the weighted graph constructed from the previously obtained mesh. The set of vertices V represent the points of X', such that the vertex $v_i \in V$ represents the point $x'_i \in X'$. Let V_0 be the set of vertices that corresponds to the initial points (X). The set of edges E is given by the triangulated mesh edges. The weight function w defines a similarity between the two vertices of each edge. This similarity is based on the position in space of the associated points, we have $w(v_i, v_j) = -exp(d(x'_i, x'_j)^2/\sigma^2)\, \forall (v_i, v_j) \in E$. With this definition, the weight function holds the spatial relations of the extended point clouds X'.

We will now present the fronts initialization and the potential field definition, both based on the same distance map $\mathcal{D}$ computed from initial point clouds. This distance is computed as follows.

Distance Map. The distance map is a function $\mathcal{D} : V \to \mathbf{R}^+$ that associates each vertex u of G with the distance between the set V_0 and u.

The distance map is computed using the Fast Marching algorithm on graphs (7)(see Sec.2), for a single front initialized on initial points (i.e., $\Omega_0 = V_0$), and with constant potential function ($P = 1$).

We recall that this algorithm produces both a label function (L) for fronts propagation and the associated arrival-time function T that can be considered as a distance map between Ω_0 and each vertex $u \in V$. In this case, function T provides distance from the single set V_0 and we have $\mathcal{D} = T$.

Inner and Outer Fronts Initialization. Inner and outer fronts are initialized equidistantly from the initial point clouds, using the previous distance function $\mathcal{D}$. The inner front Γ_i is initialized by the subset Ω_0^i of vertices that lies inside the object and at a distance k from the initial points. We have $\Omega_0^i = \{u \mid \mathcal{D} = k \pm \varepsilon \text{ and } u \text{ inside}\}$. Similarly, the outer front Γ_o is initialized by the subset Ω_0^o of vertices that lies outside the object and at a distance k from the initial points. We have $\Omega_0^o = \{u \mid \mathcal{D} = k \pm \varepsilon \text{ and } u \text{ outside}\}$.

In the case where the fronts are not equidistant, the nearest front from the surface to be reconstructed will always be favored and fronts may collapse far from the object surface.

Potential Field. In the case where the object has thin parts, the inner front will be favored and the fronts will collapse outside the object. To prevent this problem, we introduce a potential field that controls the fronts propagation such that the fronts moves very slowly near the initial points (near the object boundary) and move faster elsewhere. The better potential function is given by the distance map function $\mathcal{D}$. Indeed the distance map is almost null near initial points which guarantees that the fronts will be slowly propagated or stopped near the surface to be reconstructed. Then the potential function is defined as $P = \mathcal{D}$.

Surface Reconstruction. Once the inner and outer fronts are initialized, we set a label to each graph vertices as follows : the vertices belonging to Ω_0^i are

labeled by 1 ($L = 1$) and the vertices belonging to Ω_0^o are labeled by 2 ($L = 2$) and the rest of vertices are labeled by 0 ($L = 0$).

Then we propagate those labels on the graph using the Fast Marching algorithm on graphs (7) presented in Sec.2. The algorithm is initialized as $\Omega_0 = \Omega_0^i \cup \Omega_0^o$, and the potential function is given by $P = \mathcal{D}$.

Due to the potential field that slows down the fronts on the neighborhood of the object boundary, both fronts collapse on this boundary. Then, the vertices that lies inside the object are labeled by 1 and the vertices that lies outside the object are labeled by 2 and the resulting reconstructed surface is the vertices belonging to the inner labels and have at least one neighborhood belonging to the outer labels.

Remark. We can apply a spacial regularization on the selected vertices to adjust the vertices positions to top-up the resulting surface smoothness.

Complexity. If the graph is totally connected, the complexity of the proposed model is $O(N^3)$, where N is the number of nodes in the graph.

4 Experiments

This section demonstrate the speed and the robustness of our surface reconstruction approach by applying it on different examples. First, we explain our approach on a 2D points set for better comprehension, we show our approach results on 3D points set and we compare our approach results with some other approaches.

Figure 1 shows the initial 2D points and the adapted mesh created. The initial points number is 790 points, the adapted mesh produce 15451 points and 30868 triangles. The adapted mesh will be transformed into a weighted graph. We define a single label on the initial vertices (vertices to be reconstructed) that will propagate on the whole graph and compute the distance map of each point to the nearest initial points using the equation (7), figure 2 shows the distance map computed.

Once our distance map is computed, we define inner and outer labels using this distance map. Figure 3 shows the labels and the object reconstructed.

Figure 4 shows a cut of the face adapted mesh. One can see the high points density near the initial points and the low points density moving away from the initial points. The initial points number is 67206, the adaptive triangulation produces 952869 points. Figure 5 shows the results of our reconstruction approach, one can see the reconstructed objects smoothness. The resulting reconstructed object : face (points : 579992, triangles : 1063855), dragon (points : 451275, triangles :703488), duck (points : 498779 , triangles : 836483).

We tested our approach on a voxels image example to prove that our approach deals with multiple data types.We show our approach result on the Stanford dragon point clouds [15]. We transformed our point clouds to a voxels 3D image and constructed our weighted graph $G(V, E, w)$ such that a single vertex is associated with each point in $\mathbf{R}^3$. The weight function is constant ($w(u, v) = 1$, $\forall (u, v) \in E$).

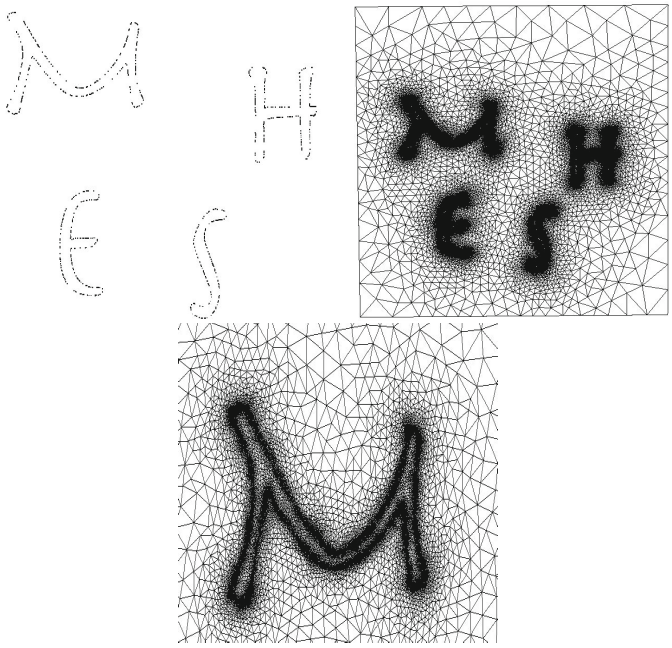


Fig. 1. The adapted mesh. Left : initial points; Right : the adapted mesh, this mesh contain 15451 points and 30868 triangles.

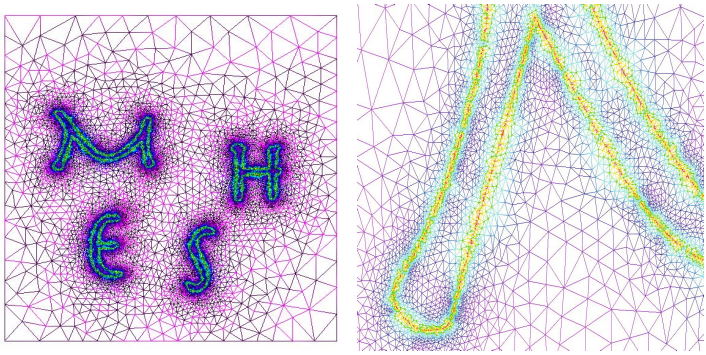


Fig. 2. Distance map computed by propagating a single front from the initial points on the whole graph. The colors represent each point distance value from the intial points. The initial points distance value is red.

Figure 6 illustrate the result of our surface reconstruction approach on the 3D voxels image (grid size : 240×169×107). The surface reconstruction algorithms computational time is 65 seconds for this example. One can see the time computational difference between this examples and the adapted meshes examples, thanks to the adaptive triangulation thats minimize the weighted graph vertices number.

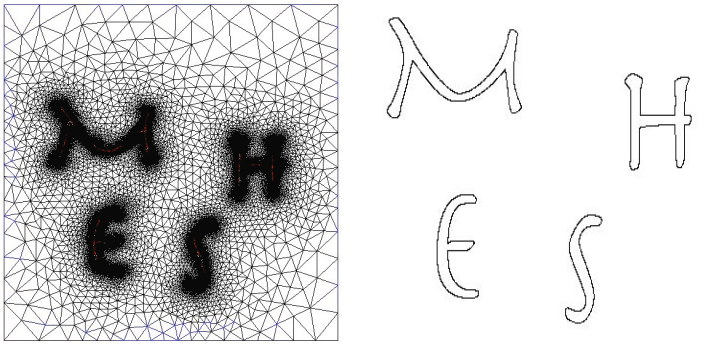


Fig. 3. Left : the labeled mesh. We assign to each label a color, the red color represent the vertices belonging to inner label ($L = 1$), The vertices colored by blue belonging to the outer label ($L = 2$), The rest of vertices colored by black are labeled by 0 ($L = 0$) and represent the propagation space of the inner and outer labels; Right : our approach surface reconstruction result that produce 6421 connected vertices.

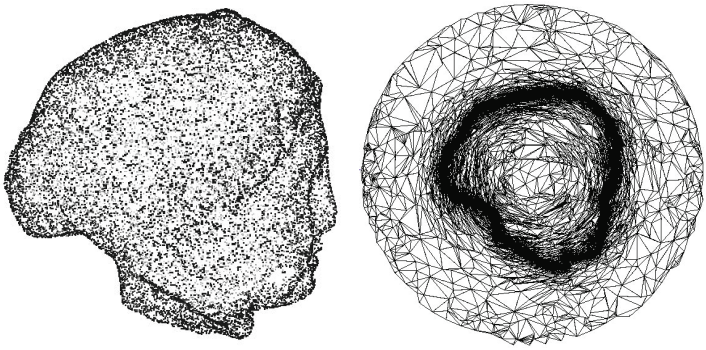


Fig. 4. Left : initial face point clouds (venus) to be reconstructed that contain 67026 points; Right : a cut of the face adapted mesh, the full face adapted mesh contain 952869 points and 5147018 triangles

Comparison with Some other Approaches. Following the quantitative information given in [9], we compare the performances of our reconstruction approach on the Stanford bunny point clouds [15] with different methods [10, 11, 12, 13], figure 7 shows our surface reconstruction result on this example. Table 2 presents the difference between our approach and the other approaches. In terms of computation time, our method is comparable to that of the MPU method, which is one of the fastest geometrically-adaptive reconstruction methods according to [12]. The Power Crust method is about 10 times slower, and the Poisson method is about 4 times slower than our approach. The FFT oppe et al. and methods are about 2 time slower but the first suffers from large memory and the second produce some holes in the final reconstructed object. In term of reconstruction quality, The method by Hoppe et al. and the Power Crust

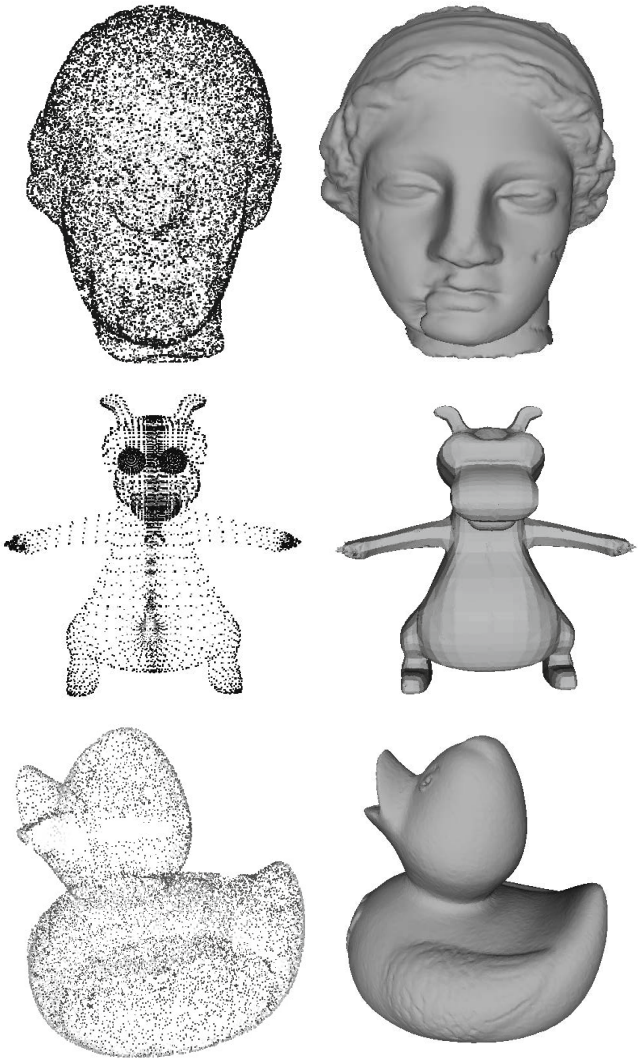


Fig. 5. Surface reconstruction result using our approach. First column : initial point clouds; Second column : our approach surface reconstruction results. One can see that the holes in the surface (for the duck example) are reconstructed.

method generate a smooth surface with some holes that are still visible, The MPU method provide a smooth surface without holes, but with some artefacts. The FFT, Poisson and our method accurately reconstruct the surface of the bunny. In terms of peak memory usage, our approach have a reasonable memory usage which corresponds to the amounts of the memory to store the dense graph constructed from the adapted mesh.

Table 1. Our method time computational for different point clouds examples

Point Cloud	Adapted points number	Reconstruction time
cart	15451	**1.50 seconds**
duck	943137	**38.38 seconds**
Face	952869	**38.98 seconds**
Dragon	991583	**39.8 seconds**

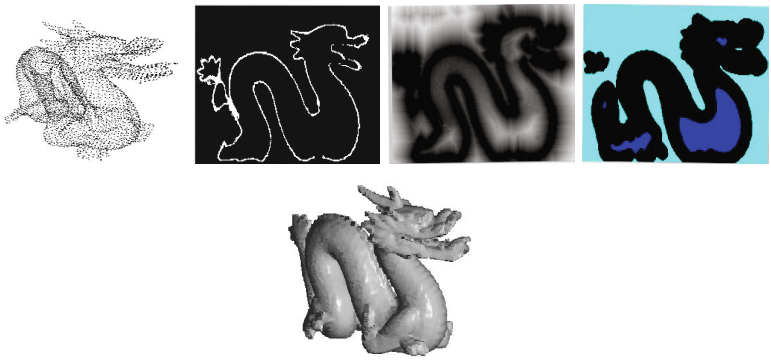


Fig. 6. Surface reconstruction example on 3D voxels image ; Left to right : dragon initial point clouds, cut of dragon voxels 3D image, distance map computed by propagation a single front starting from the initial points and evolving on the whole graph, labels definition (the blue color represent the inner labels and the green color represent the outer labels and the the black color represent the space label propagation), the surface reconstruction result of the dragon

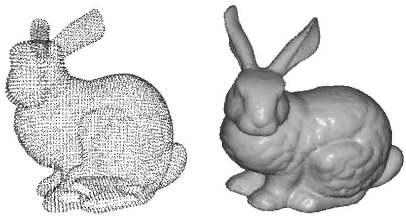


Fig. 7. Our approach result on Stanford bunny example. Left : the bunny initial point clouds; Light : our approach reconstruction result.

Advantages. This method offers several advantages. First of all, for adapted meshes, the graph representation allows to have very dense additional points only where it is necessary (near initial points), and very sparse additional points elsewhere. This significantly reduces the number of points to be treaded by the algorithm, and thus the computational time. On the contrary, traditional methods using three dimensional regular grids of voxels have to choose between precision (with a very dense grid) and computational efficiency. Second, all processes are performed using a single general algorithm which allows to deal with many

Table 2. Different methods Computational time for the Stanford bunny. Computational time (seconds), peak-memory usage (mega-bytes) and number of triangles of the reconstructed surfaces of three range data sets.

Method	Time	Peak memory	Triangles
Power Crust	504	2601	1,610,433
Poisson method	188	283	783,127
FFT method	93	1700	1,458,356
Hoppe et al.	82	230	630,345
MPU	78	421	2,121,041
Our method	45	980	2,759,146

fronts on weighted graphs of arbitrary topology, and can be used for many types of data (not only 3D point clouds). Finally, no spatial discretization is needed, thanks to the equations being directly expressed in a discrete form.

5 Conclusion

In this paper, we proposed a point clouds fast surface reconstruction algorithm based on fronts propagation over weighted graphs. We show that our approach deals with multiple types of data and produces robust results. In addition, this method is fast and does not need large memory requirements.

References

1. Sethian, J.A.: Level Set Methods and Fast Marching Methods. Some Fine Journal (1999)
2. Desquesnes, X., Elmoataz, A., L'ezoray, O.: PDEs Level Sets on Weighted Graphs. In: International Conference on Image Processing (IEEE), vol. 4(2), pp. 1–25 (2011)
3. Desquesnes, X., Elmoataz, A., Lézoray, O., Ta, V.-T.: Efficient Algorithms for Image and High Dimensional Data Processing Using Eikonal Equation on Graphs. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Chung, R., Hammound, R., Hussain, M., Kar-Han, T., Crawfis, R., Thalmann, D., Kao, D., Avila, L. (eds.) ISVC 2010, Part II. LNCS, vol. 6454, pp. 647–658. Springer, Heidelberg (2010)
4. Claisse, A., Frey, P.: A nonlinear PDE model for reconstructing a regular surface from sampled data using a level set formulation on triangular meshes q. J. Comp. Phys. (2011)
5. Zhao, H.K., Osher, S., Merriman, B., Kang, M.: Implicit and nonparametric shape reconstruction from unorganized points using variational level set method. Computer Vision and Image Understanding 80(3), 295–319 (2000)
6. Ye, J., Yanovsky, I., Dong, B., Gandlin, R., Brandt, A., Osher, S.: Multigrid Narrow Band Surface Reconstruction via Level Set Functions (2008) (submitted for publication)
7. Goldstein, T., Bresson, X., Osher, S.: Geometric Applications of the Split Bregman Method: Segmentation and Surface Reconstruction. UCLA CAM Report, pp. 09–06 (2009)

8. Mederos, B., Amenta, N., Velho, L., de Figueiredo, L.H.: Surface reconstruction from noisy point clouds. In: Proceedings of the Third Eurographics Symposium on Geometry Processing, p. 53. Eurographics Association (2005)
9. Jalba, A.C., Roerdink, J.B.T.M.: Efficient surface reconstruction using generalized coulomb potentials. IEEE Transactions on Visualization and Computer Graphics 13, 1512–1519 (2007)
10. Amenta, N., Choi, S., Kolluri, R.K.: The power crust. In: Proceedings of the Sixth ACM Symposium on Solid Modeling and Applications, SMA 2001, pp. 249–266. ACM, New York (2001)
11. Braude, I., Marker, J., Museth, K., Nissanov, J., Breen, D.: Contour-based surface reconstruction using mpu implicit models. Graphical Models 69 (2007)
12. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Proceedings of the fourth Eurographics Symposium on Geometry Processing, SGP 2006, pp. 61–70. Eurographics Association, Aire-la-Ville (2006)
13. Kazhdan, M.M.: Reconstruction of solid models from oriented point sets. In: Symposium on Geometry Processing, pp. 73–82 (2005)
14. Elmoataz, A., Lezoray, O., Bougleux, S.: Weighted Nonlocal Discrete Regularization on Graphs: a framework for Image and Manifold. Processing IEEE Transactions on Image Processing 17(7), 1047–1060 (2008)
15. Stanford 3D scanning repository, `http://graphics.stanford.edu/data/3Dscanrep/`

Apparel Classification with Style

Lukas Bossard[1], Matthias Dantone[1], Christian Leistner[1,2], Christian Wengert[1,3], Till Quack[3], and Luc Van Gool[1,4]

[1] ETH Zürich, Switzerland
[2] Microsoft, Austria
[3] Kooaba AG, Switzerland
[4] KU Leuven, Belgium

Abstract. We introduce a complete pipeline for recognizing and classifying people's clothing in natural scenes. This has several interesting applications, including e-commerce, event and activity recognition, online advertising, *etc.* The stages of the pipeline combine a number of state-of-the-art building blocks such as upper body detectors, various feature channels and visual attributes. The core of our method consists of a multi-class learner based on a Random Forest that uses strong discriminative learners as decision nodes. To make the pipeline as automatic as possible we also integrate automatically crawled training data from the web in the learning process. Typically, multi-class learning benefits from more labeled data. Because the crawled data may be noisy and contain images unrelated to our task, we extend Random Forests to be capable of transfer learning from different domains. For evaluation, we define 15 clothing classes and introduce a benchmark data set for the clothing classification task consisting of over 80,000 images, which we make publicly available. We report experimental results, where our classifier outperforms an SVM baseline with 41.38 % vs 35.07 % average accuracy on challenging benchmark data.

1 Introduction

Clothing serves for much more than covering and protection. It is a means of communication to reflect social status, lifestyle, or membership of a particular (sub-)culture. The apparel is also an important cue for describing other people. For example: "The man with the black coat", or "the girl with the red bikini". The objective of this paper is to detect, classify, and describe clothes appearing in natural scenes in order to generate such descriptions with a focus on upper body clothing. Typically this means not only recognizing the type of clothing a person is wearing, but also the style, color, patterns, materials, *etc.* An example of a desired outcome would be to label the clothing in Figure 1 as "girl wearing a summer dress with a floral pattern". Only such a combination of type and attributes comes close to the descriptions we use as humans.

Such a system has many potential applications, ranging from automatic labeling in private or professional photo collections, over applications in e-commerce,

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 321–335, 2013.

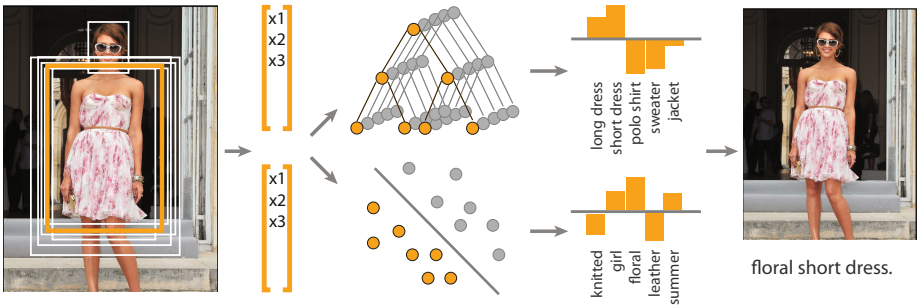


Fig. 1. Overview of our classification pipeline. First, an upper body detection algorithm is applied to the image. Then we densely extract a number of features. Histograms over the extracted features are used as input for a Random Forest (type classification) and for SVMs (attribute classification).

or contextual on-line advertising up to surveillance. Hence, systems for analyzing visual content may benefit significantly from autonomous apparel classification.

Enabling such robust classification of clothing in natural scenes is a non-trivial task that demands the combination of several computer vision fields. We propose a fully automated pipeline that proceeds in several stages (see Figure 1): First, a state-of-the art face and upper body detector is used to locate humans in natural scenes. The identified relevant image parts are then fed into two higher level classifiers, namely a random forest for classifying the type of clothing and several Support Vector Machines (SVMs) for characterizing the style of the apparel. In case of the random forest, SVMs are also used as split nodes to yield robust classifications at an acceptable speed.

Since the learning of classifiers demands large amounts of data for good generalization, but human annotation can be tedious, costly and inflexible, we also provide an extension of our algorithm that allows for the transfer of knowledge from corresponding data in other domains. E.g. knowledge from crawled web-data may be transferred to manually curated data from a clothing retail chain. We demonstrate this approach on 15 common types (classes) of clothing and 78 attributes. The benchmark data set for cloth classification is consists of over $80,000$ images.

In summary, the contributions of this work are:

- a pipeline for the detection, classification and description of upper body clothes in real-world images
- a benchmark data set for clothing classification
- an extension of Random Forests to transfer learning from related domains

The remainder of this paper is organized as follows. Section 2 discusses related work. An overview of our method is given in Section 3. In Section 4, the benchmark data set is introduced and in Section 5 our algorithms are evaluated. The paper ends with concluding remarks in Section 6.

2 Related Work

Classifying apparel or clothing is part of the wider task of classifying scenes. It is also related to detecting and describing persons in images or videos. Interestingly, in the past there has been little work on classifying clothing. Chen *et al.* [4] manually built a tree of composite clothing templates and match those to the image. Another strand of work specifically focuses on segmentation of garments covering the upper body [14]. More recently Wang *et al.* [27] also investigated segmentation of upper bodies, where the individuals occlude each other. Retrieving similar clothes given a query image was addressed by Liu *et al.* [20] and Wang *et al.* [28]. In the latter work, the authors use attribute classifiers for re-ranking the search results. Song *et al.* [24] predict people's occupation incorporating information on their clothing. Information extracted from clothing has also been used successfully to improve face recognition results [13].

Very recently, detection and classification of apparel has gained some momentum in the computer vision community. For instance, Yamaguchi *et al.* [29] show impressive results, relying strongly on state-of-the-art body pose estimation and superpixel segmentation. Their work focuses on pixelwise annotation. A somewhat limiting factor of that work is, that occurring labels are supposed to be known beforehand.

In this paper, we do not focus on clothing segmentation or similarity search, but on classification, *i.e.*, the problem of describing what type of clothing is worn in an image. To do so, we build on top of existing work [14,13,28] for clothing segmentation as described in Section 3.1, to then fully focus on the classification task. Our work is also related to learning visual attributes, which also has gained importance in recent years. They have been applied in color and pattern naming [12], object description [11], and face verification [16]. Within the context of our proposed task, attributes are obviously suited for describing the visual properties of clothing. To this end, we follow the algorithm by Farhadi *et al.* [11] for semantic attributes and extend it with s-o-a techniques as described in the following section.

3 Classifying Upper Body Clothing

In this work we focus on identifying clothing that people wear on their upper bodies, in the context of natural scenes. This demands the combination of several robust computer vision building blocks, which we will describe in the sequel.

Our apparel classification mechanism consists of two parts: one part describes the overall type/style of clothing, *e.g.*, "suit", "dress", "sweater". The other part describes the attributes of the style, such as "blue", "wool". By combining the outputs of these parts the system can come up with detailed descriptions of the clothing style, such as "blue dress". This combination is crucial for a real-world applications, because the labeling with either only the type ("dress"), or only its attributes ("blue") would be quite incomplete. The combination is also important for higher level tasks, such as event detection. For instance the knowledge that a dress is white may refer to a wedding.

More specifically, our method carries out the following steps: the first stage consists of s-o-a upper body detection as will be described in Section 3.1. After identification of upper bodies, we extract a number of different features from this region with dense sampling as explained in Section 3.2. These features are then transformed into a histogram representation by applying feature coding and pooling.

These features build the basis for classifying the type of apparel (part 1 of the system, Section 3.3) and for classification of apparel attributes (part 2 of the system, Section 3.4).

3.1 Pre-processing

Throughout this work we deal with real-world consumer images as they are found on the Internet. This entails multiple challenges concerning image quality, *e.g.*, varying lighting conditions, various image scales, *etc.* In a first pre-processing step, we address these variations by normalization of image dimensions and color distributions. This is achieved by resizing each image to 320 pixels maximum side length and by normalizing the histogram of each color channel.

As mentioned earlier, in order to identify clothing we need to identify persons first. One straightforward way to localize persons is to parametrize the upper body bounding box based on the position and scale of a detected face. In addition to this simple method, we also use the more sophisticated Calvin upper body detector [9], to generate additional bounding box hypotheses. All generated hypotheses are then combined through a non maximum suppression, in which hypotheses originating from the calvin upper body detector are scored higher than hypotheses coming only from the face position.

3.2 Features

In terms of feature extraction and coding, we follow a s-o-a image classification pipeline:

Feature extraction. Within the bounding box of an upper body found in the previous step, we extract a number of features including SURF [1], HOG [6], LBP [21], Self-Similarity (SSD) [23], as well as color information in the L*a*b space. All of those features are densely sampled on a grid.

Coding. For each of the feature types except LBP, a code book is learnt by using K-Means[1]. Subsequently all features are vector quantized using this code book.

Pooling. Finally, the quantized features are then spatially pooled with spatial pyramids [18] and max-pooling applied to the histograms.

For each feature type this results in a sparse, high-dimensional histogram.

[1] We used $1,024$ words for SURF and HOG, 128 words for color and 256 words for SSD, respectively.

3.3 Apparel Type Learning

After person detection and feature extraction, we use a classifier for the final clothing type label prediction. Since we face a multi-class learning problem with high-dimensional input and many training samples, we use Random Forests [2] as our classification method. Random Forests (RF) are fast, noise-tolerant, and inherently multi-class classifiers that can easily handle high-dimensional data, making them the ideal choice for our task.

A RF is an ensemble of T decision trees, where each tree is trained to maximize the information gain at each node level, quantified as

$$\mathcal{I}(\mathcal{X}, \tau) = H(\mathcal{X}) - \left(\frac{|\mathcal{X}_l|}{|\mathcal{X}|} H(\mathcal{X}_l) + \frac{|\mathcal{X}_r|}{|\mathcal{X}|} H(\mathcal{X}_r) \right) \tag{1}$$

where $H(\mathcal{X})$ is the entropy for the set of samples $\mathcal{X}$ and τ is a binary test to split $\mathcal{X}$ into subsets $\mathcal{X}_l$ and $\mathcal{X}_r$. Class predictions are performed by averaging over the class leaf distributions as $p(c|L) = \frac{1}{T} \sum_{t=1}^{T} P(c|l_t)$ with $L = (l_1, \ldots, l_T)$ being the leaf nodes of all trees. The term *random* stems from the fact that during training time only a random subset over the input space is considered for the split tests τ and each tree uses only a random subset of the training samples. This de-correlates the trees and leads to lower generalization error [2].

Following the idea of Yao *et al.* [30], we use strong discriminative learners in the form of binary SVMs as split decision function τ. In particular, if $\boldsymbol{x} \in \mathcal{R}^d$ is a d-dimensional input vector and $\boldsymbol{w}$ the trained SVM weight vector, an SVM node splits all samples with $\boldsymbol{w}^T \boldsymbol{x} < 0$ to the left and all other samples to the right child node, respectively. In order to enable the binary classifier to handle multiple classes, we randomly partition these classes into two groups. While training, several of those binary class partitions are randomly generated. For each grouping, a linear SVM is trained for a randomly chosen feature channel. Finally the split that maximizes the multi-class information gain $\mathcal{I}(\mathcal{X}, \boldsymbol{w})$, measured on the real labels, is chosen as splitting function, *i.e.*, $\hat{\boldsymbol{w}} = \text{argmax}_{\boldsymbol{w}} \mathcal{I}(\mathcal{X}, \boldsymbol{w})$

Random forests are highly discriminative learners but they can also overfit easily to the training data if too few training samples are available [3], an effect that tends to intensify if SVMs are used as split nodes. Therefore, in the following, we propose two extensions to the random forest algorithms of [2] and [30] that shall improve the generalization accuracy but keep the discriminative power.

Large Margin. While training, different split functions often yield the same information gain. Breaking such ties is often done by randomly selecting one split function out of the best performing splits. In this work we introduce an additional selection criterion to make more optimal decisions. It is inspired by Transductive Support Vector Machines (TSVM) [15], where the density of the feature space around the decision boundary is taken into account while solving the optimization problem for $\boldsymbol{w}$. Opposed to TSVMs however, we do not use this information while optimizing $\boldsymbol{w}$, but go after minimal feature density (or largest margin) as an additional optimality criterion for the split selection. In

other words, if several split functions perform equally well, the density of the feature space within the margin is taken into account, estimated as:

$$\mathcal{I}^m(\mathcal{X}, \boldsymbol{w}) = \frac{1}{|\mathcal{X}|} \sum_{\boldsymbol{x} \in \mathcal{X}} \max\left(0, 1 - |\boldsymbol{w}^T \boldsymbol{x}|\right) \tag{2}$$

with the decision boundary $\boldsymbol{w}$ and training examples $\mathcal{X}$. Then the optimal split can be chosen by minimizing the above equation *w.r.t.* $\boldsymbol{w}$, *i.e.*, the optimal split function is given by $\hat{\boldsymbol{w}} = \text{argmin}_{\boldsymbol{w}}\ \mathcal{I}^m(\mathcal{X}, \boldsymbol{w})$.

Transfer Forests. Another option to improve the generalization power of Random Forest is to use more training samples. However, it is often not easy to acquire more training samples along with good quality annotations. One way to achieve this is to outsource the labeling task to crowdsourcing platforms, such as Mechanical Turk [25]. Yet, this demands careful planning for an effectively designed task and an adequate strategy for quality control. It can also not be used to annotate confidential data. Therefore, previous work also studied the extension of RFs to semi-supervised learning [19,5] in order to benefit from additional unlabeled data, which is usually cheap to collect.

For our task, we can use text-based image search engines to gather large amounts of images, such that the returned images come already with prior labels $\hat{c}$. For instance, we can type *cotton, black, pastel*, *etc.* to get clothing images that probably exhibit these attributes. Similarly, we can type *jacket, t-shirt, blouse*, *etc.* to get images containing our target type classes. On the downside, these images may contain high levels of noise and originate from variable source domains. Thus, not all samples might fit to our task and $\hat{c}$ cannot be considered to flawlessly correspond to the *real* label c.

Therefore, we extend Random Forests to *Transfer Learning* (TL) [22], which tries to improve the classification accuracy in scenarios where the training and test distributions differ. In particular, assume having access to M samples from the labeled target domain $\mathcal{X}^l$ (*e.g.* a manually labeled and quality controlled data set) along with their labels $\mathcal{C}$. Additionally, in TL one has access to N samples from an auxiliary domain $\mathcal{X}^a$ (*e.g.* Google image search) together with their labels $\hat{\mathcal{C}}$. The task of TL is to train a function $f : \mathcal{X} \rightarrow \mathcal{C}$ that performs better on the target domain via training on $\mathcal{X}^l \cup \mathcal{X}^a$ than solely relying on $\mathcal{X}^l$. There exist many approaches to TL (*c.f.* [22]) and its usefulness has also been demonstrated in various vision domains, *e.g.* [26,17]. We present here a novel variant of transfer learning for Random Forests as this is our main learner.

To this end, we exploit the idea that although the source and target distributions might be different, some of the source samples $\boldsymbol{x}_i \in \mathcal{X}^a$ can still be useful for the task and should thus be incorporated during learning, while samples that may harm the learner should be eliminated. In order to accomplish such an *instance-transfer* approach (*c.f.* [22]) for Random Forests, we augment the information gain of Eq. 1 to become

$$\mathcal{I}^*(\mathcal{X}, \boldsymbol{w}) = (1 - \lambda) \cdot \mathcal{I}(\mathcal{X}^l, \boldsymbol{w}) + \lambda \cdot \mathcal{I}(\mathcal{X}^a, \boldsymbol{w}), \tag{3}$$

Table 1. List of attribute categories and the attributes therein

Colors	Patterns	Materials	Structures	Looks	Persons	Sleeves	Styles	
beige	animal print	cotton	frilly	black/white	child	long	20's	nerd
black	zebra	denim	knitted	colored	boy	short	50's	outdoor
blue	leopard	fur	ruffled	gaudy	girl	none	60's	preppy
brown	argyle	lace	wrinkled	pastel	female		70's	punk
gray	checkered	leather			male		80's	rock
green	dotted	silk					90's	romantic
orange	floral	tweed					bohemian	sports
pink	herringbone	wool					business	wedding
purple	houndstooth						casual	spring
red	paisley						dandy	summer
teal	pinstripes						hip hop	autumn
white	plaid						hippie	winter
yellow	print						mod	
	striped							
	tartan							

where the first term corresponds to Eq. 1 and $\mathcal{I}(\mathcal{X}^a, \boldsymbol{w})$ measures the information gain over the auxiliary data. The *overall* influence of $\mathcal{X}^a$ is controlled via the steering parameter $\lambda \in [0, 1]$.

The information gain $\mathcal{I}$ relies on the standard entropy measure $H(\mathcal{X}) = -\sum_c p_c \log(p_c)$ with $p_c = \frac{1}{|\mathcal{X}|} \sum_i \varphi(\boldsymbol{x}_i, \mathcal{X}_c)$, where $\varphi(\boldsymbol{x}_i, \mathcal{X}_c)$ is the indicator function and is defined as

$$\varphi_l(\boldsymbol{x}_i, \mathcal{X}_c) = \begin{cases} 1 & \text{if } \boldsymbol{x}_i \in \mathcal{X}_c \\ 0 & \text{else,} \end{cases} \tag{4}$$

with $\mathcal{X}_c$ representing the set of samples for class c. Note, the auxiliary dataset influences only the selection of the trained SVM for each node, but it is not used during the actual training of the SVM.

3.4 Clothing Attribute Learning

The slight differences in appearance of apparel are often orthogonal to the type of clothing, *i.e.*, the composition of colors, patterns, materials and/or cuttings often matter more than the information, that a particular cloth is *e.g.* a sweater. A common way to include such kind of information is to represent it by semantic attributes. We define eight attribute categories with in total 78 attributes as shown in Table 1. The training of the attributes happens for each of the eight attribute categories separately. Within each of those, the different attributes are considered mutually exclusive. Thus, within a category, we train for each attribute a one-vs-all linear SVM on the features described in Section 3.2.

4 Data Sets

For both tasks – classification of clothes and attribute detection – we collected two distinct data sets. Additionally, an auxiliary data set $\mathcal{X}^a$ was automatically crawled to be used for our transfer learning extension for Random Forests.

Table 2. Main classes and number of images per class of the benchmark data set

Category	Images	Boxes	Category	Images	Boxes	Category	Images	Boxes
Long dress	22,372	12,622	Suit	12,971	7,573	Shirt	3,140	1,784
Coat	18,782	11,338	Undergarment	10,881	6,927	T-shirt	2,339	1,784
Jacket	17,848	11,719	Uniform	8,830	4,194	Blouses	1,344	1,121
Cloak	15,444	9,371	Sweater	8,393	6,515	Vest	1,261	938
Robe	13,327	7,262	Short dress	7,547	5,360	Polo shirt	1,239	976
						Total	145,718	89,484

4.1 Apparel Type

To the best of our knowledge, there is no publicly available data set for the task of classifying apparel or clothing, respectively. The large variety of different clothing types and, additionally, the large variance of appearance in terms of colors, patterns, cuttings *etc.* necessitate that a large data set be used for training a robust classifier. However, assembling a comprehensive and high quality data set is a daunting task.

Luckily, ImageNet [8], a quality controlled and human-annotated image database that is hierarchically organised according to WordNet, contains many categories (so called *synsets*) related to clothes. Nevertheless, a closer look at ImageNet's (or rather WordNet's) structure reveals that clothing synsets often do not correspond to the hierarchy a human would expect. Therefore we handpicked 15 categories and reorganized ImageNet's synsets accordingly. Due to how ImageNet is built, some images are ambiguous and quite a few are very small. As a cleaning step, we preprocess each image as described in Section 3.1. If no face or upper body can be detected, a centered bounding box is assumed as ImageNet also contains web shop images that show pieces of clothing alone. The resulting bounding boxes smaller than 91 pixels were discarded.

An overview over the categories can be found in Table 2. As a contribution of this paper, we make the details of the data set publicly available so that the community can use this subset of ImageNet as a benchmark for clothing classification.

4.2 Transfer Forest

For each of the clothing type classes, we collected the auxiliary data set $\mathcal{X}^a$ by querying Google image search multiple times with different word combinations for the same category (*e.g.* "sweater women", "sweater men" or *e.g.* "long dress formal", "long dress casual") such that the retrieved data contains some variation. We again restricted the result to photos of a minimum size of 300×400 pixels and performed no further supervision on the 42,624 downloaded images.

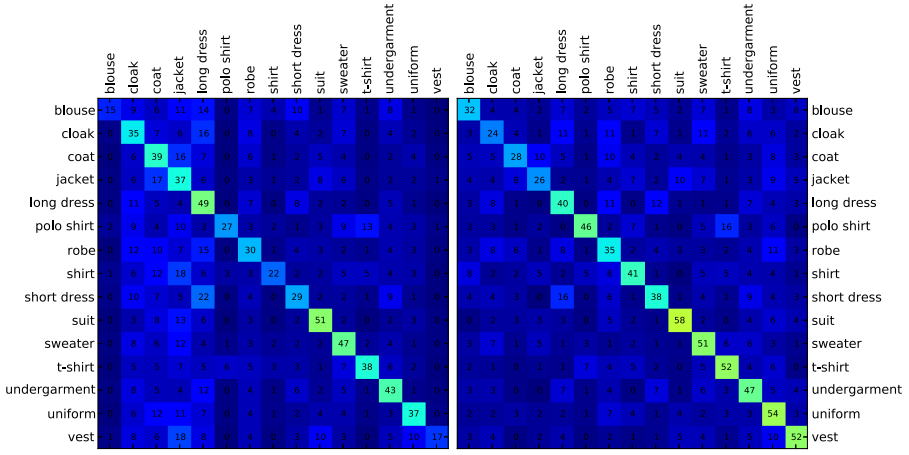


Fig. 2. Confusion matrix of our clothing classes for the best performing SVM classifier on the left side and the proposed Transfer Forest on the right side

4.3 Attributes

In order to train classifiers for visual attributes, we need a special training data set just for this task. While ImageNet provides images with attribute annotation, it only covers a small part of our defined attributes (*c.f.* Table 1). Moreover, ImageNet provides attribute annotation only for a subset of its synsets, thus making this data source not appropriate for learning our selection of attributes. Therefore we construct a third distinct data set by automatically crawling the Web. For each attribute, we let an automated script download at least 200 images using Google image search and restricted results to photos of a minimum size of 300×400 pixels. For each attribute, the script generates a query composed of the attribute label and one of the words "clothing", "clothes" or "look" as query keyword. No further supervision was applied to those 25,002 images after downloading.

5 Experiments

In this section we present experiments to evaluate our algorithm quantitatively. First we show the results for the apparel type part, then the results for the attribute part. An overview of the relevant results can be found in Table 3.

5.1 Apparel Types

We present three sets of numerical evaluations. First, using the apparel type data set introduced in Section 4.1, we trained a SVM as a baseline. Then, the results for Random Forest with SVMs as split nodes are shown. Finally, the effectiveness of the proposed Transfer Forest is demonstrated.

Table 3. Classification performance measured as average accuracy over all classes on our benchmark data set for different methods

Learner	Avg. Acc. [%]
One vs. all SVM	35.03
RF	38.29
RF + large margin on $\mathcal{X}^l$	39.31
RF on $\mathcal{X}^l \cup \mathcal{X}^a$, naïve	36.27
RF on $\mathcal{X}^l \cup \mathcal{X}^a$, Daumé [7]	35.00
Transfer Forest	41.36

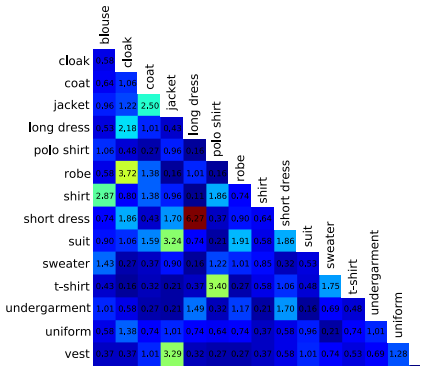


Fig. 3. Percentage of co-occurring classes in the deepest split nodes. Note how semantic similar classes often occur together.

SVM Baseline. As a baseline experiment we train a one-vs-all linear SVM for each clothing type category. We evaluated all possible feature combinations, and also L_2 regularized hinge as well as L_1 regularized logistic loss. For the evaluation of the feature combinations, the histograms of the different extracted features (*c.f.* Section 3.2) were simply concatenated. We used 80 % of the bounding boxes of each class for training and the remaining part for testing. Finally, L_1 regularized logistic loss using all available features yielded with 35.03 % average accuracy the best performance. The confusion matrix is shown in Figure 2a. There is a clear bias towards overrepresented classes.

Random Forest. To evaluate the performance of the random forest framework we define the following protocol: again we use 80 % of the images of each type class of the data set for training and the remainder for testing. Each tree has been trained on a random subset of the training set, which contains 500 images for each class, thus $7,500$ images in total.

While training, we generate at each node 50 linear SVMs with the feature type and the binary partition of the class labels chosen at random. Other than what Yao *et al.* [30] propose, we do not randomly sample subregions within the bounding boxes, but use the spatially pooled histograms (*c.f.* Section 3.2) as input for the SVMs. Each tree is then recursively split until either the information gain stops increasing, the numbers of trainings examples drops below 10, or a maximum depth of 10 is reached. In total we trained 100 trees out of which we created 5 forests by randomly choosing 50 trees. The final result is then averaged over those 5 forests to reduce the variance of the results.

Baseline. With 38.29 % average accuracy, our Random Forest with SVMs as split functions outperforms the plain SVM baseline (35.03 %) significantly. It handles

the uneven class distribution much better as can be seen in Figure 2b. These results confirm our expectation that a Random Forest is a suitable learner for our task. Figure 3 shows the co-occurrences of the different classes at the deepest levels of the tree. Interestingly, semantic similar classes often occur together.

Large Margin. Having strong discriminative learners as decision nodes renders the information gain as optimization criterion often as too weak a criteria: several different splits have the same information gain. In this case, choosing the split with the largest margin amongst the splits with the same information gain on the training data seems beneficial as performance increases about 1 % compared to the Random Forest baseline.

Transfer Forest. To assess the performance of our approach, we follow the protocol defined in the baseline Random Forest evaluation. The parameter λ of Eq. 3 was varied between $0 < \lambda < 1$ in 0.05 steps. Unfortunately, no distinct best choice for λ is obvious. Yet, our approach yields minimum and maximum improvement of 2.18 % and 3.09 % over the baseline Random Forest, respectively. On average, any choice of λ increased the performance about 2.45 % in that $0 < \lambda < 1$ interval.

To validate our assumption that transfer learning is beneficial in this case, we also trained a forest on the union of $\mathcal{X}^a$ and $\mathcal{X}^l$, thus treating the auxiliary images as they would stem from the regular data set. In this case, the performance significantly drops below that of the baseline Random Forest.

As a sanity check, we also compared to another domain adaptation method presented by Daumé [7], which comes at a cost of tripling the memory requirements and substantially longer training times, as the feature vectors that are passed on to the SVM are thrice as large. Moreover, also this approach does not improve the performance over that of the baseline Random Forest (see Table 3). This *(i)* highlights the importance of using transfer learning when incorporating data from different domains for our task and *(ii)* also shows that Random Forests are useful for transfer learning.

Table 4. Best average accuracy for each attribute category with the corresponding features. The number of attributes per category is denoted in in parentheses.

Category	Acc. [%]	Reg.	Loss	Surf	Hog	Color	Lbp	Ssd
Looks (4)	71.63	L_2	hinge	×		×	×	×
Sleeves (3)	71.52	L_1	logistic	×			×	×
Persons (5)	64.07	L_2	hinge	×	×	×		×
Materials (8)	63.97	L_1	logistic	×	×	×	×	×
Structure (4)	63.64	L_1	logistic					×
Colors (13)	60.50	L_2	hinge			×	×	×
Patterns (15)	49.75	L_1	logistic					×
Styles (25)	37.53	L_2	hinge					×

5.2 Attributes

For training and testing we assume that within a given attribute category (*e.g.* colors or patterns) attributes (*e.g.* red, white or zebra,dotted) are mutually exclusive. Furthermore attribute with the least samples constrains the number of samples for all other attributes in the same category. With this, out of the $25,002$ downloaded images, $16,155$ were used for testing and training the attributes. The data set was split in 75 % of samples for training and 25 % for testing.

We extract the features as described in Section 3.2 and train several linear one vs. all SVMs [10] with all possible feature combinations as well as with L_1 regularized logistic loss and L_2 regularized hinge loss. For the experiments, the cost was set at $C = 1$ as the classification performance stayed invariant in combination with max pooling. Results are shown in Table 4.

The classification accuracy ranges between about 38 % and 71 % depending on the category. Of course it is expected that attribute categories with less possible values (*e.g. sleeves*) perform better than those with many (*e.g. patterns*). Nevertheless a classification task such as the sleeve length is not trivial and performs surprisingly well. On the other hand color and pattern classification could probably be improved. It appears the classifier is distracted too much by background data present within the bounding box. A simple fix would be to sample data only from a small part from the center of the bounding box for categories such as colors or patterns. A large category such as *styles* with many "fuzzy" or "semantic" attribute values such as "punk" or "nerd" poses of course a challenge to even an advanced classifier.

5.3 Qualitative Results

In Figure 4 some example outputs of our full pipeline are shown. Note how we are able to correctly classify both style and attributes in many situations. This would allow a system to come up with the desired description combining attributes and style. For instance for the first example in the middle row a description such as "Girl wearing a pastel spring short dress without sleeves" could be generated. Also note how the random forest robustly handles slight variations in body pose for cloth classification (*e.g.*, in the top right example). Of course, accurate detection of the upper body is crucial for our method, and many of the failure cases are due to false upper body detections (example in the 3^{rd} row, 3^{rd} image). Another source for confusion are ambiguities in the ground truth (3^{rd} row, 1^{st} and 2^{nd} example). For attributes performance is mainly challenged by distracting background within the bounding box or lack of context in the bounding box (*e.g.*, 2^{nd} row, 2^{nd} example).

Fig. 4. Some example output of our pipeline. The header denotes the ground truth class. Each example shows the detected bounding box and the output of the type classifier. On the left side of each example, the output of the most confident attribute classifier for each attribute group is shown.

6 Conclusion

We presented a complete system, capable of classifying and describing upper body apparel in natural scenes. Our algorithm first identifies relevant image regions with state of the art upper body detectors. Then multiple features such as SURF, HOG, LBP, SSD and color features are densely extracted, vector quantized and pooled into histograms and fed into two higher level classifiers, one for classifying the type and one for determining the style of apparel. We could show that the Random Forest framework is a very suitable tool for this task, outperforming other methods such as SVM. Since there are many apparel images available on the web but they often come with noise or unrelated content, we

extended Random Forests to transfer learning. While this improved the accuracy for the task at hand, we believe that also other vision applications using Random Forests might benefit from this algorithmic extension. We also introduced a challenging benchmark data set for the community, comprising more than 80,000 images for the 15 clothing type classes. On this data set, our Transfer Forest algorithm yielded an accuracy of 41.36 %, when averaged over the type classes. This represents an improvement of 3.08 % compared to the base line Random Forest approach and an improvement of 6.3 % over the SVM baseline.

Acknowledgement. We thank Fabian Landau for his excellent work on segmentation. This project has been supported by the Commission for Technology and Innovation (CTI) within the program 12618.1.

References

1. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. In: ICCV (2006)
2. Breiman, L.: Random forests. Machine Learning, 5–32 (2001)
3. Caruana, R., Karampatziakis, N., Yessenalina, A.: An empirical evaluation of supervised learning methods in high dimensions. In: ICML (2008)
4. Chen, H., Xu, Z.J., Liu, Z.Q., Zhu, S.C.: Composite Templates for Cloth Modeling and Sketching. In: CVPR (2006)
5. Criminisi, A., Shotton, J., Konukoglu, E.: Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning. Technical Report MSR-TR-2011-114, Microsoft Research (2011)
6. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: CVPR (2005)
7. Daumé, H.: Frustratingly easy domain adaptation. Annual Meeting-Association for Computational Linguistics 45, 256 (2007)
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. Eichner, M., Ferrari, V.: CALVIN Upper-body detector for detection in still images
10. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. JMLR 9 (2008)
11. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: CVPR (2009)
12. Ferrari, V., Zisserman, A.: Learning visual attributes. In: NIPS (2008)
13. Gallagher, A.C.: Clothing cosegmentation for recognizing people. In: CVPR (2008)
14. Hu, Z., Yan, H., Lin, X.: Clothing segmentation using foreground and background estimation based on the constrained Delaunay triangulation. Pattern Recognition 41 (2008)
15. Joachims, T.: Transductive inference for text classification using support vector machines. In: ICML (1999)
16. Kumar, N., Berg, A.C., Belhumeur, P.N., Nayar, S.K.: Attribute and simile classifiers for face verification. In: ICCV (2009)
17. Lampert, C., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: CVPR (2009)
18. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR (2006)

19. Leistner, C., Saffari, A., Santner, J., Bischof, H.: Semi-Supervised Random Forests. In: ICCV (2009)
20. Liu, S., Song, Z., Liu, G., Xu, C., Lu, H., Yan, S.: Street-to-Shop: Cross-Scenario Clothing Retrieval via Parts Alignment and Auxiliary Set. In: CVPR (2012)
21. Ojala, T., Pietikainen, M., Harwood, D.: Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In: ICOR (1994)
22. Pan, S.J., Yang, Q.: A survey on transfer learning. TKDE (2010)
23. Shechtman, E., Irani, M.: Matching Local Self-Similarities across Images and Videos. In: CVPR (2007)
24. Song, Z., Wang, M., Hua, X.s., Yan, S.: Predicting occupation via human clothing and contexts. In: ICCV (2011)
25. Sorokin, A., Forsyth, D.: Utility data annotation with amazon mechanical turk. In: Workshop on Internet Vision (2008)
26. Stark, M., Goesele, M., Schiele, B.: A shape-based object class model for knowledge transfer. In: ICCV (2009)
27. Wang, N., Ai, H.: Who Blocks Who: Simultaneous clothing segmentation for grouping images. In: ICCV (2011)
28. Wang, X., Zhang, T.: Clothes search in consumer photos via color matching and attribute learning. In: MM. ACM Press (2011)
29. Yamaguchi, K., Kiapour, H., Ortiz, L., Berg, T.L.: Parsing Clothing in Fashion Photographs. In: CVPR (2012)
30. Yao, B., Khosla, A., Fei-Fei, L.: Combining randomization and discrimination for fine-grained image categorization. In: CVPR (2011)

Deblurring Vein Images and Removing Skin Wrinkle Patterns by Using Tri-band Illumination

Naoto Miura and Yoichi Sato

Institute of Industrial Science, The University of Tokyo,
4-6-1 Komaba, Meguro-ku, Tokyo, 153-8505 Japan

Abstract. We present a new method for enhancing images of blood vessels in skin tissues by using tri-band illumination. Transmitted-light vein images captured by a camera contain an image blur due to light scattering in skin. The blur can be described by a point spread function (PSF) that is a function of the thickness of skin layers in front of a vein and the extinction coefficients of skin tissues. The PSFs cannot be directly observed because the depth of a vein in skin tissue is unknown and the thickness of the skin tissues in different parts of the human body vary. Moreover, skin wrinkle patterns are observed as dark lines and need to be eliminated for clear vein imaging. We propose a method for removing image blur and skin wrinkle patterns from transmitted-light images of veins by using tri-band illumination. First, wrinkle patterns are separated from vein patterns by using a difference between the light absorbances of blood at two wavelengths. Subsequently, image blurs caused by light scattering at skin layers are removed by using a PSF estimated from two vein images. The key observations in this work are that at one of the three wavelengths to obtain the vein images the extinction coefficient of skin tissues must be twice as large as that at another of the wavelengths, and that at the third wavelength the extinction coefficient of blood must be smaller than it is at either of the other two wavelengths. This allows us to estimate true vein patterns without knowing the depth of a vein and to eliminate skin wrinkle patterns from a vein image. Our experiments show that our method can separate skin wrinkle patterns form vein patterns and that it reduces blur and improves the contrast of vein images better than a conventional method does. The results indicate that our method will contribute to the development of highly accurate personal authentication technology based on vein patterns.

1 Introduction

Near-infrared spectroscopy is widely used for biometric authentication and biomedical measurements because of its safety and the simplicity of its use [1–3]. An infrared image of finger skin shows vein patterns that can be used for biometric authentication. There are two types of methods for capturing vein images: reflected-light methods and transmitted-light methods [4]. The latter are more

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 336–349, 2013.

suitable for biometric authentication based on finger vein patterns because they yield vein images that have a higher S/N. That is, the veins are clearer in images captured from the palm side of a finger while an infrared light illuminates the other side.

The vein images obtained this way, however, contain a large amount of blur due to scattering of infrared light inside skin tissues [5]. The image blur due to light scattering can be described by point spread functions (PSFs). A PSF is a function of the depth of a vein, the extinction coefficients of the skin (at each wavelength) and the positions of the image. Moreover, skin wrinkle patterns are superimposed on the vein images, and the clarity with which they seen depends on the roughness of the skin and on the presence of extraneous substances such as cosmetics and inks. The optical parameters of wrinkle patterns therefore cannot be determined in advance.

Image blur can be removed by using deconvolution calculation if we know the PSFs, but we cannot acquire PSFs in advance because the depths of the veins at all the positions are unknown in general. Moreover, because wrinkle patterns on the skin appear as dark lines, they cannot be easily separated from vein patterns. We can use a morphological method such as top-hat transform to eliminate the dark line patterns [3, 6], but that would also eliminate vein patterns.

In this paper we propose a method for removing image blur and skin wrinkle patterns from transmitted-light images of veins by using tri-band illumination (i.e., three wavelengths). The key observations in this work are that at one of the three wavelengths the extinction coefficient of skin tissues must be twice as large as that at another of the wavelengths, and that at the third wavelength the extinction coefficient of blood must be smaller than it is at either of the other two wavelengths. This allows us to estimate true vein patterns without knowing the depth of a vein, and to eliminate skin wrinkle patterns from vein images.

Our method is different from other existing techniques using a single image (obtained at one wavelength) to eliminate wrinkle patterns and image blur. Because in a single image the appearance of vein patterns may be similar to that of skin wrinkle patterns, the conventional methods fail to eliminate only wrinkle patterns while preserving vein patterns. Moreover, the conventional methods cannot appropriately remove image blur caused by light scattering inside skins because PSFs cannot be estimated from a single image.

2 Related Works

Image enhancements of vein patterns obtained using light illumination have been studied in medical image processing and computer vision. Shimizu et al. introduced a method for estimating in-vivo depth-dependent PSFs [5]. The estimated PSF can be used for deblurring a transmitted-light image of the vein pattern, but the method cannot be applied if skin tissue is not uniform or the depths of the veins are unknown.

To get information about the detailed structure of veins, Nishidate developed a method of visualizing vein depth in skin tissues by using three wavelengths [7].

The reconstruction result showed that the depth map of a vein pattern could be obtained, but the result is affected by light scattering and it is not a clear pattern.

Kim et al. developed a method for differentiating the scattered and unscattered components of light transmitted through a scattering medium and used it for imaging finger veins [8]. The method uses an angular filter to reduce the contribution of the scattered component, but this method cannot entirely eliminate the effects of light scattering in the skin tissue.

A method for PSF estimation was proposed by Mukaigawa et al. [9], but that method cannot be used for optically dense media like a human skin because it assumes that the target object is a translucent medium in which the single scatterings can be observed.

Subcutaneous vein detection using 3D information of a human arm and multispectral imaging was proposed by Paquit et al. [10]. The detection targets are arm veins for catheter insertion, which are larger than finger veins. The article does not mention whether the method can be used for imaging small veins.

Tsumura et al. proposed a method for separating hemoglobin and melanin information in the skin [11]. In their method, parameters representing the optical characteristics of hemoglobin and melanin are given a priori. The method thus cannot be used for wrinkle elimination because the optical characteristics of wrinkle patterns cannot be determined in advance.

3 Removing Skin Wrinkles and Deblurring Veins by Using Tri-band Illumination

3.1 Light Propagation in Skin Tissues

Fig. 1 illustrates the propagation of light inside skin tissues. Incident light I_j at a wavelength of λ_j illuminates the top of the tissue, penetrates it, and reaches a vein near the bottom of the tissue. The light is then partially absorbed by hemoglobin in blood, and the rest reaches the bottom of the vein. After that the light is scattered in the skin tissue and reaches the surface of the skin. Finally, the light emitted from the bottom surface of the skin is captured by a camera. Before the emission from the skin, the transmitted light is affected by the condition of the surface of the skin. The skin surface has a lot of wrinkles with extraneous substances and roughnesses. Because wrinkles scatter or absorb the emitted light, the dark wrinkle patterns are observed in the captured image as shown in Fig. 2(a).

To make the problem tractable, we make the following assumptions.

- The light reaching a vein has been scattered enough, inside the human body, that the light s_j at the top of the vein is evenly distributed.
- The light is absorbed by the blood without scattering in accordance with the Lambert-Beer law.

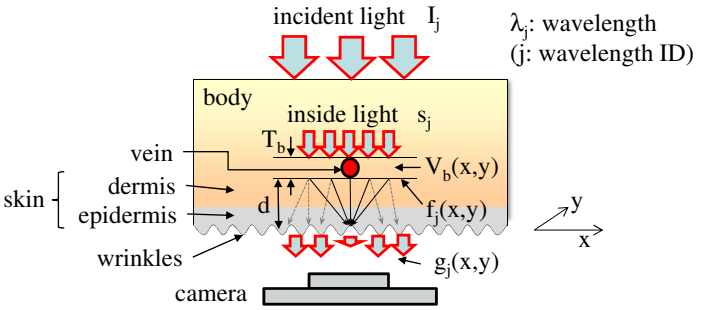


Fig. 1. Light transmission in skin tissue

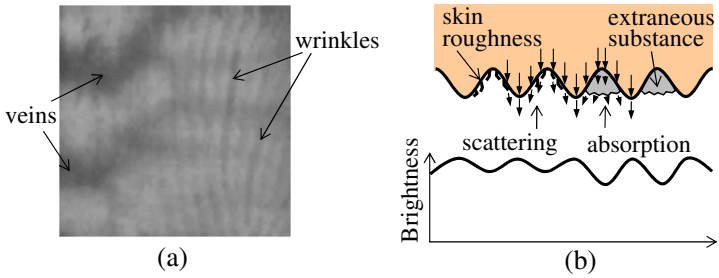


Fig. 2. Skin wrinkles on a vein image: (a) Vein image with skin wrinkles, (b) Light absorption at skin surface

Let us denote an observed image by g_j, an unknown PSF by h_j, a true vein image by f_j, the absorbance of skin wrinkles by c_j, image noise by n_j, light wavelengths by λ_j, and a wavelength ID by j. The observed image is given as

$$g_j = (f_j \otimes h_j)c_j + n_j, \tag{1}$$

where the operator $\otimes$ represents convolution. Although the images we treat in this paper are 2D, for simplicity we denote $g_j(x, y)$ by g_j. The vein image f_j can be described by the Lambert-Beer law.

$$f_j = s_j \exp(-\mu_{b_j} V_b T_b - \mu_{d_j}(1 - V_b)T_b), \tag{2}$$

where s_j is the light strength above the vein, μ_{b_j} and μ_{d_j} are respectively, the extinction coefficients of blood and dermis at a wavelength of λ_j, V_b is the volume fraction of the dermis occupied by blood at the vein layer and T_b is the thickness of the vein.

In a similar way, the absorbance c_j of skin wrinkles can be described as follows.

$$c_j = \exp(-\mu_{c_j} V_c T_c), \tag{3}$$

where μ_{c_j} is the extinction coefficient of skin wrinkles at wavelength λ_j, V_c is the volume fraction of the skin surface occupied by the skin wrinkles and T_c is the thickness of the substance. $V_c T_c$ is a spatially varying value that represents

scattering or absorption on the skin surface. The parameters μ_{c_j} and $V_c T_c$ cannot be estimated in advance because there are a lot of causes for the observation of the skin wrinkles.

3.2 Removing Wrinkles

Wrinkle patterns on skins can be observed as dark lines superimposed on a vein pattern. In general, these patterns can be eliminated by using a top-hat transform to remove dark lines [3, 6] (see Fig. 3(a)). As one can see in Fig. 2(a), however, in vein images both wrinkles and veins appear as dark lines. Therefore vein patterns would be also eliminated by a top-hat transform (see Fig. 3(b)).

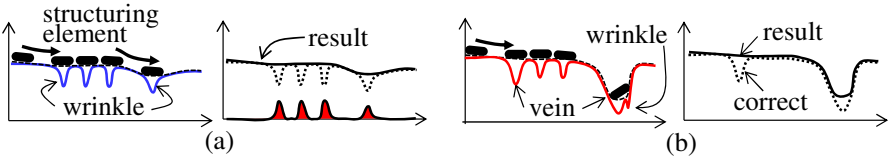


Fig. 3. Top-hat transform for eliminating dark patterns: (a) Wrinkle elimination, (b) Artifactual elimination of veins

To avoid the artifactual elimination of vein patterns, our method determines whether or not dark brightnesses are caused by veins by using the blood absorbance difference between two images captured with different wavelengths. Fig. 4 shows the concept of our method. First, images g_h and g_l are obtained (Fig. 4(a)). Let us suppose that the absorbance of blood for g_h is higher than that for g_l. The brightnesses of the two images are adjusted so as to make the average brightnesses of the images approximately the same. When we find dark lines in these images, we can determine whether or not the darkness of pixels is caused by veins by comparing the pixel values of the two images. If pixels where $g_l > g_h$ are observed, the values of these pixels are increased so that the dark line is smoothly eliminated as shown in Fig. 4(b) (the method is discussed later). A the top-hat transform then removes the dark portion, and the vein-eliminated image is obtained (Fig. 4(c)). Because our preliminary investigation shows that the darknesses of the skin wrinkles in two images are similar to each other, the top-hat transform can work well to eliminate only the skin-wrinkle portions of the image (Fig. 4(d)).

The reason our method focuses first on the vein pattern is that the modeling of skin wrinkles is more difficult than that of veins because there are several factors that make skin wrinkles clearer.

When eliminating skin wrinkle patterns, our method assumes that the light absorption in a human body conform to the Lambert-Beer. Strictly speaking, the amount of vein pattern blurring differs slightly between two images, and this difference can cause artifactual dark patterns. These artifacts are much smaller than the skin wrinkles to be eliminated, however, so in this procedure we assume the images have no blurring.

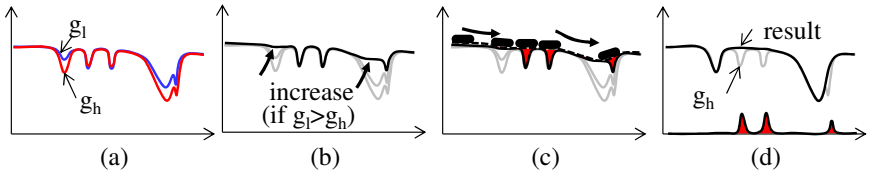


Fig. 4. Outline of proposed method for removing skin wrinkles

The observed images are given as

$$\begin{aligned} g_j &= (f_j \exp^{-\mu_{d_j} d}) c_j + n_j \\ &\approx s_j \exp^{-\mu_{b_j} V_b T_b} \exp^{-\mu_{d_j}(1-V_b)T_b} \exp^{-\mu_{d_j} d} \exp^{-\mu_{c_j} V_c T_c}, \end{aligned} \tag{4}$$

where d is the distance between the bottom of the vein and the surface of the skin.

Taking the logarithm of both sides of equation (4), we obtain

$$\ln g_j \approx \ln s_j - \mu_{b_j} V_b T_b - \mu_{d_j}(1 - V_b)T_b - \mu_{d_j} d - \mu_{c_j} V_c T_c. \tag{5}$$

If we can find regions where there are no vein patterns do not exist—that is, regions where V_b=0—the brightnesses of two images in those regions can be adjusted or corrected so that the average brightnesses there are approximately the same in both images. In this case, we can obtain the following equation:

$$\ln s_h - \mu_{d_h} T_b - \mu_{d_h} d - \mu_{c_h} V_c T_c = \ln s_l - \mu_{d_l} T_b - \mu_{d_l} d - \mu_{c_l} V_c T_c. \tag{6}$$

From equation (5) with $j = h$, equation (5) with $j = l$, and equation (6), we obtain $V_b T_b \triangleq V_{bt}$ as follows:

$$V_{bt} = \frac{\ln g_l - \ln g_h}{\mu_{b_h} - \mu_{b_l} + \mu_{d_l} - \mu_{d_h}}. \tag{7}$$

If we identify pixels where $V_{bt} \leq 0$, we can conduct the average control mentioned above. Because V_{bt} cannot be determined in advance, however, we assume V_{bt} is initially zero at all pixels and then repeatedly calculate V_{bt} and control the average of g_j.

Using equation (7), we can obtain the vein-eliminated image g_{h_e}:

$$g_{h_e} = \begin{cases} g_h \exp^{(\mu_{b_h} - \mu_{d_h}) V_{bt}}, (\text{if } V_{bt} > 0\ ) \\ g_h, (\text{otherwise}) \end{cases} \tag{8}$$

This means that the values of dark pixels in the region where $V_{bt} > 0$ are converted to higher values by using this transformation. Therefore, darkness of the pixel values caused by the absorption of blood can be eliminated and only the skin wrinkles remain in the image. A top-hat transform is applied to the image, and components of the skin wrinkles pattern c_h can be extracted.

$$c_h = TH(g_{h_e}), \tag{9}$$

where $TH(\cdot)$ means a top-hat transform operator.

Finally, subtracting the c_h with coefficient $\exp^{-(\mu_{b_h}-\mu_{d_h})V_{bt}}$ from g_{h_e}, we can obtain non-wrinkle image g'_h:

$$g'_h = g_{h_e} - c_h \exp^{-(\mu_{b_h}-\mu_{d_h})V_{bt}} . \tag{10}$$

3.3 Deblurring Veins

A vein image without skin wrinkles and obtained by incident light of wavelength λ_i can be described as follows:

$$g'_i = (f_i \otimes h_i) + n_i. \tag{11}$$

The wavelength-dependent extinction coefficients of dermis, epidermis and blood can be determined [12]. Suppose that two wavelengths λ_i and λ_j are chosen such that the extinction coefficient of skin at wavelength λ_i (defined as μ_{s_i}) is twice as large than that at the other wavelength λ_j (i.e., $\mu_{s_i} = 2\mu_{s_j}$). When light is absorbed without scattering in accordance with Lambert-Beer law, we have $\exp^{-\mu_{s_i} d} = \exp^{-\mu_{s_j} d} \exp^{-\mu_{s_j} d}$.

This means that the absorbance at λ_i can be estimated from that at λ_j. We hypothesized that the $\exp^{-\mu_{s_i} d} = \exp^{-\mu_{s_j} d} \exp^{-\mu_{s_j} d}$ would also hold approximately for scattered light situation. Consequently, the PSFs for two different wavelengths are related to each other as follows.

$$h_i \approx h_j \otimes h_j. \tag{12}$$

This shows that the PSF h_i at the depth d is approximately equal to the PSF h_j at the depth $2d$.

To confirm Eq. (12), we conducted a preliminary experiment using Monte Carlo simulation [13]. We selected the two wavelengths 620 nm and 870 nm (the selection of wavelengths is discussed later) and denoted them λ_0 and λ_2, respectively. The PSFs, h_0, h_2 and $h_2 \otimes h_2$ obtained from Eq. (12) are shown in Fig. 5, where we can see that shapes of h_0 and $h_2 \otimes h_2$ are similar to each other. This means that Eq. (12) provides a good approximation regardless of the thickness of the skin layer.

Using this relationship, we can obtain f_i from Eq. (11) in the following way. Substituting Eq. (12) into Eq. (11), we obtain

$$g'_i = (f_i \otimes h_j) \otimes h_j + n_i. \tag{13}$$

When we consider the differences between images that have different degrees of blurring and set $i = 0$ and $j = 2$, we can described the differences as follows:

$$f_2 - g'_2 = (f_2 - f_2 \otimes h_2) - n_2, \tag{14}$$

$$g'_2 - g'_0 = (f_2 - f_0 \otimes h_2) \otimes h_2 + n_2 - n_0. \tag{15}$$

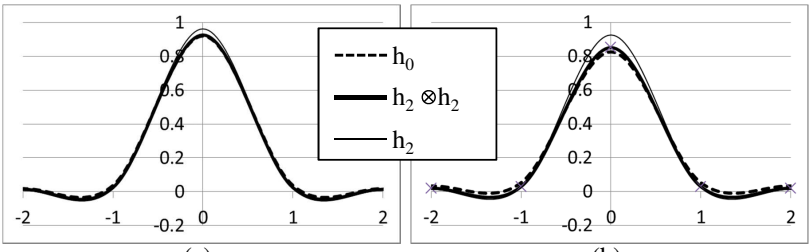


Fig. 5. Simulation results validating PSF approximation Eq.(12): (a) Dermis = 0.5 mm, (b) Dermis = 1.0 mm

If we chose two wavelengths such that the two extinction coefficients of blood are similar to each other, a relationship $f_0 \approx f_2$ can be obtained. Therefore, these two equations simplify to:

$$f_2 = g_2' + (g_2' - g_0' - n_2 + n_0) \otimes h_2^{-1} - n_2 \triangleq g_2' + f_H, \tag{16}$$

where f_H is a high-frequency component of the true vein image f_2. That is, $f_H \triangleq (g_2' - g_0' - n_2 + n_0) \otimes h_2^{-1} - n_2$, and $a \otimes b^{-1}$ means deconvolution of a and b.

The PSF h_2, which is the unknown PSF of the vein image f_2, can be obtained from Eq. (11) and Eq. (13):

$$h_2 = g_0' \otimes g_2'^{-1} + n_{h2}, \tag{17}$$

where n_{h2} is a noise component of h_2.

Therefore the image f_H can be described as following equation without h_2:

$$f_H = (g_2' - g_0') \otimes g_2' \otimes g_0'^{-1} + n_H, \tag{18}$$

where n_H is an image noise of f_H.

Finally, the true vein image f_2 can be obtained by using Eq. (16) and Eq. (18).

3.4 Illumination Wavelengths

In this work we selected the three wavelengths $\lambda_0 = 620$ nm, $\lambda_1 = 690$ nm, and $\lambda_2 = 870$ nm for the following reasons (also see Fig. 6):

1. 870 nm is a near-infrared wavelength widely used for clear imaging of finger veins [4, 14].
2. At 620 nm the extinction coefficient of skin is approximately twice as large as it is at 870 nm ($h_0 \approx h_2 \otimes h_2$) (see Fig. 6(a)).
3. At 620 nm the extinction coefficient of blood is approximately the same as it is at 870 nm ($f_0 \approx f_2$) (see Fig. 6(a)).
4. At 690 nm the extinction coefficient of blood is lower than it is at 620 or 870 nm (see Fig. 6(b)).

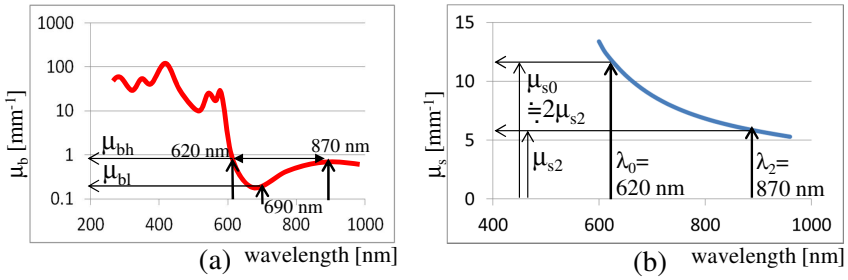


Fig. 6. Relation between wavelength and the absorption coefficient of oxyhemoglobin [15, 12]

3.5 Algorithms

Our method comprises three procedures: capturing images by using three wavelengths, removing skin wrinkles in two images and deblurring vein images.

Capturing Three Images

1. Irradiate a body with light at wavelength λ_i (i=0, 1 and 2) at light power P_i.
2. Capture an image and calculate an average of brightnesses of the image B_a.
3. Update $P_i \leftarrow (B_t/B_a)P_i$ so that the average B_a comes close to a target brightness B_t.
4. Repeat until the average becomes stable and obtain the image g_i.

We set $B_t = 128$, which is a mid-value of an 8-bit gray scale image.

Removing Wrinkles

1. Initialize $V_{bt} = 0$.
2. Correct the average brightnesses of two images g_0 and g_1.
3. Calculate the value V_{bt} by using equation (7), and smoothen V_{bt}.
4. Use the top-hat transform on image g_{0_e}, and repeat steps 2–4 once.
5. Obtain g'_0 by using Eq. (10).
6. Repeat these procedures using g_2 instead of g_0 and obtain g'_2.

The smoothing of V_{bt} is required because the V_{bt} values are affected by skin wrinkle patterns. We use an averaging filter for the smoothing process, and we use a line as a structuring element in the top-hat transform. The length of the element is sufficiently longer than the width of the typical skin wrinkle.

Deblurring Veins. In the deblurring process, two images whose skin wrinkle patterns are already removed are used for deblurring calculation.

1. Clip partial image patches from images g'_0 and g'_2 at the position (x, y), and convert them into the frequency domain by using a fast Fourier transform.

2. Calculate f_H of the patches by using the Wiener filter.
3. Obtain f_H at whole images and smoothen f_H.
4. Calculate f_2 from equation (16)

We chose the Wiener filter for calculation of f_H because it is fast and easy to implement, and we chose an image patch around 1 mm $\times$ 1 mm square in size. The smoothing procedure is also required so that the calculation results of f_H would be robust despite image noises n_H.

4 Experiments

We made an experimental prototype for capturing three vein images as shown in Fig. 7. The captured image is 256 $\times$ 256 pixels and the captured area is 10 mm $\times$ 10 mm. The captured position is on the palm side of a finger, near a joint.

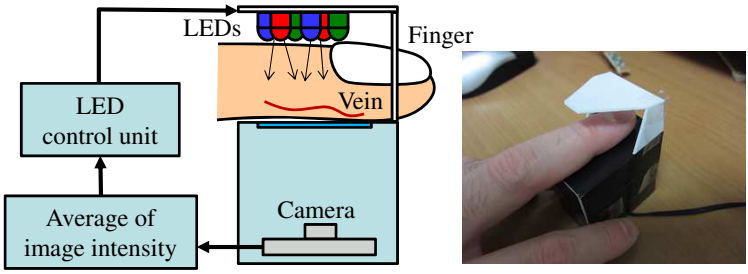


Fig. 7. Illustration of experimental equipment

4.1 Skin Wrinkle Removing and Deblurring Veins

Fig. 8 shows results obtained using the proposed method for wrinkle removal. Figs. 8(a) and (b) are the original input images g_1 and g_2, (c) is a vein-eliminated image derived from (b), (d) is a skin wrinkle image extracted by using a top-hat transform, and (e) is a wrinkle-eliminated image. For comparison, a result of a conventional method using a top-hat transform applied to g_2 is also shown in Fig. 8(f). We can see from these results that finger vein image has a lot of skin wrinkle patterns, and the removed result clearly shows the wrinkle patterns are eliminated. The cross-sectional profiles of the image indicated by arrows show that the proposed method can clearly eliminate wrinkles while they slightly remain by using the conventional method (Fig. 8(g)).

Fig. 9 shows results obtained using the proposed method for deblurring vein images. Figs. 9 (a), (b) and (c) are respectively an input image g_2, a wrinkle-removed image g_2', and vein-pattern-deblurring result f_2. Cross-sectional profiles of Figs. 9(a), (b) and (c) are shown in Fig. 9(g) in order to show the effectiveness of our method.

We define the contrast of veins as V_2/V_1, where V_1 is a darkness of a vein in g_2', and V_2 is that in f_2. As shown in Fig. 9(g), our method improved the contrast of

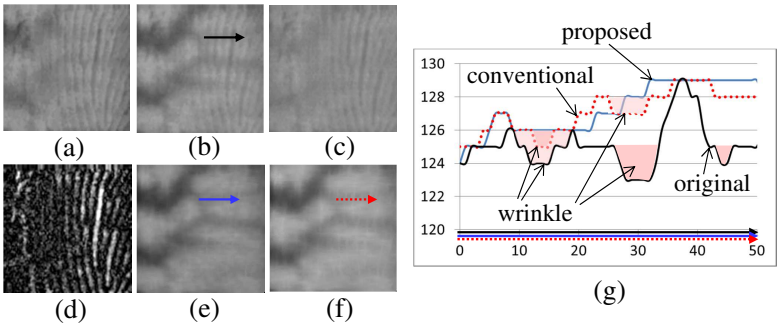


Fig. 8. Skin wrinkle elimination: (a) g_1, (b) g_2, (c) vein-eliminated image form (b), (d) eliminated wrinkles, (e) result of our method, (f) result of conventional method, (g) cross-sectional profiles of (b), (d) and (f)

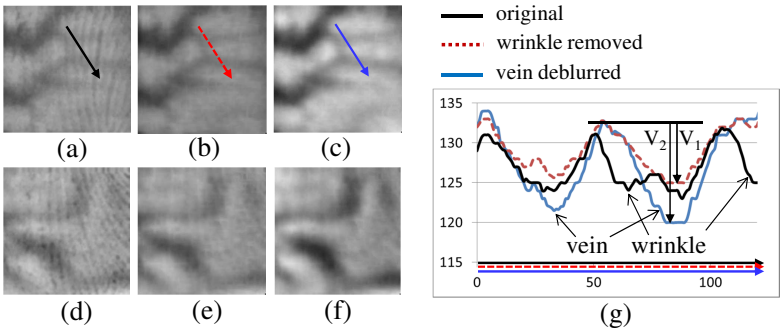


Fig. 9. Deblurring vein pattern: (a) g_2, (b) wrinkle-elimination result g_2', (c) deblurring result f_2, (d), (e) and (f) results for another sample, (g) cross-sectional profiles of (a), (b) and (c)

a not deblurred image by 55.3%. Our method is therefore expected to improve the performance of vein-extraction algorithms used in biometric identification technologies.

4.2 Identification of Fingers

To confirm the effectiveness of our method, we used it in a finger-vein-pattern-based personal identification procedure proposed by Miura et al. [1]. With that procedure, skin wrinkle patterns on fingers are an obstacle to stable and accurate extraction of finger vein patterns because the finger wrinkles patterns vary with skin conditions, the adhesion of dirt, and so on.

From four volunteers we acquired vein images of 14 fingers. For each finger we acquired one registration image and three input images. Furthermore, we assumed that skin wrinkle patterns will not be observed in the registration process

but will be clearly observed in the input trial, and that the position of the finger will be slightly different at each input trial. The reason we assume the wrinkle patterns will not be observed in the registration process is that, in actual application of a biometric security system, a system operator often checks the condition of a finger presented by a user during the registration process in order to determine whether or not the finger is suitable.

We consider that the dataset size (14 fingers, 3 trials, 588 matchings) is sufficient to demonstrate the effectiveness of our method because the experiments were intended not to evaluate the accuracy of our method but simply to demonstrate that it is more accurate than a conventional method. The number of volunteers in biometric research studies is generally larger than the number we used in our experiments because a large dataset is needed to estimate the accuracy of a personal authentication technique precisely and the estimated accuracy is extremely high.

Fig. 10 shows the results of vein pattern extraction and estimation of the error rate for finger identification. Figs. 10(a) and (b) are respectively original input images captured in the registration process and input trial. Figs. 10(c) and (d) are vein patterns extracted form (a) and (b) when not using our method, and Figs. 10(e) and (f) are patterns extracted from (a) and (b) when using our method. We can see that the vein pattern extracted when not using our method contains wrinkle patterns (indicated by dotted circles in Fig. 10(d)), whereas the vein pattern (c) extracted from the image (a) in the registration process has no wrinkles. On the other hand, both patterns (e) and (f) obtained using our method have no wrinkles and are similar to each other. These results show that our method eliminated skin wrinkles effectively and therefore can be expected to lead to good identification results.

Fig. 10(g) shows the results obtained in an experiment evaluating the original method without wrinkle removal [2], the method with the conventional top-hat transform for skin wrinkle elimination mentioned above, and our method. The graph shows ROC (receiver operation characteristic) curves, which represent the relation between the FAR (false acceptance rate) and FRR (false rejection rate). These ROCs were obtained by using statistical fitting [16]. To quantitatively evaluate these results, we calculated the EER (equal error rate), i.e., the rate of trials where the FAR equaled the FRR. The results indicated that EER was 2.13% when our method of used, was 2.94% when the original method was used and was 2.32% when the conventional top-hat transform method was used. Therefore our method could reduce the EER by 27.5% from that of the original method while the conventional method reduced it by only 21.1%.

This shows that our method can be used to make a highly accurate finger vein authentication system. Moreover, we think our method should perform equally well for other datasets. The dataset used in our study was created so that the appearance of wrinkle patterns closely resembled that observed in a real situation by adjusting the adherence of cosmetics. We therefore expect to have similar results for other datasets as long as they contain wrinkle patterns with a realistic appearance.

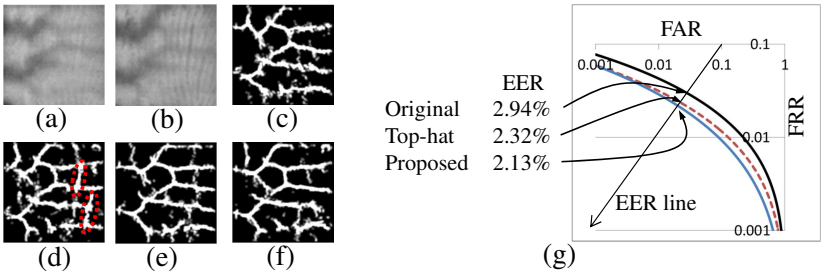


Fig. 10. Finger vein identification [2]: (a) an image obtained in a registration process, (b) an image obtained in an input trial, (c) and (d) vein patterns obtained from (a) and (b) when using the original method [2], (e) and (f) vein patterns obtained from (a) and (b) when using our method, (g) ROC curves

5 Conclusion

In this paper we propose a method for removing skin wrinkle patterns and image blur from vein images by using tri-band illumination. The experimental results showed that our method could eliminate skin structure patterns and reduce the blurring of vein pattern images. Our method resulted in better image contrast than a conventional method did and improved the accuracy of the personal identification based on finger vein patterns by 27.5%.

Our future work will include evaluating the feasibility of downsizing finger vein authentication devices by using our method. Device downsizing would be required if these devices were used in mobile equipment such as cell phones and laptop PCs. Estimating the depth of finger veins is another interesting topic. It would enable us to obtain 3D finger vein structures, which would be a key technology for next-generation tomographic finger vein authentication.

References

1. Miura, N., Nagasaka, A., Miyatake, T.: Extraction of finger-vein patterns using maximum curvature points in image profiles. IEICE Transactions on Information and Systems E90-D, 1185–1194 (2007)
2. Miura, N., Nagasaka, A., Miyatake, T.: Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. Machine Vision and Applications 15, 194–203 (2004)
3. Muramatsu, C., Hatanakab, Y., Iwasea, T., Haraa, T., Fujita, H.: Automated detection and classification of major retinal vessels for determination of diameter ratio of arteries and veins. In: Proc. of SPIE Medical Imaging 2010, vol. 7624, pp. 76240J1–76240J8 (2010)
4. Shimizu, K.: Biometrics - personal identification by measurement of biological traits -. Trans. of Japanese Society for Medical and Biological Engineering 44, 3–14 (2006)

5. Shimizu, K., Tochio, K., Kato, Y.: Improvement of transcutaneous fluorescent images with a depth-dependent point-spread function. Applied Optics 44, 2154–2161 (2005)
6. Peters, R.A.: A new algorithm for image noise reduction using mathematical morphology. IEEE Transactions on Image Processing 4, 554–568 (1995)
7. Nishidate, I., Aizu, Y., Mishina, H.: Depth visualization of a local blood region in skin tissue by use of diffuse reflectance images. Optics Letters 30, 2128–2130 (2005)
8. Kim, J., Lanman, D., Mukaigawa, Y., Raskar, R.: Descattering Transmission via Angular Filtering. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 86–99. Springer, Heidelberg (2010)
9. Mukaigawa, Y., Raskar, R., Yagi, Y.: Analysis of scattering light transport in translucent media. IPSJ Transactions on Computer Vision and Applications 3, 122–133 (2011)
10. Paquit, V.C., Tobin, K.W., Price, J.R., Me'riaudeau, F.: 3d and multispectral imaging for subcutaneous veins detection. Optics Express 17, 11360–11365 (2009)
11. Tsumura, N., Ojima, N., Sato, K., Shiraishi, M., Shimizu, H., Nabeshima, H., Akazaki, S., Hori, K., Miyake, Y.: Image-based skin color and texture analysis/synthesis by extracting hemoglobin and melanin information in the skin. Proc. ACM Transactions on Graphics, SIGGRAPH 2003 22, 770–779 (2003)
12. Jacques, S.L.: Skin optics. Oregon Medical Laser Center News (1998)
13. Wang, L., Jacques, S.L., Zheng, L.: Mcml - monte carlo modeling of photon transport in multi-layered tissues. Computer Methods and Programs in Biomedicine 47, 131–146 (1995)
14. Yu, C.B., Qin, H.F., Zhang, L., Cui, Y.Z.: Finger-vein image recognition combining modified hausdorff distance with minutiae feature matching. J. Biomedical Science and Engineering 2, 261–272 (2009)
15. Wray, S., Cope, M., Delpy, D.T., Wyatt, J.S., Reynolds, E.R.: Characterization of the near infrared absorption spectra of cytochrome aa3 and hemoglobin for the non-invasive monitoring of @cerebral oxygenation. Biochimica et Biophysica Acta (BBA) 933, 184–192 (1988)
16. Takashi, Y., Aoki, S., Ohyama, T.: Human finger vein images are diverse and its patterns are useful for personal identification. In: 56th Session of the International Statistical Society (2007)

Reconstruction of 3D Surface and Restoration of Flat Document Image from Monocular Image Sequence

Hiroki Shibayama, Yoshihiro Watanabe, and Masatoshi Ishikawa

Graduate School of Information Science and Technology, University of Tokyo, Japan
{Hiroki_Shibayama,Yoshihiro_Watanabe,
Masatoshi_Ishikawa}@ipc.i.u-tokyo.ac.jp

Abstract. There is a strong demand for the digitization of books. To meet this demand, camera-based scanning systems are considered to be effective because they could work with the cameras built into mobile terminals. One promising technique proposed to speed up book digitization involves scanning a book while the user flips the pages. In this type of camera-based document image analysis, it is extremely important to rectify distorted images. In this paper, we propose a new method of reconstructing the 3D deformation and restoring a flat document image by utilizing a unique planar development property of a sheet of paper from a monocular image sequence captured while the paper is deformed. Our approach uses multiple input images and is based on the natural condition that a sheet of paper is a developable surface, enabling high-quality restoration without relying on the document structure. In the experiments, we tested the proposed method for the target application using images of different documents and different deformations, and demonstrated its effectiveness.

1 Introduction

There is a growing demand for camera-based document analysis and recognition. Book digitization, a relatively new application of camera-based document capturing which images all of the pages in a book, has a rapidly expanding market globally and is attracting various types of potential users, including libraries, corporations, and general users of books, official documents, and notes. However, the conventional technology cannot meet the demands for ease-of-use and high-speed book digitization. One emerging solution that meets these demands is *Book Flipping Scanning* [1]. This is a new style of scanning in which all pages of a book are captured while a user continuously flips through the pages without stopping at each page. Although this new technology has had a tremendous impact in the field of book digitization, a compact, expensive, high-speed 3D sensing module would be required to allow a massive number of potential users to try this scanning style with portable terminals such as mobile phones.

Focusing on this promising new approach, we attempted to implement *Book Flipping Scanning* with a single camera. Fig. 1 shows our concept. With this

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 350–364, 2013.

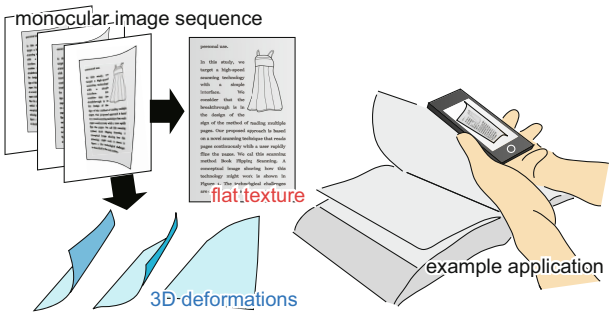


Fig. 1. Our method reconstructs surface deformations and flat document images using a monocular image sequence

system, the user can hold a camera and merely flip through a document held in front of it, allowing all information to be captured rapidly and in high definition.

For this purpose, the rectification problem described above is also essential. In our target application, multiple images are assumed as the input. This assumption enables a new possibility of allowing a flat document to be restored even if the document structure is unknown. However, because paper is a non-rigid surface, a deformation model is needed to realize this approach. A sheet of paper exhibits the characteristics of a so-called developable surface [2–5], an interesting property that can be exploited to meet this requirement.

Based on the design concept, this paper describes a new rectification technique for reconstructing 3D deformations and restoring a flat document image from multiple monocular images captured from a deformed sheet of paper. The deformations and restoration of a flat document image are not independent; however, the complementary relationships described by the developable property of a sheet of paper enables us to solve this problem. The proposed method employs a different approach from the conventional ones where the document structure in a single captured image is utilized. In particular, this technique is suitable for high-speed book digitization with a single camera, an application which is expected to show explosive growth.

2 Related Work

Book Digitization Using a Camera. Obtaining the 3D surface shape of a document simplifies unwarping of the captured image because this operation can be described as a planar development. For example, methods employing a stereo-camera system [6] and the so-called Shape from Shading technique [7, 8] are proposed. As for the scanning styles employed, these are typically the same as those used in flatbed scanners, requiring the user to scan page-by-page. In order to speed up book digitization, *Book Flipping Scanning* was developed to scan all of the pages in a book while a user flips through the pages, by obtaining the 3D surface shape using a high-speed, structured-light-based sensing system [1]. In addition, distorted images with various deformations can be restored to a

flat document image using a model-based estimation involving the developable-surface condition [3]. These systems realize camera-based book scanning based on 3D measurement techniques. In contrast, our challenge here is to develop a new book digitization technique similar to *Book Flipping Scanning* but based on a single-camera configuration.

Rectification of Warped Document Images. Recently, optical character recognition (OCR) has found new potential in a vast array of applications involving capturing document images with mobile phone cameras. However, OCR is less accurate when the characters are distorted. Although new OCR techniques have been proposed recently [9, 10], basically a rectification step is required. The typical methods utilize lines of texts. The rectification can be described using the assumption that the lines of text are horizontal in the flat document. However, extraction of lines of text is still challenging in various page layouts in which text and figures are mixed [4, 11, 12]. Also, there are two types of rectification techniques, describing the estimation as 2D warping [4, 13] and 3D reconstruction [12]. These methods solve the rectification problem under the assumption that the document contents and its structure are known. In addition, although there is a method utilizing the known bounding contour [14], it is difficult to detect it in case of book and there are typical cases where only insensible differences of contours between images can be found in spite of the large differences in 3D deformations. Instead of those approach, we assume the use of multiple images of deformed sheets of paper and describe the deformation as a fundamental property of paper in terms of a developable surface.

Non-rigid Surface Estimation. There have been some studies dealing with non-rigid objects, such as registration techniques between a reference image on a flat surface and a warped image on a deformed surface [15–17]. In addition, there are methods of reconstructing 3D surfaces from monocular video by using this kind of image registration, such as a learning-based approach [18] and a model-based approach [19, 20]. In particular, the model-based approach utilizes the inextensibility of the target material as a constraint. The limitation of those methods is that they require a template image captured in advance with known deformation of the target. In contrast, a method that requires no reference image has been proposed [21]. Although the preconditions for the methods meet our target application, high-quality restoration of a flat image is not the main target of that work. Also, the surface deformation is modeled based on a mesh structure, which increases the complexity of the inverse problem depending on the resolution of the mesh.

3 Reconstruction of Deformation of 3D Surface and Its Flat Texture

3.1 Overview of the Proposed Method

In this paper, the restoration of a single page is considered. An entire book can be digitized by applying this method to each captured page. In our method,

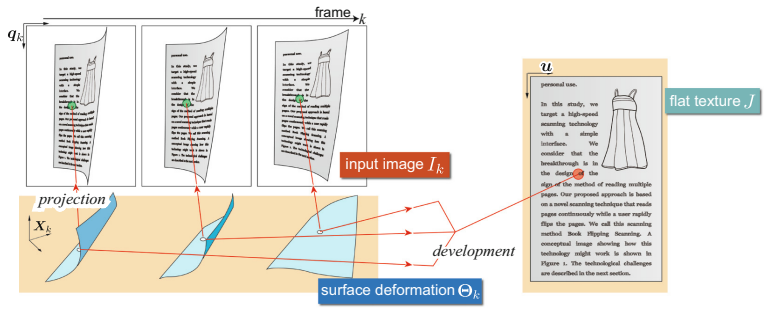


Fig. 2. Problem structure illustrating the correspondence mappings

the inputs are N images $\{I_k\}_{k=1}^N$ of a deformed page captured by a monocular camera. The outputs are N deformation shapes $\{\boldsymbol{\Theta}_k\}_{k=1}^N$ and a single restored flat document image J. Each shape $\boldsymbol{\Theta}_k$ is observed in each input image I_k.

The assumptions for this data are as follows. A single camera whose intrinsic parameters are known in advance captures the input images. The output restored image J is assumed to be a texture on the surface when it is developed into a plane. The target object is assumed to be developed into a plane without any extension, contraction, or tearing, which means that the Gaussian curvature is identically zero over the entire surface [2]. Also, in order to improve the accuracy, the aspect ratio of the paper is known in advance. Considering actual applications such as book digitization, these assumptions are considered to be acceptable.

The observation situation is explained as follows. At every frame, the shape of the target object continues to be deformed. The camera captures the texture deformation depending on the shape deformation in each input image I_k. The captured texture deformation contains clues to reconstruct the shape $\boldsymbol{\Theta}_k$ and the original flat texture J. This is illustrated in Fig. 2.

We model the transformation between two coordinate systems, those of the k-th camera image and the flat document image, as follows:

$$\boldsymbol{q}_k = \boldsymbol{g}\left(\boldsymbol{u}; \boldsymbol{\Theta}_k\right) = \boldsymbol{g}_k(\boldsymbol{u}) \tag{1}$$

In this equation, the point $\boldsymbol{u}$ on the flat image is projected onto the point $\boldsymbol{q}_k$ on the input image plane through process $\boldsymbol{g}_k$ involving the deformation and observation process, which is controlled by the deformation parameter $\boldsymbol{\Theta}_k$ in our task. Using Equation (1), all of the pixels can be projected onto the captured image. Therefore, each camera image I_k can be formulated as follows:

$$I_k\left(\boldsymbol{q}\right) = \sum_{\boldsymbol{u} \in \{\boldsymbol{u} \mid \|\boldsymbol{g}_k(\boldsymbol{u}) - \boldsymbol{q}\| \leq \epsilon\}} B\left(\|\boldsymbol{g}_k(\boldsymbol{u}) - \boldsymbol{q}\|\right) J(\boldsymbol{u}) \tag{2}$$

Here, $I_k(\boldsymbol{q})$ and $J(\boldsymbol{u})$ represent the brightness value at each point $\boldsymbol{q}$ and $\boldsymbol{u}$, respectively. Also, $B\left(d\right)$ is the lens blurring effect (the Point Spread Function, or PSF). In this paper, we define the PSF by using a Gaussian operation.

The target can deform its shape freely under the developable conditions. Although our method allows such free deformation, the developed flat image J is

unique, as shown in Fig. 2. In this way, introducing the idea of development gives temporal consistency between the captured images. Therefore, increasing the number of input images increases the number of conditions, and this framework enables us to construct an environment for solving the inverse estimation problem for the surface deformations and the developed flat document image. Also, as the number of input images increases, the synthesized information for estimating a single restored, unwarped image becomes more rich, so that a high-resolution image can be recovered. In addition, the high-quality developed flat image J can recursively contribute to improvement of the accuracy in surface estimation.

Based on this approach, we propose estimating the surface deformations and a developed image from multiple images captured by a monocular camera based on the following framework:

$$\min_{\{\tilde{\boldsymbol{\Theta}}_k\}_{k=1}^N, \tilde{\boldsymbol{J}}} \sum_k \sum_{\boldsymbol{q}} \|I_k(\boldsymbol{q}) - \tilde{I}_k(\boldsymbol{q}; \tilde{\boldsymbol{\Theta}}_k, \tilde{\boldsymbol{J}})\|^2 + f(\tilde{\boldsymbol{J}}) \tag{3}$$

$\tilde{\boldsymbol{J}}$ is the vector aligning the brightness value of the image $\tilde{J}$. Other vector notations have similar definitions. $\tilde{I}_k$ is generated from the estimated shape $\tilde{\boldsymbol{\Theta}}_k$ and the restored image $\tilde{\boldsymbol{J}}$. The second term is a regularization term, and the function f constrains the restored image J. In our experiments, the square norm of the Laplacian image of the restored image was used.

In order to formulate this framework concretely, it is crucially important to describe the developable non-rigid body with a small parameter set so as not to increase the problem scale. In this paper, we introduce the concept of a developable surface used in the field of differential geometry. If another typical model is used to represent the deformation, uniqueness is no longer ensured, and it is difficult to maintain stable estimation accuracy. Details of the developable surface modeled in this paper are shown in Section 3.2.

Also, in the problem represented in Equation (3), simultaneous estimation of both the deformation $\{\tilde{\boldsymbol{\Theta}}_k\}_{k=1}^N$ and the restored image $\tilde{J}$ is required. However, such simultaneous estimation complicates the inverse problem, making the solution difficult to reach. Therefore, in our method, the surface and the developed texture are estimated alternately. In addition, our method estimates the solution based on nonlinear optimization, so that initial estimation parameters need to be given. Based on these aspects, our method is organized as follows.

1. **Initial deformation estimation.** In this step, the initial deformation parameters are estimated. The estimation is allowed to be rough.
2. **Reconstruction of the 3D deformation.** Using the input image I_k and the restored image J, each 3D surface deformation $\boldsymbol{\Theta}_k$ is estimated.
3. **Reconstruction of the flat texture.** Using all of the input images $\{I_k\}_{k=1}^N$ and the estimated deformations $\{\boldsymbol{\Theta}_k\}_{k=1}^N$, a developed flat image J is restored.
4. **Iteration** Steps 2 and 3 are iterated until convergence.

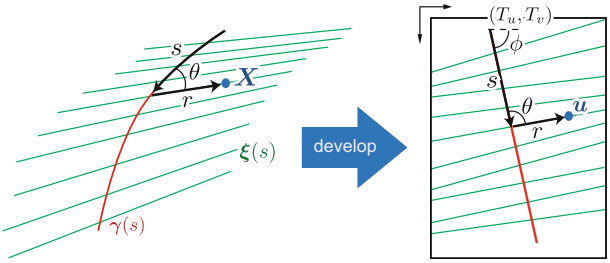


Fig. 3. Developable surface and its planar development

3.2 Developable Surface for Monocular Reconstruction

The developable surface is parameterized by the family of lines $(\boldsymbol{\gamma}(s), \boldsymbol{\xi}(s))$. Here, $\boldsymbol{\gamma}$ is a curve called the *directrix*. Also, $\boldsymbol{\xi}$ is a vector called the *ruling*. The Gaussian curvature is identically zero over the entire surface [2]. This condition makes it possible to develop the surface to a plane. An example developable surface is shown in Fig. 3.

A 3D point $\boldsymbol{X}$ on the surface is described by two parameters (s, r), as in the following equations:

$$\boldsymbol{X}(s,r) = \boldsymbol{\gamma}(s) + r\boldsymbol{\xi}(s), \quad \boldsymbol{\xi}(s) = \frac{\boldsymbol{\gamma}''(s) \times \boldsymbol{\gamma}'''(s)}{\|\boldsymbol{\gamma}''(s) \times \boldsymbol{\gamma}'''(s)\|} \tag{4}$$

Parameters (s, r) mean the lengths of the *directrix* and *ruling*, respectively. In other words, the 3D point $\boldsymbol{X}$ on the surface can be reached after moving by s along the *directrix* and moving by r along the *ruling*.

In this definition, once the curve $\boldsymbol{\gamma}$ is defined, the entire surface is determined. This means that in the surface reconstruction, we need to estimate only the parameters describing the *directrix*. This is a powerful advantage because the introduction of the developable surface model does not make the solution complex.

In addition, in the developable surface, on the planar development, *directrix* and *ruling* are transformed to straight lines. Therefore, as shown in Fig. 3, the coordinates on the flat image are as follows:

$$\boldsymbol{u} = \begin{bmatrix} T_u \\ T_v \end{bmatrix} + s \begin{bmatrix} \cos\phi \\ \sin\phi \end{bmatrix} + r \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \tag{5}$$

Here, the registration parameters are set to $\boldsymbol{\Phi} = (T_u, T_v, \phi)$, and θ is the angle between the transformed *directrix* and *ruling*, which can be obtained from the surface model.

As the curve $\boldsymbol{\gamma}$, we use the following Bezier curve:

$$\boldsymbol{B}(t) = \sum_{i=0}^{n-1} \boldsymbol{p}_i \binom{n-1}{i} t^i (1-t)^{n-1-i} \tag{6}$$

Here, the points $\boldsymbol{P} = (\boldsymbol{p}_0, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_{n-1})$ are the control points of the Bezier curve. The developable surface deformation can be defined uniquely only if the control points are provided. The parameter t in the Bezier curve can be transformed to the arc length s. In this case, a position on the surface is defined using the 2D coordinates $\boldsymbol{\eta} = (t, r)$.

Based on these definitions, the deformation parameter $\boldsymbol{\Theta}$ is defined as follows:

$$\boldsymbol{\Theta} = \{\boldsymbol{\Phi}, \boldsymbol{P}\} \tag{7}$$

3.3 Initial Estimation for Recovering Rough Surfaces

This section describes the initial estimation step. Our method estimates each deformation by using two input images. First, we get the corresponding feature points in two images. Also, in the steps of our deformation recovery, we utilize parts of the page outline, including the four corners and some points on the edges. The numbers and positions of the points on the edges are arbitrary. In the experiments, we set the image pair as two successive images in the captured video. Also, in order to remove the outliers caused by the same characters appeared in a page, we applied adaptive patch segmentation in the images and found the corresponding points only between the patches located in the same position in two images. As feature descriptors, SURF features were used [22].

The corresponding positions on the two input images I_A and I_B are $\boldsymbol{q}_A$ and $\boldsymbol{q}_B$, respectively. Similarly, the coordinates on the surfaces are $\boldsymbol{\eta}_A$ and $\boldsymbol{\eta}_B$, and the coordinates on the restored flat image are $\boldsymbol{u}_A$ and $\boldsymbol{u}_B$. The values with the tildes mean estimated ones.

If the two points $\boldsymbol{q}_A$ and $\boldsymbol{q}_B$ are corresponding points, the estimated points $\tilde{\boldsymbol{u}}_A$ and $\tilde{\boldsymbol{u}}_B$ also are required to be located at the same position. The estimation errors for the feature points, the corners, and the edges are defined as Equations (8), (9) and (10), respectively:

$$E_{fp} = \sum_{i=1}^{N_{fp}} v_{fp}(\|\boldsymbol{q}_A^i - \tilde{\boldsymbol{q}}_A^i\|^2 + \|\boldsymbol{q}_B^i - \tilde{\boldsymbol{q}}_B^i\|^2) + w_{fp}\|\tilde{\boldsymbol{u}}_A^i - \tilde{\boldsymbol{u}}_B^i\|^2 \tag{8}$$

$$E_c = \sum_{I=A,B} \sum_{j_I=1}^{N_{cI}} v_c\|\boldsymbol{q}_I^{j_I} - \tilde{\boldsymbol{q}}_I^{j_I}\|^2 + w_c\|\boldsymbol{u}_I^{j_I} - \tilde{\boldsymbol{u}}_I^{j_I}\|^2 \tag{9}$$

$$E_e = \sum_{I=A,B} \sum_{k_I=1}^{N_{eI}} v_e\|\boldsymbol{q}_I^{k_I} - \tilde{\boldsymbol{q}}_I^{k_I}\|^2 + w_e\|\boldsymbol{u}_I^{k_I} - \tilde{\boldsymbol{u}}_I^{k_I}\|_{(x,y)}^2 \tag{10}$$

Here, N_{fp}, N_{cI}, and N_{eI} are the total numbers of each point; i, j_I, and k_I are indices; and v_{fp}, ω_{fp}, v_c, ω_c, v_e, and ω_e are constant weights. The target position of the flat-image coordinates for the contour points are calculated from the given aspect ratio. In the second term of Equation (10), the distance only in the x or y coordinate is calculated, depending on the edge direction.

We assume a sample with various deformations described using N_{pca} principal vectors based on principal component analysis (PCA), as follows:

$$\boldsymbol{P} = \sum_{i=1}^{N_{pca}} \alpha_i \boldsymbol{\rho}_i \tag{11}$$

This sample deforms its shape at a fixed position. Therefore, in order to fit the sample deformation to the observed one, a coordinate transformation including rotation $\boldsymbol{R}$ and translation $\boldsymbol{T}$ must be estimated.

Based on this configuration, the initial deformations are estimated using two images:

$$\min_{\tilde{\boldsymbol{\alpha}}^A, \tilde{\boldsymbol{\alpha}}^B, \tilde{\boldsymbol{H}}^A, \tilde{\boldsymbol{H}}^B, \tilde{\boldsymbol{\Phi}}^A, \tilde{\boldsymbol{\Phi}}^B, \tilde{\boldsymbol{R}}, \tilde{\boldsymbol{T}}} E_{fp} + E_c + E_e \tag{12}$$

Here, $\tilde{\boldsymbol{H}}^A$ and $\tilde{\boldsymbol{H}}^B$ are the sets of the corresponding points $\tilde{\boldsymbol{\eta}}^A$ and $\tilde{\boldsymbol{\eta}}^B$ on the surface. Normally the estimated points $\tilde{\boldsymbol{u}}$ on the restored flat image can be calculated from the observed points $\boldsymbol{q}$ and the deformation $\boldsymbol{\Theta}$. However, this calculation cannot be analytically defined. Therefore, instead of employing this calculation, we add the points $\boldsymbol{\eta}$ for estimation of the mapping between the observed point and the flat-image point. This problem is solved by using the Levenberg–Marquardt method.

3.4 Reconstruction of 3D Surface Deformation

In the surface reconstruction, the developed flat texture J is fixed. Each deformation $\boldsymbol{\Theta}_k$ for each captured image I_k is individually reconstructed. This reconstruction requires solving the following equation:

$$\min_{\{\tilde{\boldsymbol{\Theta}}_k\}_{k=1}^N} \sum_k \sum_{\boldsymbol{q}} \|I_k(\boldsymbol{q}) - \tilde{I}_k(\boldsymbol{q}; \tilde{\boldsymbol{\Theta}}_k)\|^2 + f(\boldsymbol{J}) \tag{13}$$

In order to search for the solution $\boldsymbol{\Theta}_k$, an iterative technique is essential. However, as the surface parameters change, the correspondence mapping $\boldsymbol{g}_k$ in Equation (1) must also change. In this case, its calculation must be repeated at every iterative update of the parameters. This leads to the critical problem of a long calculation time.

Therefore, in a similar way to the initial deformation estimation, instead of brightness evaluation, we evaluate the distance between the corresponding feature points in the input image I_k and the estimated captured image $\tilde{I}_k$, which is generated based on Equation (2), to reconstruct the surface deformation. Although the evaluation function is different from the reconstruction of a flat image, the meanings are essentially equivalent.

First, using the deformation parameter $\tilde{\boldsymbol{\Theta}}_k$ estimated in the previous iteration and the restored image J, the estimated captured image $\tilde{I}_k$ is generated. Between two images I_k and $\tilde{I}_k$, the corresponding feature points are obtained. Also, using the deformation parameter, the point $\boldsymbol{u}_i$ on the flat image for the point $\boldsymbol{q}_i$ on the observed image is calculated. In addition, corners and edge points are obtained

from the input images. However, if this is the first time for this step after the initial estimation, the deformation recovery is achieved by using only outline information.

The minimization problem for this step is described as follows:

$$\min_{\tilde{\boldsymbol{\Theta}}_k, \tilde{\boldsymbol{H}}} \sum_{i=1}^{N_p} v_i \|\boldsymbol{q}_i - \tilde{\boldsymbol{q}}_i\|^2 + w_i \|\boldsymbol{u}_i - \tilde{\boldsymbol{u}}_i\|^2 \tag{14}$$

Here, N_p is the total number of points, and i is the index, and v_i and w_i are constant weights. Three sets of weights are switched for the feature points, corners, and edge points. For the same reason as described in Section 3.3, the point set $\tilde{\boldsymbol{H}}$ is added for efficient estimation. The second term allows us to fix the flat image J. In the case of edge points, this term calculates the distance only in x or y direction as described in Section 3.3. This problem is solved by using the Levenberg–Marquardt method.

3.5 Flat Document Image Restoration

In this step, the target optimization problem is formulated as follows:

$$\min_{\tilde{\boldsymbol{J}}} \sum_k \sum_{\boldsymbol{q}} \|I_k(\boldsymbol{q}) - \tilde{I}_k(\boldsymbol{q}; \tilde{\boldsymbol{J}})\|^2 + f(\tilde{\boldsymbol{J}}) \tag{15}$$

Based on this minimization problem, the input images can be synthesized. This synthesis requires the function $\boldsymbol{g}_k$ shown in Equation (2). This function establishes the correspondence between the camera coordinates and the flat texture coordinates.

This correspondence process is decomposed into two stages. The first stage is to establish the correspondence between the 2D point $\boldsymbol{q}$ on the camera image and the 3D point $\boldsymbol{X}$ on the surface. The second stage is to establish the correspondence between the 3D point $\boldsymbol{X}$ on the surface and the 2D point $\boldsymbol{u}$ on the developed plane. This relationship is described in Section 3.2.

As a result, the correspondence between the camera coordinates and the flat texture coordinates is established, so that the problem formulated as Equation (15) can be solved. In this paper, we solve this problem by using the conjugate gradient method.

4 Experiments

We captured, in advance, various deformations while flipping through a book and generated sample deformations for the initial estimation. We set the number of control points for the *directrix* described in Equation (6) to three. The weights in the surface estimation were set to focus on the feature points compared with the points on the contour. The repeat count described in Section 3.1 was 5. As a comparison, we used a non-rigid registration technique based on a thin-plate spline (TPS) [23]. The unwarping was tested by giving the corner correspondences between the input image and the restored image.

4.1 Evaluations Using Various Datasets

In the experiment, the method was applied to a synthetic input dataset. We used 10 document images whose resolutions were 1,032 × 1,458 pixels. The original images are shown in Fig. 4. Using each image, the observed image sequences were created by applying deformations and rigid motion. We prepared four sets of sequences (two deformations and two rigid motions). The observed sequence consisted of three images. The input resolutions of the images shown in the figure were 2,048 × 2,048.

The restored results are shown in Fig. 5. We set the resolution at 1,032 × 1,458. In the figure, the input images, the image unwarped by TPS, and the image unwarped by the proposed method are shown. Also the example reconstructed

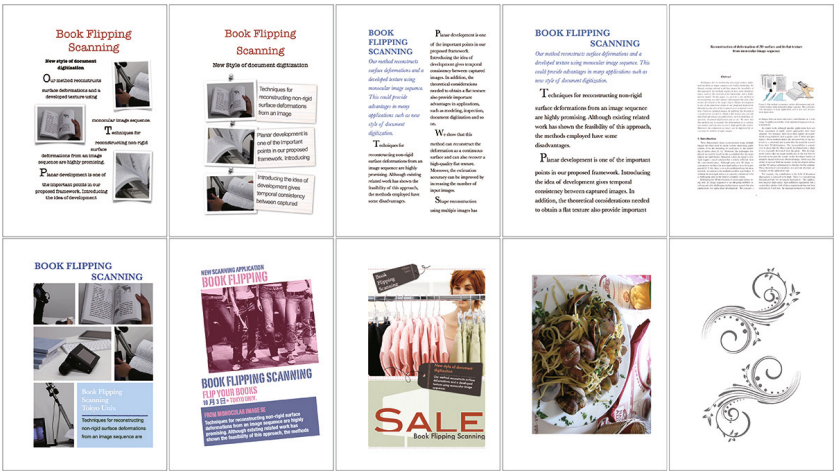


Fig. 4. The used original document images

Fig. 5. The synthesized input images and the restored images. Four example results are shown. In each example, the input image and the enlarged image (left), image unwarped by TPS (center), and image unwarped by the proposed method (right) are shown.

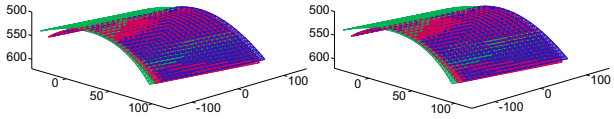


Fig. 6. The example reconstructed deformation. Left: the reconstructed deformation. Right: The true deformation.

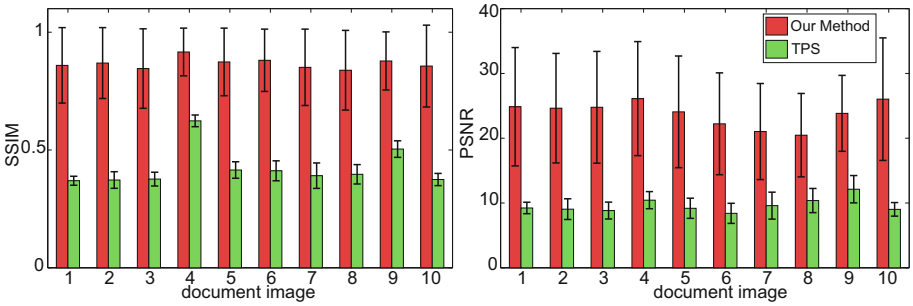


Fig. 7. The evaluated image quality. The label of the document image on the horizontal axis corresponds to the order in Fig. 4 (the upper left to the lower right).

deformation is shown in Fig. 6. In this figure, the reconstructed deformation and the true one are compared. The deformations observed in the three input images are drawn in different color meshes.

We also evaluated the image quality using peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) [24]. The evaluated quality is shown in Fig. 7. The 10 document images with four different input set were evaluated. Compared with TPS warping, our method restored high-quality images. Also, our method achieved rectification with various deformations, and changes of the target document did not significantly affect the accuracy. Restoration succeed even when only a part of the document had a small illustration.

4.2 Experiments Using Book Flipping Images

The experimental setup is shown in Fig. 8. The camera captured 2,048 × 2,048-resolution images at 180 fps while the user flipped the pages. In this setup, a pattern projector (multi-line pattern) was also provided to obtain the true deformation based on a structured light technique. The pattern was switched on and off alternately for every captured frame [1]. In this experiment, only the images in which the pattern was not projected were used. The input sequence consisted of three images.

Fig. 9 shows the restored image after the initial estimation. As shown in the figure, the quality in this step was not high enough. Also, Fig. 9 shows an example of feature point correspondence between the input image and the estimated one in the deformation reconstruction step.

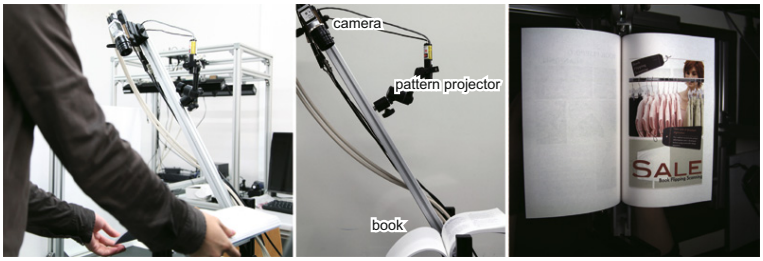


Fig. 8. Experimental setup and the captured images

Fig. 9. Restored images after initial estimation and the example of feature point correspondence in the deformation reconstruction

Fig. 10. Restored images of six example documents. In each example, the input image (left), image unwarped by TPS (center), and the image unwarped by the proposed method (right) are shown.

Fig. 10 shows the restoration results. The resolution of the rectified images was 495 × 740. In the figure, the input images and the images unwarped using TPS are also shown. Fig. 11 shows the examples of the errors between the input images and the estimated ones. The errors in the brightness values are shown.

Fig. 12 shows an example of the reconstructed deformation. In the figure, the reconstructed deformation and the true deformation measured based on a structured light technique are compared. Since the absolute error is difficult to be obtained because the reconstruction is based on a single camera, the quantitative comparison is shown. Fig. 13 shows the image quality evaluating the flipped 11-page documents. As the correct images, those shown in Fig. 4 were used. On the reconstructed image, non-uniform shading effect was left. This reduced the qualitative performance even when we achieved the geometric rectification

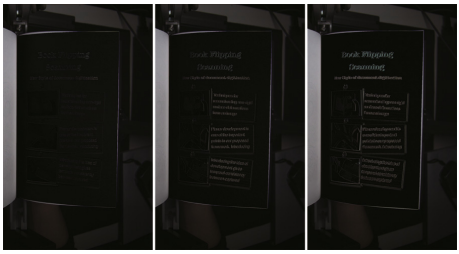


Fig. 11. Example of the error between the input image and the estimated one. The images shown are the difference images obtained by subtracting these images.

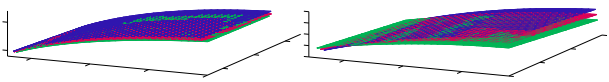


Fig. 12. Example of the reconstructed deformation. Left: the reconstructed deformation. Right: the measured deformation using the active stereo technique.

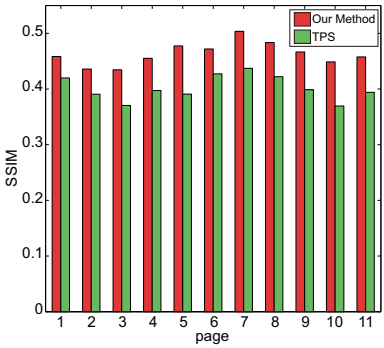


Fig. 13. The evaluated image quality

successfully. Therefore, the visibly good results in Fig. 10 was not reflected to the evaluation perfectly.

5 Conclusion

We have presented a new method for reconstructing 3D deformation and restoring an unwarped flat document image from multiple images. The planar development feature involved in the paper allows us to solve the problem based on temporal consistency between input images. Our method can handle various types of documents without knowing their structure and layout in advance. Experiments showed that our method worked well and reconstructed both the surface and the developed texture. In addition, the image quality was evaluated based on the PSNR and SSIM, and the results were considered to be acceptable. We confirmed that our proposed method was effective in realizing a new book scanning system.

Acknowledgments. This research was funded by Strategic Information and Communications R&D Promotion Programme (SCOPE).

References

1. Nakashima, T., Watanabe, Y., Komuro, T., Ishikawa, M.: Book flipping scanning. In: Adjunct Proceedings of UIST, pp. 79–80 (2009)
2. Carmo, M.P.D.: Differential Geometry of Curves and Surfaces. Prentice Hall (1976)
3. Watanabe, Y., Nakashima, T., Komuro, T., Ishikawa, M.: Estimation of non-rigid surface deformation using developable surface model. In: Proceedings of ICPR, pp. 197–200 (2010)
4. Cao, H., Ding, X., Liu, C.: Rectifying the bound document image captured by the camera: A model based approach. In: Proceedings of ICDAR, pp. 71–75 (2003)
5. Liang, J., DeMenthon, D., Doermann, D.: Unwarping Images of Curved Documents Using Global Shape Optimization. In: Proceedings of CBDAR, pp. 25–29 (2005)
6. Yamashita, A., Kawarago, A., Kaneko, T., Miura, K.T.: Shape reconstruction and image restoration for non-flat surfaces of documents with a stereo vision system. In: Proceedings of ICPR, pp. 482–485 (2004)
7. Zhang, Z., Tan, C.L., Fan, L.: Estimation of 3D shape of warped document surface for image restoration. In: Proceedings of ICPR, pp. 486–489 (2004)
8. Prados, E., Camilli, F.: A unifying and rigorous shape from shading method adapted to realistic data and applications. Journal of Mathematical Imaging and Vision 25, 307–328 (2006)
9. Hase, H., Shinokawa, T., Yoneda, M., Suen, C.: Recognition of rotated characters by eigen-space. In: Proceedings of ICDAR, pp. 731–735 (2003)
10. Narita, R., Ohyama, W., Wakabayashi, T., Kimura, F.: Three dimensional rotation-free recognition of characters. In: Proceedings of ICDAR, pp. 824–828 (2011)
11. Liang, J., DeMenthon, D., Doermann, D.: Geometric rectification of camera-captured document images. IEEE Transactions on Pattern Analysis and Machine Intelligence 30, 591–605 (2008)

12. Tian, Y., Narasimhan, S.G.: Rectification and 3D Reconstruction of Curved Document Images. In: Proceedings of ICCV, pp. 377–384 (2011)
13. Zhang, L., Tan, C.L.: Warped Image Restoration with Applications to Digital Libraries. In: Proceedings of ICDAR, pp. 192–196 (2005)
14. Gumerov, N.A., Zandifar, A., Duraiswami, R., Davis, L.S.: 3d structure recovery and unwarping of surfaces applicable to planes. International Journal of Computer Vision 66, 261–281 (2006)
15. Crum, W.R., Hartkens, T., Hill, D.L.G.: Non-rigid image registration: theory and practice. The British Journal of Radiology 77, 140–153 (2004)
16. Bartoli, A., Zisserman, A.: Direct estimation of non-rigid registrations. In: Proceedings of BMVC (2004)
17. Gay-Bellile, V., Bartoli, A., Sayd, P.: Direct estimation of nonrigid registrations with image-based self-occlusion reasoning. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 87–104 (2010)
18. Salzmann, M., Urtasun, R., Fua, P.: Local deformation models for monocular 3D shape recovery. In: Proceedings of CVPR (2008)
19. Salzmann, M., Pilet, J., Ilic, S., Fua, P.: Surface deformation models for nonrigid 3D shape recovery. IEEE Transactions on Pattern Analysis and Machine Intelligence 29, 1481–1487 (2007)
20. Brunet, F., Hartley, R., Bartoli, A., Navab, N., Malgouyres, R.: Monocular Template-Based Reconstruction of Smooth and Inextensible Surfaces. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part III. LNCS, vol. 6494, pp. 52–66. Springer, Heidelberg (2011)
21. Varol, A., Salzmann, M., Tola, E., Fua, P.: Template-free monocular reconstruction of deformable surfaces. In: Proceedings of the ICCV, pp. 1811–1818 (2009)
22. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
23. Sprengel, R., Rohr, K., Stiehl, H.S.: Thin-plate spline approximation for image registration. In: Proceedings of the IEEE Engineering in Medicine and Biology Society, pp. 1190–1191 (1996)
24. Hore, A., Ziou, D.: Image quality metrics: PSNR vs. SSIM. In: Proceedings of ICPR, pp. 2366–2369 (2010)

Utilizing Optical Aberrations for Extended-Depth-of-Field Panoramas

Huixuan Tang and Kiriakos N. Kutulakos*

Dept. of Computer Science, University of Toronto
{hxtang,kyros}@cs.toronto.edu

Abstract. Optical aberrations in off-the-shelf photographic lenses are commonly treated as unwanted artifacts that degrade image quality. In this paper we argue that such aberrations can be useful, as they often produce point-spread functions (PSFs) that have greater frequency-preserving abilities in the presence of defocus compared to an ideal thin lens. Specifically, aberrated and defocused PSFs often contain sharp, edge-like structures that vary with depth and image position, and become increasingly anisotropic away from the image center. In such cases, defocus blur varies spatially and preserves high spatial frequencies in some directions but not others. Here we take advantage of this fact to create extended-depth-of-field panoramas from overlapping photos taken with off-the-shelf lenses and a wide aperture. We achieve this by first measuring the lens PSF through a one-time calibration and then using multi-image deconvolution to restore anisotropic blur in areas of image overlap. Our results suggest that common lenses may preserve frequencies well enough to allow extended-depth-of-field panoramic photography with large apertures, resulting in potentially much shorter exposures.

1 Introduction

Optical aberrations—deviations of a lens system from the predictions of paraxial optics—occur in all photographic lenses as a compromise between image quality, lens complexity and cost. Aberrations affect image quality to various degrees and become especially significant when capturing photos with a wide aperture [1].

Optical aberrations are typically considered undesirable artifacts that cause well-focused subjects to appear blurry in a photo. Although recent work showed that it may be possible to restore the appearance of in-focus subjects by undoing this blur [2], the impact of lens aberrations on a photo's *out-of-focus* regions has not been explored by the vision community. Here we argue that aberrations and defocus in off-the-shelf lenses interact in ways that can have a significant impact on their point spread function (PSF): as shown in Figs. 1 and 2b, aberrations can introduce sharply-defined edges in the PSF of defocused points, making them non-uniform, anisotropic, and spatially varying. This represents a significant departure from the spatially-invariant PSF of an ideal thin lens, which is always a uniform-intensity disk (*i.e.,* a "pillbox function").

* This work was supported by the Natural Sciences and Engineering Research Council of Canada under the Discovery, Accelerator, GRAND-NCE and RGPIN programs.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 365–378, 2013.

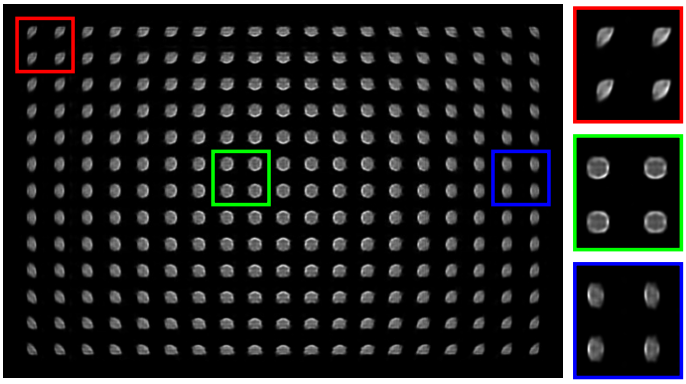


Fig. 1. The PSF of a Canon 50mm f/1.2L lens focused at $2m$ for a plane at depth $1.5m$. Note the PSF's spatially-varying and strongly-anisotropic structure.

A key consequence of this PSF structure is that it confers a greater ability to preserve high spatial frequencies in out-of-focus regions of a scene. We empirically analyze the frequency-preserving behavior of such PSFs using a simplified lens aberration model that takes into account Seidel aberrations up to third order. More specifically, we pay attention to three properties of aberrated and defocused PSFs: (1) the way they vary in radial directions away from the image center, (2) their anisotropic structure near the image border and (3) asymmetries in how they vary as functions of object depth relative to the in-focus plane.

We argue that these properties of aberrated PSFs are useful for light-efficient acquisition of extended-depth-of-field panoramas. In particular, instead of stitching photos taken with a small aperture—which gives a wide depth of field but needs long exposure times because of light inefficiency—we stitch wide-aperture, short-exposure shots. We then extend the panorama's depth of field by restoring out-of-focus blur. This is possible because aberrations cause out-of-focus scene points to be blurred differently depending on their position on the image plane. Hence, in areas of image overlap, where the same scene point is blurred differently in each photo, aberrations can preserve frequencies well enough in all directions to enable significant deblurring and depth-of-field extension.

Our approach is related to four lines of recent work. First, work on aberration modeling [1,3], PSF estimation [4], and aberration correction of in-focus subjects [2,5] has also observed that real-lens PSFs vary spatially. Here, however, our emphasis is on studying the structure of aberrated *and defocused* PSFs for the purpose of counteracting defocus. Second, our work is similar in spirit to aberration-coding methods for depth-of-field extension [6,7]. Instead of using specialized optics, however, we study standard photographic lenses, whose PSFs have different properties. For instance, we observe that aberrated PSFs preserve high spatial frequencies only for object depths on one side of the in-focus plane (*i.e.,* either closer to or farther from that plane, but not both). Third, deblurring from multiple images with complementary PSFs has been studied in coded-aperture photography and motion deblurring [8,9]. We use the same

underlying principle but in a completely different imaging domain, where complementary PSFs come from aberrations rather than coded apertures or camera motions. Last but not least, our work can be thought of as a generalized mosaicing procedure [10], where scene points are imaged under different conditions in a dense collection of photos captured while panning a specially-equipped camera. Instead of relying on specialized hardware to independently modulate the appearance of each scene point, we exploit aberrations inherent in ordinary camera lenses. Moreover, we require a relatively small number of images: as in conventional panorama stitching [11], photos must overlap enough to enable registration and to ensure that individual scene points are imaged in at least two photos.

2 Modeling Aberrated Lens PSFs

In this section we consider the problem of modeling and estimating the non-stationary, depth-varying PSF of a lens. We first measure its "ground truth" PSF for different defocus levels in a controlled lab setting and then propose a parametric model that behaves consistently with our acquired data.

2.1 PSF Measurement

Our procedure acquires a dense 3D grid of "local" 2D PSFs that capture PSF variation over the image plane as well as over object depth (*i.e.,* defocus level). A 2D grid of local PSFs corresponding to a single depth is shown in Fig. 1.

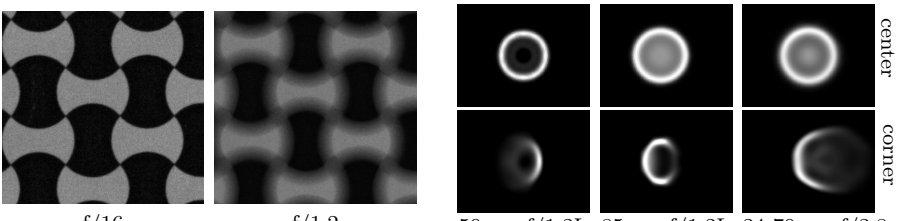
To estimate spatially-varying PSFs at a specific object depth, we use the deconvolution approach of Joshi *et al.* [4]. The idea is to take an image of a patterned fronto-parallel plane, use the corners of the known pattern to align it to the image, and then estimate local PSFs using non-blind deconvolution. We used this approach because it allowed us to compute a dense grid of local PSFs from a single image and because it yielded PSFs of reasonable quality.[1]

Since precise localization of corners is difficult for large levels of defocus (*e.g.,* PSF width > 50 pixels), we capture a second photo at each object depth with a narrow $f/16$ aperture and use this photo for image-to-pattern alignment (Fig. 2a). To capture the full 3D grid of local PSFs, we fix the zoom and focus settings of the camera and translate it along its optical axis using a translation stage to vary the object-to-camera distance.

We applied this procedure to three Canon lenses: a 50mm f/1.2L, an 85mm f/1.2L and a 24-70mm f/2.8 lens with its zoom setting set to 70mm. In all cases, we used the widest-possible lens aperture and acquired a 19×13 grid of local PSFs for each of 20 to 60 object depths away from the in-focus plane, until the PSF's diameter was approximately 80 pixels.[2] The defocused PSFs of all three lenses have evident non-uniform, edge-like features, with significant differences between center and corner PSFs (Fig. 2b). The depth ranges where these features

[1] We also tried to directly acquire the PSF by taking photos of pinhole arrays, but this is prone to noise for large levels of defocus and can be affected by diffraction.

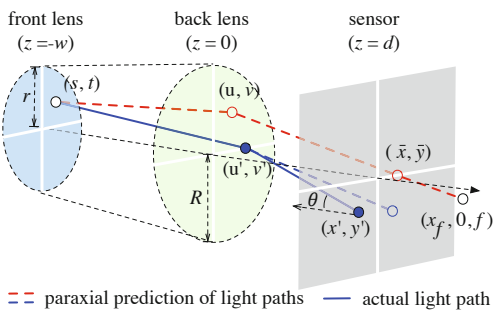
[2] This is approximately $0.5mm$ on the sensor of the Canon 1Ds Mk3 we used.



(a) Patches from a narrow/wide-aperture image pair.

(b) Measured PSFs for three Canon lenses at two positions on the image plane.

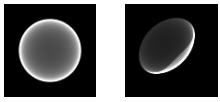
Fig. 2. Measuring the lens PSF for object distances outside the depth of field. For the PSFs in (b), lenses were focused at infinity, $0.95m$ and $0.38m$, respectively, with objects at a distance of $2m$, $1.2m$ and $0.5m$.



(a) The aberrated lens model.

F	focal length of back lens
r	aperture radius of front lens
R	aperture radius of back lens
c_F	aberration coefficient of front lens
c_B	aberration coefficient of back lens
w	length of lens barrel
d	lens-sensor distance
f	back lens -focal plane separation

(b) Model parameters.



(c) PSFs synthesized from the model.

Fig. 3. Accounting for lens aberrations with a variable-cone doublet model

appear, however, are specific to each lens: the 50mm lens yields defocused PSFs with sharp features when the lens is focused at infinity whereas the other two lenses yield such PSFs for object depths that are more distant than the in-focus plane. It follows that to take full advantage of the lenses' frequency-preserving properties, it is important to choose a focus setting that places all objects of interest on the "frequency-preserving side" of the in-focus plane.

2.2 Modeling Lens Aberrations

The structure exhibited by our measured PSFs is primarily due to monochromatic aberrations and vignetting. Monochromatic aberrations are deviations from the paraxial lens approximation that warp the light paths passing through the lens.[3] Vignetting causes a reduction in brightness at the periphery of the

[3] Lenses also suffer from chromatic aberrations [12]. Although we do not consider them here, they can be taken into account by capturing and modeling local lens PSFs separately for each color channel.

image and is due to physical ray occlusion from the lens mechanics as well as natural light fall-off from oblique rays.

We now consider a simplified lens model that captures both monochromatic aberrations and vignetting. Our model consists of two thin lenses enveloped in a variable-cone-shaped barrel (Fig. 3a) and its parameters are summarized in Fig. 3b. To model aberrations, we assume that each lens is rotationally symmetric so that deviations from the paraxial approximation of a ray can be expressed relative to the *meridional plane*, which contains both the paraxial ray and the optical axis. We also restrict our model to third-order optics, where the angle of a refracted ray is expressed as a cubic polynomial of the ray's height at the lens interface. To model vignetting, we use the variable-cone aperture model of Asada *et al.* [13] in which ray occlusion (and thus vignetting) is not affected by aberrations.

Fig. 3c shows example defocused PSFs produced by our lens model. These PSFs are qualitatively very similar to the captured PSFs shown in Fig. 2b. This suggests that our simplified model is rich enough to capture the overall structure of aberrated PSFs that we observe in practice.

We now sketch the derivation of our PSF model; a detailed derivation can be found in the supplementary materials [14]. By definition, a local PSF is the image of an idealized isotropic point light source. This image is determined by the cone of rays exiting the source. We parameterize these rays by their intersections $(u,\ v)$ with the back lens (Fig. 3a) and assume, without loss of generality, that the point source forms a focused image at $(x_f,\ 0,\ f)$ under the paraxial approximation. Together, $(u,\ v)$ and $(x_f,\ 0,\ f)$ completely determine the ray's path, $(s,\ t) \to (u,\ v) \to (\bar{x},\ \bar{y})$, under paraxial optics as well as its path $(s,\ t) \to (u',\ v') \to (x',\ y')$ under our third-order model.

It follows that the local 2D PSF $k(x,\ y)$ can be expressed as an irradiance integral of aberrated rays

$$k(x,\ y) = \int_{u^2+v^2 \le R^2} \delta(x - x'(u,\ v),\ y - y'(u,\ v))\ \ p(u,\ v)\ \mathrm{d}u\mathrm{d}v \qquad (1)$$

where $\delta(\cdot)$ denotes Dirac's delta and $p(u,\ v)$ is the *pupil function* which models vignetting. This function is zero if the corresponding ray is occluded by the lens and is $\cos^4\theta$ otherwise, where θ is the angle between the ray and the normal of the sensor plane, *i.e.*, $\theta = \cos^{-1}\left(f/\sqrt{(x_f-u)^2+v^2+f^2}\right)$.

From the thin-lens law, the intersection $(s,\ t)$ of the paraxial ray and the front lens is

$$\begin{pmatrix} s \\ t \end{pmatrix} = m_P \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} s_0 \\ 0 \end{pmatrix},\ \text{where}\ \ m_P = w\left(\frac{1}{w} + \frac{1}{f} - \frac{1}{F}\right)\ \text{and}\ s_0 = \frac{-x_f w}{f}\ . \qquad (2)$$

Restricting $(s,\ t)$ to its circular aperture determines the pupil function in Eq. (1):

$$p(u,v) = \begin{cases} \cos^4\theta, \text{ if } (m_P u + s_0)^2 + (m_P v)^2 \le r^2 \text{ and } u^2+v^2 \le R^2 \\ 0 \qquad \text{otherwise .} \end{cases} \qquad (3)$$

We can now derive the ray displacement on the image plane using Eq. (2):

$$\begin{pmatrix} \Delta x(u,v) \\ \Delta y(u,v) \end{pmatrix} \doteq \begin{pmatrix} x' \\ y' \end{pmatrix} - \begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix} = \underbrace{\alpha_1(u^2+v^2)\begin{pmatrix} u \\ v \end{pmatrix}}_{\text{spherical}} + \underbrace{\alpha_2 x_f \begin{pmatrix} 3u^2+v^2 \\ 2uv \end{pmatrix}}_{\text{coma}} + \underbrace{\alpha_3 x_f^2 \begin{pmatrix} u \\ 0 \end{pmatrix}}_{\text{astigmatism}} + \underbrace{\alpha_4 x_f^2 \begin{pmatrix} u \\ v \end{pmatrix}}_{\text{field curvature}} + \underbrace{\alpha_5 x_f^3 \begin{pmatrix} 1 \\ 0 \end{pmatrix}}_{\text{field distortion}} \tag{4}$$

where

$$\alpha_1 = d\left(c_F \frac{F-w}{F} m_P^3 + c_B\right), \quad \alpha_2 = -c_F\left(w + d\frac{F-w}{F}\right) m_P^2 \left(\frac{w}{f}\right),$$
$$\alpha_3 = 2c_F\left(w + d\frac{F-w}{F}\right) m_P \left(\frac{w}{f}\right)^2, \alpha_4 = c_F\left(w + d\frac{F-w}{F}\right) m_P \left(\frac{w}{f}\right)^2 \tag{5}$$
$$\text{and} \quad \alpha_5 = -c_F\left(w + d\frac{F-w}{F}\right)\left(\frac{w}{f}\right)^3 .$$

Eq. (4) provides a compact model of ray displacements as a linear combination of five displacement vector fields. These fields are referred to as the primary Seidel aberrations [1]. The five fields approximate the law of refraction up to third order and are usually good enough to model both spherical and aspherical lenses used in contemporary commercial lenses. When the lens settings are fixed and objects lie on a single depth plane, it suffices to use Eq. (4) to describe monochromatic aberrations by treating the parameters $\alpha_1, \ldots, \alpha_5$ as constants. However, Eq. (5) is useful if multiple focus-settings and object depths are involved because it explicitly models the displacement fields' dependence on depth. Note that both m_P and s_0 depend on $1/f$, the inverse focal distance, which depends on object depth. This factor can significantly affect the displacement of aberrated rays and explains why the PSF changes drastically with depth.

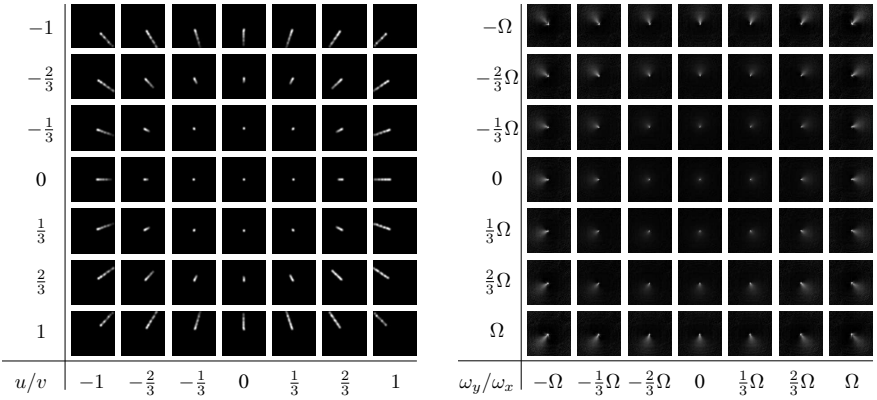
3 Frequency-Preserving Properties of Aberrated PSFs

We now use the lens model of Section 2.2 to examine the frequency-preserving properties of aberrated PSFs. To do this, we employ a light field analysis similar to that of Levin *et al.* [15].

Let us consider again the image of a point source whose paraxial image is at $(x_f,\ 0,\ f)$. We parameterize rays incident on the sensor by their intersection with two reference planes, one aligned with the back lens and the other aligned with the in-focus plane. We take the origin of the in-focus plane to be at $(x_f,\ 0,\ f)$. Paraxial optics predict that all rays from the source converge at $(x_f,\ 0,\ f)$. Thus, the light field incident on the sensor under paraxial optics is given by

$$l_0(x,\ y,\ u,\ v) = \delta(x,\ y)\ p(u,\ v)\ . \tag{6}$$

Note that the coordinates $(x,\ y)$ correspond to points on the in-focus plane and should not be confused with the image-plane coordinates in Eq. (1).



(a) The spherically-aberrated light field $\hat{l}(x, y, u, v)$. Each subplot corresponds to a 2D slice $\hat{l}(\cdot, \cdot, u, v)$.

(b) The Fourier spectrum of the 4D light field in (a). Each subplot is a 2D slice $\mathcal{F}[\hat{l}](\omega_x, \omega_y, \cdot, \cdot)$.

Fig. 4. 4D light field of an isotropic point source under our aberrated-lens model

In the presence of aberrations, the light field is distorted according to

$$l(x,\ y,\ u,\ v) = \delta\Big(x - \Delta x_f(u,\ v),\ y - \Delta y_f(u,\ v)\Big)\ p(u,\ v) \tag{7}$$

where $(\Delta x_f,\ \Delta y_f)$ are obtained from Eqs. (4) and (5) by setting $d = f$.

From the Generalized Fourier Slice Theorem [16], the Fourier transform of the lens PSF, $k()$, is a slice of the 4D Fourier transform of the light field:

$$\mathcal{F}[k](\mu,\ \nu) = \mathcal{F}[l]\left(\alpha_f\mu,\ \alpha_f\nu,\ (1-\alpha_f)\mu,\ (1-\alpha_f)\nu\right), \quad \text{where } \alpha_f = d/f\ . \tag{8}$$

Now, let $\hat{l}(x,\ y,\ u,\ v) = \delta\Big(x - \Delta x_f(u,\ v),\ y - \Delta y_f(u,\ v)\Big)$ be the light field we get when we ignore vignetting. From the convolution theorem, we have

$$\mathcal{F}[l](\omega_x, \omega_y, \omega_u, \omega_v) = \mathcal{F}[\hat{l}](\omega_x, \omega_y, \omega_u, \omega_v) \otimes \mathcal{F}[p](\omega_u, \omega_v) \tag{9}$$

where $\otimes$ denotes convolution.

Eq. (9) tells us that the Fourier transform of the lens PSF is a convolution of two light fields—one that depends only on aberrations and one that depends only on vignetting. In the following, we first discuss how the five Seidel aberrations affect the aberrated light field and then discuss the effects of vignetting.

The spherical aberration term, $\alpha_1(u^2 + v^2)\binom{u}{v}$, causes rays passing through the boundary of the lens pupil to converge at a different depth compared to rays passing through a circle near the pupil's center. These displacements, which are independent of the local PSF's position on the image plane, cause sharp features in the PSF and break the PSF's symmetry relative to the in-focus plane. An example of a spherically-aberrated light field and its Fourier transform is shown in Fig. 4. From Eq. (8), it follows that the region in 4D frequency space that

contributes to the PSF is the domain of $\mathcal{F}[\hat{l}](\alpha_f\mu, \alpha_f\nu, (1-\alpha_f)\mu, (1-\alpha_f\nu))$. Since the aberrated light field is non-zero outside of this domain, it wastes some energy outside of the "focal manifold." Thus it does not extend the depth-of-field as much as some recent computational cameras do.[4] Nevertheless, spherically-aberrated light fields do concentrate energy near the focal manifold and preserve high frequencies even when the scene is not on the in-focus plane. Importantly, observe that the Fourier spectrum in Fig. 4b is large only for frequencies ω_x, ω_u (and ω_y, ω_v) having the same sign, *i.e.,* when $d < d_f$. This indicates that high frequencies are only preserved at object depths one side of the in-focus plane.

The coma aberration term, $\alpha_2 x_f \binom{3u^2+v^2}{2uv}$, can be thought of as a change in magnification that depends on the ray's position on the pupil plane, *i.e.,* the u-v plane. Thus, rays passing through the periphery of the pupil may cross the image plane at different distances from the paraxial image of the point source. This causes asymmetries in the local PSF that vary radially away from the image center. Coma aberrations produce a cubic phase delay in the optical wavefront, resulting in defocused PSFs that preserve high spatial frequencies in a way analogous to wavefront coding [6,7].

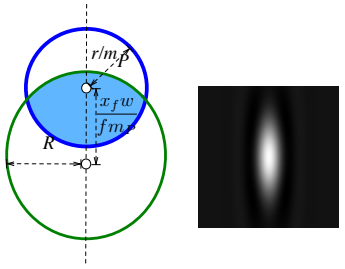
Both *the astigmatism term,* $\alpha_3 x_f^2 \binom{u}{0}$, and *the field curvature term,* $\alpha_4 x_f^2 \binom{u}{v}$, correspond to displacements that are linear functions of the ray's position on the pupil. Since ray displacements due to defocus are also linear functions of pupil position, these aberrations can be differentiated from defocus only by the fact that they increase quadractically with distance from the image center. In addition, astigmastim displaces rays only in the radial direction, yielding elliptically-shaped PSFs near the image corners.

The field distortion term, $\alpha_5 x_f^3 \binom{1}{0}$, causes a non-linear, radially-symmetric image distortion. The term, however, does not affect the local 2D PSF because it is completely independent of the pupil position. It can therefore be safely ignored when analyzing interactions between aberrations and defocus.

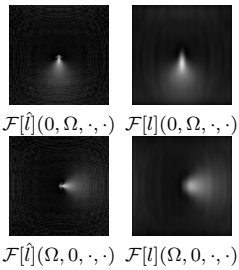
Effect of the pupil function. Eq. (3) suggests that the pupil function has a "cat eye"-shaped support on the pupil plane, with its shortest axis aligned with the v-axis (Fig. 5a). This causes its Fourier spectrum to be elongated along that axis. Since the pupil function $p(u,\ v)$ is a 2D function that is independent of $(x,\ y)$, its Fourier transform in the 4D light field space will have its spectrum concentrated on the $\mathcal{F}[p](0, 0, \omega_u, \omega_v)$ slice. Therefore, frequency-domain blurring due to the pupil function occurs only within each 2D subplot in Fig. 4b. An example of such a frequency-domain blur is shown Figs. 5a and 5b.

In summary, we observe that (1) when defocus is present, spherical and coma aberrations yield frequency-preserving PSFs; (2) this occurs only for depths on one side of the in-focus plane and thus depth-of-field extension—although possible—is asymmetric relative to that plane; (3) astigmatism and field curvature modulate defocus in a spatially-varying manner but do not affect the PSF's frequency-preserving properties; and (4) vignetting makes the PSF even more anisotropic, with frequencies in the radial direction (*i.e.,* along the v-axis) preserved more than others.

[4] See [15] for a thorough discussion.



(a) Geometry of the pupil function and its Fourier spectrum. The pupil's v-axis is shown as a dashed vertical line.



(b) 2D slices of the Fourier-transformed light field before (left) and after (right) accounting for the pupil-induced blur.

Fig. 5. The effect of vignetting. The Fourier-transformed pupil function in (a) is convolved with the individual 2D slices shown in Fig. 4b. Two of those slices are shown in the left column of (b). Note that vignetting blurs these slices primarily in the vertical direction.

4 Building Light-Efficient Panoramas

We now take advantage of these properties to create extended-depth-of-field panoramas in a "light-efficient" way, *i.e.,* using photos taken with large apertures and short exposure times. In particular, wherever these photos overlap, the underlying sharp image will be blurred by several defocused PSFs, each of which preserves high spatial frequencies in some directions but not others. By combining these photos we can therefore restore spatial frequencies in many directions, extending the depth of field of the final panorama.

To do this, we first calibrate the lens by recovering a 3D grid of local PSFs, as explained in Section 2.1. We then capture a sequence of photos using a wide aperture and a focus setting that places the scene of interest on the frequency-preserving side of the in-focus plane. These photos are aligned geometrically by estimating pairwise homographies with Autostitch [11]. Finally, we use the multi-image restoration procedure described below to compute the panorama.

Because the PSF is spatially varying, we restore the panorama patch by patch, assuming that (1) each patch may contain objects at multiple depths and (2) for a given depth, the PSF is spatially-invariant within each patch. We restore individual patches using joint, non-blind deblurring using the pre-calibrated local 2D lens PSFs. Since these PSFs vary with depth, we run the restoration procedure once for each depth hypothesis and then use the restoration results across all hypotheses to compute a per-pixel depth map for each patch.

Let ψ be a patch in the underlying sharp panorama and let $\varphi_1, \varphi_2, \ldots, \varphi_N$ be the patches corresponding to ψ in N overlapping photos. We assume that each of these patches is formed by blurring the hidden patch ψ with a depth-dependent PSF that is specific to each photo and has been pre-warped by the homography that maps photo pixels to panorama pixels. When the entire patch contains scene points at just one depth, the observed patch is given by

$$\varphi_j = k^j_{\lambda *} \otimes \psi + n \ , \tag{10}$$

where $k^j_{\lambda *}$ is the pre-warped PSF corresponding to the j-th photo, $\lambda *$ is the true depth of the patch and n denotes Gaussian noise of variance η^2. When the patch contains points at multiple depths, Eq. (10) generalizes to a layered model, where each layer's appearance is described by this equation.

Under a Gaussian image prior and a depth hypothesis λ, we can obtain an estimate $\tilde{\psi}_\lambda$ of the hidden patch using joint Wiener deconvolution:

$$\tilde{\psi}_\lambda = \mathcal{F}^{-1}\left[\frac{1}{\eta^2}\sum_j \left(\mathcal{F}[\varphi_j]\overline{\mathcal{F}[k^j_\lambda]}\right) V_\lambda^{-1}\right] \text{ where } V_\lambda^{-1} = \frac{1}{\eta^2}\sum_j \left|\left|\mathcal{F}[k^j_\lambda]\right|\right|^2 + S^{-1} \ . \quad (11)$$

Here S is the variance of $\mathcal{F}[\psi]$ and $\overline{\mathcal{F}[]}$ denotes the complex conjugate of $\mathcal{F}[]$.[5]

To assign a depth λ to each pixel p within a patch, we construct a piece-wise smooth depth map using a Markov random field approach [17]. We use per-pixel reconstruction error as the data term

$$E_\lambda(p) = \sum_j \left(\varphi_j(p) - [k^j_\lambda \otimes \tilde{\psi}_\lambda](p)\right)^2 \quad (12)$$

and use the $L1$-norm between neighboring depths as the smoothness term (with a weight of 0.1). To synthesize the final panorama $\tilde{\psi}$ from the depth map $\lambda(p)$, we copy pixels from the restored patch at the optimal depth: $\tilde{\psi}(p) = \tilde{\psi}_{\lambda(p)}(p)$.

5 Experiments

5.1 PSF Evaluation

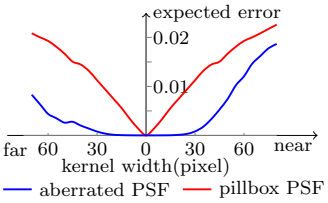
We start by considering the advantage conferred by aberrated PSFs over the standard pillbox PSF. In order to evaluate the extended-depth-of-field performance regardless of the scene, we use two criteria: (1) the expected restoration error over the distribution of hidden images and (2) the power of PSFs to discriminate between different depths.

According to Hasinoff *et al.* [18], the expected mean-squared error of the restored patch is given by

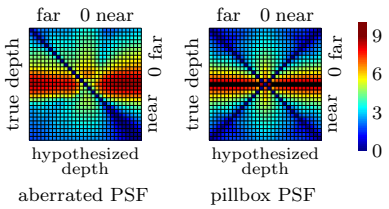
$$\mathrm{E}\left[\left|\left|\psi(p) - \tilde{\psi}(p)\right|\right|^2\right] = \sum_{\mu,\nu} V_\lambda(\mu,\nu) \ , \quad (13)$$

where $V_\lambda(\mu,\nu)$ is defined in Eq. (11). Fig. 6a plots this term as a function of PSF size for a local 2D PSF estimated from our Canon 50mm f/1.2L lens, and for the pillbox PSF at the same defocus level. The expected reconstruction error of the pillbox PSF grows significantly with defocus, whereas the error curve for the real-lens PSF remains flat near the in-focus position. In addition, reconstruction error increases at a slower rate for object depths behind the in-focus plane. This is consistent with our observation in Section 3, that the frequency-preserving ability of aberrated lenses is generally asymmetric relative to the in-focus plane.

[5] We set $S = 1$ and $\eta^{-2} = 1e-4$ in our implementation.



(a) Expected mean squared restoration error as a function of PSF width. The curves represent an average over nine randomly-chosen image locations.



(b) KL divergence between images of different defocus level, computed from PSFs at nine randomly-chosen locations.

Fig. 6. Comparative evaluation of aberrated and pillbox PSFs

To evaluate the PSF's depth-discrimination power, we calculate the Kullback-Leibler (KL) divergence between images at different defocus levels, for both the aberrated and the pillbox PSF. A high KL divergence is desirable because it indicates a smaller chance of incorrect depth estimation. As we show in the supplemental materials [14], the KL divergence between images defocused by k_1^j and k_2^j corresponding to object depths λ_1 and λ_2, respectively, is given by

$$\mathrm{KL}(\lambda_1,\lambda_2) = \frac{1}{2}\sum_{\mu,\nu}\left[\log\left(\frac{\eta^2 + S\sum_j |\mathcal{F}[k_2^j]|^2}{\eta^2 + S\sum_j |\mathcal{F}[k_1^j]|^2}\right) + \frac{S}{\eta^2}\sum_j |\mathcal{F}[k_2^j]|^2 - \frac{\sum_j |\mathcal{F}[k_1^j]|^2}{\sum_j(\eta^2/S + |\mathcal{F}[k_1^j]|^2)} - \frac{S}{\eta^2}\frac{|\sum_j \overline{\mathcal{F}[k_2^j]}\mathcal{F}[k_1^j]|^2}{\sum_j(\frac{\eta^2}{S} + |\mathcal{F}[k_1^j]|^2)}\right](\mu,\nu)\ . \tag{14}$$

Fig. 6b plots the KL divergence for our aberrated lens and for an aberration-free lens. The figure shows that the KL divergence is higher in the aberrated case, suggesting that depth recovery is easier in presence of aberrations. This is because the PSF's frequency spectrum varies inhomogeneously with depth, resulting in a smaller correlation between blurred object textures at different depths. The figure also shows that unlike the pillbox PSF, where it is impossible to tell if a defocused scene point is in front or behind the in-focus plane, aberrated PSFs do not suffer from this depth-reflection ambiguity.

5.2 Panorama Restoration

Simulation. We synthesized ten images of a scene containing three depth layers (Fig. 7). All images are focused at the same depth, with each having a specific, randomly-chosen PSF from our Canon 50mm 1.2L lens calibration data. The PSF sizes ranged from 20 to 60 pixels. We restored the scene with the algorithm discussed in Sec. 4. Despite the challenging defocus blur, our algorithm successfully restored image details and recovered an approximate depth map.

Real Data. We captured three datasets under different imaging conditions, from macro to landscape. We used the Canon 50mm f/1.2L lens to capture the

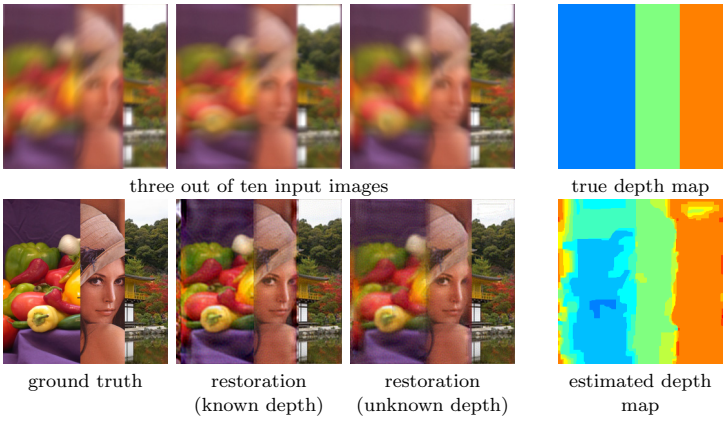


Fig. 7. Multi-image restoration on simulated images

indoor "magazine" scene (Fig. 8) and the outdoor "portrait" scene (Fig. 9). For both datasets we focused the lens at the far end of the scene's depth range so that the aberrated PSF preserves frequencies. We used the Canon 24-70mm f/2.8L zoom lens to capture the close-up "static" scene (Fig. 10). In this case, the lens was focused at $0.38m$ with its focal length set to $70mm$. We always captured photos at the largest possible aperture (f/1.2 or f/2.8).

Individual photos are significantly contaminated by defocus in all three examples. Despite this, multi-image restoration was able to successfully recover scene details outside the original depth of field (*e.g.,* text and facial features) and to obtain a reasonable depth segmentation. For the magazine scene, we also show patch restoration results from a single photo. Although some details are recovered even in this case, multi-image restoration is of much higher quality.

We used a rather simple restoration procedure that was not designed to handle depth discontinuities. As a result, our restorations of the portrait and static scene

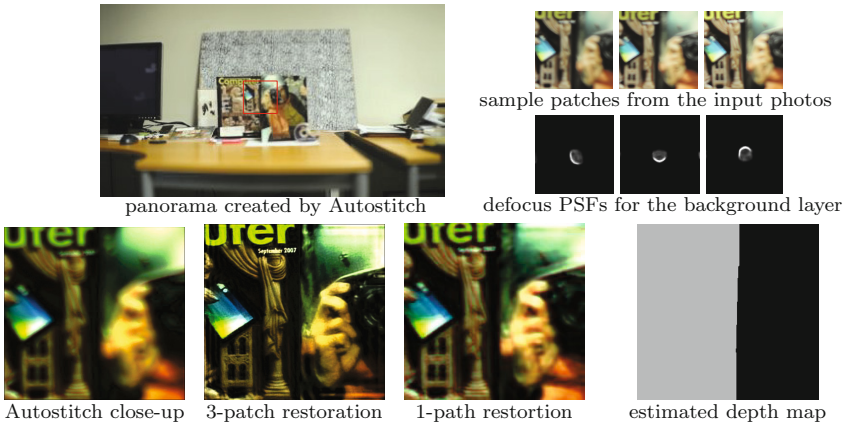


Fig. 8. The "magazine" scene. The lens is focused at $2m$ whereas the patches shown are approximately $1.7m$ away. Note that the lens depth of field at $2m$ is less than $1cm$.

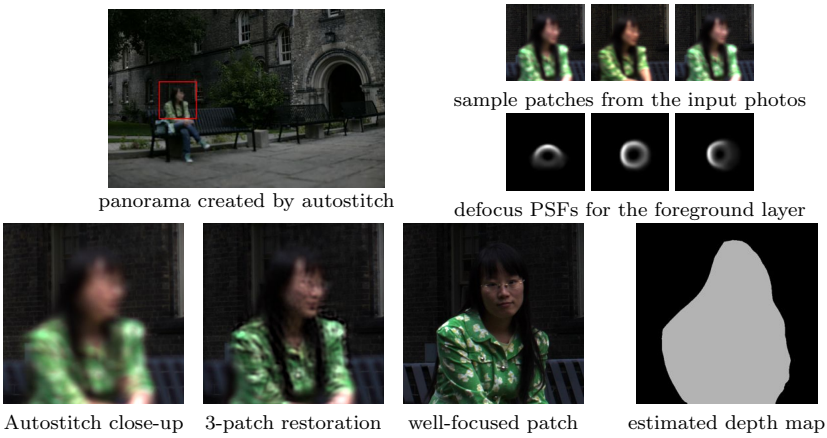


Fig. 9. The "portrait" scene. The lens is focused at infinity with subject at $10m$.

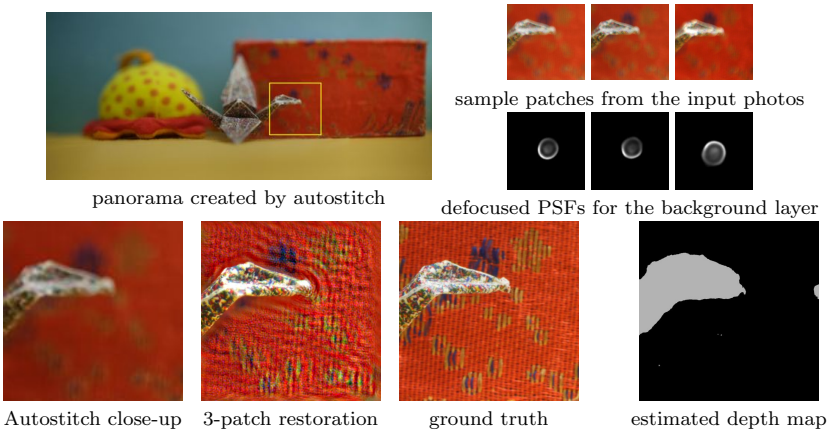


Fig. 10. The "static" scene. The lens is focused at its minimum focusing distance and the foreground region in the patch shown is roughly $3cm$ in front of the in-focus plane.

suffer from ringing. In addition, scenes with significant depth variation cannot be precisely aligned with a single homography. These issues may be resolved by incorporating multi-view geometry into the image formation model and by relying on more advanced deconvolution algorithms. Nevertheless, our restoration method recovers many details not visible in the directly-stitched panorama.

6 Conclusion

Our goal was to show that aberrations in many photographic lenses can be *significant* and *useful* in analyzing defocus under certain conditions. Specifically, optical aberrations are capable of preserving high frequencies and can be used for depth-of-field extension as well as depth estimation. Moreover, aberrated

PSFs exhibit several anisotropies, both on the image plane and across depth, that must be taken into account for accurate results.

We are currently in the process of analyzing the aberration properties of several off-the-shelf photographic lenses and are exploring several directions for future work. These include (1) studying the general depth-from-defocus problem for real, aberrated lenses, (2) estimating the aberration properties of a lens from a single photo without prior calibration, and (3) developing a unified framework for one-shot depth-from-defocus and blind image deblurring.

References

1. Smith, W.J.: Modern Optical Engineering. McGraw Hill (2000)
2. Schuler, C., Hirsch, M., Harmeling, S., Scholkopf, B.: Non-stationary correction of optical aberrations. In: Proc. ICCV 2011, pp. 659–666 (2011)
3. Janssen, A.J.E.M.: Extended nijboer-zernike approach for the computation of optical point-spread functions. J. Opt. Soc. Am. A 19, 849–857 (2002)
4. Joshi, N., Szeliski, R., Kriegman, D.J.: PSF estimation using sharp edge prediction. In: Proc. CVPR 2008 (2008)
5. Kee, E., Paris, S., Chen, S., Wang, J.: Modeling and removing spatially-varying optical blur. In: ICCV 2011 (2011)
6. Cathey, W.T., Dowski, E.R.: New paradigm for imaging systems. Applied Optics 41, 6080–6092 (2002)
7. Dorronsoro, C., Guerrero-Colon, J.A., Fuente, M.C., Infante, J.M., Portilla, J.: Low-cost wavefront coding using coma and a denoising-based deconvolution. In: Proc. SPIE, vol. 6737 (2007)
8. Chen, J., Yuan, L., Tang, C.K., Quan, L.: Robust dual motion deblurring. In: Proc. CVPR 2008 (2008)
9. Zhou, C., Lin, S., Nayar, S.K.: Coded aperture pairs for depth from defocus. In: Proc. ICCV 2009 (2009)
10. Schechner, Y.Y., Nayar, S.K.: Generalized mosaicing. In: Proc. ICCV 2001, pp. 17–24 (2001)
11. Brown, M., Lowe, D.: Autostitch home page (2005), `http://www.autostitch.net`
12. Cossairt, O., Nayar, S.K.: Spectral focal sweep: Extended depth of field from chromatic aberrations. In: Proc. ICCP 2010 (2010)
13. Asada, N., Amano, A., Baba, M.: Photometric calibration of zoom lens systems. In: Proc. ICPR 1996, pp. 186–190 (1996)
14. Tang, H., Kutulakos, K.N.: Supplementary materials (2012), `http://www.cs.toronto.edu/hxtang/projects/edof_panorama/`
15. Levin, A., Hasinoff, S.W., Green, P., Durand, F., Freeman, W.T.: 4D frequency analysis of computational cameras for depth of field extension. ACM TOG 28 (2009)
16. Ng, R.: Fourier slice photography. ACM TOG 24, 735–744 (2005)
17. Agarwala, A., Dontcheva, M., Agrawala, M., Drucker, S., Colburn, A., Curless, B., Salesin, D., Cohen, M.: Interactive digital photomontage. ACM TOG 23, 294–302 (2004)
18. Hasinoff, S., Kutulakos, K.N., Durand, F., Freeman, W.T.: Time-constrained photography. In: Proc. ICCV 2009 (2009)

Motion-Invariant Coding Using a Programmable Aperture Camera

Toshiki Sonoda, Hajime Nagahara, and Rin-ichiro Taniguchi

Kyushu University, Fukuoka, Japan

Abstract. A moving object or camera causes motion blur in a conventional photograph, which is a fundamental problem of a camera. In this research, we propose to code a motion-invariant blur using a programmable aperture camera. The camera can realizes virtual camera motion by translating the opening, and as a result, we obtain a coded image in which motion blur is invariant with the object velocity. Thereby, we recover motion blurs without estimation of the motion blur kernels or knowledge of the object speeds. We model the projection of the programmable aperture camera, and also demonstrate that our proposed coding works for a prototype camera.

1 Introduction

Motion blur is the blur that results from either camera shake or object motion in a scene. It often results in photographs that the photographer did not intend. Motion blur should be avoided to ensure a clear image, since such blur loses high-frequency information.

Various methods have been proposed to avoid motion blur. The simplest solution is short exposure photography. A camera has a shutter in front of an imager, and in short exposure photography, the imager is exposed for only a short period of time (exposure time) when the shutter is opened to capture an image. If the exposure time is short, we can ignore the object motion of a scene and avoid motion blur. However, there is unavoidable trade-off between motion blur and the signal-to-noise ratio (SNR) of the image since the shorter exposure darkens the captured image.

Another solution is lens or sensor shifting, which has been implemented by some modern cameras to stabilize an image. Here, a mechanical actuator is controlled to shift a lens or sensor in real time during the exposure to compensate for motion of the camera [1]. This system is applicable only to motion blur caused by camera motion and not to motion blur caused by object motion.

An approach that restores the unblurred scene through deconvolution in image processing has been proposed [2]. However, the blur kernels for deconvolution vary according to object motion and it is difficult to estimate the kernels or motions. To tackle this problem, some methods attempt to estimate the point spread functions (PSFs) and restore a sharp image from a single input image [3][4][5][6] or multiple input images [7][8][9]. However, the typical motion blur

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 379–391, 2013.

kernel contains many zero-crossings in the Fourier domain. Therefore, the kernel loses image information and the deconvolution becomes ill-conditioned. To address this issue, several attempts have been made in the field of computational photography to control the motion-blur PSF using special optics or hardware so that the PSF estimation and motion deblurring can be handled easily.

Raskar et al. [10] proposed the method of a fluttering shutter that modifies the motion-blur PSF to achieve a broader-band frequency response and to avoid the zero-crossings. The method allows the stabilization of deconvolution results. However, there is still a requirement of precise knowledge of motion segmentation boundaries and object velocities. Agrawal et al. [11] improved the fluttering shutter to estimate the object motion or deconvolution kernel robustly by modifying the shuttering pattern. There is an intuitive disadvantage in terms of the image SNR when employing these shuttering methods, since half the incoming light is blocked in engineering the PSF.

Levin et al. [12] proposed the parabolic-motion coding of a camera. The parabolic motion of the camera makes the motion blur invariant to the object speed and the kernel has a broadband frequency in the Fourier domain. This approach was described as motion-invariant photography. The method has an advantage in terms of the image SNR since it uses the camera motion to engineer the motion-blur PSF and the shutter is totally open during the exposure. However, the invariance and broadband property of motion blur is only for one-dimensional horizontal motion. This method requires a mechanical mechanism, such as the use of cams and gears, to implement the parabolic camera motion. This should be avoided because of the difficulties of practical implementation and the limitation of the motion speed restricted by inertia of the element.

McCloskey et al. [13] proposed the implementation that achieves motion-invariant photography using lens shifting. This method achieves motion-invariant photography more practically than that of using camera body motion. Cho et al. [14] extended Levin's parabolic motion coding to two-dimensional object motion. Similarly, Bando et al. [15] extended the coding using circular camera motion. These methods engineer a broadband frequency response of the PSFs, but they require motion estimation since motion invariance is not realized in contrast to the case for Levin's parabolic motion.

In this paper, we propose a novel method that achieves motion-invariant PSF coding using a programmable aperture camera [16]. The camera can change its aperture pattern at high speed using a liquid-crystal-on-silicon (LCoS) device and electrical signal. The camera achieves virtual camera motion by translating patterns of an aperture opening. Hence, we realize Levin's motion-invariant photography without using a mechanical mechanism unlike the conventional method that moves the camera body or a part of its elements . Accordingly, our method increases the utility and practicability of motion-invariant photography. We also model the projective geometry of the programmable aperture camera, the virtual camera motion and the generated PSFs. We analyze the parameter settings for the aperture pattern and optical parameters of the camera in simulation experiments. We confirm that the proposed method realizes motion-invariant photography using a prototype camera in experiments.

2 Motion-Invariant Photography Using a Programmable Aperture Camera

2.1 Realization of Camera Motion Using a Programmable Aperture Camera and Modeling the Motion-Blur PSF

In a conventional photograph, objects moving at different speeds cause varying motion blur of different shapes and lengths. To remove such motion blur, we must estimate the speeds of the objects one by one. To tackle this problem, Levin et al. [12] realized motion-invariant photography that can make the PSF invariant to motion and almost the same shape by parabolic camera motion during an exposure. Because of this invariance, we can remove all blur for all moving objects by deconvolution using a single PSF. In the Levin's work [12], the obtained PSF that is invariant to motion is expressed as:

$$\phi(x) = \frac{\lambda(x)}{2T\sqrt{s_i^2 - 2a_i x}}, \tag{1}$$

$$\lambda(x) = \begin{cases} 2, & \frac{s_i^2}{2a_i} \leq x < s_i T - a_i T^2, \\ 1, & s_i T - a_i T^2 < x < s_i T + a_i T^2, \\ 0, & \text{otherwise}, \end{cases}$$

where they assume that the image has the acceleration a_i it comes from camera motion, and the velocity s_i it comes from object motion. Here, x is the position in the image, and $2T$ is exposure time. These assumptions are expressed as:

$$x(t) = s_i t + \frac{a_i t^2}{2}. \tag{2}$$

In the paper [12], you can see the detailed process that the Eq. 1 obtained.

In Eq. 1, the all parameters is considered in image space. Then we consider the capturing scene as shown in Fig. 1 to disclose the relations between the camera acceleration in real scene a_c and the camera acceleration in image a_i, and the object velocity in real scene s_o and the object velocity in image s_i.

In this scene shown in Fig. 1, the principal point of the lens is on the origin of the world coordinates, the optical axis is coincident with the z-axis. This camera moves on the x axis of the world coordinate during the exposure, and an arbitrary point in the scene is denoted $P(X, Y, Z)$. The point $P(X, Y, Z)$ is projected to the image point $p(x, y)$. Figure 2 shows an X-Z slice of the projection for simplicity. $P(X, Z)$ is projected to $p(x)$ on the imager plane($Z = -Z_p$) by a pinhole camera model. This can be expressed as

$$x = \alpha X, \;\; \alpha = \frac{-Z_p}{Z}, \tag{3}$$

where $P(X, Z)$ is the point on the moving object in a scene. We assume that the point moves parallel to the x-axis and the position is expressed as $X(t)$. Similarly, if the camera moves parallel to the x-axis and the position is expressed as $X_c(t)$,

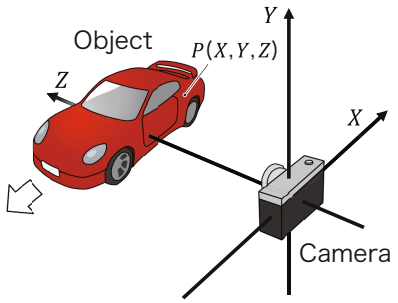


Fig. 1. Coordinate system of a camera

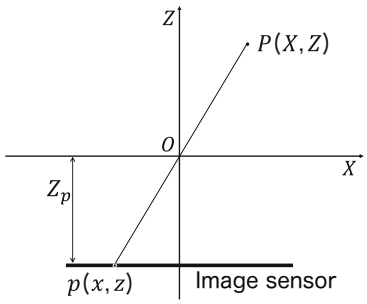


Fig. 2. Projective geometry of a normal camera

the coordinate time motion $x(t)$, which is the position of the projective point $p(x)$ on the image space relative to P, can be expressed as

$$x(t) = \alpha(X(t) - X_c(t)). \tag{4}$$

It is shown that the distance in the image correspond to that in the real scene with coefficient α from this Eq. 4. Thus we obtain these relations:

$$a_c = \frac{1}{\alpha} a_i, \tag{5}$$

$$s_o = \frac{1}{\alpha} s_i. \tag{6}$$

The center of the camera aperture is also the center of the projection in the projective geometry, and a projective change can thus be realized by motion of this aperture position. In this research, we realize the virtual camera motion that is needed for motion-invariant photography by temporally changing aperture patterns. Figure 3 shows the projective geometry of the programmable aperture camera. For simplicity and in a manner similar to Fig. 2, Fig. 3 shows the geometry on $X - Z$ space, where a point in the scene is $P(X, Z)$, the aperture is the plane $Z = 0$, and a point on the image space ($Z = -Z_p$) is $p(x)$. The lens focal distance is set as f. The distance Z_p between the lens plane and the focal point Q is then expressed as

$$\frac{1}{f} = \frac{1}{Z} + \frac{1}{Z_q}. \tag{7}$$

When we set the position of the pinhole aperture at $A(X_a, 0)$, as shown in Fig. 3, the ray radiating from P goes toward the focal point Q through lens refraction via A and is projected to the point p on the image. The relation of the projection of the projective point p can be modeled as

$$x(t) = \alpha X(t) - \beta X_a(t), \tag{8}$$

$$\beta = Z_p \left(\frac{1}{f} - \frac{1}{Z} - \frac{1}{Z_p} \right). \tag{9}$$

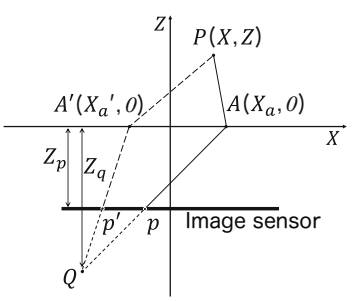


Fig. 3. Virtual camera motion model for a programmable aperture

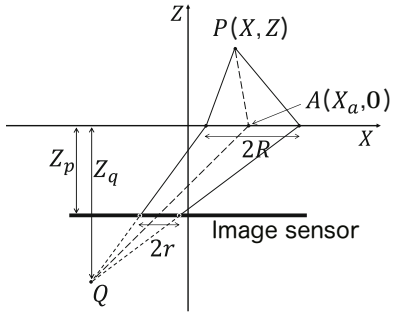


Fig. 4. Relation between the programmable aperture and static PSF

From Eq.s 4 and 8, it is found that the motion of the camera varies from that of the aperture while the motion is the same as the object motion in the projection of the programmable aperture camera. Since it is also shown that the distance in the image correspond to that in the real scene with coefficients α and β, we obtain this relation:

$$a_a = \frac{1}{\beta} a_i, \tag{10}$$

When we set the aperture motion to a constant acceleration as below equation obtained from Eq.s 4 and 8, we can represent Motion-Invariant Photography with aperture motion.

$$X_a(t) = \frac{\alpha}{\beta} X_c(t) = \frac{\alpha}{\beta} a_c t^2. \tag{11}$$

Since this proposed method imitates the camera motion in Levin's work [12], the limitation that the object motion must have constant acceleration, and one-dimensional horizontal motion corresponding to camera motion are also the same as the work.

2.2 Relation between Aperture and Static PSF Sizes

An actual camera aperture is an opening of finite radius $R > 0$ and is not a pinhole like that in Fig. 3. The radius r that the static PSF generates on the image space is proportional to R. Figure 4 shows this relation. The radius r of the static PSF projected on the projective point p on the image can be modeled as

$$r = -Z_p\left(\frac{1}{f} - \frac{1}{Z} - \frac{1}{Z_p}\right)R = -\beta R. \tag{12}$$

When the coefficient β is zero, there is no parallax. Therefore, we must accept a depth blur for generating a parallax to make $|\beta| > 0$, since the static PSF is

no more impulse function ($r > 0$). The generated static PSF can be modeled as pillbox function whose radius is r,

$$\psi(x,y) = \begin{cases} \frac{1}{\pi r^2}, & x^2 + y^2 \leq r^2, \\ 0, & \text{otherwise.} \end{cases}$$

The motion blur caused by our proposed method is modeled by the combination of the static PSF and temporal PSF, and is expressed as:

$$\Phi(x,y) = \phi(x) * \psi(x,y). \tag{13}$$

We use this $\Phi(x,y)$ for restoring motion blur as a deconvolution PSF.

The deconvolution of the image captured by our proposed coding is difficult to recover than that of Levin's coding [12], since our coding must allow the depth blur caused by the static PSF while Levin's method assumed the static PSF is an impulse function. This is a different part from Levin's work [12] realized by camera motion. There is a same limitation to use a single PSF for the deconvolution without PSF estimation. The size of PSF should not be changed within same range of the object depth such as depth of field (DOF), whatever the radius of static PSF is within one pixel (Levin's method) or some size of pixels (our method).

2.3 Relation between Aperture Size and Object Speed

The aperture position must be moved to realize motion-invariant photography using programmable aperture camera. However, the maximum size of the aperture radius of the R_{max} is restricted from the lens and the caliber of the optical system. We must set a smaller size of the aperture R, when the aperture we require the large motion of the aperture ΔX, as shown in Fig. 5.

$$\Delta X_a = 2(R_{max} - R). \tag{14}$$

As a result, a larger motion of the aperture ΔX_a produces a lager acceleration,

$$a_a = \frac{2\Delta X_a}{T^2}. \tag{15}$$

Thus, we need to use a large aperture acceleration a_a to restore a large motion blur. When we set a large acceleration a_a, the radius R become small and amount of light become decreasing simultaneously. Therefore it is shown that they have trade-off between the acceleration a_a and SNR ratio of captured image.

$$a_a = \frac{4}{T^2}(R_{max} - R). \tag{16}$$

We have discussed about the acceleration that depend on the size of the radius R, however, the motion-invariance of this proposed method depends on the acceleration in the image space a_i more directly than the physical aperture acceleration a_a, so we rewrite Eq. 16 to Eq. 17.

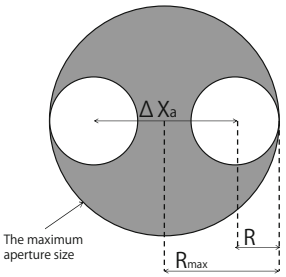


Fig. 5. Relation between the aperture size and the length of the aperture motion

$$a_i = \frac{4}{T^2}\beta(R_{max} - R). \tag{17}$$

From this equation, when we set β to larger value by setting the lens focal distance f larger, the acceleration a_i that is presented in the image space also becomes larger. However, as shown in Eq. 12, the larger β also produces the larger static PSF (or the smaller size of the aperture) that reads the restored image quality worse. Hence, there is another trade-off that if we would like to use larger aperture acceleration in the image space a_i that depend on the fixed size of the aperture and the variable lens focal distance, we must allow the static PSF becomes larger.

3 Discussion by Simulation

We discussed about the limitations of our proposed method and optimizations of the parameters by simulation experiments. We evaluated the restored images by using PSNR for discussing them. We used thirty natural images downloaded from Flickr. We set s_i to 0.01-4.0 pixels/ms for emulating the moving object. We added Gaussian noise with zero mean and standard deviation of 0.001 to the images for emulating a market camera (Panasonic FZ28 with ISO 100, F4, 45 ms exposure).

3.1 Discussion of Camera Acceleration and Object Velocity

We evaluated how much acceleration we can apply to recover the motion blur with the object speed. In the experiment, we set camera accelerations appeared in the image a_i to 0.008, 0.016, 0.032, 0.064, 0.128, 0.256, and we generated coded images that captured scenes containing objects with various motions. We used the PSF of static object ($s_i = 0$) as the deconvolution PSF under the assumption of motion invariance. We calculated PSNR between the original image and the deconvoluved image for evaluating the restored image qualities. We evaluated the image qualities and invariances according to the peak signal-to-noise ratio (PSNR).

Figure 6 shows the PSNR against the object speeds. The horizontal axis indicates motion blur length as easy to understand how fast the motion is. It is calculated by $45 \times s_i$ and appeared in the image when we assume to use the regular camera for capturing the object with 45 ms exposure. Figure 6 shows that the PSNR is high when the object velocity s_i is low. When we use higher setting of the acceleration of camera a_i, the PSNR is getting flatter across the object speeds. This shows that the higher a_i gives more invariant to wide range of the object speeds, however, there is a trade-off between the acceleration a_i and the peak quality of the restored image, since the larger acceleration gives a larger motion coding and make the deconvolution much difficult. Hence, we should set a_i to fit the maximum velocity that it is supposed that the scene will contain. Here, we assume the models of setting for the velocity as;

$$a_i = \frac{2}{T} s_{max}, \tag{18}$$

where s_{max} is the maximum velocity of the objects s_i in a scene. Under the assumption, Fig. 6 experimentally shows the motion invariance of the range under the maximum object speed and minimum quality of the image is guaranteed as 22dB PSNR through the various settings.

We used this assumption in following simulation experiments.

3.2 Relation between Quality of Restored Image and Static PSF Size

We must accept a depth blur in a image to realize the parallax with the changing aperture position in our proposed method. We examined how does the static PSF size affects the quality of the restored image in a simulation experiment. We used a pillbox function with a radius r for the static PSF model and set r to 1, 4, 8 pixels. The acceleration setting a_i was decided under the assumption described by Eq. 18. The size of the aperture R can be calculated from the acceleration a_i and the static PSF radius r as described in Eq.s 12 and 17. The SNR of the input image is changed based on the amount of the light, when the aperture size R is changed. We emulated the SNR changes that calculated by the ratio of R and R_{max}.

Figure 7 shows PSNR across the blur length in the image with the different radius of the static PSF r. We also shows that PSNR with short exposure case as a comparison in this figure. We set the exposure time not to appear the motion blur and the amount of the light loses with inverse proposition to the the velocity s_i for emulating noise ration of the short exposure case. Figure 7 shows that PSNRs of the proposed method are better than short exposure, although we allow the static blur PSF. As a result, we confirmed that the proposed method gives a better quality of restored image than that of regular short exposure photography.

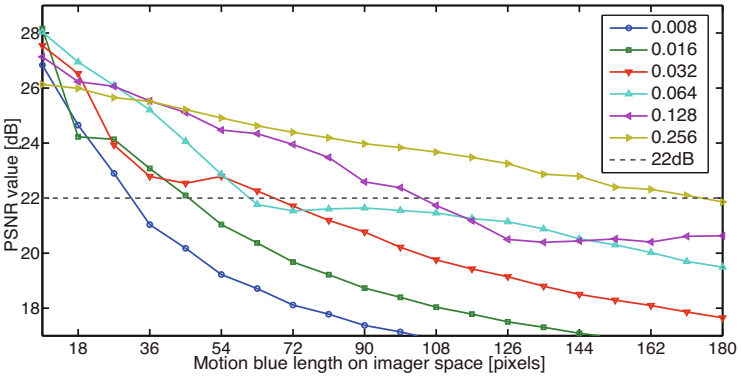


Fig. 6. Relation between the PSNR and object velocity at various camera acceleration

3.3 Comparison with the Restored Images in Various Noise Levels

We often use high ISO setting for capturing in a dark scene. However, higher ISO setting produces higher noise level of the image. We evaluated the trade-off between the image quality and noise levels. We used standard deviation of 0.01 as a standard noise and added two different settings, 0.1 and 0.001 for emulating the different ISO setting in the simulation.

Figure 8 shows the PSNRs with three noise conditions. We also showed that the PSNRs with short exposures with the noise conditions for comparison. This figure shows that the PSNR of the proposed method is higher than that of the short exposure in the case that motion blur length is more than 117 pixels with 0.001 noise. We also confirmed that the PSNR of the proposed method with 0.01 and 0.1 is higher than that of the short exposure in case of more than 18 and 12 pixels, respectively. From these results, we can conclude that our proposed method has an advantage to the short exposure photography in the case when the blur length is larger or noise level is higher.

4 Real Experiment

We demonstrated motion-invariant coding of a scene using a prototype camera as shown in Fig. 10. The specifications of the prototype are given in Table 9.

In this experiment, we used a toy train for the scene, and this experiment arrangement is shown in Fig. 11. The train on the railroad track is in a distance of 0.85 m from camera. Also we set the picture board for background and the miniature car to the distance of 0.9 m and 0.83 m, so that these objects are in the DOF. When the image is captured by normal photography, the focal point is set on the object. All captured images (Fig.s 12-15) have been adjusted to almost same intensity. We shot the scene and obtained the image as shown in Fig. 12. For this experimental setup, the motion of the train appeared as linear motion of 0.6 pixels/ms in the image space. Since we set the camera

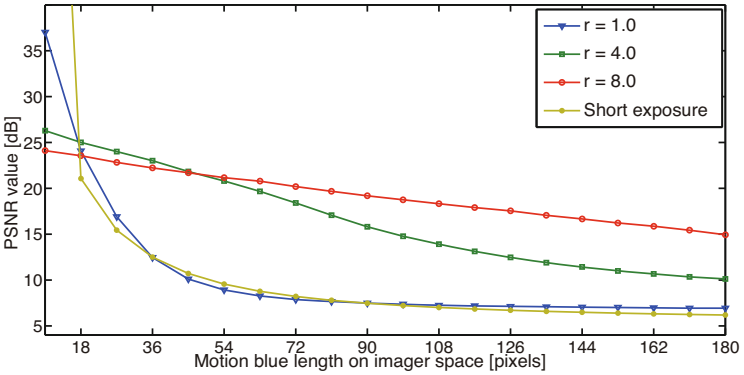


Fig. 7. Relation between the PSNR and object velocity at various static PSF radius

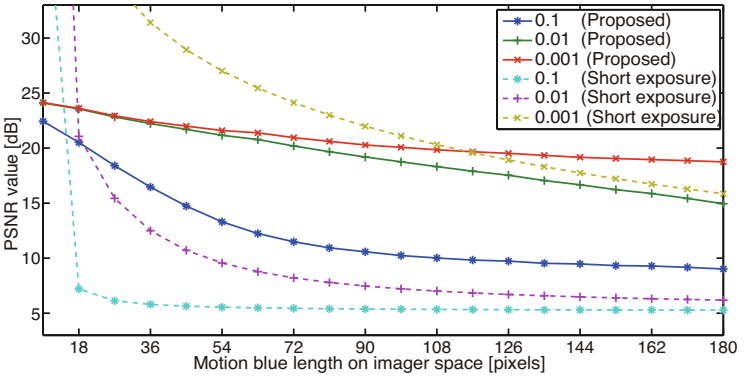


Fig. 8. Relation between the PSNR and object velocity at another noise level

exposure time as 45 ms, motion blur appeared with length of about 27 pixels in a normal photograph. (The length is obtained from Fig. 12.) The frame rate of the aperture changes is 365 fps for the prototype camera. Thus, 16 aperture patterns can be displayed during the exposure time. We set the radius of the displaying aperture pattern R as 145.5 pixels (F14.08) against the width of the aperture pattern display of 1280 pixels to make static PSF size $r = 2$ pixels. From these conditions, acceleration 0.053 pixels/ms^2 can be presented in the imager plane. We coded the motion blur using this acceleration. Figure 13 is the image captured by our motion-invariant photography. In this figure, the moving train and static background are equally blurred . We used the measured PSF, which was captured in advance, for deconvolution. (For deconvolution, we used BM3D deconvolution proposed by Dabov et al. [17]) Figure 14 shows the restoration result obtained by deconvolution of the captured image (Fig. 13) with the single measured PSF. As a result, it is demonstrated that we can remove the motion blur through deconvolution since the edge of the image is sharper than that in the

Image resolution	1280 × 960
Image aquisition frame rate	15fps
Aperture resolution	1280 × 1024
Aperture frame rate	365fps
Minimum F-number	2.8
FOV	46°
Actual aperture contrast	372:1
Light transmittance	16.49°
Focal distance	24.8mm

Fig. 9. Specifications of the prototype camera

Fig. 10. Prototype camera

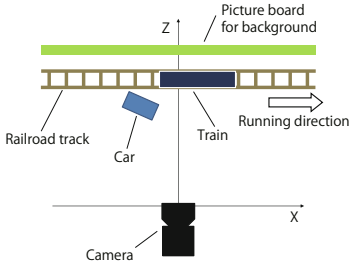
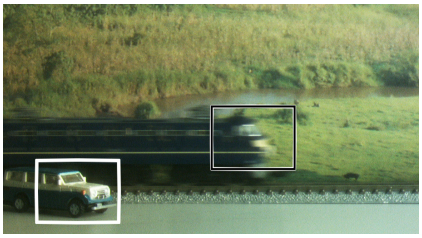


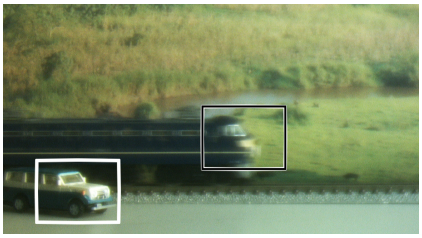
Fig. 11. Target scene



a. Entire image

b. Magnified image
(Left: Moving object, Right: Static object)

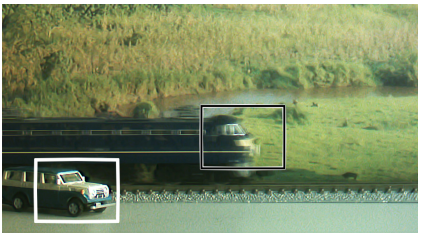
Fig. 12. Image captured by normal photography



a. Entire image

b. Magnified image
(Left: Moving object, Right: Static object)

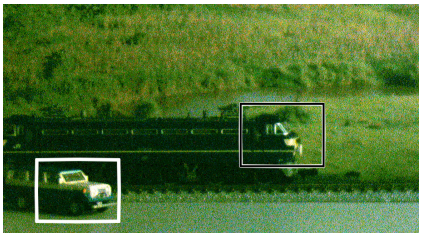
Fig. 13. Blurred image recorded by motion-invariant photography



a. Entire image

b. Magnified image
(Left: Moving object, Right: Static object)

Fig. 14. Deconvolved image



a. Entire image

b. Magnified image
(Right: Moving object, Left: Static object)

Fig. 15. Image captured with a short exposure

normal photography (Fig. 12). In addition, we could remove the blur of the static background equally without motion estimation or segmentation. Furthermore, we compared our result with the simple short exposure photograph. Figure 15 shows the image obtained in short exposure photography by setting the exposure time to 1.665 ms so that we can ignore all motion in the scene. Here we used F4 that is maximum radius setting of the programmable aperture camera. It is observed that the short exposure makes the image noisy and loses gradation, because the amount of light is reduced to 0.4582 times of that for the normal photograph and the noise was emphasized by intensity adjustment. Our coded and deconvolved result shows a better image that is sharper than the burred image and brighter the short exposure image.

5 Conclusion

In this research, we proposed a novel method to code a motion-invariant PSF using a programmable aperture camera. The camera can dynamically change the aperture pattern at a high frame rate and realizes virtual camera motion by translating the opening, and as a result, we obtain a coded image in which motion blur is invariant with the object velocity. Thereby, we recover motion blurs without estimation of the motion blur kernels or knowledge of the object speeds. To realize this, we modeled the projective geometry of the programmable aperture camera, virtual motion of the camera and the generated PSFs. We analyzed setting and optical parameters that are required for the proposed motion-invariant photography and discussed the range of the parameters that proposed method is superior to short exposure in a simulation experiment. Moreover, we experimentally demonstrated that our proposed coding works for a prototype camera.

References

1. Canon Inc: EF Lens Work III, The Eyes of EOS. (Lens Product Group)
2. Jansson, P.: Deconvolution of image and spectra, 2nd edn. Academic Press (1997)
3. Fergus, R., Singh, B., Hertzmann, A., Roweis, S.T., Freeman, W.T.: Removing camera shake from a single photograph. ACM Trans. Graph. 25(3), 787–794 (2006)
4. Joshi, N., Szeliski, R., Kriegman, D.: PSF estimation using sharp edge prediction. In: Proc. CVPR (2008)
5. Shan, Q., Jia, J., Agarwala, A.: High-quality motion deblurring from a single image. ACM Trans. Graph. 27(3), 1 (2008)
6. Yuan, L., Sun, J., Quan, L., Shum, H.Y.: Progressive interscale and intra-scale non-blind image deconvolution. ACM Trans. Graph. 27(3), 1 (2008)
7. Ben-ezra, M., Nayar, S.K.: Motion-based motion deblurring. IEEE Trans. Pattern Recognition and Machine Intelligence (2004)
8. Yuan, L., Sun, J., Quan, L., Shum, H.Y.: Image deblurring with blurred/noisy image pairs. ACM Trans. Graph. 26(3), 1 (2007)
9. Bar, L., Berkels, B., Sapiro, G., Rump, M.: A variational framework for simultaneous motion estimation and restoration of motion-blurred video. In: Proc. ICCV (2007)
10. Raskar, R., Agrawal, A., Tumblin, J.: Coded exposure photography: motion deblurring using fluttered shutter. ACM Trans. Graph. 25(3), 795–804 (2006)
11. Agrawal, A., Xu, Y.: Coded exposure deblurring: optimized codes for psf estimation and invertibility. In: Proc. CVPR (2009)
12. Levin, A., Sand, P., Cho, T.S., Durand, F., Freeman, W.T.: Motion-invariant photography. ACM Trans. Graph. 27(3), 71:1–71:9 (2008)
13. McCloskey, S., Muldoon, K., Venkatesha, S.: Motion invariance and custom blur from lens motion. In: Proc. ICCP (2011)
14. Cho, T.S., Levin, A., Durand, F., Freeman, W.: Motion blur removal with orthogonal parabolic exposures. In: Proc. ICCP (2010)
15. Bando, Y., Chen, B.Y., Nishita, T.: Motion deblurring from a single image using circular sensor motion. Computer Graphics Forum 30(7), 1869–1878 (2011)
16. Nagahara, H., Zhou, C., Watanabe, T., Ishiguro, H., Nayar, S.K.: Programmable Aperture Camera Using LCoS. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part VI. LNCS, vol. 6316, pp. 337–350. Springer, Heidelberg (2010)
17. Dabov, K., Foi, A., Egiazarian, K.: Image restoration by sparse 3d transform-domain collaborative filtering. In: Proc. SPIE Electronic Imaging, vol. 6812, no. 6812–1D (2008)

Color-Aware Regularization for Gradient Domain Image Manipulation

Fanbo Deng[1], Seon Joo Kim[2], Yu-Wing Tai[3], and Michael S. Brown[1]

[1] National University of Singapore
[2] SUNY Korea
[3] Korea Advanced Institute of Science and Technology

Abstract. We propose a color-aware regularization for use with gradient domain image manipulation to avoid color shift artifacts. Our work is motivated by the observation that colors of objects in natural images typically follow distinct distributions in the color space. Conventional regularization methods ignore these distributions which can lead to undesirable colors appearing in the final output. Our approach uses an anisotropic Mahalanobis distance to control output colors to better fit original distributions. Our color-aware regularization is simple, easy to implement, and does not introduce significant computational overhead. To demonstrate the effectiveness of our method, we show the results with and without our color-aware regularization on three gradient domain tasks: gradient transfer, gradient boosting, and saliency sharpening.

1 Motivation and Related Work

Gradient domain manipulation is the cornerstone of many image processing algorithms from image editing to texture transfer to image fusion. For an overview of gradient domain algorithms and applications we refer readers to [1]. As the name implies, gradient domain algorithms do not operate in the 0th order domain (i.e. color domain), but instead impose changes to the 1st order derivatives of the input image, i.e. the image gradient. When left unchecked, gradient domain processing can result in noticeable color shifts in the 0th domain output image. To ameliorate color-shifting artifacts, most gradient domain approaches impose an additional 0th order constraint either at the boundary of the processed region or over the entire region.

Early gradient domain processing approaches (e.g. [2–5]) were formulated using the Poisson equation (see [6]) which incorporates a 0th order boundary constraint on the solution, i.e. the Dirichlet boundary condition. While generally sufficient for most processes, this method can, from time to time, exhibit very noticeable color shifts inside the processed region. As a result, other approaches, especially more recent ones (e.g. [1, 7–11]) impose a regularization over the entire 0th order solution. This is typically done using an L_2 norm regularization on one or more of the 0th order image channels. This solution results in a bi-objective function that tries to manipulate the image gradient while minimizing the Euclidean error (i.e. L_2) between the original and output 0th order domains.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 392–405, 2013.

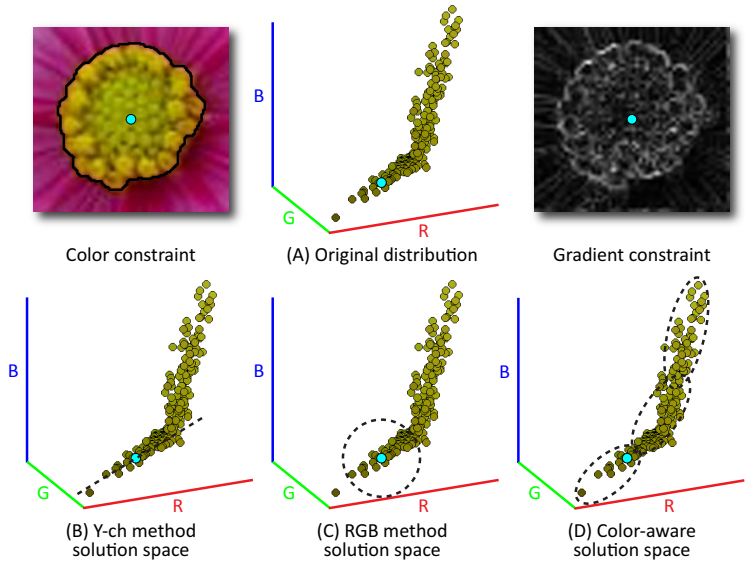


Fig. 1. Solution spaces (denoted by the dotted line) of the marked pixel using different 0th domain regularization methods

This paper targets this latter regularization strategy which is applied in one of two manners, either by 1) first converting the input space (presumably RGB) to a new color space (e.g. YUV or LAB) that separates the luminance and chrominance components and processing only the luminance channel (we refer to this manner as Y-ch method in the rest of this paper); or by 2) applying the L_2 regularization to all three channels separately (we refer to this manner as RGB method). When only one channel is processed, the regularization effectively constrains the output solution so that each pixel is restricted to a 1-D space (Figure 1(B)). Although this approach does not shift the chromacity, it can produce outputs that appear flat and less vivid. This can be seen in Figure 2(B). When all three channels are processed, the per pixel solution space is constrained to lie within the sphere about its original value as shown in Figure 1(C). This conventional regularization is applied irrespective to how the scene colors are distributed in the original input. As a result, satisfying the regularization constraint may also introduce colors that are quite different than those in the original image. This can be seen in Figure 2(C) where the solution of the gradient boosting has resulted in noticeable color shifts.

Our work is motivated by the observation that objects' RGB colors in natural images follow unique distributions. For example, in Figure 1(A), the pixel marked in cyan is plotted with all other pixels belonging to the same object. It is easy to see that the pixel belongs to a distinct color distribution in the RGB space. Such unique distributions observed by Omer and Werman [12] have shown that colors in natural images tend to form elongated clusters (referred to as lines) in the RGB space. Our color-aware approach constrains the

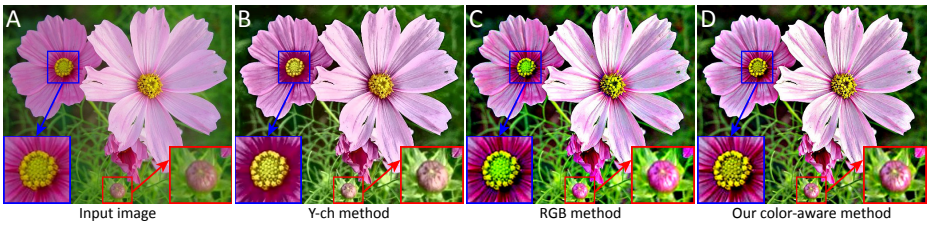


Fig. 2. This figure compares conventional 0th domain regularization applied to an image that has had its gradient boosted. A) Input image. B) Result using L_2 regularization over the Y channel only. C) Result using L_2 regularization over all three channels of the RGB input. D) Our color-aware regularization result. Note the flat output colors exhibited by Y-ch method in B, and the subtle color-shifting exhibited by RGB method in C.

solution space to more tightly follow the original distribution in order to avoid color shifting as shown in Figure 1(D) and Figure 2(D).

Contribution. Our contribution is the introduction of a regularization term that more faithfully follows the distribution of colors in the input image. Our approach applies a simple segmentation to the input image to assign each pixel to a color distribution represented as a Gaussian mixture model (GMM). Using these GMMs we can formulate the color-aware regularization as an anisotropic Mahalanobis distance [13] which can be expressed as a linear system. This color-aware regularization constrains the output solution to better fit the original input color distributions thereby avoiding color shifts. Our approach can be easily incorporated into existing gradient-domain formulations. We demonstrate the effectiveness of this regularization on a variety of inputs using three selected applications, gradient transfer, gradient boosting and saliency sharpening. We compare our results with conventional L_2 regularization approaches (Y-ch method and RGB method) as used by [1, 7–11].

2 Color-Aware Regularization Framework

2.1 Overview

An overview of our framework is shown in Figure 3. Each pixel is first assigned to a color distribution via segmentation. We found that a simple superpixel segmentation [14] followed by k-means clustering [13] is enough to find the underlying color distributions. These individual color distributions are then fit with a series of 3D Gaussian distributions in the RGB color space. The input to our algorithm is an image where each pixel is assigned to a single distribution represented by a series of Gaussians, i.e. $\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_m$.

A bi-objective function is then used to transfer the new gradients to the input while regularizing each output pixel to lie within a minimum distance from one of

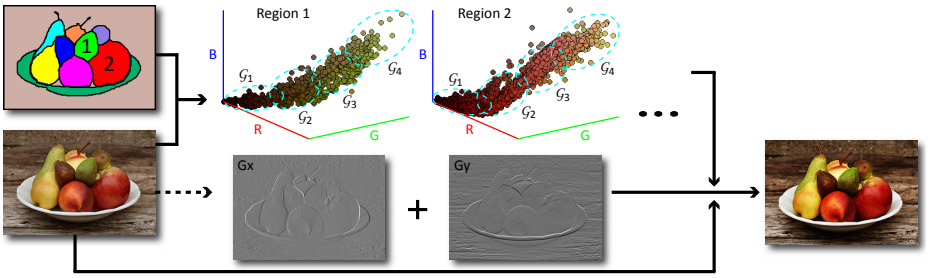


Fig. 3. The overall workflow of our color-aware regularization framework

the Gaussian distributions used to model its associated color distribution. This regularization is formulated as an optimization problem similar to [1, 7–11].

2.2 Conventional Optimization Framework

Taking gradient transfer application as example, we review the conventional optimization framework based on an L_2 regularization term. The purpose of gradient transfer is to transform gradients from the source image to the target image while preserving the original look-and-feel of the target image.

Given two images f and g, we seek a new image u whose colors (from one or more color channels) are as close as possible to f, and at the same time, has gradients that are as close as possible to g. More formally, the final result u is generated by minimizing the following bi-objective cost function

$$E(u) = \sum_{p \in u} (\lambda E_d(p) + E_s(p)), \tag{1}$$

where p is the pixel index of image u; E_d is the 0th order color constraint term and E_s is the 1st order gradient constraint term; λ is used for the balance between E_d and E_s. These two terms are defined as:

$$E_d(p) = (u_p - f_p)^2, \tag{2}$$

$$E_s(p) = \left(\frac{\partial u}{\partial x} - c \cdot \frac{\partial g}{\partial x}\right)_p^2 + \left(\frac{\partial u}{\partial y} - c \cdot \frac{\partial g}{\partial y}\right)_p^2, \tag{3}$$

where $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$ denote the partial derivative operators in x- and y-direction; c is a scaling factor used in gradient boosting and saliency sharpening application and is set to 1.0 for gradient transfer application.

2.3 Color-Aware Regularization Term

As shown in Figure 1, the solution space of each pixel in the resulting image u is constrained either to lie on a 1-D solution space if only a single channel is processed (Y-ch method), or to lie within a sphere centered at each pixel if applied

to all three channels (RGB method). Since the Euclidean distance is blind to the inherent correlation among variables, neither of these methods is able to take into account the color distribution information of the input image f. This can lead to flattened colors or noticeable color shifts in the output image. To solve this problem, we change the conventional L_2 regularization to an anisotropic Mahalanobis distance that more tightly fits the original color distribution. By using the Mahalanobis distance, 0th domain solutions along the shorter axis of each pixel's associated Gaussian model are penalized. This forces the solution to move along the longer axis, thus constraining the solution to lie closer to the original color distribution.

Single Gaussian Model. We first consider the case where we can model a color distribution using a single Gaussian distribution. We define our color-aware 0th order regularization term as:

$$E_{mdd}(p) = (\boldsymbol{u}_p - \boldsymbol{f}_p)^T S_p^{-1} (\boldsymbol{u}_p - \boldsymbol{f}_p), \tag{4}$$

where p is the pixel index; both $\boldsymbol{u}_p$ and $\boldsymbol{f}_p$ are the RGB pixel values represented by 3D column vectors; S_p is a 3×3 covariance matrix of the Gaussian that pixel p is assigned to. The term E_{mdd} is the squared Mahalanobis distance, which is a dissimilarity measure between the two vectors $\boldsymbol{u}_p$ and $\boldsymbol{f}_p$. The benefit of the Mahalanobis distance is that, unlike the Euclidean distance, it considers the correlation of data elements in the vector, in our case the pixels' RGB values.

Combining Eq. 3 and Eq. 4 using matrix notation we can write our quadratic form bi-objective cost function as

$$\begin{aligned} E(u) &= \lambda E_{mdd}(u) + E_s(u) \\ &= \lambda (u - f)^T \Sigma (u - f) \\ &\quad + (G_x u - c \cdot G_x g)^T (G_x u - c \cdot G_x g) \\ &\quad + (G_y u - c \cdot G_y g)^T (G_y u - c \cdot G_y g), \end{aligned} \tag{5}$$

where u, f and g are RGB images reshaped into the column vector form (e.g. $[R_1 G_1 B_1 \dots R_N G_N B_N]^T$); Σ is a 3N×3N (N is the number of pixels) block-diagonal matrix containing the 3×3 inverse covariance matrices of Gaussian models that each pixel is assigned to; the matrices G_x and G_y are discrete forward differentiation operators. Note that gradient constraint g does not necessarily form a 3-channel image since we may transfer gradients of a grayscale image to a color image (see Section 3). In that case, image g is extended to an RGB image by copying itself three times. Minimizing Eq. 5 amounts to taking its derivative, setting it to zero, and solving for vector u that is uniquely defined as the solution of the linear system:

$$(\lambda \Sigma + G_x^T G_x + G_y^T G_y) u = \lambda \Sigma f + c \cdot (G_x^T G_x g + G_y^T G_y g). \tag{6}$$

To solve this linear system, we use the conjugate gradient (CG) method [15] that is also used by [16] and [1]. Note that further improvement can be made

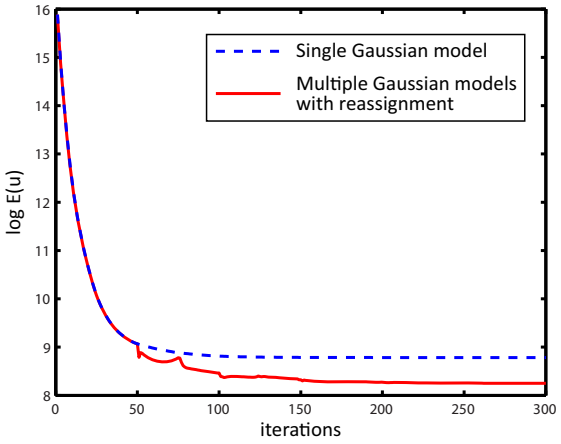


Fig. 4. Comparison of cost values (with spatial-varying weights applied) when using single Gaussian model (blue dashed line) and multiple Gaussian models (red solid line). For multiple Gaussian models, the reassignment operation is carried out every 50 iterations ($t = 50$ in CG solver).

to the 1st order term $E_s(u)$ in Eq. 5 since the L_2 norm is known to be sensitive to noise and may result in haloing artifacts in the output image. To solve this problem, we add two spatial-varying weights to $E_s(u)$ using the same weighting scheme presented in [1]:

$$w^x(p) = \left(\left| \frac{\partial f}{\partial x} - \frac{\partial g}{\partial x} \right|_p + 1 \right)^{-\alpha} \tag{7}$$

$$w^y(p) = \left(\left| \frac{\partial f}{\partial y} - \frac{\partial g}{\partial y} \right|_p + 1 \right)^{-\alpha} \tag{8}$$

where the parameter α (typically $1.2 \leq \alpha \leq 3$) determines the sensitivity of $E_s(u)$ to noise. By using this per-pixel weighting scheme halo artifacts are effectively reduced.

Multiple Gaussian Models. Instead of using a single Gaussian model per color distribution, we use several Gaussian models to represent each color distribution more precisely. As shown in Figure 3, each pixel is first assigned to a color distribution (region) via segmentation. Each color distribution is represented by a series of 3D Gaussian models $\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_m$ and each pixel is initially assigned to its nearest Gaussian model $\mathcal{G}_i$ via Eq. 4. All pixels in the same region (color distribution) share the same set of Gaussian models and can be reassigned to any Gaussian models within this set when iteratively solving the output image. We integrate this reassignment scheme with the conjugate gradient algorithm and show that it can further decrease the objective cost function (see Figure 4).

Assume that we divide the input image into k color distributions and each distribution is represented by m 3D Gaussian models, resulting in $k \times m$ Gaussian models in total; $\mathcal{G}_{i,j}$ is the jth Gaussian model of the ith color distribution

Algorithm 1. Gaussian model reassignment

Require: input image f and g, initial assignment map of all pixels ASG (a matrix), maximum number of reassignment operations T, number of CG solver iterations t, small tolerance $\epsilon > 0$

$u = f$
for $reselect = 1$ to T **do**
  $u = \text{conjugate\_gradient\_solver}(f, g, u, ASG, t)$; $ASG\_old = ASG$;
  for $i = 1$ to k **do**
    for all $p \in Region(i)$ **do**
      $j_0 = \underset{j \in [1,m]}{\text{argmin}} \ (\boldsymbol{u}_p - \boldsymbol{f}_p)^T S_{i,j}^{-1} (\boldsymbol{u}_p - \boldsymbol{f}_p)$
      reassign p to the Gaussian model $\mathcal{G}_{i,j_0}$ (one element of ASG is updated)
    end for
  end for
  if $\|ASG - ASG\_old\|_F < \epsilon$ **then**
    break
  end if
end for
return the output image u

$(1 \leq i \leq k; 1 \leq j \leq m)$ and $S_{i,j}$ is the covariance matrix of $\mathcal{G}_{i,j}$. The expression $\|A\|_F$ denotes the Frobenius norm of matrix A. We now outline the overall algorithm of our reassignment approach as shown in Algorithm 1.

Convergency Analysis. Without using spatial-varying weights on the 1st order constraint term, minimizing the conventional bi-objective cost function reviewed in Section 2.2 is known to be a convex problem. Our color-aware optimization framework (using single Gaussian model) differs from the conventional formulation by only introducing a block-diagonal matrix Σ on both sides of the linear system $Au = b$ (see Eq. 6). We know that the covariance matrix Σ is positive-semidefinite. As a result, introducing the matrix Σ does not violate the convex property of this optimization problem and a global optimal solution exists.

When using multiple Gaussian models and the reassignment scheme, the convex property remains intact. As shown in Algorithm 1, the reassignment scheme is actually a combination of several independent conjugate gradient solving procedures. After each reassignment step is done, the cost value is guaranteed to be decreased (or at least remain unchanged) by reassigning each pixel to the Gaussian model whose covariance matrix can minimize the 0th order term E_{mdd} while keeping the 1st order term E_s unchanged.

However, the optimization problem is no longer convex once the spatial-varying weights are used. In this case, the global optimum solution may not exist, but we can still use conjugate gradient method to find an appropriate solution. In practice, we find our framework works well to minimize Eq. 5 within 250 iterations (see Section 3.1). Two plots of the cost values during conjugate gradient iterations are shown in Figure 4. As we can see, with the help of multiple Gaussian models and the reassignment scheme, the cost value has been further

decreased compared to the result achieved by the single Gaussian model. Note that the cost values are shown in *log* scale.

3 Experiments

We compare results obtained by our color-aware regularization against those obtained using a conventional optimization framework [1, 7–11] based on L_2 0th order regularization in the two manners previously discussed (i.e Y-ch method and RGB method). The fast deconvolution algorithm presented by [17] is used to perform the conventional optimization. Comparisons are conducted on three selected tasks including gradient transfer, gradient boosting and saliency sharpening. Before carrying out experiments we briefly explain the parameters we used for these tasks.

3.1 Experiment Setups

For all the three methods, the gradient scaling factor c is set to 1.0 in gradient transfer task and 2.0 or 3.0 in gradient boosting/saliency sharpening tasks. To keep the comparisons fair, we adjust the balancing factor λ for each method to make sure that a comparable amount of gradient has been transferred or boosted for each example (see quantitative comparison in Section 3.3).

For our color-aware regularization method, we use over segmentation algorithm followed by k-means clustering to detect underlying color distributions of an image and k is chosen from [10, 15] range (see Section 3.3 for detail explanation). The number of Gaussian models used to represent each color distribution is fixed to $m = 5$. We restrict the number of Gaussian reassignment operations within 5 times ($T = 5$) and set 50 iterations for the CG solver ($t = 50$). With the above settings, the running time for an 800×600 image is around 3 minutes (Matlab implementation on an Intel Core 2 Duo 2.8GHz computer). We note using more than 5 Gaussian models does not significantly improve the results.

3.2 Image Gradient Manipulation Tasks

Gradient Transfer. The first two examples demonstrate gradient transfer of the gradients from a near-infrared (NIR) image to an ordinary RGB image. Such gradient transfer has been demonstrated to improve some forms of photography [8, 18] since NIR often contain more details that cannot be seen in the visible spectrum. In the first example, we show an example of an outdoor scene of a castle where the clouds and other textures are notably stronger in the NIR image. Two input images (NIR and RGB) are shown in Figure 5(A-a) and Figure 5(A-b). Figure 5(A-c) shows the result generated by the Y-ch method. While the desired gradients (clouds) are transferred, the color of the green plants below the castle change to cyan. Figure 5(A-d) shows the result produced by the RGB method. Note that the red color of the plants and rocks changes to green. Our result is

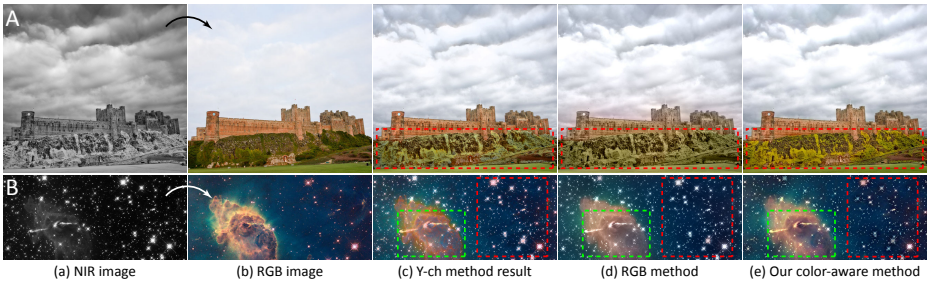


Fig. 5. Examples of gradient transfer: (a) input NIR image; (b) input RGB image; (c) result using L_2 regularization over the Y channel only; (d) result using L_2 regularization over R/G/B channels; (e) our color-aware regularization result. Regions with color-shifting problem have been highlighted in red and green dashed boxes.

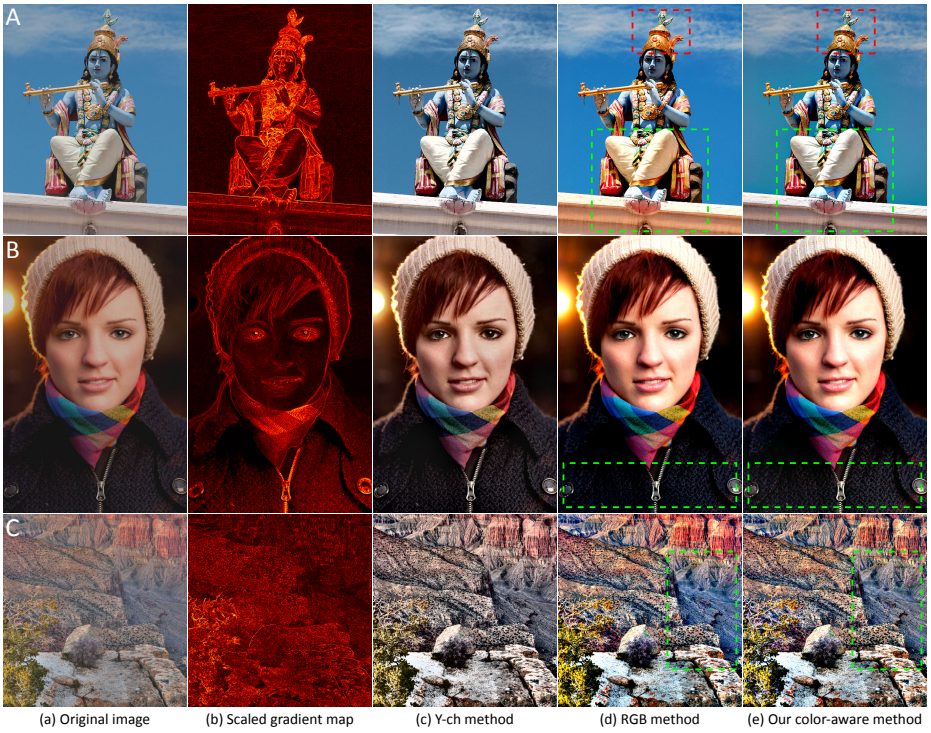


Fig. 6. Examples of gradient boosting: (a) input RGB image; (b) scaled gradient map providing target gradients; (c) result using L_2 regularization over the Y channel only; (d) result using L_2 regularization over R/G/B channels; (e) our color-aware regularization result. Regions with color-shifting problem have been highlighted in green dashed boxes.

Fig. 7. Examples of saliency sharpening: (a) input RGB image; (b) saliency map of the input image; (c) result using L_2 regularization over the Y channel only; (d) result using L_2 regularization over R/G/B channels; (e) our color-aware regularization result. Regions with color-shifting problem have been highlighted in red and blue dashed boxes.

shown in Figure 5(A-e). The colors of both the red rocks and green plants are preserved well. Another example is shown in the second row of Figure 5. Note that the color of the nebula (highlighted by a green dashed box) changes significantly in Figure 5(B-c) and the color of the stars (highlighted by a red dashed box) is washed out in Figure 5(B-d). Our method achieves a better result in Figure 5(B-e) with colors that are more similar to the input RGB image.

Gradient Boosting. The second example targets gradient boosting that is aimed to enhance image contrast. In Figure 6, column (a) shows original input images; column (b) shows the scaled gradient magnitudes after boosting (rendered as a *hot map* for better visualization); column (c), (d) and (e) are the results generated by the three different methods. We can see that when using the RGB method (column (d) in Figure 6), the results suffer from noticeable color-shifting in some regions. For instance, the color of the wall and the Buddha's legs in example A become yellowish; the color of the woman's clothing in example B changes from brown to blue; the color of the valley in example C also shifts to blue. Although less color shifts is noticeable when using the Y-ch method (column (c) in Figure 6), the overall color of these images seems to be flattened and less vivid. Our results (column (e) in Figure 6) show the images with boosted contrast and without color shifts. In addition, our results are more vivid and colorful compared to the Y-ch method.

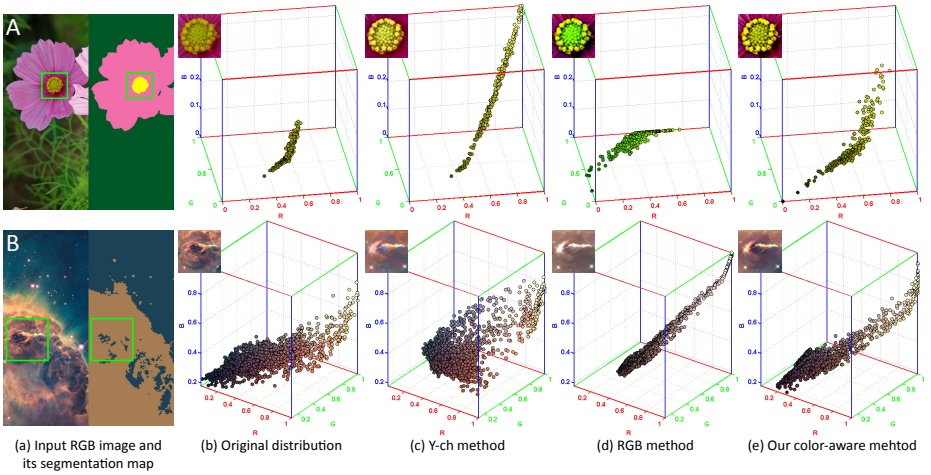


Fig. 8. Distributions of the solutions using different 0th domain regularization methods: (a) input RGB image and its segmentation map; (b) original color distribution of the selected region (highlighted in green solid boxes); (c) resulting distribution using L_2 regularization over the Y channel only; (d) resulting distribution using L_2 regularization over R/G/B channels; (e) our color-aware regularization distribution. Note that our distribution better maintains the shape and trend of the original.

Saliency Sharpening. Saliency sharpening is similar to gradient boosting application. The only difference is that the gradient boosting globally enhances gradients by a factor c, while saliency sharpening strengthens gradients in a spatially varying manner based on the image saliency map. We adopted the gradient attenuation function proposed in [2] to generate a grayscale saliency map M (brighter regions indicate larger scale factors and stronger boosting). In this case, the global scale factor c in Eq. 5 will be replaced by $(1 + c \cdot M)$. As shown in Figure 7, our method produces results visually more appealing compared to the other two methods. Note the visible color-shifting on the wall behind the tiger (Figure 7(A-d)), the cloud above the rock (Figure 7(B-d)) and the sunflower (Figure 7(C-d)). Again, results from the Y-ch method (column (c) in Figure 7) appear flat similar to the examples in gradient boosting application. However, our results (column (e) in Figure 7) successfully preserve the original color of input images after saliency sharpening process.

3.3 Evaluation and Analysis

In order to show how our color-aware regularization method preserves the original color distribution more faithfully than the other two methods, we plot the original color distribution of a selected region in the input image and compare it with color distributions of the same region in three output images. In Figure 8, column (a) shows the input image and its color-coded segmentation map; column (b) plots the color distribution (data points are randomly sub-sampled for better visualization) of the selected region in the RGB space; column (c), (d)

Fig. 9. Comparison of other color spaces: (a) input RGB images; (b), (c) gradient boosted results using L_2 regularization over the luminance/brightness channel of YIQ/HSV color spaces; (d) result of L_2 regularization over all channels of LAB color space; (e) our color-aware regularization result

and (e) plot different results generated by the Y-ch method, the RGB method and our method respectively. The plots show that the color distribution of our output image (selected regions) preserves the original distribution much more faithfully than the other two in terms of shape and trend.

Other than YUV and RGB spaces, we also compared our method with traditional L_2 regularization applied on other commonly-used color spaces. Similar to the Y-ch method, we convert the input image into YIQ/HSV color space and regularize the luminance/brightness channel only. As shown in Figure 9(b, c), the results are similar to that of the Y-ch method and also suffer from flattened colors due to the limitation that the output pixels are restricted to a 1-D space (refer to Figure 1(B)). Similar to the RGB method, we convert the input image into LAB color space and regularize three channels separately. Using LAB color space we get the result (Figure 9(d)) that also appears flat and less colorful compared to our result (Figure 9(e)).

Table 1. This table shows the overall amount of gradient transferred by each method (average L_2 difference between output and input gradients) is similar for all example images shown in Figure 5(A, B), Figure 6(A, B, C) and Figure 7(A, B, C)

Methods	Figure 5		Figure 6			Figure 7		
	A	B	A	B	C	A	B	C
Y-ch method	0.0040	0.0047	0.0369	0.0148	0.0899	0.0849	0.1344	0.0518
RGB method	0.0041	0.0036	0.0372	0.0123	0.0591	0.0757	0.1244	0.0467
Our method	0.0044	0.0046	0.0340	0.0113	0.0533	0.0747	0.1182	0.0453

We also want to examine the amount of gradient effectively transferred by each method. To do so, we compare the average per-pixel Euclidean distance of the gradient maps of three output images with the constrained gradient map. Table 1 lists the amount of gradient transferred for each example. Note that all methods transfer a comparable amount of gradient. This verifies that 1) our approach is able to transfer gradient as effective as the other methods; and 2) the results shown are fairly compared because they have each transferred approximately the same amount of gradient.

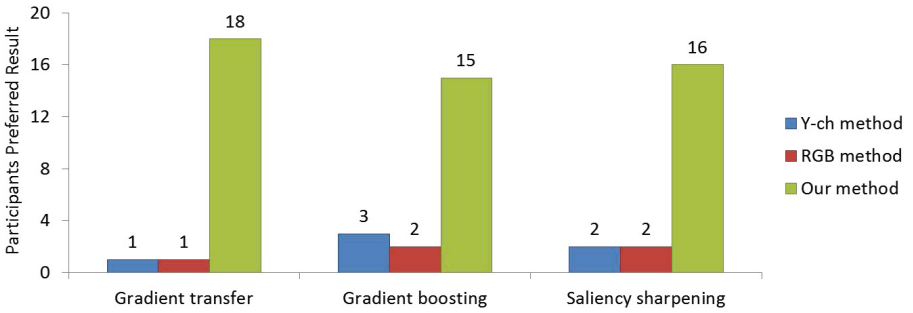


Fig. 10. Participants preferred results of three different methods

Our color distributions are determined by over segmentation followed by k-means clustering, resulting in k distributions, each of which is further decomposed into GMMs. To determine the sensitivity of our results to the choice of k, we performed experiments with k ranging from 5 to 40. We found the results dose not vary too much for values of k greater than 15. As a result, we advocate using the range [10, 15].

Lastly, since our approach is subjective in nature, we performed a simple user study on user's preference of the results on 14 examples (3 for gradient transfer, 7 for gradient boosting and 4 for saliency sharpening). Twenty participants (average age around 25) were asked to choose their preferred results out of the outputs of the three different methods. Participants were not trained before the experiment, but over half of them had experience with image editing software such as Photoshop. Our user study showed that 18 participants preferred our results for the gradient transfer application, and 15 participants preferred our results for the gradient boosting application. For saliency sharpening application, 16 participants preferred the results produced by our color-aware regularization method. Figure 10 shows a graph of these results.

4 Conclusion

We have presented a straight-forward approach to perform 0th domain regularization in a manner that more faithfully follows the original input color distribution. This results in gradient transfer that avoids color shifting while still producing vivid results. While our approach requires an initial segmentation to determine the distinct color distributions in the image, we found that the segmentation stage is not a crucial issue and any basic over segmentation algorithm (e.g. watershed [19] or superpixel [14]) gave good results. More sophisticated segmentation algorithm like Ridge-based Distribution Analysis [20] were tried but generated similar results. We also note that our approach is not significantly slower than conventional techniques and can be easily incorporated into existing image gradient manipulation methods.

Acknowledgement. This work was supported by the NUS Young Investigator Award, R-252-000-379-101, and the IT Consilience Creative Program of the Ministry of Knowledge Economy, Korea.

References

1. Bhat, P., Zitnick, C., Cohen, M., Curless, B.: Gradientshop: a gradient-domain optimization framework for image and video filtering. ACM Transactions on Graphics 29, 1–14 (2010)
2. Fattal, R., Lischinski, D., Werman, M.: Gradient domain high dynamic range compression. ACM Transactions on Graphics 21, 249–256 (2002)
3. Jia, J., Sun, J., Tang, C., Shum, H.: Drag-and-drop pasting. ACM Transactions on Graphics 25, 631–637 (2006)
4. Raskar, R., Ilie, A., Yu, J.: Image fusion for context enhancement and video surrealism. ACM SIGGRAPH Courses (2005)
5. McCann, J., Pollard, N.: Real-time gradient-domain painting. ACM Transactions on Graphics 27, 1–7 (2008)
6. Pérez, P., Gangnet, M., Blake, A.: Poisson image editing. ACM Transactions on Graphics 22, 313–318 (2003)
7. Zeng, Y., Chen, W., Peng, Q.: A novel variational image model: Towards a unified approach to image editing. Journal of Computer Science and Technology 21 (2006)
8. Krishnan, D., Fergus, R.: Dark flash photography. ACM Transactions on Graphics 28 (2009)
9. Yang, W., Zheng, J., Cai, J., Rahardja, S., Chen, C.: Natural and seamless image composition with color control. IEEE Transactions on Image Processing 18 (2009)
10. Xiao, X., Ma, L.: Gradient-preserving color transfer. Computer Graphics Forum 28, 1879–1886 (2009)
11. Shibata, T., Iketani, A., Senda, S.: Image Inpainting Based on Probabilistic Structure Estimation. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part III. LNCS, vol. 6494, pp. 109–120. Springer, Heidelberg (2011)
12. Omer, I., Werman, M.: Color lines: Image specific color representation. IEEE Computer Vision and Pattern Recognition (2004)
13. Duda, R., Hart, P., Stork, D.: Pattern classification, vol. 2. Wiley, New York (2001)
14. Ren, X., Malik, J.: Learning a classification model for segmentation. In: IEEE International Conference on Computer Vision (2003)
15. Avriel, M.: Nonlinear programming: Analysis and methods. Dover Publishing (2003)
16. Levin, A., Fergus, R., Durand, F., Freeman, W.: Image and depth from a conventional camera with a coded aperture. ACM Transactions on Graphics 26 (2007)
17. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-laplacian priors. In: Advances in Neural Information Processing Systems, vol. 22, pp. 1–9 (2009)
18. Zhang, X., Sim, T., Miao, X.: Enhancing photographs with near infra-red images. In: IEEE Computer Vision and Pattern Recognition (2008)
19. Vincent, L., Soille, P.: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence 13, 583–598 (1991)
20. Vazquez, E., Baldrich, R., van de Weijer, J., Vanrell, M.: Describing reflectances for color segmentation robust to shadows, highlights, and textures. IEEE Transactions on Pattern Analysis and Machine Intelligence (2010)

Local Covariance Filtering for Color Images

Keiichiro Shirai[1], Masahiro Okuda[2], Takao Jinno[3], Masayuki Okamoto[1], and Masaaki Ikehara[4]

[1] Shinshu Univ.
[2] Univ. of Kitakyushu
[3] Toyohashi Tech.
[4] Keio Univ.

Abstract. In this paper, we introduce a novel edge-aware filter that manipulates the local covariances of a color image. A covariance matrix obtained at each pixel is decomposed by the singular value decomposition (SVD), then diagonal eigenvalues are filtered by characteristic control functions. Our filter form generalizes a wide class of edge-aware filters. Once the SVDs are calculated, users can control the filter characteristic graphically by modifying the curve of the characteristic control functions, just like tone curve manipulation while seeing a result in real-time. We also introduce an efficient iterative calculation of the pixel-wise SVD which is able to significantly reduce its execution time.

1 Introduction

As a basic image processing tool, edge-aware smoothing plays an important role in various applications. Many other image processing techniques such as detail manipulation and contrast compression for tone mapping are based on the smoothing operation. Moreover the performances are dependent on the characteristic of the smoothing filters. One of well-known smoothing filters, the bilateral filter [1] has been used for many applications, and there has been proposed many methods to improve their processing speed [2,3]. But the bilateral filter often causes perceptual artifacts near edges, such as ringing and halos.

As for high performance filters, some optimization-based filters are proposed [4,5]. The weighted least squares (WLS) filter [4] generates halo-free decompositions of input images. However this method requires a time-consuming procedure to solve large linear systems, and the iterative optimization is required especially when using the large size filter kernel. The guided-filter [5] is one of the state-of-the-art edge-aware filters based on the local linear system of the image matting method [6], and works with low computational complexity independent of the kernel size by virtue of using the integral image [7].

In this paper, we newly develop a more general form of the edge-aware filter based on the guided-filter [5] and extend the filter performance. In our framework, the guided-filter can be considered a special case of our class. Our contribution is listed as follows:

Filter Expression via SVD and Characteristic Control Functions. To control the texture detail and base color, the singular value decomposition (SVD)

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 406–417, 2013.

is applied to the covariance matrices of local regions. Then the eigenvalues in the diagonal matrix are transformed by two "characteristic control functions" (*cc*-functions). Manipulating the shape of the functions, various types of the edge-aware filters are obtained.

Flexible Graphical Manipulation for Optimization-Based Filters. Unlike conventional parameter tuning by slide bars, the filtering effect can be controlled more flexibly such as the well-known tone curve manipulation and the detail manipulation for the Laplacian pyramid [8].

Pixel-Wise SVD Calculation. We also propose an alternative pixel-wise SVD decomposition algorithm exploiting the characteristic of the SVD for the symmetric covariance matrix, and explain how to accelerate execution speed in our class of filters, including the conventional guided-filter.

The rest of this paper is organized as follows. Sec. 2 describes the derivation of our filter and shows the *cc*-functions, Sec. 3 describes the effective method for improving the processing speed. Sec. 4 shows experimental results and criteria for controlling the amount of effects.

2 Derivation

2.1 Energy Function Based on Guided-Filter [5]

The basic energy function of our filter is based on the "self smoothing mode" of the guided-filter. The guided-filter uses a guidance image $\mathbf{I}$ to smooth a degraded input image $\mathbf{p}$ (e.g. a noisy non-flash image $\mathbf{p}$ can be denoised while keeping its color by using edge information of a flash image $\mathbf{I}$ [5,9]). The restoration is performed by transforming the local color distribution of $\mathbf{I}$ by a 3×3 transform matrix $\mathbf{A}$ and a 3×1 translate vector $\mathbf{b}$ so as to imitate the distribution of $\mathbf{p}$ at each pixel:

$$\{\mathbf{A}_k^*, \mathbf{b}_k^*\} = \arg\min_{\mathbf{A},\mathbf{b}} \sum_{j \in \mathcal{N}(k)} \|\mathbf{A}_k \mathbf{I}_j + \mathbf{b}_k - \mathbf{p}_j\|^2 + \eta\epsilon \|\mathbf{A}_k\|^2_{\text{trace}}, \tag{1}$$

where $\mathbf{I}_j$ and $\mathbf{p}_j$ ($\in R^3$) consists of the RGB values at pixel j of the guidance and input images, respectively. ϵ is a regularization parameter, η is the number of pixels in a filter window. The locally optimal $\mathbf{A}_k^*$ and $\mathbf{b}_k^*$ are first calculated at each pixel k by using color information of neighboring pixels $j_{\in \mathcal{N}(k)}$. Then resulting pixel colors $\mathbf{q}_i^*$ are calculated at each pixel i by using $\mathbf{A}_k^*$ and $\mathbf{b}_k^*$ obtained at neighboring pixels $k_{\in \mathcal{N}(i)}$:

$$\mathbf{q}_i^* = \arg\min_{\mathbf{q}} \sum_{k \in \mathcal{N}(i)} \|\mathbf{A}_k^* \mathbf{I}_i + \mathbf{b}_k^* - \mathbf{q}_i\|^2. \tag{2}$$

Meanwhile the "self smoothing mode" is a special case which quotes an input image itself as the corresponding guidance image: $\mathbf{p}_j = \mathbf{I}_j$. In this case, the regularization parameter ϵ works as a smoothing controller, and as a result this filter behaves like the edge-aware smoothing filter. The least squares solution of Eq.1 under the condition $\mathbf{p}_j = \mathbf{I}_j$ is given by:

$$\begin{gathered}\mathbf{q}_i^* = (\frac{1}{\eta}\sum_{k\in\mathcal{N}(i)}\mathbf{A}_k^*)\ \mathbf{I}_i + (\frac{1}{\eta}\sum_{k\in\mathcal{N}(i)}\mathbf{b}_k^*), \\ \text{where } \mathbf{A}_k^* = \mathbf{C}_k\left(\mathbf{C}_k + \epsilon\mathbf{I_d}\right)^{-1},\ \ \mathbf{b}_k^* = (\mathbf{I_d} - \mathbf{A}_k^*)\boldsymbol{\mu}_k,\end{gathered} \tag{3}$$

the matrix $\mathbf{C}_k = (\frac{1}{\eta}\sum_{j\in\mathcal{N}(k)}\mathbf{I}_j^T\mathbf{I}_j) - \boldsymbol{\mu}_k^T\boldsymbol{\mu}_k$ is a covariance matrix, $\boldsymbol{\mu}_k = \frac{1}{\eta}\sum_{j\in\mathcal{N}(k)}\mathbf{I}_j$ is a mean color vector, and $\mathbf{I_d}$ is an identity matrix. In this equation, all summations $(\frac{1}{\eta}\sum\cdot)$ are calculated by box-filtering using the integral images [7]. For more detail of the derivation, please refer to [5].

2.2 Local Covariance Filtering via SVD

Our new type of filtering is characterized by the following filter equation composed of a linear combination of an input image $\mathbf{I}$ and the filtered image $\mathbf{q}^*$:

$$\mathbf{J}_i = \alpha\mathbf{I}_i + \beta\mathbf{q}_i^*, \tag{4}$$

where a filtered image $\mathbf{J}$ is obtained by the pixel-wise operation. This equation can be interpreted as a general form for the edge-aware filters, for example, the contrast enhancement [5]: $\mathbf{J}_i = s(\mathbf{I}_i - \mathbf{q}_i^*)^{\text{Detail}} + \mathbf{q}_i^{*\text{Base}}$ where parameters are set as $s > 0$, $\alpha = s$, $\beta = 1 - s$, and the contrast compression for tone-mapping [2]: $\mathbf{J}_i = (\mathbf{I}_i - \mathbf{q}_i^*)^{\text{Detail}} + t\mathbf{q}_i^{*\text{Base}}$ where $t < 0$, $\alpha = 1$, $\beta = t - 1$.

Eq. 4 results in the following equation by applying the SVD to the symmetric covariance matrix $\mathbf{U}_k\mathbf{D}_k\mathbf{U}_k^T = \text{SVD}(\mathbf{C_k})$, and employing two functions to the diagonal matrices:

$$\mathbf{J}_i = \{\frac{1}{\eta}\sum_{k\in\mathcal{N}(i)}(\mathbf{U}_k f_{\text{Detail}}(\mathbf{D}_k)\mathbf{U}_k^T)\}\ \mathbf{I}_i + \{\frac{1}{\eta}\sum_{k\in\mathcal{N}(i)}(\mathbf{U}_k f_{\text{Base}}(\mathbf{D}_k)\mathbf{U}_k^T)\ \boldsymbol{\mu}_k\}, \tag{5}$$

where f_{detail} is the edge variance control function and f_{base} is the mean color control function, and we call them the characteristic control functions (*cc*-functions). Furthermore the summation of the functions satisfies:

$$f_{\text{Detail}} + f_{\text{Base}} = \alpha + \beta\ (= \text{const.}), \tag{6}$$

and it controls the tone scale of output images.

The remarkable feature is that the shapes of functions $f_{\text{Detail}}, f_{\text{Base}}$ intuitively give the filter characteristic graphically, and enable users to control the amount of effects like tone-curve (explained in 2.3).

Proof of Eq. 5. Substituting $\mathbf{q}_i^*$ in Eq. 3 to Eq. 4 and manipulating $\mathbf{b}_k$, Eq. 4 becomes:

$$\begin{aligned}\mathbf{J}_i &= \alpha\mathbf{I}_i + \beta(\tfrac{1}{\eta}\textstyle\sum_k \mathbf{A}_k^*)\mathbf{I}_i + \beta(\tfrac{1}{\eta}\textstyle\sum_k \mathbf{b}_k^*) \\ &= \{(\tfrac{1}{\eta}\textstyle\sum_k \alpha\mathbf{I_d}) + (\tfrac{1}{\eta}\textstyle\sum_k \beta\mathbf{A}_k^*)\}\mathbf{I}_i + (\tfrac{1}{\eta}\textstyle\sum_k \beta\mathbf{b}_k^*) \\ &= \{\tfrac{1}{\eta}\textstyle\sum_k(\alpha\mathbf{I_d} + \beta\mathbf{A}_k^*)\}\mathbf{I}_i + \{\tfrac{1}{\eta}\textstyle\sum_k(\beta\mathbf{I_d} - \beta\mathbf{A}_k^*)\boldsymbol{\mu}_k\}\end{aligned} .$$

Meanwhile, applying the SVD to the covariance matrices of $\mathbf{A}^* = \mathbf{C}\,(\mathbf{C} + \epsilon\mathbf{I_d})^{-1}$ (hereafter we omit the subscript $_k$ for simplicity) and considering the SVD's characteristics of orthogonality $\mathbf{I_d} = \mathbf{U}\mathbf{U}^T = \mathbf{U}\mathbf{I_d}\mathbf{U}^T$ and its inverse $(\mathbf{U}\mathbf{D}\mathbf{U}^T)^{-1} = \mathbf{U}\mathbf{D}^{-1}\mathbf{U}^T$, the matrix $\mathbf{A}^*$ is rewritten as:

$$\mathbf{A}^* = \mathbf{U}\mathbf{D}\mathbf{U}^T(\mathbf{U}\mathbf{D}\mathbf{U}^T + \epsilon\mathbf{U}\mathbf{I_d}\mathbf{U}^T)^{-1} = \mathbf{U}\mathbf{D}(\mathbf{D} + \epsilon\mathbf{I_d})^{-1}\mathbf{U}^T.$$

Using this result, the elements in the summations of $\mathbf{J}$ are given as:

$$\begin{aligned}\alpha\mathbf{I_d} + \beta\mathbf{A}^* &= \alpha\mathbf{U}\mathbf{I_d}\mathbf{U}^T + \beta\mathbf{U}\mathbf{D}(\mathbf{D} + \epsilon\mathbf{I_d})^{-1}\mathbf{U}^T \\ &= \mathbf{U}\{\alpha\mathbf{I_d} + \beta\mathbf{D}(\mathbf{D} + \epsilon\mathbf{I_d})^{-1}\}\mathbf{U}^T \\ \beta\mathbf{I_d} - \beta\mathbf{A}^* &= \mathbf{U}\{\beta\mathbf{I_d} - \beta\mathbf{D}(\mathbf{D} + \epsilon\mathbf{I_d})^{-1}\}\mathbf{U}^T\end{aligned} .$$

In addition, elements in $\{\cdot\}$ in the above equation are diagonal matrices. Therefore this part can be rewritten by using the diagonal elements $\mathbf{D} = \mathrm{diag}(\{d_m\}_{m=1}^3)$ and $\epsilon\mathbf{I_d} = \mathrm{diag}(\{\epsilon, \epsilon, \epsilon\})$ as follows:

$$\begin{aligned}\alpha\mathbf{I_d} + \beta\mathbf{A}^* &= \mathbf{U}\mathrm{diag}(\{\alpha + \beta\tfrac{d_m}{d_m+\epsilon}\}_{m=1}^3)\mathbf{U}^T \\ \beta\mathbf{I_d} - \beta\mathbf{A}^* &= \mathbf{U}\mathrm{diag}(\{\beta - \beta\tfrac{d_m}{d_m+\epsilon}\}_{m=1}^3)\mathbf{U}^T\end{aligned} .$$

Finally, comparing Eq. 5 and the above equation, we obtain $f_{\mathrm{Detail}}(x) = \frac{(\alpha+\beta)x+\alpha\epsilon}{x+\epsilon}$, $f_{\mathrm{Base}}(x) = \frac{\beta\epsilon}{x+\epsilon}$ as the *cc*-functions.

2.3 Characteristic Control Functions and the Shapes

We examine the relationships between some typical filters and the *cc*-functions in Fig. 1.

Edge-Aware Smoothing[5] : $\mathbf{J}_i = \mathbf{q}_i^{*\mathrm{Base}}$ is given by:

$$f_{\mathrm{Detail}}(x) = \frac{x}{x+\epsilon}, \qquad f_{\mathrm{Base}}(x) = 1 - \frac{x}{x+\epsilon}. \tag{7}$$

In Fig. 1(a), small edges are cut out since $\lim_{x\to 0} f_{\mathrm{Detail}}(x) = 0$. In contrast large edges remain since $\lim_{x\to\infty} f_{\mathrm{Detail}}(x) = 1$. The curve around $x = 0$ is controlled by ϵ.

Edge-Aware Enhancing[5] : $\mathbf{J}_i = s(\mathbf{I}_i - \mathbf{q}_i^*)^{\mathrm{Detail}} + \mathbf{q}_i^{*\mathrm{Base}}$ s.t. $s > 1$ is given by:

$$f_{\mathrm{Detail}}(x) = \frac{x+s\epsilon}{x+\epsilon}, \qquad f_{\mathrm{Base}}(x) = 1 - \frac{x+s\epsilon}{x+\epsilon}. \tag{8}$$

In Fig. 1(a), small edges are enhanced since $\lim_{x\to 0} f_{\mathrm{Detail}}(x) = s$. In contrast large edges remain since $\lim_{x\to\infty} f_{\mathrm{Detail}}(x) = 1$.

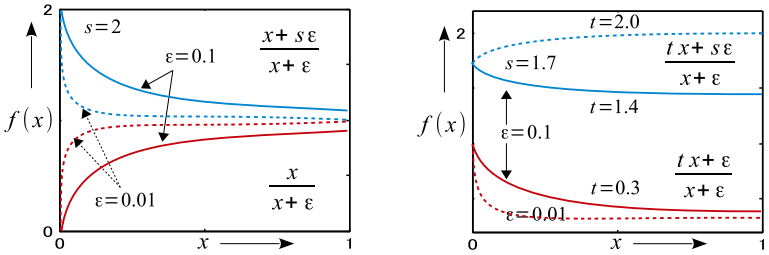


Fig. 1. The shape of characteristic control functions (only $f_{\mathrm{Dtail}}(x)$ is shown). (a) Smoothing (bottom) and Enhancing (top), (b) Base compression (bottom) and Our flexible control (top).

Contrast Compression for Tone Mapping[2] : $\mathbf{J}_i = (\mathbf{I}_i - \mathbf{q}_i^*)^{\mathrm{Detail}} + t\mathbf{q}_i^{*\mathrm{Base}}$ s.t. $t < 1$ is given by:

$$f_{\mathrm{Detail}}(x) = \frac{tx+\epsilon}{x+\epsilon}, \qquad f_{\mathrm{Base}}(x) = t - \frac{tx+\epsilon}{x+\epsilon}. \tag{9}$$

As one can see from Fig. 1(b), f_{Detail} implicitly works same as the enhancing function in Eq. 8, while scaling down the total contrast.

Flexible Detail & Base Control : We introduce a more general form: $\mathbf{J}_i = s(\tilde{\mathbf{I}}_i - \tilde{\mathbf{q}}_i^*)^{\mathrm{Detail}} + t\tilde{\mathbf{q}}_i^{*\mathrm{Base}}$ The filter can be flexibly controlled by:

$$f_{\mathrm{Detail}}(x) = \frac{tx^\alpha+s\epsilon}{x^\alpha+\epsilon}, \qquad f_{\mathrm{Base}}(x) = t - \frac{tx^\alpha+s\epsilon}{x^\alpha+\epsilon}. \tag{10}$$

where α controls the kurtosis of the curve $f(x)$. The functions satisfy the condition $f_{\mathrm{Detail}} + f_{\mathrm{Base}} = t$. The starting point at $x = 0$ is determined by s, and the convergence point at $x = 1$ can be adjusted by t (see Fig. 1(b)). In Sec. 4, we show how to find good parameters.

Other Effect Control Functions. Similarly, the other functions like sigmoid and Gaussian functions. Furthermore multinomial and spline functions can be represented by f_{Detail} and its corresponding $f_{\mathrm{Base}} = \mathrm{const.} - f_{\mathrm{Detail}}$. In Fig. 4(a), to flexibly control the effect, the curve of a spline function specified by a user is used as a smoothing function.

3 Performance Improvement

3.1 Efficient Alternative Calculation of SVD by EVD

In Sec.2, we use the SVD to obtain matrices **U** and **D**. The calculation cost of the pixel-wise SVD is inherently a bottleneck of our filter. The guided-filter [5]

has a similar problem in its inverse calculation. But actually matrices $\mathbf{U}$ and $\mathbf{D}$ consist of $\mathbf{C}$'s eigen-pairs, that is, eigenvalues $\mathbf{D} = \mathrm{diag}([d_1, d_2, d_3])$ and their corresponding eigenvectors $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3]$. We can also obtain the eigen-pairs via the eigenvalue decomposition (EVD).

EVD by Power Iteration with Orthogonalization. As an alternative calculation, we use the power iteration algorithm with the Gran-Schmidt orthogonalization [10] in order to find the m-th largest eigenvalue d_m and its corresponding eigenvector $\mathbf{u}_m$. Although the accuracy of this traditional EVD method is a slightly worse than the recent analytical methods implemented on LAPACK [11], the algorithm converges after a few iterations and requires a little amount of calculation (as one can see from Eq.11). Moreover our algorithm and [11] have little perceptible difference in resulting images.

The original power iteration is given as follows and modified a little later:

$$\begin{array}{lll}
1.\ \text{update eigenvector:} & \tilde{\mathbf{u}}_m = \mathbf{C}\mathbf{u}_m^{(t)}{}_{m-1} \\
2.\ \text{orthogonalization:} & \hat{\mathbf{u}}_m = \tilde{\mathbf{u}}_m - \sum_{p=1}^{} (\tilde{\mathbf{u}}_m^T \mathbf{u}_p)\mathbf{u}_p \\
3.\ \text{update eigenvalue:} & d_m^{(t+1)} = \|\hat{\mathbf{u}}_m\|^{}_{p=1} \\
4.\ \text{normalization:} & \mathbf{u}_m^{(t+1)} = \hat{\mathbf{u}}_m / d_m^{(t+1)} \\
5.\ \text{convergence check:} & \text{break if } |\mathbf{u}_m^{(t)T}\mathbf{u}_m^{(t+1)} - 1| < \tau
\end{array} ,$$

where t is the number of iterations, and an initial eigenvector is set by random values normalized to $\|\mathbf{u}_m^{(0)}\| = 1$. As the angle tolerance for the convergence condition, we set $\tau = 0.0001$ in the experiment.

To further reduce the number of multiplications, we modify the orthogonalization step as follows:

$$\hat{\mathbf{u}}_m = \mathbf{I_d}\tilde{\mathbf{u}}_m - (\sum_{p=1}^{m-1} \mathbf{u}_p\mathbf{u}_p^T)\tilde{\mathbf{u}}_m^T = \left[\{\mathbf{I_d} - (\sum_{p=1}^{m-1} \mathbf{u}_p\mathbf{u}_p^T)\}\mathbf{C}\right] \mathbf{u}_m^{(t)}, \tag{11}$$

where the part of $[\cdot]$ in Eq.11 can be calculated in advance [1]. Note that the 3rd eigenvectors are calculated by the cross product of the 1st and 2nd eigenvectors in our method.

Propagation of Initial Eigenvectors. Since color pixels have correlation among its neighbors, their eigen-pairs also tend to be similar in its neighborhoods. Thereby the number of iterations at a current pixel can be reduced by setting a well converged neighboring eigenvector to the initial eigenvector of a current pixel.

To propagate an eigenvector at a pixel to the whole image, multi-scale methods such as the multi-grid [12] may be applied. But in practice, the cost of calculation and memory reference for generating pyramidical images were larger than

[1] Actually the original calculation $(\tilde{\mathbf{u}}_m^T\mathbf{u}_p)\mathbf{u}_p$ via the dot product is often used for matrices with huge dimension to avoid the tensor product $\mathbf{u}_p\mathbf{u}_p^T$, which results in saving memory. However in our case, the dimension is only 3. Thus we can ignore the memory saving.

the cost of the power iterations. Therefore instead of the multi-scale approach, we simply use the image at the finest resolution and propagate an obtained eigenvector at a current pixel i to the next adjacent pixel $i+1$ as the initial eigenvector:

$$\mathbf{u}_{m,i+1}^{(0)} = \mathbf{u}_{m,i}^{(\text{converged})} \tag{12}$$

4 Experimental Results

The results of our method are shown in from Fig. 2 to Fig. 6. One can see that the obtained images have natural halo-free appearances. Here we mainly mention the efficiency of the *cc*-functions.

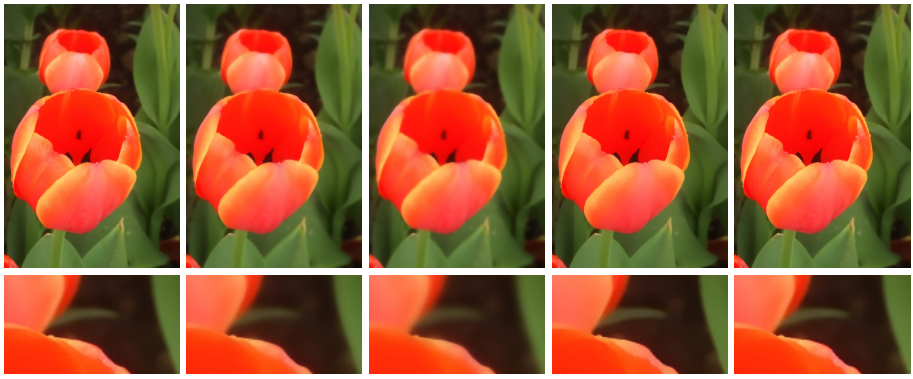
4.1 Comparison with Conventional Methods

Fig. 2 and Fig. 3 show the comparison with our local covariance (LC) filter (a-c) including the guided-filter [5] (b), the WLS filter [4] (d), and the domain-transform (DT) filter [3] (e). Form the results, our LC filter seems to have the equivalent performance to the WLS and DT filters. Compared to the guided-filter (b), our filter is able to further control the edge-preserving degree by manipulating the shape of function. The other recognizable difference is discolorations around edges in WLS and DT filters. In the LC filter, the colors are varied along the principal axis of the local color distribution in the RGB space, and the color difference $\|\mathbf{I}-\mathbf{q}\|$ along the principal axis tends to become smaller than WLS and DT filters.

4.2 Effectiveness of Characteristic Controle Functions

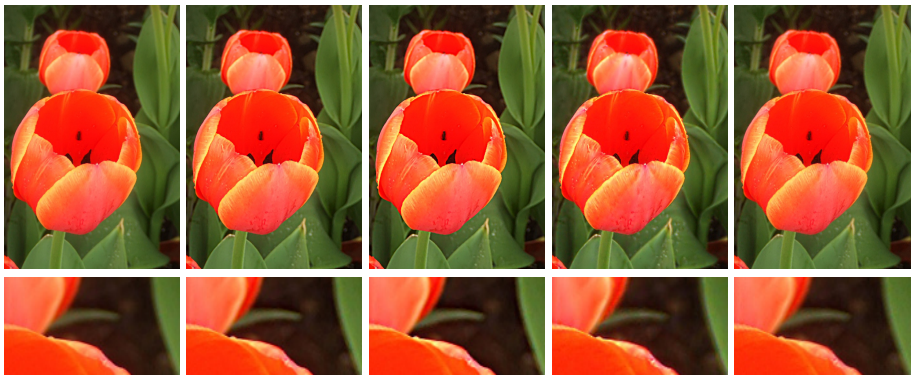
Graphical Manipuration. One of the advantages in our method is that filtering effect can be easily visualized and graphically manipulated. The example of our graphical image manipulation is shown in Fig. 4, where (a-c) show the histogram of the original diagonal elements of $\mathbf{D}_k$ (a), and its histograms (b,c) controlled by $f_{\text{Detail}}, f_{\text{Base}}$ respectively, where f_{Detail} is a spline function specified with control points manually. (d,e) are the generated scale and offset images corresponding to $\{(d)\}\mathbf{I}_i + \{(e)\}$ in Eq. 5, and (f) jointly shows the original and the resulting images. Users can manually control the curve of f_{Detail} of (a) while seeing the result (f), where f_{Base} is automatically determined as $1-f_{\text{Detail}}$.

Parameter Selection by Histogram Balance. Instead of selecting parameters by time-consuming trial and error, the histograms can be used as a criterion of the parameter selection. Fig. 5 shows the results of the detail manipulation and their histograms below (same order as Eq.4(a-c)). In this figure, (a) has the almost same appearance with the original image and has a biased histogram. In contrast, (b-c) have extended histograms, especially (b) has non-saturated histogram and a photographic look.



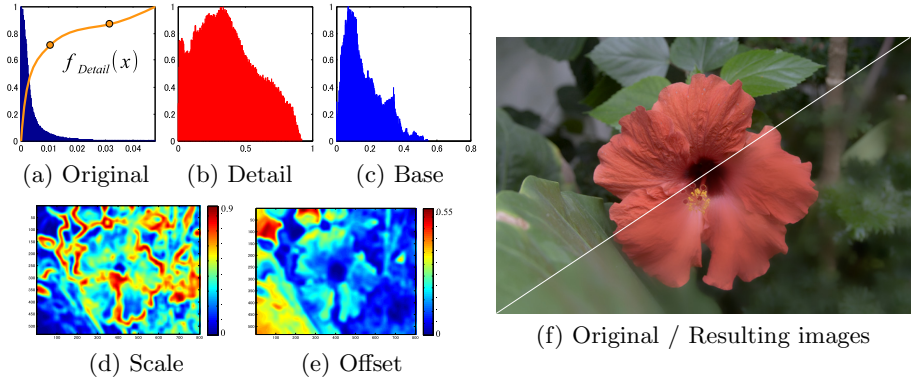
(a) LC $\alpha = 0.5$ (b) LC $\alpha = 1.0$ (c) LC $\alpha = 1.5$ (d) WLS (e) DT

Fig. 2. Resulting images of smoothing. (a-c) Local Covariance filter including the guided-filter (b), (d) WLS filter [4], (e) domain transform recursive filter [3]. The setting of the LC filter is as follows: radius $r = 5$, $f_{\mathrm{Detail}}(x) = \frac{x^\alpha}{x^\alpha + \epsilon}$ where $\epsilon = 0.025$, α is shown below the images. Parameters of other filters are as follows: WLS filter are $\alpha = 1.2$, $\lambda = 0.5$, DT filter are $\sigma_s = 3$, $\sigma_r = 0.2$.



(a) LC $\alpha = 0.5$ (b) LC $\alpha = 1.0$ (c) LC $\alpha = 1.5$ (d) WLS (e) DT

Fig. 3. Resulting images of enhancing. (a-c) Local Covariance filter including the guided-filter (b), (d) WLS filter [4], (e) domain transform recursive filter [3]. The parameter settings are the same as Fig. 2. Each image is enhanced by $3(\mathbf{I}-\mathbf{q})^{\mathrm{Detail}}+\mathbf{q}^{\mathrm{Base}}$.



(a) Original (b) Detail (c) Base

(d) Scale (e) Offset

(f) Original / Resulting images

Fig. 4. Graphical manipulation of the local covariance of color distribution (e.g. smoothing). The filter setting: radius $r = 10$, and $f_{\mathrm{Detail}}(x)$ is a spline function specified manually.

4.3 Saliency Based Flexible Detail Manipulation

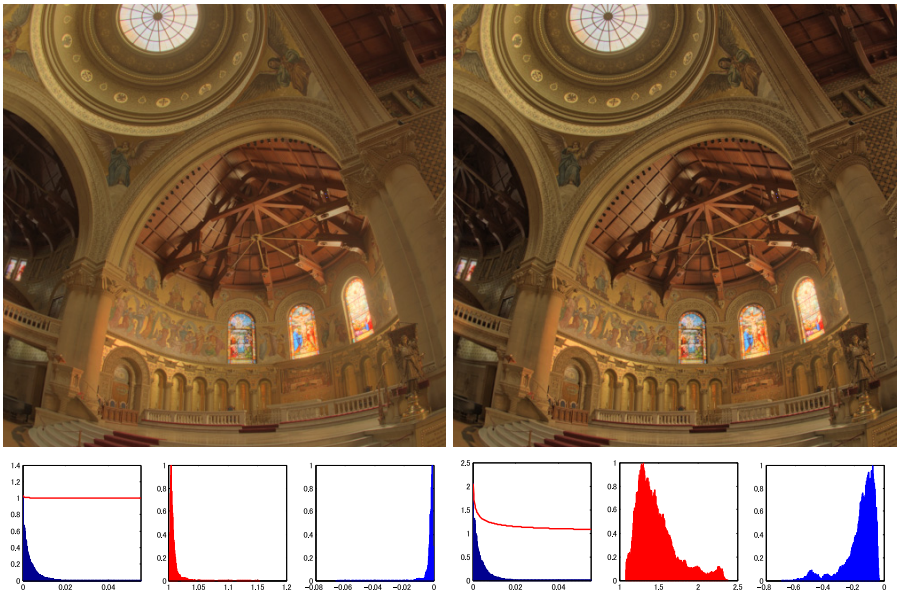
By changing the shape of the *cc*-functions at each pixel smoothly, one can simultaneously realize partial smoothing and partial enhancing (hereafter we call it mixture filtering) as shown in Fig. 6. In this figure, the original images (a), the results of the mixture filtering (b) and its saliency maps as a parameter map of *cc*-function (c) are shown. Saliency maps are generated by [13], and the pixel values (normalized in the range of $[0, 2]$) are used as a parameter s of Eq.10. In a case $s \to 0$, the *cc*-functions become Eq.7 and behave as a smoothing filter. In contrast $s \to 2$, the *cc*-functions become Eq.8 and behave as an enhancing filter. Thus regions with low saliency in background are smoothed while main subjects with high saliency are enhanced. This method is effective to generate aesthetic pictures that matches the human perception.

4.4 Processing Speed

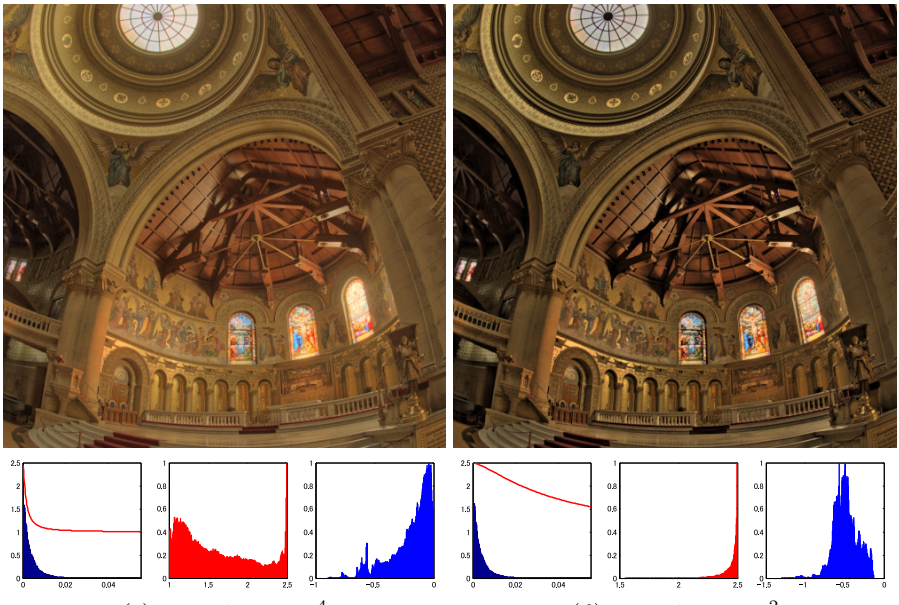
Our method was implemented in MATLAB (mex C++ is partially used) and tested on Core i7 2.67GHz PC (single thread). The comparison in the execution

Table 1. Execution time of the SVD and EVD (sec/Mpix *RGB*colors) with smoothing functions in Eq. 7 that correspond to the guided-filter [5].

Pre-processing	before SVD	0.18
matrix inverse [5]	by LAPACK	3.13
SVD	by LAPACK	5.89
EVD	by power iteration (Sec. 3.1)	0.75
EVD	with vector propagation (Sec. 3.1)	0.54
Post-processing	after SVD	0.16



(a) $\alpha = 0.6, \epsilon = 10^{-4}$ (b) $\alpha = 0.6, \epsilon = 10^{-2}$



(c) $\alpha = 1.4, \epsilon = 10^{-4}$ (d) $\alpha = 1.4, \epsilon = 10^{-2}$

Fig. 5. Correlation between appearances and the balacance of histograms (e.g. detail manipuration). The filter setting: radius $r = 10$, $f_{\text{Detail}}(x) = \frac{x^\alpha + s\epsilon}{x^\alpha + \epsilon}$ where $s = 2.5$, the other parameters are shown at the bottom of each image.

(a) (b) (c)

Fig. 6. Saliency adaptive local covariance filtering. (a) Original images, (b) results of mixture filtering, (c) saliency maps [13] as a parameter map of *cc*-functions. The filter setting is as follows: radius $r = 10$, $f_{\mathrm{Detail}}(x) = \frac{x^{\alpha}+s\epsilon}{x^{\alpha}+\epsilon}$ where $s \in [0, 2]$ is a normalized pixel value of the saliency map (c), other parameters are set as $\alpha = 1.2$ and $\epsilon = 0.1$.

time of some SVD calculation methods is shown in Table. 1. In our method, all summations ($\frac{1}{\eta}\sum\cdot$) are calculated by box-filtering using the integral images [7] in the same way as the guided-filter [5]. Therefore its complexity does not depend on the filter size. The SVD section is accelerated five times compared to the conventional methods. The average PSNR between the SVD and EVD is 80 (dB). Thus their resulting images are almost the same.

5 Conclusion

In this paper, we show a novel general edge-aware filtering which controls the local covariance of images. Once the SVD is calculated at each pixel, users can control the filter characteristic graphically while seeing the result in real-time. As the future work, we will develop the $f_{\mathrm{Detail}}, f_{\mathrm{Base}}$ of multi-scale processing for tone-mapping to create natural images based on human perception.

References

1. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: Proc. of IEEE ICCV, pp. 836–846 (1998)
2. Durand, F., Dorsey, J.: Fast bilateral filtering for the display of high-dynamic-range images. ACM TOG (Proc. of SIGGRAPH) 21, 257–266 (2002)
3. Gastal, E.S.L., Oliveira, M.M.: Domain transform for edge-aware image and video processing. ACM TOG (Proc. of SIGGRAPH) 30, 69:1–69:12 (2011)
4. Farbman, Z., Fattal, R., Lischinski, D., Szeliski, R.: Edge-preserving decompositions for multi-scale tone and detail manipulation. ACM TOG (Proc. of SIGGRAPH) 27, 67:1–67:10 (2008)
5. He, K., Sun, J., Tang, X.: Guided Image Filtering. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 1–14. Springer, Heidelberg (2010)
6. Levin, A., Lischinski, D., Weiss, Y.: A closed-form solution to natural image matting. IEEE Trans. on PAMI 30, 228–242 (2008)
7. Crow, F.C.: Summed-area tables for texture mapping. In: Proc. of the 11th Annual Conf. on Comp. Graph. and Interactive Techniques (SIGGRAPH), pp. 207–212 (1984)
8. Paris, S., Hasinoff, S.W., Kautz, J.: Local laplacian filters: edge-aware image processing with a laplacian pyramid. ACM TOG (Proc. of SIGGRAPH) 30, 68:1–68:12 (2011)
9. Shirai, K., Ikehara, M., Okamoto, M.: Noiseless no-flash photo creation by color transform of flash image. In: Proc of IEEE ICIP, pp. 3437–3440 (2011)
10. Sharma, A., Paliwal, K.K.: Fast principal component analysis using fixed-point algorithm. In: Pattern Recognition Letters, pp. 1151–1155 (2007)
11. Netlib Repository: Linear algebra package (LAPACK), `www.netlib.org`
12. Briggs, W.L., Henson, V.E., McCormick, S.F.: A multigrid tutorial, 2nd edn. SIAM (2000)
13. Harel, J., Koch, C., Perona, P.: Graph-based visual saliency. In: Proc. of Advances in Neural Information Proc. Systems, pp. 545–552 (2007)

A New Projection Space for Separation of Specular-Diffuse Reflection Components in Color Images

Jianwei Yang[1], Zhaowei Cai[1], Longyin Wen[1], Zhen Lei[1], Guodong Guo[3], and Stan Z. Li[1,2,⋆]

[1] CBSR & NLPR, Institute of Automation, Chinese Academy of Sciences, China
[2] China Research and Development Center for Internet of Thing
[3] Dept. of Computer Science and Electrical Engineering, West Virginia University
{jwyang,zwcai,lywen,zlei,szli}@cbsr.ia.ac.cn, gdguo@cs.wisc.edu

Abstract. In this paper, we propose a new reflectance separation model to separate the diffuse and specular reflection components. The model is based on a two-dimensional space called Ch-CV space, which is spanned by maximum chromaticity (Ch) and the coefficient of variation (CV) of RGB color. The space exhibits a more direct correspondence to diffuse and specular reflection components than the RGB color space, as well as the HSI color space. Under the whitened illumination, the surface points with the same diffuse chromaticity have the same slope in Ch-CV space. Based on these properties, we propose a slope-based region growing method to implement an image segmentation in the specular regions, and to separate the reflection components for each segmented region. The comparison experiments with several state-of-the-art algorithms show its superior capability to separate the specular and diffuse reflection components.

1 Introduction

In recent years, separating the diffuse and specular reflection components in color images has become an important research topic. Lots of highlight detection and removal methods have been proposed. In these method, the dichromatic reflectance model [1] has been widely utilized with the assumption that the reflected light can be separated into specular and diffuse reflections, respectively.

In terms of the quantity of input data, the reflection component separation algorithms can be categorized into two groups [2]: multi-image based and single-image based methods. In the early multi-image based methods, polarization method was introduced in conjunction with color information [3–5]. Later, Sato and Ikeuchi [6] introduced a temporal-color space to analyze the diffuse and specular reflections based on colors and image intensity. Lin and Shum [7, 8] changed the light source direction to produce two photometric images and used

⋆ Corresponding author.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 418–429, 2013.

linear basis functions to separate the specular components. Based on the observation that the shift of the highlights in sequential images is generated by the shift of the light source, Feris et al. [9] proposed a multi-flash method. In addition, [10] and [11] also use multiple images to separate specular and diffuse reflection components. However, multiple images are not always available under many circumstances in practice.

Many reflection separation methods which rely on a single image have been proposed. Klinker [12] extended the dichromatic model by introducing a T-shaped color space. In this model, the diffuse component of highlight is estimated through projecting the highlight limb to the diffuse one. However, this T-shaped distribution will degrade much in image areas with uniform hue but various saturation. Alternatively, Bajscy et al. [13] proposed a specified three-dimensional space called S space. However, to construct S_0 axis of S space, they must use a spectro-photometer to measure the scene radiance, which is not practical in many cases. Different from the previous works, Mallick et al. proposed a data-driven color space called SUV space [14]. In this space, the specular and diffuse components are separated into S channel and UV channel, then the highlights are removed by iteratively eroding the specular channel using either a single image or video sequences [15]. Actually, when the input image is under whitened illumination, all the analysis in the SUV space can be turned to RGB space. Therefore, it is also vulnerable to a multi-colored or textured images.

Different from the previous approaches which are based on three-dimensional space, Tan and Ikeuchi [16] proposed a novel mechanism based on a two-dimensional Maximum Chromaticity-Intensity space. In their method, the diffuse component is obtained by locally iterative calculation based on a specular-free (SF) image for each pixel. Unfortunately, though no prior image segmentation is needed, this method leads to a much higher computational complexity and much color distortions in image, especially at edge areas. Considering the computational complexity, Yang et al. [17] exploited a fast bilateral filtering technique. This method estimates the maximum diffuse chromaticity by directly applying low-pass filter. However, it results in much more color distortions at edges, as well as inside the region of uniform color. On the other hand, Yoon et al. [18] also proposed an iterative framework based on the comparison of local ratios. A modified specular-free (MSF) image was introduced, the reflection components were achieved by comparing local ratios between input and MSF images, followed by making those ratios equal in an iterative framework. The MSF image was also exploited in the work of Shen et al. [19, 20]. Unfortunately, the type of methods based on MSF image may wrongly estimate the reflection components due to color discontinuities in surface edge regions.

In this paper, we exploit a two-dimensional space called Ch-CV space to separate the reflection components in a single image, the space which is spanned by the maximum chromaticity and the coefficient of variation of color intensity. There are three major properties about the proposed space: i) under the whitened illumination, the Ch-CV space provides a linear description of the specular and diffuse reflections; ii) the polar coordinate values exhibit a more direct

relationship to reflection components; iii) there is a one-to-one correspondence between HSI color values and the polar coordinate values in Ch-CV space: the surface with identical hue and saturation also has identical characteristics in Ch-CV space.

As mentioned before, Tan et al. [16] also proposed a two-dimensional space called Maximum Chromaticity-Intensity space so that the separation can be described as a closed-from. However, the non-linearity of their space leads to the necessariness of an iterative algorithm to removal the highlights in an image. In contrast, the Ch-CV space provide a linear description of the specular and diffuse reflections, which can avoid the iterative estimation process [16, 18]. Based on its properties, we propose a region growing method to segment specular regions, and then obtain the maximum diffuse chromaticity for each segmented region. In this way, the interferences among neighboring surfaces of various color in the previous work are avoided in our method. As a result, the separation results are more reliable and accurate than those in previous works.

2 Surface Reflection Model

Based on the dichromatic reflection model [1], the color intensity of pixels in an image can be computed by the integration over the light spectrum as follows:

$$I_c(x) = \omega_d(x) \int \tau_c(\lambda) S_d(\lambda, x) E(\lambda) d\lambda + \omega_s(x) \int \tau_c(\lambda) S_s(\lambda, x) E(\lambda) d\lambda \qquad (1)$$

where $S_d(\lambda, x)$ and $S_s(\lambda, x)$ are the spectral distribution function of diffuse reflection and specular reflection, respectively; $E(\lambda)$ is the spectral power distribution of illumination light (assume there is a single light source); $\tau_c(\lambda)$ is the transmittance function of the camera sensor, and the subscript $c \in \{r, g, b\}$, representing three color channels: red, green and blue; $\omega_d(x)$ and $\omega_s(x)$ are the geometric scale factors of diffuse reflection and specular reflection, respectively, which merely depend on the geometry of a surface point.

We define the diffuse chromaticity $\boldsymbol{\Lambda} = \{\Lambda_r, \Lambda_g, \Lambda_b\}$ and specular chromaticity $\boldsymbol{\Gamma} = \{\Gamma_r, \Gamma_g, \Gamma_b\}$ as those in [16]. For each channel, $\Lambda_c(x) = J_c^d / \sum_c J_c^d$, and $\Gamma_c = J_c^s / \sum_c J_c^s$. Here, $J_c^d = \int \tau_c(\lambda) S_d(\lambda, x) E(\lambda) d\lambda$ and $J_c^s = \int \tau_c(\lambda) S_s(\lambda, x) E(\lambda) d\lambda$. Then the color intensity of pixels for each channel $c \in \{r, g, b\}$ becomes:

$$I_c(x) = m_d(x) \Lambda_c(x) + m_s(x) \Gamma_c \qquad (2)$$

As explained in [16], both the sum of diffuse chromaticity vector $\boldsymbol{\Lambda}$ and that of specular chromaticity vector $\boldsymbol{\Gamma}$ are equal to 1, that is, $\sum_c \Lambda_c = \sum_c \Gamma_c = 1$. As a result, the sum of color intensity will be $\sum_c I_c(x) = m_d(x) + m_s(x)$.

3 The Proposed Reflection Separation Model

3.1 Illumination Chromaticity Normalization

In the real world, most illumination light are not pure white because of the non-uniform spectral distribution of light source and different transmittance function

of camera sensors. In this paper, we utilize the normalization approach introduced in [21] to obtain a normalized specular-diffuse color image, which is derived by dividing the estimated illumination chromaticity in both sides of Eq. (2):

$$I'_c(x) = m'_d(x)\Lambda'_c(x) + m'_s(x)\Gamma'_c = m'_d(x)\Lambda'_c(x) + \frac{m'_s(x)}{3} \tag{3}$$

where $m'_d(x) = m_d(x)\sum_c \frac{\Lambda_c(x)}{\Gamma_c^e}$, $m'_s(x) = 3m_s(x)$. Obviously, the sum of $\boldsymbol{\Lambda}'$ is still equal to 1, and the same to $\boldsymbol{\Gamma}' = \{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\}$. Upon the completion of normalization, the reflection separation can be conducted to achieve the normalized reflection components, followed by transferring the components to the previous un-normalized components [16].

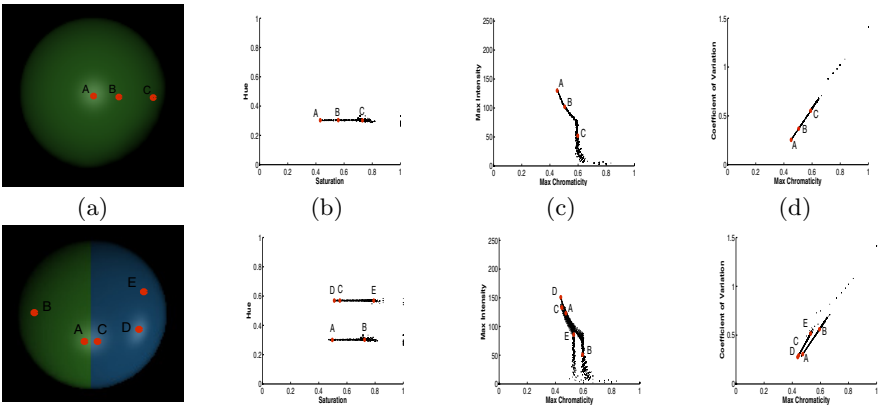
3.2 The Illustration of Ch-CV Space

In this part, by comparing the Ch-CV space with the Maximum Chromaticity-Intensity space [16] and HSI color space [22], we demonstrate the claimed properties of Ch-CV space. At first, we explain the linearity in Ch-CV space.

Given a normalized input image, the maximum chromaticity is defined as:

$$\sigma(x) = \frac{\max(I'_r(x), I'_g(x), I'_b(x))}{\sum_c I'_c(x)} \tag{4}$$

The coefficient of variation (CV), which is defined as the ratio of the standard deviation to the mean of color intensity $\boldsymbol{I} = \{I'_r, I'_g, I'_b\}$ in a normalized image, is with the following formulation:



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 1. Projections of single-colored image and bi-colored image into the three spaces. (a) and (e): Synthetic *green ball* and synthetic *blue-green ball*. (b) and (f): Projection of images into Hue-Saturation space. (c) and (g): Projection of images into Maximum Chromaticity-Intensity space. (d) and (h): Projection of images into Ch-CV space.

$$CV(x) = \frac{\sqrt{\frac{1}{3}\sum_c (I'_c(x) - I'_m(x))^2}}{I'_m(x)} \tag{5}$$

where $I'_m(x)$ is the mean of RGB color. According to Eqs. (3), (4) and (5), all of the specular only points locate exactly at $(1/3, 0)$ in the Ch-CV space because their color intensity $I'(x) = \{\frac{m'_s(x)}{3}, \frac{m'_s(x)}{3}, \frac{m'_s(x)}{3}\}, m'_s(x) \neq 0$. We regard the point $(1/3, 0)$ as the origin in the following analysis.

Substituting Eq. (3) into Eq. (5), the CV of surface point x has the following formulation:

$$CV(x) = \frac{\sqrt{3\sum_c I'^2_c(x) - (\sum_c I'_c(x))^2}}{m'_d(x)(\max_c \Lambda'_c - \frac{1}{3})}(\sigma(x) - \frac{1}{3}) \tag{6}$$

In Eq. (6), if the fraction at the right side is a constant, then the CV will be linear with the maximum chromaticity σ. In other word, the distribution of transformed RGB image in the Ch-CV space assembles a sector which consists of a cluster of straight lines with different slopes. In addition, these lines will exactly intersect at the specular only point $(1/3, 0)$ if ruling out the absolute black points. In the following analysis, the absolute black surface points are filtered out.

According to Eq. (6), for each surface point x in a normalized image, its slope is described as:

$$k(x) = \frac{\sqrt{3\sum_c I'^2_c(x) - (\sum_c I'_c(x))^2}}{m'_d(x)(\max_c \Lambda'_c - \frac{1}{3})} \tag{7}$$

Furthermore, based on Eq. (3), the slope of surface point x can be simplified to be:

$$k(x) = \frac{\sqrt{3\sum_c \Lambda'^2_c(x) - 1}}{(\max_c \Lambda'_c(x) - \frac{1}{3})} \tag{8}$$

Eq. (8) explains that the slopes of surface points in the Ch-CV space are merely determined by their diffuse chromaticities; therefore, the surface points that have the same diffuse chromaticity will have identical slope in the Ch-CV space. Meanwhile, the range of slope is exactly $[3\sqrt{2}/2, 3\sqrt{2}]$, where $3\sqrt{2}/2$ and $3\sqrt{2}$ correspond to the case that one component of $\boldsymbol{\Lambda'}$ is equal to 1 and the case that two components of $\boldsymbol{\Lambda'}$ are equal to 0.5, respectively. Clearly, the permutation of the values of $\boldsymbol{\Lambda'}$ among RGB channels will result in the same slope. To avoid the confusions among three channels, we split the original space into six sub-spaces according to the relationship of RGB value. From the last column in Fig. 1, we can find the projection of the normalized *green ball* in the Ch-CV space composes a single straight line, and there are two principal lines for the normalized *blue-green ball*. It should be noted that we present only one Ch-CV sub-space rather than all the six ones since there is no overlapped lines in the space.

Another property of Ch-CV space is that there exist one-to-one correspondence between the HSI color values and the polar coordinate values in Ch-CV space. The surface points with identical hues also have identical slopes in Ch-CV space, and the saturation corresponds to the horizonal coordinate as well as polar radius .

Based on the Eq. (3), the hue of surface point in normalized image is reformulated as:

$$H(x) = cos^{-1}[\frac{3\sqrt{2}}{2}\frac{\Lambda'_r(x) - \frac{1}{3}}{\sqrt{3\sum_c \Lambda'^2_c(x) - 1}}] \tag{9}$$

Comparing Eq. (8) with Eq. (9), we can find the slope in Ch-CV space corresponds to the hue component of HSI color. Therefore, the surface points with the same diffuse chromaticity $\boldsymbol{\Lambda'}$ have not only the same slope in Ch-CV space, but also the same hue in HSI space. Comparing the second and forth columns in Fig. 1, we can see that the surface points with identical hue construct a slant straight line in the Ch-CV space.

Though the level of specular component is irrelevant to the slope in the Ch-CV space (or hue in HSI space), it does decide the surface points' location in a single line. By constituting Eq. (3) into Eq. (4), the maximum chromaticity for each surface point can be written as:

$$\sigma(x) = \frac{m'_d(x)\max_c \Lambda'_c(x) + \frac{1}{3}m'_s(x)}{m'_d(x) + m'_s(x)} \tag{10}$$

Clearly, $\sigma(x)$ is equal to $1/3$ when $m'_d(x) = 0$, and it is equal to $\max_c \Lambda'_c(x)$ when $m'_s(x) = 0$. In other word, given a color surface with a certain $\boldsymbol{\Lambda'}$, $\max_c \Lambda'_c = \sigma$ for the diffuse only points. More generally, in a homogeneous color surface, the larger the specular component is, the smaller σ is. Actually, the distribution of a homogeneous color surface in the Ch-CV space is a segmented line, which starts from $(1/3, 0)$ and ends at the point $(max_c\Lambda'_c, \sqrt{3\sum_c \Lambda'_c - 1})$. Consequently, as for a uniform color image, there is only one segmented line with a certain slope, whereas for a multi-colored image, there may exist several overlapped segmented lines with the same slope in the Ch-CV space, and each of them corresponds to an unique diffuse chromaticity $\boldsymbol{\Lambda'}$, in spite of their same slopes.

The above analysis indicates that the maximum chromaticity σ also represents a characteristic of surface color, which is analogous to the saturation value in HSI color space. Specifically, the suturation is formulated as follows:

$$S = 1 - (\frac{3}{I'_r + I'_g + I'_b})\min_c I'_c \tag{11}$$

Substituting I'_c in Eq. (3) into Eq. (11), we have:

$$S(x) = \frac{m'_d(x)}{m'_d(x) + m'_s(x)}(1 - 3\min_c \Lambda'_c(x)) \tag{12}$$

Meanwhile, the maximum chromaticity for point x can be re-formulated as:

$$\sigma(x) = \frac{1}{3} + \frac{m'_d(x)}{m'_d(x) + m'_s(x)}(\max_c \Lambda'_c(x) - \frac{1}{3}) \tag{13}$$

According to the last two equations, given a group of surface points with identical diffuse chromaticity $\boldsymbol{\Lambda'}$, both S and σ are maximized by the diffuse only points ($m'_s = 0$), and their values are $(1 - 3\min_c \Lambda'_c)$ and $\max_c \Lambda'_c$, respectively. Moreover, by combining Eq. (8) and the normalization condition of $\boldsymbol{\Lambda'}$, we will

obtain the unique solution of $\min_c \Lambda'_c$ if given the value of $\max_c \Lambda'$, and versa vice. Therefore, the surface points with the same hue and saturation will also have the same slope and maximum chromaticity in Ch-CV space. This relationship also holds between saturation and the polar radius in Ch-CV space. The labeled points in Fig. 1 proves our numerical analysis.

At this point, we have proved the claimed properties of the Ch-CV space. Though similar to HSI color space, there is a crucial difference between such two spaces. In HSI space, m'_d, a vital intermediate factor for reflection separation, cannot be derived from the hue formula given the $\min_c \Lambda'_c$ in Eq. (12), whereas it is computable in Eq. (7) if the value of $\max_c \Lambda'_c$ is obtained from Eq. (13). Another property of the Ch-CV space is that we can achieve maximum diffuse chromaticity and $m'_d(x)$ directly by making use of the linearity in Ch-CV space. Therefore, as for the reflection separation task based on dichromatic reflectance model, the Ch-CV space is superior than the aforementioned spaces, including S space, Maximum Chromaticity-Intensity space and HSV color space.

4 Specular-Diffuse Reflection Components Separation

This section will focus on how to separate specular and diffuse reflection components based on Ch-CV space. According to Eq. (3), given a normalized image, to separate the reflection components means to decompose the color intensity into two partitions, $m'_d(x)\Lambda'_c(x)$ and $\frac{1}{3}m'_s(x)$ for each color channel $c \in \{r, g, b\}$ in specular regions.

Before the separation, we choose the specularity detection method presented in [20] to determine the specular surface points in an image. Similarly, we dilate the original detected specular region into a larger one containing both specular and diffuse surface points in general case, and we call such regions diffuse-specular connected regions. To separate the reflection components in the specular regions, we first transform the normalized image into the Ch-CV space. According to Eq. (7), $m'_d(x)$ can be written as:

$$m'_d(x) = \frac{\sqrt{3\sum_c I'^2_c(x) - (\sum_c I'_c(x))^2}}{k(x)(\max_c \Lambda'_c(x) - \frac{1}{3})} \tag{14}$$

Eq. (14) illustrates that $m'_d(x)$ can be derived for every surface point given its slope and maximum diffuse chromaticity among thee color channels. Therefore, the derivation of m'_d can be divided into two stages: a) obtain $k(x)$; b) decide the $\max_c \Lambda'_c(x)$ for the surface points.

In a normalized image, the slopes of specular surface points can be calculated simply by $CV(x)/(\sigma(x) - 1/3)$. After obtaining the slopes of specular surface points, we can determine their maximum diffuse chromaticity based on the proof that it is identical to the maximum chromaticity of the the diffuse only points with the same hue. However, there may be no diffuse only points for some surfaces in practice. In such case, we can regard the points with the maximum σ as the diffuse only ones, which is rational due to the fact that we care more about removing the highlights in the input image, rather than getting an exactly diffuse

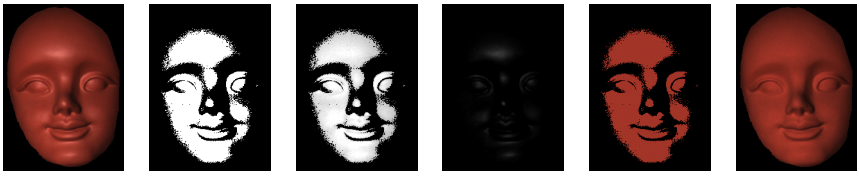
only image. Unfortunately, there may exist diffuse surfaces with the same hue yet different saturation in a multi-colored image. In such case, the surface points having the same slope in a Ch-CV sub-space may correspond to distinctive $\boldsymbol{\Lambda'}$. As a result, we should estimate the corresponding diffuse chromaticity for each surface with various saturation in each line optimally, rather than assign the single value of maximum σ to the $\max_c \Lambda'_c$ of all the surface points in a line.

In this paper, we use a 8-connected region growing method [23] to segment the specular regions so that surface points in each segmented region have similar hue. In our algorithm, we define an uniformity parameter η for the segmentation, and we set it to be a constant value 0.12, which is suitable for most images. After conducting the algorithm, we can obtain all the connected regions with similar hue values (or slopes) within the constraint of uniformity parameter. In each connected region, we assume the surface points share the same diffuse chromaticity $\boldsymbol{\Lambda'}$. Inspired by [20], the smoothness of diffuse reflection component are considered in our algorithm. In each diffuse-specular connected regions, the optimal $\max_c \Lambda'_c$ can be obtained under the condition that the difference between the mean RGB color intensity diffuse component in the specular region and that of surrounding diffuse region is minimized. Based on the Least Square Error (LSE) algorithm, we can obtain the optimal $\max_c \Lambda'_c$ for each specular surface point, then m'_d can be calculated by Eq. (14), and m'_s can be calculated for each surface point by using the equation $m'_s(x) = \sum_c I'_c(x) - m'_d(x)$. Afterward, the diffuse component of surface points can be derived according

$$m'_d(x)\Lambda'_c(x) = I'_c(x) - \frac{m'_s(x)}{3}. \tag{15}$$

Furthermore, the diffuse chromaticity $\boldsymbol{\Lambda'}$ can be obtained by dividing the diffuse component by m'_d. Fig. 2 shows the derived components of image *head*.

In practice, it is probably that all of the surface points in a connected region are specular ones. In this case, we estimate their specular reflection component directly. Based on the continuity of specularity in an image, we determine m'_s for each specular surface point by calculating the mean value of specular components in its surrounding 5×5 neighborhood.

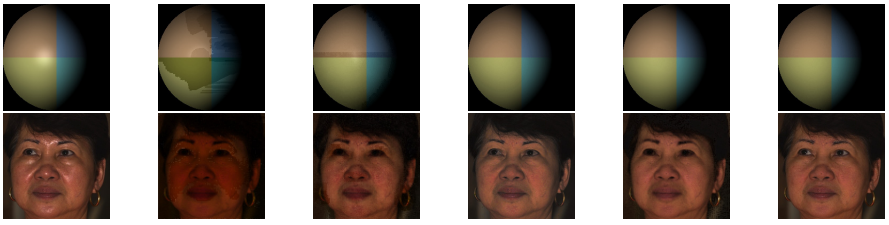


(a) (b) (c) (d) (e) (f)

Fig. 2. (a) Input image *head*. (b) Detected specular region. (c) m'_d, (d) m'_s and (e) $\boldsymbol{\Lambda'}$ in the specular region. (f) Diffuse component of *head*.

5 Experiments

The proposed method is compared with five previous methods [16–20]. In our experiments, 13 test images from previous works are used in our experiment, and two other images are synthesized to give a quantitative comparisons by using the PBRT v2.0 software [24]. All the experiments are performed on a PC with Intel Core i5, CPU 2.67 GHz, 2G RAM. Because of the limited space, we only present the better experimental results from Shen's two papers [19] and [20].



(a) (b) (c) (d) (e) (f)

Fig. 3. Comparison of diffuse components of *synth* and *lady*. Form left to right: (a) input images. (b) Diffuse components from [16]. (c) Diffuse components from [18]. (d) Diffuse components from [19, 20]. (e) Diffuse components from [17]. (f) Diffuse components from the proposed method.

In Fig. 3, we first use a synthetic image *synth* and a real-world image *lady* to evaluate the separation performance. Then two real-world images with multicolored and textured surfaces are used to compare the performance of five methods. Fig. 4 shows the diffuse components of real-world input image *fish* from five methods. As we can see, the diffuse component from the proposed method has the least color distortion, and the highlights in the image are removed effectively as well. In contrast, the method proposed by Tan et al. [16] fails to find the correct maximum diffuse chromaticity at color edges, leading to an obvious color distortions at edges. Moreover, because of its neighbor-based iteration algorithm, the original color distortion at edges spread inside. Though the method in [17] accelerates the separation process significantly by introducing inter-patch based algorithm rather than inter-pixel based, the result has much more color distortion than [16] because more color interferences between surfaces of different diffuse chromaticities are caused by inter-patch algorithm. The methods in [18–20] utilize a new specularity-invariant color image representation, MSF image. Yoon et al. [18] introduced an iteration scheme for neighboring pixels, which leads to much color distortions as well. To reduce the distortions, Shen et al. [19, 20] detect the highlight regions first, and then conduct local least-squares technique for each highlight region. However, though the results are acceptable in the specular regions of uniform color, the specular components are wrongly estimated in textured specular regions, such as the region around the eye of *fish* in Fig. 4d.

In comparison, the proposed method can obtain accurate and robust separate reflection components for images with both uniform color and highly textured

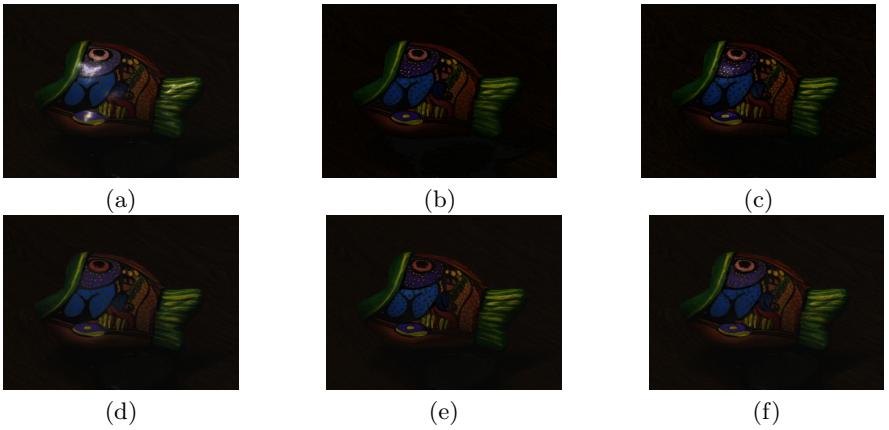


Fig. 4. (a) Input image *fish*. (b) Diffuse component from [16]. (c) Diffuse component from [18]. (d) Diffuse component from [19, 20]. (e) Diffuse component from [17]. (f) Diffuse component from the proposed method.

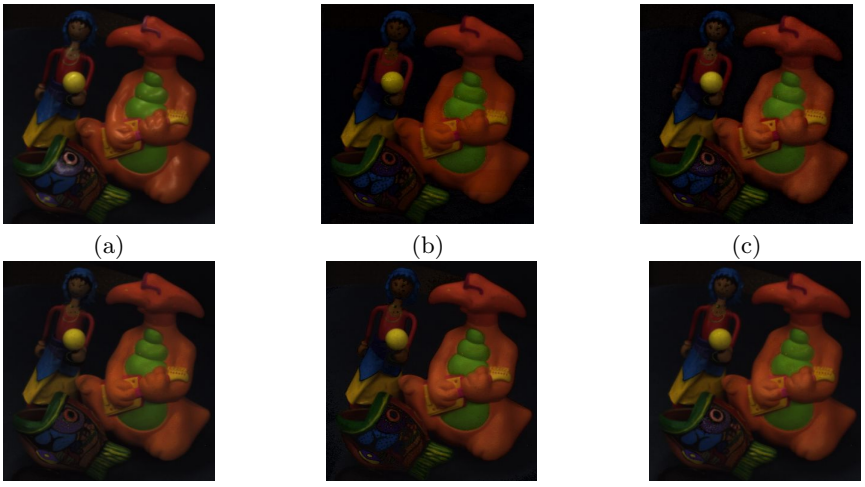


Fig. 5. (a) Input image *toys*. (b) Diffuse component from [16]. (c) Diffuse component from [18]. (d) Diffuse component from [19, 20]. (e) Diffuse component from [17]. (f) Diffuse component from the proposed method.

specular regions. By introducing a reflectance separation model based on the Ch-CV space and region growing algorithm, we can derive accurate diffuse chromaticity for each connected region, and rule out the interferences among regions of different diffuse chromaticities. The diffuse component of another real world image *toys* presented in Fig. 5 also supports to our claim.

We adopt the peak signal-to-noise ratio (PSNR) to evaluate the methods quantitatively. The experiments are conducted on our self-synthesized images. As shown in Fig. 6, the proposed method achieves higher PSNR values than the other methods. Moreover, we compare computational cost of different methods.

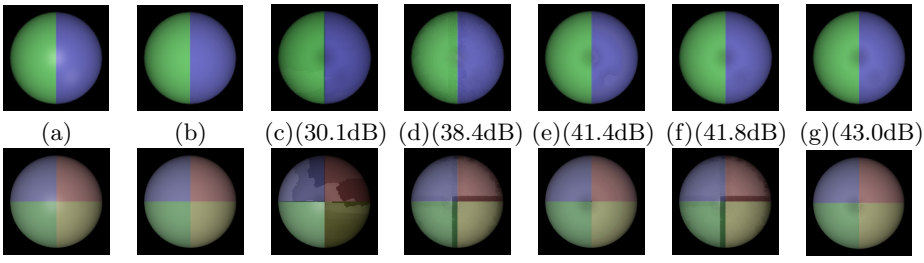


Fig. 6. Form left to right, the images are: input images, ground truth, diffuse components from [16], [18], [19,20], [17] and the proposed method. The corresponding PSNR value are reported below each image.

We compute the average time costs over all the test images: *ball-green*, *ball-blue-green*, *synth*, *head*, *pear*, *fish*, *toys*, *bear*, *bear2*, *red-pear*, *red-pear2*, *train*, *lady* and the other two self-synthesized images. The average time costs for all the methods are: 37.42s [16], 88.36s [18], 5.82s [19], 0.18s [20], 0.10s [17] and 6.25s (proposed). Because the proposed algorithm does not involve iterative process, the time cost is lower than the iterative methods [16, 18]. In our algorithm, the derivation of diffuse chromaticity should be conducted for each connected specular region, therefore, the time cost is comparable to [19], and higher than the methods in [17, 20].

6 Conclusion

In this paper, a new two-dimension space, called Ch-CV space is proposed. In the space, images are transformed to be a cluster of straight lines intersecting at a single point, leading to a fast and accurate derivation of the maximum diffuse chromaticity for all specular surfaces with various colors. Compared with the previous methods, the proposed one exploits a physical description of natural color, which facilitates the effectiveness of highlight removal in images, especially the multicolored and textured images. The further work will be focused on accelerating the reflection components separation process without additional color distortions.

Acknowledgement. This work was supported by the Chinese National Natural Science Foundation Project #61070146, #61105023, #61103156, #61105037, #61203267, National IoT R&D Project #2150510, Chinese Academy of Sciences Project No. KGZD-EW-102-2, European Union FP7 Project #257289 (TABULA RASA http://www.tabularasa-euproject.org), and AuthenMetric R&D Funds.

References

1. Shafer, S.: Using color to separate reflection components. Color Research and Applications 10, 210–218 (1985)
2. Artusi, A., Banterle, F., Chetverikov, D.: A survey of specularity removal methods. Comput. Graph. Forum 30, 2208–2230 (2011)

3. Wolff, L.B.: Polarization-based material classification from specular reflection. IEEE Trans. Pattern Anal. Mach. Intell. 12, 1059–1071 (1990)
4. Wolff, L.B., Boult, T.E.: Constraining object features using a polarization reflectance model. IEEE Trans. Pattern Anal. Mach. Intell. 13, 635–657 (1991)
5. Nayar, S.K., Fang, X.S., Boult, T.E.: Separation of reflection components using color and polarization. International Journal of Computer Vision 21, 163–186 (1997)
6. Sato, Y., Ikeuchi, K.: Temporal-color space analysis of reflection. JOSA 11, 2990–3002 (1994)
7. Lin, S., Shum, H.Y.: Separation of diffuse and specular reflection in color images. In: CVPR (1), 341–346 (2001)
8. Lin, S., Li, Y., Kang, S.B., Tong, X., Shum, H.-Y.: Diffuse-Specular Separation and Depth Recovery from Image Sequences. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part III. LNCS, vol. 2352, pp. 210–224. Springer, Heidelberg (2002)
9. Feris, R., Raskar, R., Tan, K.H., Turk, M.: Specular reflection reduction with multi-flash imaging. In: SIBGRAPI, pp. 316–321 (2004)
10. Umeyama, S., Godin, G.: Separation of diffuse and specular components of surface reflection by use of polarization and statistical analysis of images. IEEE Trans. Pattern Anal. Mach. Intell. 26, 639–647 (2004)
11. Chen, T., Goesele, M., Seidel, H.P.: Mesostructure from specularity. In: CVPR (2), pp. 1825–1832 (2006)
12. Klinker, G., Shafer, S.A., Kanade, T.: The measurement of highlights in color images. International Journal of Computer Vision 2, 7–32 (1988)
13. Bajcsy, R., Lee, S.W., Leonardis, A.: Detection of diffuse and specular interface reflections and inter-reflections by color image segmentation. International Journal of Computer Vision 17, 241–272 (1996)
14. Mallick, S.P., Zickler, T., Kriegman, D.J., Belhumeur, P.N.: Beyond lambert: Reconstructing specular surfaces using color. In: CVPR (2), pp. 619–626 (2005)
15. Mallick, S.P., Zickler, T., Belhumeur, P.N., Kriegman, D.J.: Specularity removal in images and videos: A pde approach. In: ECCV (1), pp. 550–563 (2006)
16. Tan, R.T., Ikeuchi, K.: Separating reflection components of textured surfaces using a single image. IEEE Trans. Pattern Anal. Mach. Intell. 27, 178–193 (2005)
17. Yang, Q., Wang, S., Ahuja, N.: Real-Time Specular Highlight Removal Using Bilateral Filtering. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 87–100. Springer, Heidelberg (2010)
18. Yoon, K.J., Choi, Y., Kweon, I.S.: Fast separation of reflection components using a specularity-invariant image representation. In: ICIP, pp. 973–976 (2006)
19. Shen, H.L., Zhang, H.G., Shao, S.J., Xin, J.H.: Chromaticity-based separation of reflection components in a single image. Pattern Recognition 41, 2461–2469 (2008)
20. Shen, H.L., Cai, Q.Y.: Simple and efficient method for specularity removal in an image. Applied Optics 48
21. Tan, R., Nishino, K., Ikeuchi, K.: Color constancy through inverse intensity chromaticity space. JOSA 21, 321–334 (2004)
22. Gonzalez, R.C., Woods, R.E.: Digital image processing. Addison-Wesley (1992)
23. Hojjatoleslami, S.A., Kittler, J.: Region growing: a new approach. IEEE Transactions on Image Processing 7, 1079–1084 (1998)
24. Pharr, M., Humphreys, G.: Physically Based Rendering: From Theory to Implementation, 2nd edn. Morgan Kaufmann (2010)

Hand Vein Recognition Based on Oriented Gradient Maps and Local Feature Matching

Di Huang[1], Yinhang Tang[1], Yiding Wang[2], Liming Chen[3], and Yunhong Wang[1]

[1] Laboratory of Intelligence Recognition and Image Processing, School of Computer Science and Engineering, Beihang University, Beijing 100191, China
[2] College of Info. Eng., North China University of Technology, Beijing, 100041, China
[3] Université de Lyon, CNRS, Ecole Centrale Lyon, LIRIS, Lyon, 69134, France

Abstract. The hand vein pattern as a biometric trait for identification has attracted increasing interests in recent years thanks to its properties of uniqueness, permanence, non-invasiveness as well as strong immunity against forgery. In this paper, we propose a novel approach for back of the hand vein recognition. It first makes use of Oriented Gradient Maps (OGMs) to represent the Near-Infrared (NIR) hand vein images, simultaneously highlighting the distinctiveness of vein patterns and texture of their surrounding corium, in contrast to the state-of-the-art studies that only focused on the segmented vein region. SIFT based local matching is then performed to associate the keypoints between corresponding OGM pairs of the same subject. The proposed approach was benchmarked on the NCUT database consisting of 2040 NIR hand vein images from 102 subjects. The experimental results clearly demonstrate the effectiveness of our approach.

1 Introduction

Driven mainly by an increasing requirement for security against terrorist activities, sophisticated crimes, as well as electronic frauds, biometric based solutions have witnessed an accelerated pace of growth in the global market of security over the last several decades. Recently, a new biometric, named hand veins, has emerged for the purpose of people identification.

Anatomically, the veins are the blood carrying vessels interweaved with muscles and bones, and the fundamental function of the vascular system is to supply oxygen to each part of the body. The spatial arrangement of the vascular network in human body is quite stable and unique, and vein patterns of individuals are different, even between identical twins [1]. In this work, we concentrate on the vein patterns of back of the hand since they are distinctly visible, easy to acquire, and efficient to process. As compared with other popular biometric traits, like face or fingerprint, hand vein patterns possess several main merits, in particular the following ones:

- Direct liveness test. As hand veins are imaged by using far or near infrared lighting to capture temperature differences between the flow of hot blood in the

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 430–444, 2013.

veins and surrounding skin, they can only be imaged on live body and the images taken on non-live bodies cannot capture their spatial vein arrangement;

- Safety. Blood vessel patterns are hardwired underneath the skin at birth; they are hence much harder for intruders to forge.

Vein pattern as biometric trait is relatively recent. It was not exploited until 1990 when MacGregor and Welford [2] came up with their system called vein check for people identification. Despite the vast vascular network in human body, hand veins are largely favored for their simplicity in processing and there exists an increasing amount of work in the last decade, using hand vein patterns of the palm part [3,4], back of the hand [5,6,7] or finger veins [8] (refer to the standard ISO/IEC 19794-9 for more detailed definitions of different hand vein patterns).

Most tasks in the literature followed the framework that first segments the region of interest and hand subcutaneous vascular network from hand vein images, and then extracts local geometric features for matching, such as the positions and angles of short straight vectors [9], endpoints and crossing points [10], dominant points [5], vein minutiae and knuckle shape [7]. All these methods demonstrate reasonable recognition accuracies on small databases ranging from 32 [5] to 100 subjects [1,7], however, when regarding the problem of back of the hand vein recognition, the above techniques suffer from limited local features because compared with the palm and finger part, the number of vein minutiae on the back of the hand is really few, leading to a deficiency in capturing differences of vein networks between subjects. Meanwhile, NIR imaging systems deliver vein patterns along with the surrounding texture of corium in the same image, but the texture information of corium is rarely used. Intuitively, if its details can be highlighted, there should be distinctive cues for improved performance.

In this paper, we propose a novel approach for back of the hand vein recognition. It adopts Oriented Gradient Maps (OGMs) originally proposed to describe 3D face models (i.e. range and texture images) under the term of Perceived Facial Images (PFIs) [11], to represent hand vein images, simultaneously highlighting the distinctiveness of the vascular pattern as well as the texture of its surrounding corium. Using these OGMs instead of the original raw hand vein images, SIFT-based local matching is then carried out to associate local features between the backs of two hands, and to account for slight geometrical transformations (e.g. rotations, translations) and possible lighting variations (NIR intensity changes) that can occur on hand vein images. The proposed method was benchmarked on Near Infrared (NIR) hand-dorsa vein images in NCUT, one of the largest hand vein databases so far known in the literature, consisting of 2040 right and left hand-dorsa vein images of 102 subjects. The achieved experimental result clearly demonstrates the effectiveness of the proposed approach.

2 NIR Vein Image Acquisition

Hand veins can be imaged either by Far-Infrared (FIR) or Near-Infrared (NIR) imaging techniques, thereby providing a manner of contactless and non-invasive data acquisition. Wang and Leedham [12] made a study in depth to compare

FIR and NIR imaging techniques for vein pattern biometrics, and concluded that FIR imaging techniques are sensitive to ambient conditions such as temperature, humidity and human body condition, and hence do not deliver a stable quality. Meanwhile, they pointed out that NIR imaging techniques produce good quality images that are tolerant to variations in environmental and body condition. In this work, back of the hand vein images were sensed by a NIR imaging system developed by Wang et al. [13,14].

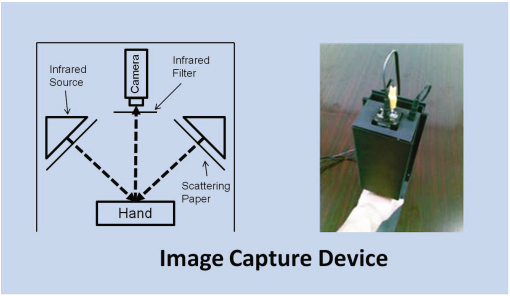


Fig. 1. The setup of the NIR imaging system

Fig.1 illustrates the system setup where an LED array lamp is utilized to shine infrared light onto the back of the hand. Because the incident infrared light can penetrate into the biological tissue with an approximate depth of 3mm and venous blood generally absorbs and scatters more incident infrared radiation than surrounding tissue, vein patterns can be imaged by a CCD camera attached an IR filter where vein appears darker. In order to avoid the major hand vein image registration issue, a handle is pre-mounted at the bottom of the device to position the hand. The hand vein images are hence roughly aligned, but still differ by slight translations and rotations. Fig.2 shows a back of the hand vein image captured with a resolution of 640 by 480.

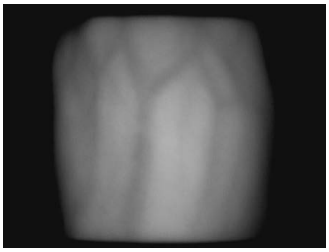


Fig. 2. A back of the hand vein image

Using this setup, a dataset of 2040 hand vein images was acquired under the natural lighting condition (indoor office environment) [6]. It was named as North

China University of Technology hand-dorsa vein database or NCUT dataset for short. In detail, 10 right and 10 left back of the hand vein images were captured from all 102 subjects, aged from 18 to 29, of which 50 were male while 52 were female. It makes the dataset one of the largest ones for hand vein biometrics. As the vein pattern is best defined when the skin on the back of the hand is taut, subjects were asked to clench their fists as acquiring vein patterns. There were no major illumination variations, but slight lighting changes still can occur since the vein images were acquired at different time.

As we can see from Fig.2, major hand vein patterns are captured and appear darker within the NIR image. Widths of these vein profiles vary in the range of 30 to 50 pixels. Even though the vein spatial arrangements are visible, they are not so distinguishable from the bio-tissue background. Moreover, local features, in terms of endpoints and crossing points, are limited and usually vary from 5 to 10, thus making local feature-based approach questionable for their ability of discriminative power. On the other hand, Fig. 2 delivers vein patterns along with the surrounding texture of corium, but the usefulness of texture information of corium is rarely investigated. Intuitively, the corium region contains critical cues for identification tasks as well, and if its details can be sufficiently highlighted, the performance of hand vein recognition could be improved.

3 Oriented Gradient Maps (OGMs)

In order to simultaneously increase the distinctiveness of the vein region as well as the texture of its surrounding corium, we propose a novel method to represent hand vein images, which makes use of Oriented Gradient Maps (OGM), originally applied to describe texture and range information in 3D face recognition [11].

The objective of OGM is to provide a visual description simulating the human visual perception and such a concept was inspired by the study of Edelman et al. [15], who proposed a representation of complex neurons in the primary visual cortex. These complex neurons respond to a gradient at a particular orientation and spatial frequency, but the location of the gradient is allowed to shift over a small receptive field rather than being precisely localized.

3.1 Representation of Complex Neuron Response

The proposed representation aims at simulating the response of complex neurons and it is based on a convolution of gradients in specific directions in a pre-defined neighborhood. Since veins of different hands have a diversity of patterns, in this study, for process simplicity, we just make use of a circular neighborhood R, as illustrated in Fig.3. The precise radius value of the circular area needs to be fixed experimentally. The response of a complex neuron at a certain pixel location is a set of gradient maps in different orientations convolved with a Gaussian kernel.

Specifically, given an input image I, a certain number of gradient maps G_1, G_2, ..., G_o, one for each quantized direction o, are firstly computed. They are defined as:

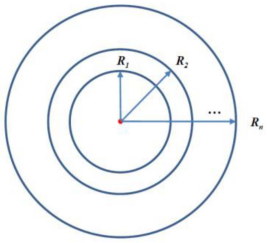


Fig. 3. The neighborhood of the complex neurons is a circular region; its radius can be changed according to the scale

$$G_o = \left(\frac{\partial I}{\partial o}\right)^+ \tag{1}$$

The "+" sign means that only positive values are kept to preserve the polarity of the intensity changes, while the negative ones are set to zero.

Each gradient map describes gradient norms of the input original image in an orientation o at every pixel. We then simulate the response of complex neurons by convolving its gradient maps with a Gaussian kernel G. The standard deviation of G is proportional to the value of radius of the given neighborhood area, R, as in eq. 2.

$$\rho_o^R = G_R * G_o \tag{2}$$

The purpose of the convolution with Gaussian kernels is to allow the gradients to shift in a neighborhood without abrupt changes.

At a given pixel location (x, y), we collect all values of the convolved gradient maps at that location and form the vector $\rho^R(x, y)$, and it thus has a response value of complex neurons for each orientation o.

$$\rho^R(x, y) = \left[\rho_1^R(x, y),\ \cdots,\ \rho_O^R(x, y)\right]^t \tag{3}$$

This vector, $\rho^R(x, y)$, is further normalized to unit norm vector, which is called response vector and denoted by $\underline{\rho}^R$.

3.2 Oriented Gradient Maps by Response Vectors

According to the response vectors, an image can be represented by its perceived values of complex neurons. In this work, the original input image is NIR back of the hand vein image. Specifically, given a raw image I, we generate an Oriented Gradient Map (OGM) J_o using complex neurons for each orientation o defined as in eq. 4.

$$J_o(x, y) = \underline{\rho}_o^R(x, y) \tag{4}$$

Fig.4 depicts such a process applied to a NIR back of the hand vein image. In our work, we generate eight OGMs for eight pre-defined quantized directions. Instead of original NIR back of the hand images, these OGMs are thus used for the subsequent matching for hand vein identification.

3.3 The Properties of Distinctiveness and Invariance

The OGMs potentially offer high distinctiveness as they highlight the details of local texture variations. Meanwhile, they also possess some interesting properties of robustness to affine lighting variations

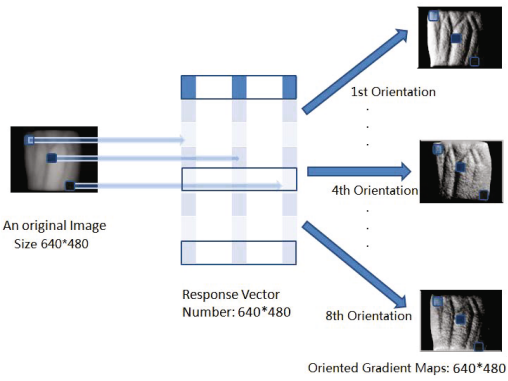


Fig. 4. The OGMs describe a perceived NIR hand vein image in 8 orientations

When applied OGMs to 2D texture images, e.g. NIR back of the hand images, it offers the property of being robust to affine illumination transformations. Indeed, an OGM J_o is simply normalized convolved gradient maps at the orientation o, while monotonic illumination changes often adds a constant intensity value, as a result, it does not affect the computation of gradients. Furthermore, a change in image contrast in which the intensities of all the pixels are multiplied by a constant will lead to the multiplication of gradient computation; however, this contrast change will be cancelled by the normalization of response vectors.

OGMs can be made even rotation invariant if we chose to quantize directions starting from the principal gradient direction of all the gradients within a given neighborhood. Nevertheless, we do not perform such rotation normalization for saving computational cost as NIR back of the hand vein images in NCUT were already roughly aligned.

3.4 Difference Discussion with the State of the Art

In the literature, a few Gaussian filter based descriptors also exist, such as Gabor filter [16] and Daisy descriptor [17]. Gabor filters, which are spatially localized and selective to spatial orientations and scales, are comparable to the receptive

fields of simple cells in mammalian visual cortex [18]. Whilst the Daisy descriptor convolves these gradient maps of pre-defined directions with Gaussian filters of different kernel sizes arranged based on a daisy-style neighborhood to compose a comprehensive representation of an image.

Compared with the two descriptors mentioned above, i.e. Gabor filters and daisy, the proposed OGMs fundamentally share the same biological vision principal to build complex neurons for image representation. Considering that the input image required by the following keypoint detection should have sufficient details, to only highlight the distinctiveness of these hand vein images and avoid the very smooth results given by larger Gaussian kernels, we experimentally set the radius of the circular neighborhood at a specific value rather than introduce the multi-scale strategy as applied by Daugman [19], which computes average gradient variations between adjacent circular rings according to a coarse-to-fine scheme supported by varying kernel based Gaussian smoothing. As a result, the computation of the OGMs is more efficient.

Finally, the OGMs are a set of output images whose details are greatly enhanced with all preserved spatial information (see Fig.4), which is suitable for the subsequent framework of local matching and distinct from the feature vector based one as in [20].

4 Local Feature Matching

The local feature extraction directly on original hand vein images leads to few local features with low distinctiveness. On the other hand, the OGMs of a hand vein image contain much more details of local texture variations, thus simultaneously enhancing the distinctiveness of the vascular network and surrounding corium. Once OGMs of hand vein images are generated, we extract the widely-used SIFT features [21] to associates keypoints between two hand images and to account for the changes in rotation and lighting conditions. These local features are further employed for similarity score computation and final decision making.

4.1 Keypoint Detection

SIFT applies the scale-space Difference-of-Gaussian (DoG) to detect keypoints in images. A given image is repeatedly convolved with Gaussians of different scales separated by a constant factor k to produce an octave in scale space. As for an original input image, $I(x, y)$, its scale space is defined as a function, $L(x, y, \sigma)$, produced by convolution of a variable scale Gaussian $G(x, y, \sigma)$ with the image I, and the DoG function $D(x, y, \sigma)$ can be computed from the difference of two nearby scales:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (5)$$

The extrema of $D(x, y, \sigma)$ can be detected by comparing each pixel value to those of its 26 neighbors within a 3×3 area at current and adjacent scales. At

each scale, gradient magnitude, $m(x, y)$, and orientation, $\theta(x, y)$, is computed by using pixel differences in eq.6 and eq.7.

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (6)$$

$$\theta(x,y) = \tanh(L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y)) \quad (7)$$

For each detected keypoint, a feature vector is extracted as a descriptor from the gradients of sampling points within its neighborhood. In order to achieve orientation invariance, coordinates and gradient orientations of sampling points in the neighborhood are rotated relative to keypoint orientation. Then a Gaussian function is used to assign a weight to gradient magnitude of each point. Points close to the keypoint are given more emphasis than the ones far from it (see [21] for more details of SIFT parameter setting). The orientation histograms of 4×4 sampling regions are calculated, each with eight orientation bins. Thus a feature vector with a dimension of 128 ($4 \times 4 \times 8$) is produced.

SIFT operates on each OGM separately. Because OGMs highlight local texture characteristics of hand vein images, much more keypoints are detected for the following SIFT matching step than those in the original NIR images. Some statistical work was done along with the experiments on the NCUT database. The number of keypoints extracted from each of OGM can rise up to 627, while that from the original hand vein image can be as few as 4. Fig.5 illustrates this phenomenon.

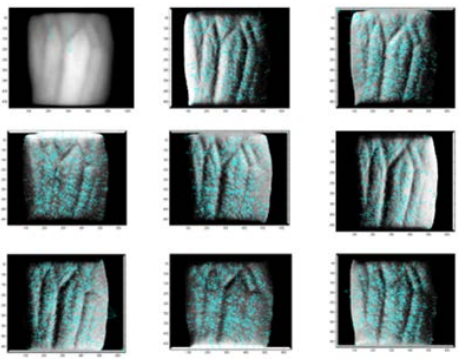


Fig. 5. SIFT-based keypoint detection. The upper row displays the original hand vein image along with the OGMs in the first two directions. The two bottom rows display the OGMs in the left six directions. All images are with their detected keypoints.

4.2 SIFT-Based Local Feature Matching

Given these local features extracted from each OGM pair of the gallery and probe hand vein images respectively, two sets of hand vein keypoints can be matched. Matching one keypoint to another is accepted only if similarity distance is less than a predefined threshold t times the distance to the second closest match. In this work, t is empirically set at 0.6 as in [21]. We can hence denote the number of the matched keypoints by N_o in an OGM pair in the the o_{th} direction. The bigger N_o is the more likely the underlying hand vein images are the same.

This similarity measure, N_o, is with a positive polarity (a bigger value means a better matching relationship). A hand vein image in the probe set is matched with every hand vein images in the gallery set. The nth element in each matching score vector corresponds to the similarity measure between the probe and the n_{th} gallery hand vein image. Each vector is normalized to the interval of [0, 1] by using the max-min rule. In order to achieve complete analysis, the matching scores of OGM pairs in all orientations are fused using a basic weighted sum rule:

$$S = \sum_{i=1}^{o} w_i \cdot N_i \tag{8}$$

The corresponding weight w_i is calculated dynamically during the online step using the scheme as in [22]:

$$w_i = \frac{\max_1(N_i) - mean(N_i)}{\max_2(N_i) - mean(N_i)} \tag{9}$$

where the operators $max_1(S)$ and $max_2(S)$ produce the first and second maximum value of the score S respectively. The gallery hand vein image which owns the maximum value is declared as the identity of the probe hand vein image.

5 Experimental Results

To comprehensively evaluate the proposed method, we designed 4 experiments that are detailedly introduced in the following subsections. The experiments were conducted in the scenario of identification as in the state-of-the-art tasks using the NCUT database described in section 2. Recall that this dataset is one of the largest datasets on NIR back of the hand vein images as it contains 10 right and 10 left hand vein images respectively for each of the 102 subjects enrolled, thus making up a dataset of 2040 hand vein images. All the hand vein images are roughly aligned thanks to the hardware configuration, but they still have slight rotation and lighting variations.

5.1 The Discriminative Power of OGMs

As it was found out that the hand vein pattern is unique to some level for each person as well as each hand [23], we hence considered three different experimental setups, namely, i) left hand vein images only, ii) right hand vein images only, iii) both the left and right hand vein images but as if we had 204 subjects each of which has 10 vein images in the dataset. For each setup, we followed a popular protocol that the first 5 images were used in the gallery set while the remaining 5 images were exploited as probes. We computed the recognition rate of each OGM and their combination as displayed in Table 1.

Table 1. Performance of each OGM and their combination in the setup of left-hand only, right-hand only and both-hands on the NCUT database

Directions	Left Hand Only	Right Hand Only	Both Hands
OGM-1	92.94%	93.53%	93.04%
OGM-2	81.18%	79.41%	78.53%
OGM-3	75.88%	77.45%	75.10%
OGM-4	73.14%	60.78%	66.18%
OGM-5	97.57%	92.55%	91.57%
OGM-6	78.82%	74.90%	75.49%
OGM-7	77.65%	84.51%	80.69%
OGM-8	78.63%	82.16%	78.82%
Fusion	99.02%	99.02%	99.02%

As we discussed in section 4, each hand vein image has very limited number of keypoints if applying DoG directly on original images, and a reasonable accuracy cannot be achieved, which concludes the fact that enhancing the distinctiveness of hand vein images by OGMs is a necessity because each OGM contributes to identification. From Table.1 we can see that the fusion of all 8 OGMs leads to a much better result than any of the single one. This fact accords with our preliminary study for this issue adopting subspace techniques [24]. Unfortunately, in that work, due to the sensitivity of holistic approaches to NIR intensity changes and hand geometric transformations, only about 70%-80% rank-one recognition rates were reported even with a easier experimental setting, and it is not accurate enough for a biometric system. On the other hand, we compared the results in the three columns, and found out that the performance only using left hand images was comparable to that only using right hand ones. When left and right hand vein images were both used and considered as captured from different subjects, the result generally remains stable, suggesting that our method works well as the class size is doubled.

5.2 The Impact of Gallery Size

Many efforts have been focused on how to improve the system accuracy, however, it seems that most of them neglect the potential problem which may stem from

the hand vein database where there are only limited sample images per person enrolled, possibly due to the difficulties of sample collection or the bottleneck of storage capability of the systems. Therefore, we vary the size of gallery samples of each person from 1 to 9 (since at least 1 sample per person should be used in the probe set) to analyze the impact of the gallery size to the proposed method, using the third protocol of both hand images defined in the last experiment. As illustrated in Fig. 6, we can see that the accuracies of each single OGM as well as their combination are generally improved as the gallery size increases, and our method can even achieve a recognition rate of 91.67% when only 2 samples are enrolled in the gallery set of each subject.

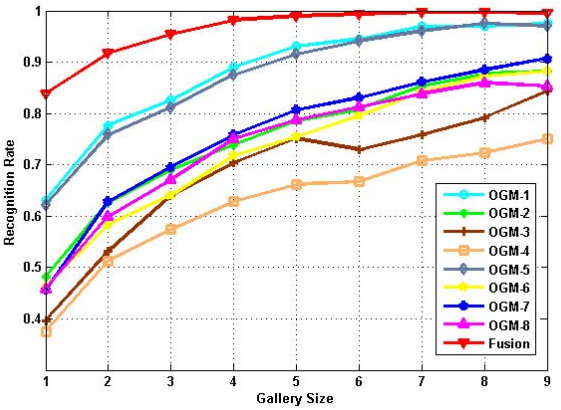


Fig. 6. Accuracy curves with respect to the gallery size of each subject

The cumulative match characteristic (CMC) curves with different numbers (from 1 to 9) of gallery samples of each subject are provided in Fig. 7.

5.3 The Comparison with the State of the Art

Table 2 compares the proposed method with the state of the art that evaluated their accuracies on the NCUT database. In [25], Wang et al. followed the way that firstly detects the vein region on each back of the hand and represents the region as a binary image, and then applies SIFT on the generated binary image for the matching step, originally introduced by Ladoux et al. [4] for palm vein recognition. we can see that if only using the vein region, their accuracy is only 78.68%, far behind the one achieved by the proposed method, showing that there indeed exists some useful information in the corium region that can contribute to the final result if described properly. Fig. 8 demonstrates an example of SIFT based keypoint matching using corresponding OGM pairs of two left hand images belonging to the same subject. It can be seen in Fig. 8 that due to the utilization of the OGMs, the details of the entire back of hand are highlighted, leading to a robust matching result. On the other hand, these matched keypoints of different

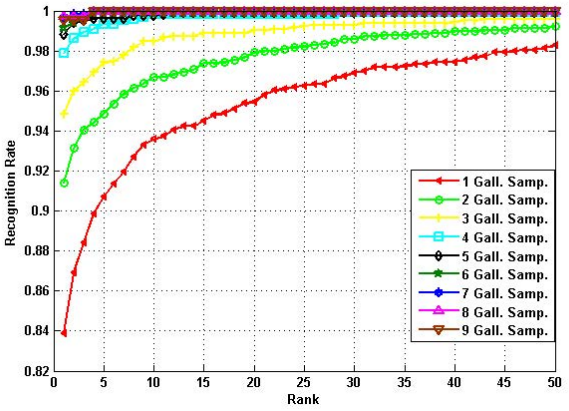


Fig. 7. Accuracy curves with respect to the number of gallery samples of each subject

OGM pairs are located in various positions, proving the fact once again that the similarity measurements of different orientations are complementary. Moreover, the matched keypoints distribute in the vein (marked in yellow) and surrounding corium (marked in red) area, indicating both the regions contribute to the final performance. Our accuracy is also slightly higher than the best one reported in [25] that was achieved by adopting the relationship of multiple gallery samples of each subject. In [14], Wang et al. employed the circular partition LBP (thus namely LCP), and achieved a recognition rate of 90.88% with 5 hand vein images in the gallery set and the remaining 5 ones used as probe. In our case, using the same protocol, much better performance is obtained.

Table 2. The Comparison with the state of the art on the NCUT dataset

Local feature	Class Num	Gallery/Probe	Results
OGM+SIFT	204	816/1224	**98.20%**
Binary+SIFT [25]	204	816/1224	78.68%
Best Binary+SIFT [25]	204	816/1224	97.95%
OGM+SIFT	204	1020/1020	**99.02%**
LCP [14]	204	1020/1020	90.88%

5.4 The Complementation of Left and Right Hands

Since vein patterns are different to some level for both hands of the same person [23], intuitively, the left and right hands of one person should possess complementary information for identification. In this experiment, we further investigate the answer to this problem by fusing the similarity measurement of each hand using the weighted sum rule as for combining the similarity of single OGM. We can see from Table 3 that the accuracy based on the fusion of both hands (in the

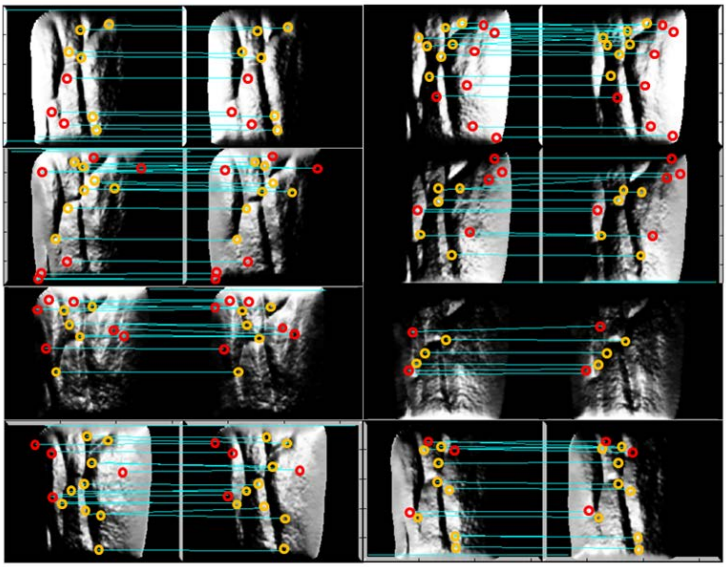


Fig. 8. A matching example using the corresponding OGM pairs of two left hands of the same subject. The left column from the top to the bottom: OGM1 to OGM4; while the right one with the same order: OGM5 to OGM8. The matched keypoints marked in yellow are located in the vein region; the red ones are in the corium area.

Table 3. The results of left hand only, right hand only, and their fusion using different numbers of gallery samples

Number of Gallery	Left Hands	Right Hands	Fusion of Both Hands
1	84.53%	84.31%	95.32%
2	92.28%	91.18%	97.92%
3	95.66%	94.96%	99.16%
4	97.55%	98.53%	100.00%
5	99.02%	99.02%	100.00%
6	99.22%	99.51%	100.00%
7	99.35%	100.00%	100.00%
8	99.51%	100.00%	100.00%
9	99.02%	100.00%	100.00%

3rd column) always outperforms that based on either of single hand, and we can achieve a rank-one recognition rate of 95.32% even using one enrolled hand vein sample. This fact clearly proves that left and right hands provide complementary cues for identifying persons.

6 Conclusions and Perspectives

In this paper, we presented the Oriented Gradient Maps (OGMs) as an effective representation of NIR back of the hand vein images to enhance the distinctiveness along with a local feature SIFT-based matching scheme to ameliorate the accuracy of back of the hand vein identification. Experimental results achieved on the NCUT databases clearly demonstrated the effectiveness of the proposed approach.

In future work, we will investigate other fusion strategies to combine different orientation scores for possible improvements on the final performance.

Acknowledgement. This work was supported in part by the National Basic Research Program of China under grant 2010CB327902; the National Natural Science Foundation of China (No. 61273263, No. 61202237, No.61271368); and the Fundamental Research Funds for the Central Universities.

References

1. Kumar, A., Hanmandlu, M., Gupta, H.M.: Online biometric authentication using hand vein patterns. In: IEEE Symposium on Computational Intelligence for Security and Defence Applications (2009)
2. MacGregor, P., Welford, R.: Imaging for security and personnel identification. Advanced Imaging 6, 52–56 (1991)
3. Malki, S., Spaanenburg, L.: Hand veins feature extraction using dt-cnns. In: SPIE International Symposium on Microtechnologies for the New Millennium, vol. 6590 (2007)
4. Ladoux, P., Rosenberger, C., Dorizzi, B.: Palm vein verification system based on sift matching. In: IEEE International Conference on Biometrics (2009)
5. Lin, C., Fan, K.: Biometric verification using thermal images of palm-dorsa vein patterns. IEEE Transactions on Circuits and Systems for Video Technology 14, 199–213 (2004)
6. Zhao, S., Wang, Y., Wang, Y.: Biometric identification based on low quality hand vein pattern images. In: IEEE International Conference on Machine Learning and Cybernetics (2008)
7. Kumar, A., Prathyusha, K.: Personal authentication using hand vein triangulation and knuckle shape. IEEE Transactions on Image Processing 18, 2127–2136 (2009)
8. Miura, N., Nagasaka, A., Miyatake, T.: Feature extraction of finger-vein pattern based on repeated line tracking and its application to personal identification. Machine Vision and Applications 15, 194–203 (2004)
9. Cross, J., Smith, C.: Thermographic imaging of the subcutaneous vascular network of the back of the hand for biometric identification. In: IEEE International Carnahan Conference on Security Technology (1995)
10. Wang, K., Yuan, Y., Zhang, Z., Zhuang, D.: Hand vein recognition based on multi-supplemental features of multi-classifier fusion decision. In: IEEE International Conference on Mechatronics and Automation (2006)
11. Huang, D., Ben Soltana, W., Ardabilian, M., Wang, Y., Chen, L.: Textured 3d face recognition using biological vision-based facial representation and optimized weighted sum fusion. In: IEEE International Conference on Computer Vision and Pattern Recognition Workshop on Biometrics (2011)

12. Wang, L., Leedham, G.: Near- and far-infrared imaging for vein pattern biometrics. In: IEEE International Conference on Advanced Video and Signal-based Surveillance (2006)
13. Zhao, S., Wang, Y., Wang, Y.: Extracting hand vein patterns from low-quality images: a new biometric technique using low-cost devices. In: IEEE International Conference on Image and Graphics (2007)
14. Wang, Y., Li, K., Cui, J., Shark, L., Varley, M.: Study of hand-dorsa vein recognition. In: IEEE International Conference on Intelligent Computing (2007)
15. Edelman, S., Intrator, N., Poggio, T.: Complex cells and object recognition. Unpublished manuscript (1997), http://kybele.psych.cornell.edu/~edelman/archive.html
16. Gabor, D.: Theory of communications. Journal of the Institute of Electrical Engineers 93, 429–457 (1946)
17. Tola, E., Lepetit, V., Fua, P.: Daisy: an efficient dense descriptor applied to wide baseline stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 815–830 (2010)
18. Marcelja, S.: Mathematical description of the responses of simple cortical cells. Journal of the Optical Society of America 70, 1297–1300 (1980)
19. Daugman, J.: High confidence visual recognition of persons by a test of statistical independence. IEEE Transactions on Pattern Analysis and Machine Intelligence 15, 1148–1161 (1993)
20. Liu, C., Wechsler, H.: Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. IEEE Transactions on Image Processing 11, 467–476 (2002)
21. Lowe, D.G.: Distinctive image features from scale-invariant key-points. International Journal of Computer Vision 60, 91–110 (2004)
22. Mian, A.S., Bennamoun, M., Owens, R.: Keypoint detection and local feature matching for textured 3d face recognition. International Journal of Computer Vision 79, 1–12 (2008)
23. Badawi, A.M.: Hand vein biometric verification prototype: A testing performance and patterns similarity. In: International Conference on Image Processing, Computer Vision and Pattern Recognition, pp. 3–9 (2006)
24. Chen, L., Huang, D., Chiheb, H., Ben Amar, C.: Increasing the distinctiveness of hand vein images by oriented gradient maps. In: International Conference of the Special Interest Group on Biometrics and Electronic Signatures (2011)
25. Wang, Y., Fan, Y., Liao, W., Li, K., Shark, L., Varley, M.: Hand vein recognition based on multiple keypoints sets. In: International Conference on Biometrics (2012)

Fusing Warping, Cropping, and Scaling for Optimal Image Thumbnail Generation

Zhan Qu[1], Jinqiao Wang[1], Min Xu[2], and Hanqing Lu[1]

[1] National Laboratory of Pattern Recognition, Institute of Automations, Chinese Academy of Sciences, Beijing, China
[2] iNEXT, School of Computing and Communications, University of Technology, Sydney, Australia

Abstract. Image retargeting, as a content aware technique, is regarded as a logical tool for generating image thumbnails. However, the enormous difference between the size of source and target usually hinders single retargeting method from obtaining satisfactory results. In this paper, an unified framework is proposed to fuse three popular retargeting strategies, i.e. warping, cropping, and scaling, for thumbnail generation. Complementing each other, three retargeting strategies work together efficiently. Firstly, cropping selectively discards the unimportant regions in order to free up more space for displaying important content aesthetically. Next, warping helps to incorporate as much as possible visual information into thumbnails by rearranging important content more compactly through non-uniform deformation. Finally, scaling retrains the important content at an optimal size rather than undergoing an improper shrinkage. In our solution, warping, cropping and scaling are encoded as three energy terms of the objective function respectively, which can be solved efficiently by numerical optimization. Both qualitative and quantitative comparison results demonstrate that the proposed method achieves an excellent trade-off among smoothness, completeness and distinguishableness in thumbnail generation. Through these results, our method shows obvious superiority over state-of-the-art techniques.

1 Introduction

With the development of multimedia and Internet techniques, massively increasing visual data, such as image and video, play an important role in modern computer application. Therefore, how to present and browse image data efficiently becomes an urgent issue to be resolved. Thumbnail, as a small-size generalization of source image, has been used broadly across various digital display platforms, from PC, PDA, cell phone to digital album. Most of the current image tools generate thumbnails through scaling the source uniformly. However, this intuitive strategy often causes noticeable distortion and shrinkage of important content in image. Consequently, the generated thumbnails can hardly deliver meaningful information, which cannot satisfy either the users' intuitional browsing experience or intelligent image searching and recognition.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 445–456, 2013.

As a crowded research topic, content-aware image retargeting is originally designed for changing the aspect ratio of image to accommodate various display devices. The core of technique focuses on preserving the visual information of important content as possible while resizing the images arbitrarily. Recently, a variety of approaches have been published, which can be further categorised as discrete methods —seam carving [1,2,3], continuous methods —cropping [4,5], warping [6,7,8], shift map [9], and hybrid approaches [10,11,12]. Due to the content aware property, image retargeting is considered as a reasonable tool for thumbnail generation. However, in practice, the existing retargeting approaches are better for resizing images to a comparable size. Once the target size is set too small, especially for the thumbnail of 100×100 or so, single retargeting approach can hardly secure viewers' browsing experience.

Cropping methods [4,5] return a target-size window, which covers the most salient content. When applied for thumbnails, the cropping window has to only retain part of the important content while discard the other regions. This often destroy the integrity of the objects, and the result thumbnails can not provide meaningful information. Seam carving methods [1,2,3] alter image size by removing the unimportant pixel chain in both horizontal and vertical directions iteratively. However, the quite difference between the size of source and target brings significant damage to the geometric structure of content. Warping methods [6,7,8] continuously transform the image to target size while decentralize distortion to non-salient regions. Compared to other methods, warping based methods maximally preserve the geometric structure. However, the continuity of warping transformation not only is the key of visual smoothness, but also permits the unimportant region to occupy more or less room in output. As a result, there is usually not enough space to absorb the distortion in warping, and an obvious scale shrinkage of the important objects will appear. In the case of thumbnail, the objects even become undistinguishable. In addition, shift-map methods [9] are proposed for reconstructing image through cropping and blending the important regions. Most recently, some hybrid approaches [10,11,12] are developed to further improve the performance of retargeting. Without the consideration about small-size target, these recent works subject to the same difficulties mentioned above, and are not appropriate for handling the thumbnail problems.

Some researchers have noticed the limitation of existing retargeting strategies on generating thumbnail. Sun *et al.* [13] proposed an approach for addressing this problem, where seam carving and warping are used in combination. They first employ cyclic seam carving(CSC) to adapt images to thumbnail size. Subsequently, the result guides the thumbnail generation with a thin plate spline(TPS), which forms a continuous mapping from the source to target. Although this method preserves the smoothness in thumbnail and makes the most use of the limited space, the two-stage combination leads to too much time cost to meet the practical needs. As discussed above, direct scaling totally neglects the distortion and shrinkage of content; cropping may damage the completeness; warping often weakens the distinguishableness. Motivated to

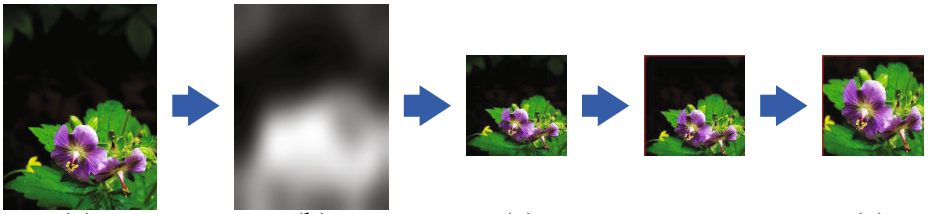


Fig. 1. The thumbnails generated by each step of our approach. From left to right: (a) Source image (b) Importance map (c) Warping the image just considering shape distortion (d) The result of cropping + warping (e) The result of cropping + warping + scaling.

integrate the advantages of cropping, warping and scaling while guide them to complement each other, in this paper, an unified thumbnail generating model is proposed, where three retargeting strategies are encoded in terms of the energy terms respectively. Firstly, warping distributes deformation across images non-homogeneously, which rearranges the salient content more compactly as well as preserving the geometric structure of objects smoothly. This property actually helps the cropping window to include more important information while to avoid the loss of completeness. Secondly, cropping automatically removes the relatively unimportant margin of images. Consequently more space in thumbnail is reserved for warping to absorb the distortion due to resizing and makes it possible to produce the output more aesthetically. Moreover, scaling is added as a constraint for restraining the important content at an optimal size, which aims at striking a balance between distinguishableness and completeness. As shown in Fig.1, in our solution, the above three methods are fused into an unified optimization, which can be solved efficiently and specially appropriate for the image tools across various digital platforms.

2 The Unified Framework

To achieve an efficient solution, we fuse three retargeting methods as an unified framework. By encoding warping, cropping, and scaling as three energy terms, the total objective function is formulated as follows:

$$D_{total} = D_W + \lambda D_C + \mu D_S \tag{1}$$

where D_W indicates warping and evaluates the shape distortion of content in thumbnail; D_C indicates cropping and decides how to discard the unimportant regions; D_S indicates scaling and is employed for displaying content at an optimal size. λ and μ are two parameters responsible for adjusting the weights of them.

Our algorithm is implemented on the basis of the grid structure. The first step is to partition the source image into $m \times n$ grids uniformly, and denote the set of grids as $Q = \{q_{11},\ q_{12}, \ldots,\ q_{mn}\}$. $V_{ij} = \{v_{ij}^1,\ v_{ij}^2,\ v_{ij}^3,\ v_{ij}^4\} \subseteq R^2$ is defined as the vertex coordinate set of q_{ij}. $\tilde{V}_{ij}$ represents the deformed coordinate

set in thumbnail correspondingly. D_{total} is actually the sum of various energy loss of all grids. Minimizing of D_{total} finally results in new vertex coordinates. The deformed vertex coordinates and cropping window of target size determine the output together. In our solution, the minimizing is solved as a convex programming problem, which can be resorted to the numerical optimization plan efficiently.

2.1 Warping

In this section, we employ grid based warping to preserve the spatially important content from obvious shape distortion, which is suppose to retain the geometric structure smoothly. We employ the *similarity transformation* as in [14] to evaluate the shape distortion energy in resizing, which can be formulated as follows:

$$D_w(q) = \sum_{l}^{4} \parallel s_q(v_q^l) - \tilde{v}_q^l \parallel_2^2 \tag{2}$$

where s represents the unique *similarity transformation* for each grid. This transformation is essentially formed from the affine projection in 2D, which can be defined in terms of four parameters as follows:

$$s(v) = \begin{bmatrix} c & -d \\ d & c \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}, \quad v = \begin{bmatrix} x \\ y \end{bmatrix} \tag{3}$$

By plugging Eq.3 within, Eq.2 can be reformulated as a linear least-quare problem in terms of $[c,\ d,\ t_x,\ t_y]^T$, where

$$A_q = \begin{bmatrix} x_q^1 & -y_q^1 & 1 & 0 \\ y_q^1 & x_q^1 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_q^4 & -y_q^4 & 1 & 0 \\ y_q^4 & x_q^4 & 0 & 1 \end{bmatrix}, \quad b_{\tilde{q}} = \begin{bmatrix} \tilde{x}_q^1 \\ \tilde{y}_q^1 \\ \vdots \\ \tilde{x}_q^4 \\ \tilde{y}_q^4 \end{bmatrix} \tag{4}$$

It is intuitive to obtain the least-square solution $[c,\ d,\ t_x,\ t_y]_q^T = (A_q^T A_q)^{-1} A_q^T b_{\tilde{q}}$ and the shape distortion energy $D_w(q) = \parallel (A_q (A_q^T A_q)^{-1} A_q^T - I) b_{\tilde{q}} \parallel_2^2$, more details can be found in [14]. The shape distortion energy is finally formulated as:

$$D_W = \sum_{i,j}^{m,n} I_{ij} D_w(i,j) \tag{5}$$

Up to now, the target has converted to solve a quadratic programming(QP) problem, which can be handled efficiently. We quantify the importance of grids as I, which is normalized to $[0.2, 2]$ for avoiding undue deformation. Some boundary constraints can make the deformed grids have specified size, however, in this solution, we just restrict the deformed image to have rectangle appearance.

2.2 Cropping

As discussed before, the products of single warping usually subject to the small size of target and hardly make a good meeting to human vision. Thus, by cropping some regions out selectively, we provide system more space for presenting the content in result aesthetically. At the same time, the cropping procedure is requested to satisfy two requirements: 1) the most important information of image should be preserved preferentially; 2) the proposed scheme should be naturally integrated into our framework, while not influencing the smooth geometric structure achieved by warping.

To achieve this, we first define a spatial rectangle of target size as the cropping window. The deformed grids, which are located outside the window, are deemed as "cropped" and would not appear in thumbnail. Then, we encode the information loss caused by cropping as a piecewise function as follow:

$$D_C = \sum_{i,j}^{m,n} = (D_c^x(i,j) + D_c^y(i,j)) \tag{6}$$

and

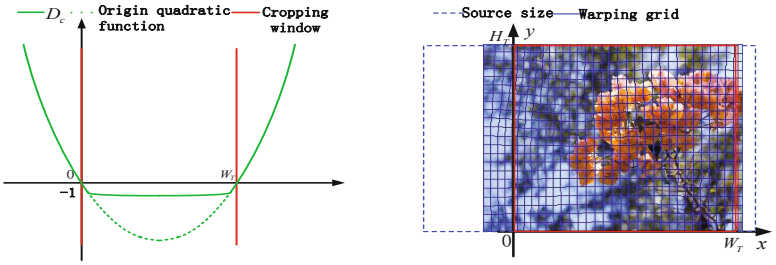
$$D_c^x(q) = \begin{cases} (-1) \cdot (x_q \cdot (W_T - x_q)) \cdot I_q & \text{if } x_q \notin [0, W_T] \\ (-1) \cdot (x_q \cdot (W_T - x_q))^{\frac{1}{\delta}} \cdot I_q & \text{if } x_q \in [0, W_T] \end{cases} \tag{7}$$

$$D_c^y(q) = \begin{cases} (-1) \cdot (y_q \cdot (H_T - y_q)) \cdot I_q & \text{if } y_q \notin [0, H_T] \\ (-1) \cdot (y_q \cdot (H_T - y_q))^{\frac{1}{\delta}} \cdot I_q & \text{if } y_q \in [0, H_T] \end{cases} \tag{8}$$

where W_T, H_T are the width and height of thumbnail respectively. $[x_q,\ y_q]^T$ represents the centroid coordinate of q, which can be calculated in terms of V_q easily. And δ is a positive parameter correlated to the size of target. For 120×120 thumbnail, it works well when $\delta = 15$.

As shown in Fig.2(a), the first term of piece-wise function ensures that the function fetches a lower value when the grid lies inside the cropping window than outside, which would help the most important content to be preserved in output preferentially. The second term of Eq.7 or Eq.8 guarantees D_c not to make significant differences when the grid vertices move inside the cropping window. That is, the resulted coordinates of grid within cropping window are still determined by the warping procedure chiefly.

As discussed in last section, D_W is quadratic and the global solution can be resolved. Fig.2(a) illustrates that the original form of D_c is quadratic as well. When it become the piece-wise version, both pieces are still convex respectively. It is easy to prove that the boundary points make no effect on the convexity of entire D_c, and the function is numerical continuous. Although the cropping cost energy is incorporated, the global solution of unified framework still exists and can be solved through numerical optimization. Fig.2(b) shows an example of deforming the grids via combining warping with cropping.



(a) (b)

Fig. 2. (a) The function curve of D_c^x, and D_c^y have same form. (b) The result coordinates of grid vertexes obtained by combining cropping with warping.

2.3 Scaling

Through removing the relatively unimportant margin, cropping frees up more space for system to present the important content more aesthetically. However, the warping procedure takes only shape distortion into account, where the indifference of scale usually prevents the combination of warping and cropping from satisfactory result, especially for small-size thumbnail. As shown in Fig.1 and Fig.3, the thumbnails generated without consideration about scale generally display the content at a small size. Although the content in output may be accurate(have same shape as in source) and complete, all the regions suffer such huge shrinkage that the important content becomes undistinguishable. And it is difficult to reflect the advantage of combination of warping and cropping. To address this problem, we further incorporate scaling into our solution, which is responsible for balancing distinguishableness with completeness.

As discussed before, the *similarity transformation* of each grid is uniquely determined by 4 parameter as $[c,\ d,\ t_x,\ t_y]_q^T$, which factually results from the affine projection in computer vision and can be rewritten as $[r\cos\theta,\ r\sin\theta,\ t_x,\ t_y]_q^T$. θ and r decide the rotation and scaling of transformation respectively, and the latter is positive. When r equals to 1, the grid is considered deformed without scale difference. According to this, we make the scaling take effect as a constraint, which can be formulated in terms of r. We define $U_q = [u_1, u_2, u_3, u_4]^T = (A_q^T A_q)^{-1} A_q^T$, while c, d can be calculated as $c = u_1 \cdot b_{\tilde{q}}$ and $d = u_2 \cdot b_{\tilde{q}}$. For generating thumbnails, the grids generally undergo shrinkage and the corresponding r is supposed to range from 0 to 1. As a result, the scale energy can be formulated as:

$$D_S = sz \cdot \sum_{i,j}^{m,n} I_{ij} \cdot (1 - r_{ij}^2) = sz \cdot \sum_{i,j}^{m,n} I_{ij} (1 - \left\| [u_1, u_2]^T \cdot b_{\tilde{q}} \right\|_2^2) \tag{9}$$

Where sz measures the length of original grid diagonal, and guarantees D_S is comparable to D_W and D_C in magnitude. Eq.9 doesn't accord with norm form, and can be negative. To avoid scaling up objects incorrectly, we impose the following constraints to ensure all $m \times n$ terms of Eq.9 fetch positive values:

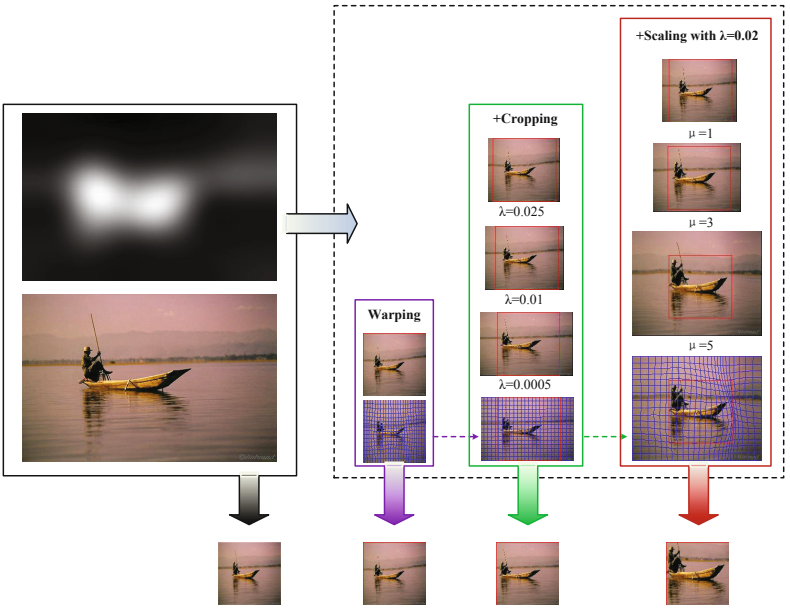


Fig. 3. The thumbnail generating process fusing three strategies together gradually. *The full-line box*: the source image and the importance map. *The dashed-line box*: the thumbnails yielded with different strategies and parameters; the cropping window and grids corresponding to the optimal effect. *The bottom row*: thumbnail results with different strategies.

$$0 < r < 1 \Longrightarrow \begin{cases} -1 < c < 1 \\ -1 < d < 1 \end{cases} \Longrightarrow \begin{cases} -1 < u_1 \cdot b_{\tilde{q}} < 1 \\ -1 < u_2 \cdot b_{\tilde{q}} < 1 \end{cases} \tag{10}$$

All the constraints are linear, which reduce little efficiency in optimization.

2.4 Energy Minimization

As discussed above, D_W and D_S are quadratic, and D_C is convex. Although the latter makes it impossible to solve Eq.1 via sparse linear system, minimizing still can be achieved by numerical algorithm under some linear constraints. In this solution, an active-set method is employed.Minimizing becomes a convex programming problem. Once a local solution is resolved, the global solution is yielded. And we initialize the optimization by placing the vertices of grids inside the cropping window uniformly, where the linear constraints are all satisfied.

Fig.3 shows the thumbnails obtained using different strategies. By adjusting λ and μ, different focuses are reflected in output. When only the grid-based warping is employed, the improvement compared to direct scaling is limited. Although warping tries best to preserve the geometric structure, the ship appears to be so small that the viewers can hardly discriminate it from simple scaling. In the cases of incorporating cropping with warping, a higher λ encourages less regions to be

discarded, which achieves similar result with using warping only; while a lower λ permits abandoning the unimportant regions liberally to improve the aesthetic effect of output. We notice that the cropping helps to improve the visual effect as well as to preserve the most salient object(boat) preferentially. However, since the scale difference is not counted into the shape distortion energy, all the deformed grids(both inside and outside cropping window) usually suffer huge shrinkage, where the cropping window often fails to yield distinguishable output. Finally, when scaling participates in, a trade-off among smooth visual effect, distinguishableness and completeness is achieved. According to the experiments, in Eq.1, the λ can be between 0.01 and 0.08 when fixing $\mu = 6$, and it works well when $\lambda = 0.025$ in most cases.

Fig.3 further demonstrates how warping, cropping and scaling take effect. In order to demonstrate the advantages of our algorithm comprehensively, more examples can be found in Fig.4.

3 Experimental Results

To demonstrate the effectiveness and efficiency of our framework, there are 420 images are used for generating thumbnails of 120×120. There are 240 images collected from PASCAL VOC2008[15], 100 from ImageNet[16], and 80 from RetargetMe[17]. The test images range from photos containing one or more salient foregrounds to landscapes with relatively scattered importance distribution. In this paper, the importance of pixels are computed through combining the visual attention-based method [18] with the face detector[19]. Based on the grid structure, the proposed method achieves better visual effect but consumes more time when employing a finer grid partition. According to practice, grid size of 20×20 strikes a good balance between thumbnail quality and computational complexity in most cases.

To provide a comprehensive evaluation, our method is compared with popular state-of-the-art approaches, including improved seam carving (ISC)[2] for discrete method, shape-preserving warping (SPW)[14] for continuous method and scale and object aware retargeting (SOAR)[13] for combinational method, in both qualitative and quantitative fashions. In addition, the speed of different methods are presented for further demonstrating the efficiency and feasibility of our method. All the experiments are completed on a computer with Intel Pentium 2.33GHz dual core, 4GB memory.

3.1 Speed

Table.1 gives the average time cost of various methods. On one side, the seam based methods show a poor performance in comparison: single seam carving procedure consumes about $0.5s$ on average; for SOAR[13], which employs seam carving as a component of two-stage combination, the total time even exceeds 1s. In fact, the performances of seam based methods are specially dependent on the size of source: when the size reaches 700×500, the computational time

Table 1. Speed of various methods for adapting 500×400 source images to 120×120 thumbnails

Method	SPW[14]	ISC[2]	SOAR(CSC+TPS)[13]	ours
Time	0.02s	0.6s	1.25s(0.5s+0.75s)	0.3s

quickly reaches more than $2.5s$, which is generally deemed inappropriate for any online applications. On the other side, as the grid based approaches, although both SPW[14] and our method employ grids of 20×20 and consequently have the same number of variables in optimization, SPW[14] works more efficiently than ours. This is because the objective function of SPW[14] is quadratic and can be converted to a sparse linear system. Our solution, of which D_C is not quadratic but just convex, has to resort to numerical optimization. Fortunately, the computational cost of our method is still acceptable. Our method outperforms seam based methods obviously and can achieve a satisfactory trade-off between effectiveness and efficiency.

3.2 Qualitative Results

Some intuitional comparisons are shown in Fig.4, where various types of images are exhibited. For SPW[14], the output images are usually smooth and natural. And yet, the important foregrounds have to face a borderless shrinkage at the same time. Especially for the purpose of thumbnail, the foreground become so small that viewers can hardly discriminate them from scaling. This occurs in almost all examples. ISC[2] produces the thumbnails by removing the least important pixel chain iteratively, which from another angle helps the foreground maintain a understandable size. However, this arbitrary strategy of discarding often damages the geometric structure of output severely. As shown in Fig.4, unacceptable distortions occur in the 1st, 2nd, 3rd, 6th, 8th, 9th, 10th rows on the left side, and the 1st, 2nd, 3rd, 4th, 5th, 7th, 8th rows on the right. SOAR[13], making use of discrete seam carving to guide continuous mapping, yields compatible results with ours in some cases, which prevents the foreground from significant shrinkage and achieves the smooth global visual effect simultaneously. Nevertheless, unnatural distortions still happen occasionally, such as the 1st, 6th, 8th, 9th, 10th rows on the left, and the 5th, 7th, 8th, 10th rows on the right. The reason can be explained as that TPS is insufficient for completely repairing the geometric structure damaged by CSC.

Generally speaking, the continuity of warping not only is the key of visual smoothness, but also permits the unimportant region to occupy more or less room in thumbnail. The too much space occupied by unimportant content impels the foreground to undergo a fain shrinkage, which causes the degeneration of distinguishableness definitely. On the contrary, the discrete methods reserve as much space as possible for preserving important content, while the geometric structure cannot be saved effectively. SOAR[13], as a two-stage solution, combines the advantages of discrete approaches and continuous approaches. In

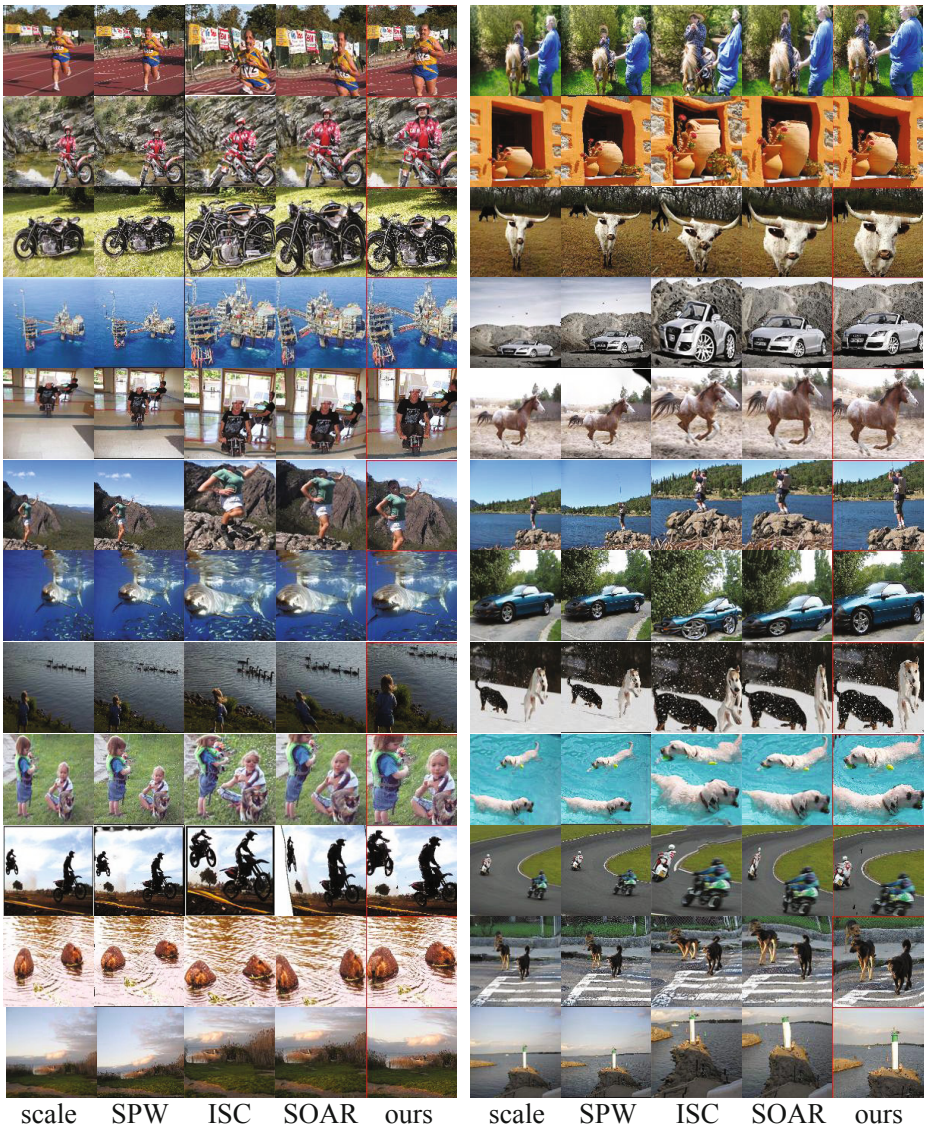


Fig. 4. The thumbnail results produced by various methods. There is a single salient foreground in the images placed on the first 7 rows. Those in the 8–11th row contain multiple salient objects. Two landscapes are arranged on the last row.

Table 2. The statistics of user study. Beside the time cost and accuracy on thumbnail browsing, the user study takes the visual preference into account.

Method	SPW[14]	ISC[2]	SOAR[13]	ours
Time	97.25s	92.79s	84.37s	80.55s
% of accuracy	93.21	94.73	96.52	98.15
% of preference	9.93	15.46	25.84	48.77

essence, our solution is based on the similar consideration. Instead of carving, our solution employs cropping to discard the irrelevant image margin selectively, which reduces the impact of unimportant regions on displaying the important content.

3.3 Quantitative Result

In order to reflect the effectiveness and efficiency of thumbnail browsing objectively, we conducted a user study and collected the quantity statistics for 4 thumbnail generation schemes. There are altogether 72 college students participating in this user study, 50 of which come from the natural science realm while others come from social science realm. Each participator is designated to 4 groups of data, which are generated by various methods respectively. Each group includes 50 thumbnails, and the source are selected at random without overlap. The participators are asked to browse all data and choose the category for each thumbnail from a set of predefined options, which should reflect the content of thumbnail furthest. We counted the time cost and accuracy of each group. In addition, there are 30 other images incorporated into the quantitative comparison, where the thumbnails generated by various methods are displayed to the viewer in a random order simultaneously. The participator are requested to show their aesthetic preference.

The final statistics of all the feedback are shown in Table.2. The quantitative results reflect the superiority of our method over the others. Our approach not only provides higher efficiency and reliability for thumbnail browsing but also attracts more popularity in visual aesthetics than others.

4 Conclusion

In the proposed solution, cropping is incorporated to discard the irrelevant image margin selectively, which factually reduces the impact of unimportant regions on displaying the important content. Warping preserves the shape of foreground smoothly as well as rearranges the important content compactly, which factually encourages the important regions to be preserved completely. Scaling helps to maintain the important foreground distinguishable. As result, our system is able to deliver as much visual information as possible to viewers within a very limited size of thumbnail, and achieves an excellent trade-off among smoothness, completeness and distinguishableness. The qualitative and quantitative results demonstrate the effectiveness and efficiency of our approach.

Acknowledgment. This work was supported by the National Natural Science Foundation of China (Grant No. 60833006, 61070104 and 60905008).

References

1. Avidan, S., Shamir, A.: Seam carving for content-aware image resizing. ACM Transaction on Graphics 26 (2007)
2. Rubinstein, M., Shamir, A., Avidan, S.: Improved seam carving for video retargeting. ACM Transaction on Graphics 27 (2008)
3. Mansfield, A., Gehler, P., Van Gool, L., Rother, C.: Scene Carving: Scene Consistent Image Retargeting. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 143–156. Springer, Heidelberg (2010)
4. Liu, H., Xie, X., Ma, W.Y., Zhang, H.J.: Automatic browsing of large pictures on mobile devices. In: Proceedings of ACM Multimedia, pp. 148–155 (2003)
5. Santella, A.M., Decarlo, D., Salesin, D., Cohen, M.: Gaze-based interaction for semiautomatic photo cropping. In: Proceedings of CHI, pp. 771–780 (2006)
6. Gal, R., Sorkine, O., Cohen-Or, D.: Feature-aware texturing. In: EGSR, pp. 297–303 (2006)
7. Wang, Y.S., Tai, C.L., Sorkine, O., Lee, T.Y.: Optimized scale-and-stretch for image resizing. ACM Transaction on Graphics 27 (2008)
8. Li, B., Chen, Y.M., Wang, J.Q., Duan, L.Y., Gao, W.: Fast retargeting with adaptive grid optimization. In: ICME, pp. 1–4 (2011)
9. Pritch, Y., Kav-Venaki, E., Peleg, S.: Shift-map image editing. In: ICCV, pp. 151–158 (2009)
10. Rubinstein, M., Shamir, A., Avidan, S.: Multi-operator media retargeting. ACM Transaction on Graphics 28 (2009)
11. Dong, W.M., Zhou, N., Paul, J.C., Zhang, X.P.: Optimized image resizing using seam carving and scaling. SIGGRAPH (2009)
12. Liu, L.G., Chen, R.J., Wolf, L., Cohen-Or, D.: Optimize photo composition. Computer Graphic Forum 29 (2010)
13. Sun, J., Ling, H.B.: Scale and object aware image retargeting for thumbnail browsing. In: ICCV, pp. 1511–1518 (2011)
14. Zhang, G.X., Cheng, M.M., Hu, S.M., Martin, R.R.: A shape-preserving approach to image resizing. Computer Graphics Forum 28, 1897–1906 (2009)
15. Everingham, M., Van-Gool, L., Williams, C., Winn, J., Zisserman, A.: The pascal visual object classes challenge (2008)
16. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Li, F.F.: Imagenet: A large-scale hierarchical image database. In: CVPR, pp. 248–255 (2009)
17. Rubinstein, M., Guterrez, D., Sorkine, O., Shamir, A.: A comparative study of image retargeting. SIGGRAPH Asia (2010)
18. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. TPAMI (1998)
19. Viola, P., Jones, M.J.: Robust real-time face detection. Int. J. Computer Vision (2004)

Shift-Map Based Stereo Image Retargeting with Disparity Adjustment

Shaoyu Qi and Jeffrey Ho

University of Florida

Abstract. This paper introduces a novel image retargeting algorithm for 3D images given as pairs of stereo images. In the context of 3D image retargeting, the novel viewpoint advocated in this paper is that the geometric consistency in the form of preserving disparity values should not be an overpowering objective formulated as hard constraints. Instead, for maximizing viewing experience and comfort, it is desirable to simultaneously retarget the images as well as adjust the disparity values. The proposed retargeting algorithm is based on the methods of shift-map and importance filtering, and the main technical contribution of this paper is a successful extension of these earlier techniques to 3D images. We have evaluated the proposed method extensively, and the results demonstrate the efficiency of the proposed method as well as its potential for producing high-quality outputs. In particular, comparing with the state-of-the-art, the proposed method has a considerably shorter running time, and at the same time, it produces the retargeted 3D images that are more agreeable and pleasing for viewing.[1]

1 Introduction

Current image retargeting methods (e.g.,[1]) in the computer vision literature aim to resize the image by changing its aspect ratio while minimizing the resulting distortion to the image content and the structures of salient objects therein. The development of this technology is amply justified by the increasing diversity in the size and shape of the display devices that often require the image to be retargeted to conform with the device's display area. While working well for regular images, it is difficult to extend these methods, in their current forms, to retarget 3D images typically given as pairs of rectified stereo images. Recent development in 3D acquisition technology and the rapid proliferation of 3D contents provide the impetus and motivation for developing next generation image retageting algorithm that can efficiently retarget 3D images for maximizing viewing experience. In realizing this futuristic vision, this paper proposes and evaluates a novel 3D image retargeting algorithm that makes a small but substantial step forward in this direction.

An algorithm for retargeting 3D images given as pairs of stereo images has been reported in a very recent paper by Basha *et al.* [2]. The overall aim of this

[1] This work is partially supported by NSF IIS0916001.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 457–469, 2013.

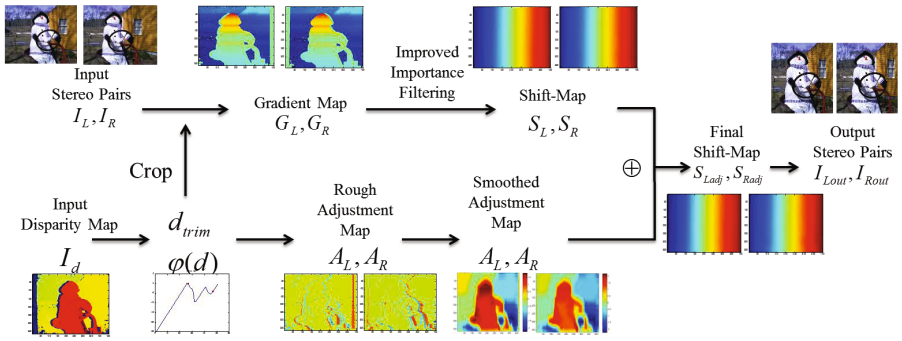


Fig. 1. Outline of The Proposed Stereo Image Retargeting Algorithm: The input is a pair of stereo images, I_L, I_R, and a disparity map I_d. The shift-maps S_L and S_R for retargeting, and the adjustment-maps A_L, A_R for disparity adjustment are initially computed independently. The two pairs of maps are combined at the last step to yield the final shift-maps for generating the retargeted images.

algorithm is to preserve geometric consistency between the image pairs in terms of disparity values preservation. Based on seam carving, the algorithm enforces the geometric consistency by carving only seams consisting of unoccluded pixels, and the left and right carved seams are in exact correspondence. For many applications such as 3D shape reconstruction and depth range estimation, exact disparity values are important and their preservation is worthwhile. However, for image retageting with the overall aim of maximizing viewing experience, the goal is less about the precision of the actual geometric details but more about their *perceptions*, and the latter can be affected by various factors including basic human perceptions and psychological responses.

Extensive work has been done on evaluating various different factors that can affect human perception of 3D images [3]. In particular, there is a certain range of disparity values, the *comfort zone*, that maximizes the acuity of human perception. [4] has proposed a method that adjusts the disparity values based on the 3D image content and saliency. The same argument can (and should) be applied to retargeting applications in that the preservation of disparity values should not be enforced as hard constraints but instead, considered as soft requirements that can be adjusted and modified like image pixels. In particular, disparity values do not enjoy a privileged status in our approach to 3D image retargeting, and our method trades some inaccuracies in disparity values (through adjustments) for the overall objective of maximizing viewing experience and comfort.

This paper proposes a novel algorithm for 3D image retargeting that incorporates retargeting the stereo images as well as adjusting the disparity values. The proposed algorithm is based on the methods of shift-map [5] and importance filtering [1] introduced earlier for retageting 2D images. However, their extensions to stereo images are not straightforward, and significant modifications are required and at places, entirely new sets of ideas have been introduced, and the

latter comprises the main technical contribution of this paper. Figure 1 illustrates the general outline of the proposed algorithm. The inputs are a pair of rectified stereo images, I_L and I_R, and a disparity map I_d. Image retargeting and disparity adjustment, originally proceed independently, produce the shift-maps for the image pair S_L, S_R and the disparity *adjustment-maps* A_L, A_R independently. The final step combines both estimates together by adding the corresponding shift-map and adjustment-map to yield the final shift-map S_{Ladj} and S_{Radj}. The final output stereo image pairs, I_{Lout} and I_{Rout}, is then generated by image interpolation using the two final shift-maps. We evaluate the proposed method extensively in the experimental section , and our results demonstrate our method's efficiency as well as its potential for producing high-quality outputs, with a considerably shorter running time.

2 Related Works

Image/Video retergeting algorithms have attracted considerable amount of attention recently in the vision community. Early retargeting work such as [6] [7] aim to preserve important and salient objects in the image by segmentation, cropping and pasting. While laying the groundwork for future development, the effectiveness of these methods, particularly in an automated and unsupervised setting, is somewhat limited due to their simplicity. For instance, important notions such as preserving scene consistency are more difficult to manage using cropping and pasting that usually alter the image in a non-incremental way.

Seam carving [8] is perhaps the first image retargeting method to gain wide popularity. The method incrementally alters the image by carving seams, sequences of pixels running vertically across the image. Its simplicity allows many extensions in different ways. Rubinstein *et al.* applies seam carving to video retargeting [9], and further incorporate seam carving with cropping and homogeneous resizing to develop a multi-operator image retargeting algorithm [10].

Image warping has been applied to image retargeting and it has generated a whole new family of retargeting algorithms. The idea of image warping and retargeting is to achieve an optimal deformation of the image that preserves salient regions by computing the global minima of a quadratic energy function [11,12,13]. The shift-map method, first introduced by Pritch, *et al.*[5], aims to manipulate values in the 'shift-map' to rearrange the image content, and this particular feature makes it ideally suited for image retargeting [5,14]. Very recently, Ding, *et al.* [1] propose another shift-map based image retargeting method, by introducing an 'importance filter' to compute the shift-map by integrating shift-map gradient across scanline.

3 Our Algorithm

In this section, we detail the proposed 3D image retageting algorithm based on shift-map and importance filtering. Without loss of generality, we will assume that retargeting will decrease the image width: given an input image I of width

W and height H, the retargeted image has width $W' < W$. The inputs are a pair of stereo images I_L, I_R and the disparity map I_d. Using disparity map, the set of occluded pixels on both images can be determined. Furthermore, we assume that targeted range Ω for the disparity values is also given, and the algorithm simultaneously retargets the stereo image pairs and preserves the disparity values, i.e., the retargeted images have the desired size and the resulting disparity values lie in the desired range Ω.

The shift-map was first introduced in [5] and we refer the reader to the original paper for more specific details. We will consider the shift-map $S(x, y)$ as a $W \times H$ real-valued matrix, elements of which define the locations of the pixels after retargeting. $S(x, y)$ essentially provides a transformation (deformation) between $W \times H$ and $W' \times H$-image grids, and the retargeted image is then generated by image interpolation according to the 2D deformation given by $S(x, y)$. There are two basic constraints for $S(x, y)$: boundary constraint $S(1, y) = 1$ and $S(W, y) = W'$ and monotonicity constraint that $S(x, y) \leqslant S(x + 1, y)$ for maintaining the scanline order. Further constraints can be readily formulated for $S(x, y)$. For instance, suppose (x_L, y) and (x_R, y) are a pair of corresponding pixels from a pair of rectified stereo images with disparity value $d = x_L - x_R$. This particular disparity value can be preserved by enforcing the constraint $S_L(x_L, y) - S_R(x_R, y) = d = x_L - x_R$. In actual computation, it is the gradients of the shift-maps that are computed directly from image data and the final shift-map is obtained by integrating the gradient using importance filtering [1].

This paper proposed a new algorithm for both stereo image retargeting and disparity adjustment. Three main novel features of our algorithm are illustrated as follows:

1. We have incorporated two different tasks, stereo image retargeting and disparity adjustment, into one framework. Our final shift-map is capable of both retargeting and disparity adjusting, and is easily computed by adding the shift-map for retargeting (section 3.1) and adjustment-map (section 3.2) together.
2. We have improved the importance filtering [1] algorithm for stereo image retargeting, with special concern of occluded pixels and disparity preservation.
3. We have invented another way to adjust the disparity values for better viewing experience other than [4] by using trimming and adjustment-map.

3.1 Improved Importance Filtering for Stereo Images

Pixel Saliency and Shift-Map Gradient. Denote $G = \nabla_x S$ the horizontal gradient of the shift map S. $G(x, y) = 1$ will indicate that no deformation occurs at (x, y), while $G(x, y) = 0$ means that (x, y) will be removed. Other values of $G(x, y) \in (0, 1)$ represent different amounts of shrinkage at the given pixel. For occluded pixels, the shift-map gradient is always set to one and the formula for $G(x, y)$ is given by

$$G(x, y, E_s(x, y)) = \begin{cases} C_y \cdot e^{-2\alpha^2 \left(\frac{1-E_s(x,y)}{\sigma^2}\right)^2} & (x, y) \notin O \\ 1 & (x, y) \in O \end{cases} \quad (1)$$

where $\alpha = W'/W$, E_s is the normalized saliency map and O is the set occluded pixels. σ is a tuning parameter whose value is between 0.2 to 0.5. To satisfy the boundary constraint, the sum of the gradient in each row should be W'

$$\sum_{(x,y)\in O} 1 + \sum_{(x,y)\notin O} G(x, y, E_s(x, y)) = W' \qquad \forall y, \tag{2}$$

and this determines the normalization constant C_y for row y in Equation (1)

$$C_y = \frac{(W' - \sum_{(x,y)\in O} 1) \cdot \beta}{\sum_{(x,y)\notin O} G(x, y, E_s(x, y))}. \tag{3}$$

Since importance filtering could not enforce the boundary constraint with occluded pixels incorporated, in Equation (3) a balance factor β is added to adjust the sum of shift-map gradient. The alue of β is related to the amount of occlusion, and for most cases $\beta \in [0.8, 1]$.

Improved Importance Filtering. Importance filtering [1] computes the shift-map S by weighted integration of a given gradient map G. By setting the span of the filter to be one quarter of the image height, the up/bottom neighboring shift-map values will be similar. With the extra constraint of preserving occluded areas, our improved importance filter is as follows:

$$S(x, y) = \begin{cases} \frac{\sum_{j=y-r}^{y+r} w(x,j)[S(x-1,j)+G(x,j)]}{\sum_{j=y-r}^{y+r} w(x,j)} & (x, y) \notin O \\ S(x-1, y) + 1 & (x, y) \in O \end{cases} \tag{4}$$

with the weight $w(x, y)$ given by

$$w(x, y) = e^{E_s(x,y)}. \tag{5}$$

Equation (4) implies that shift-map values of non-occluded pixels will be computed from a one dimension filter of size $(2r+1)$ by taking the weighted average of the predicted values, which is the sum of shift-map value from the previous column and the shift-map gradient. Meanwhile, for occluded pixels, we use the same assumption as [2] that no size shrinkage would happen within these pixels, thus their shift-map values are computed directly by adding 1 to shift-map values of their left neighbours.

We use a two-step approach to obtain the left and right shift-map S_L and S_R: First we compute the left and right shift-maps independently by Equation (4). Second, we refine the shift-map values as follows: Suppose p and q are a pair of corresponding pixels in the left and right images, and f_p and f_q are their preliminary shift-map values from the first step, then their final shift-map values could be computed by (6):

$$\begin{cases} S_L(p) = \frac{1}{2}(f_p + f_q + d_{p,q}) \\ S_R(q) = \frac{1}{2}(f_p + f_q - d_{p,q}) \end{cases} \tag{6}$$

Equation (6) ensures that information from both I_L and I_R are utilized to compute shift-maps, and disparity values are explicitly preserved. Shift-map value of occluded pixels will be filled in according to equation (4) afterwards.

3.2 Disparity Adjustment

For disparity adjustment, the goal is to map the original disparity value $d \in \Omega' = [d'_{min}, d'_{max}]$ to a targeted range $d \in \Omega = [d_{min}, d_{max}]$. A direct but unsatisfactory solution is to find a direct mapping function $d = \varphi(d')$, and warp the image by moving corresponding pixel pairs to adjust the disparity value. However, if the difference between Ω and Ω' is large, this method invariably requires large pixel deformation that can cause serious image distortions. Instead, we proposed an alternative solution that follows the idea of trimming and mapping using the non-linear mapping function $d = \varphi(d' - d_t)$.

The basic idea for dealing with large difference in disparity values is to first correct it with a large constant shift (d_t) in disparity values followed by a non-linear mapping φ to map from $\Omega'_n = [d'_{nmin}, d'_{nmax}] = [d'_{min} - d_t, d'_{max} - d_t]$ to Ω. The large constant shift in disparity values can be accomplished, specifically for image retargeting applications, by trimming away image regions near the borders. With a properly chosen value of d_t, Ω'_n and Ω will have significant overlaps, which minimizes the amount of pixel movement (usually to less than 10). We note that trimming is peculiar to image retargeting and under a different context (e.g., [4]), large deformation and the resulting distortion are usually unavoidable.

Trimming. Trimming is done by cropping image columns. If d_t is larger than zero, then the leftmost d_t columns in I_L and the right most d_t columns in I_R are cropped, and vice versa. This is reasonable because in most cases salient regions in the image are usually away from the boundary. Furthermore, the number of cropped columns also contribute to the total size reduction. The offset constant d_t is closely related to the disparity ranges Ω and Ω'. In our implementation, d'_{min} and d'_{max} in Ω' are fixed to be the disparity value ranked exactly on the top 1% and bottom 1% of all disparity values in I_d (disparity values at occluded pixels do not count), and d_t is set to $d_{min} - d'_{min}$ to match Ω and Ω' roughly.

Adjustment-Function $\phi(d_n)$ and Adjustment-Map. The disparity mapping functions are fomulated using operators in [4], and by applying different operators (linear/nonlinear, continuous/discontinuous), we are able to adjust the disparity values with great flexibility. In our method, the nonlinear mapping function $\varphi(x)$ is calculated as

$$\varphi(d_n) = d'_{min} + \int_{d_{nmin}}^{d_n} \varphi'(x)dx = d'_{min} + \frac{1}{n}\int_{d_{nmin}}^{d_n} h(x)\,dx \tag{7}$$

In (7), $h(d_n)$ is a histogram counting the number of pixels with a certain disparity value d_n. The value of $h(d_n)$ is regarded as proportional to the first derivative of $\varphi(d_n)$, and n is the normalization factor to ensure that

$$\begin{aligned} \varphi(d_{nmin}) &= \varphi(d_{min} - d_t) = d'_{min} \\ \varphi(d'_{nmax}) &= \varphi(d_{max} - d_t) = d'_{max}. \end{aligned} \tag{8}$$

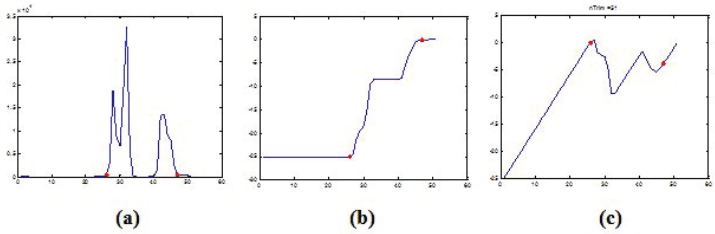


Fig. 2. This figure shows how adjustment-function is calculated. $\phi(d)$, d_{min} and d_{max} are labelled by the red dots. (a) the histogram of the disparity value, (b) the mapping function $\varphi(d_n)$ by integrating and normalizing the histogram, (c) the adjustment-function $\phi(d_n) = d_n - \varphi(d_n)$.

We define the adjustment-function that quantifies the amount of additional pixel movement needed for a pixel with disparity d_n as $\phi(d_n) = d_n - \varphi(d_n)$. An example is shown in Figure 2.

In our algorithm, nonlinear disparity adjustment is done by adding a novel *adjustment-map* to the shift-map. Suppose $p : (x_L, y)$ and $q : (x_R, y)$ are a pair of correspondent pixels in the left and right image, where x and y is the column and row coordinate, the disparity value between them are defined as $d_{p,q} = x_L - x_R$. The disparity adjustment model then could be illustrated as follows:

$$d'_{p,q} = \varphi(d_{p,q}) = d_{p,q} - \phi(d_{p,q}) = x_L - x_R - \phi(d_{p,q}) \tag{9}$$

To minimize the distortion caused by disparity adjustment, the amount of shift is evenly distributed to p and q. We define $\phi_L(d) = \phi_R(d) = \phi(d)/2$, then the shifting model is:

$$\begin{cases} x'_L = x_L - \phi_L(d_{p,q}) \\ x'_R = x_R + \phi_R(d_{p,q}) \end{cases} \tag{10}$$

The adjustment-map is constructed by applying $\phi(d)$ to all the pixels:

$$A_L(p) = \begin{cases} \phi_L(d_{p,q}) & p \in I_L \wedge p \notin O_L \\ 0 & p \in I_L \wedge p \in O_L \end{cases} \tag{11}$$

$$A_R(q) = \begin{cases} \phi_R(d_{p,q}) & q \in I_R \wedge q \notin O_R \\ 0 & q \in I_R \wedge q \in O_R \end{cases} \tag{12}$$

O_L and O_R are the set of occluded pixels in I_L and I_R. Once the shift-maps S_L and S_R are obtained, the final shift-map for both retargeting and disparity adjustment could be generated directly by:

$$\begin{cases} S_{Ladj} = S_L - A_L \\ S_{Radj} = S_R + A_R \end{cases} \tag{13}$$

By interpolation from the final shift-map S_{Ladj} and S_{Radj}, the retargeted and disparity adjusted stereo image pair I_{Lout} and I_{Rout} could be obtained.

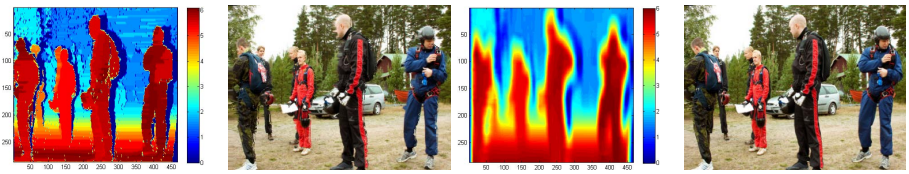


Fig. 3. Effectiveness of Adjustment-Map Smoothing The top row displays the un-smoothed adjustment-map and visible defects can be observed around human torsos and legs. The bottom row shows the smoothed adjustment-map with a much better improvement in visual quality.

Smoothing the Adjustment-Map. Applying the adjustment-map from (11) and (12) directly without considering smoothness usually results in significant visual defects, and experimental results have suggested that smoothing the adjustment-map can provide significant improvement in the quality of the final result. Horizontal smoothing is preformed on the gradient domain of the adjustment-map, $\nabla_x A$, by smoothing the pixels with absolute gradients deemed too large. The first step is to scan the pixels in $\nabla_x A$ row by row from top to bottom. Once we spot a pixel (x_0, y_0) where $|\nabla_x A(x_0, y_0)| > \tau$, the following horizontal smoothing function is applied, while keeping the sum of each row in $\nabla_x A$ unchanged:

$$\begin{aligned} &\nabla_x A(x, y_0) = \tfrac{1}{2r+1} \textstyle\sum_{i=x_0-r}^{x_0+r} \nabla_x A(i, y_0) \\ &x = x_0 - r, x_0 - r + 1, \cdots, x_0 + r \end{aligned} \tag{14}$$

In (14), r defines the size of the horizontal smoothing window and in our experiment it is set to $ceil(\frac{|\nabla_x A(x,y)|}{\tau})$. τ denotes the amount of smoothing. Larger τ will enhance adjustment accuracy, but impair visual effects. A reasonable value of τ is found to be 0.3. The horizontally-smoothed adjustment-map A_{sh} can be easily recomputed by integrating the smoothed version of $\nabla_x A$. Vertical smoothing is done easily by applying a filter with the size of $1 \times (2r + 1)$ to A_{sh} so that each points will take the average of both itself and its r upper/lower neighbors. Meanwhile, since we smooth the adjustment-map instead of the disparity map, the depth relationship would not be messed up.

Figure 3 shows the significant improvement in the output quality after horizontal and vertical smoothing. It is inevitable that after smoothing the disparity values for some pixels will no longer equal to $\varphi(d_{p,q})$. But since only pixels with abrupt changes in *adjustment-map* will be affected, the trade-off between inaccurate $\varphi(d_{p,q})$ and image quality clearly favors the latter.

4 Experimental Results and Discussions

We have implemented the proposed algorithm in MATLAB and C without serious code optimizations, and it takes less than 3s to generate a 300×400 retargeted stereo image pairs on a 3.4GHz computer. In comparison, Basha *et al.*'s

method [2] will take around 20s to iteratively carve all the seams. In the following subsection, we will first demonstrate the correctness of our stereo image retargeting algorithm without disparity adjustment, and then demonstrate how viewing effect could be enhanced using with disparity adjustment.

4.1 Stereo Image Retargeting

We test our algorithm on the stereo retargeting dataset from [2] and *Middlebury* stereo datasets. Since computing the saliency map directly from intensity values has always been tricky, in this experiment, we do not use any particular saliency map in order to avoid degrading the quality of the output images with suboptimal saliency map. Instead, we use the normalized disparity map combined with the image gradient as the saliency map to compute the two gradient maps G_L and G_R required for the importance filtering.

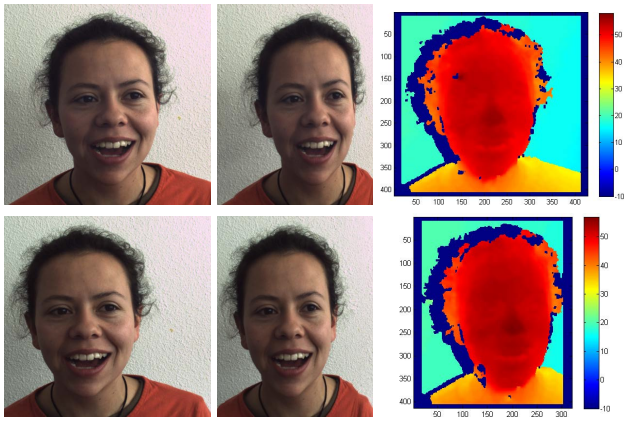
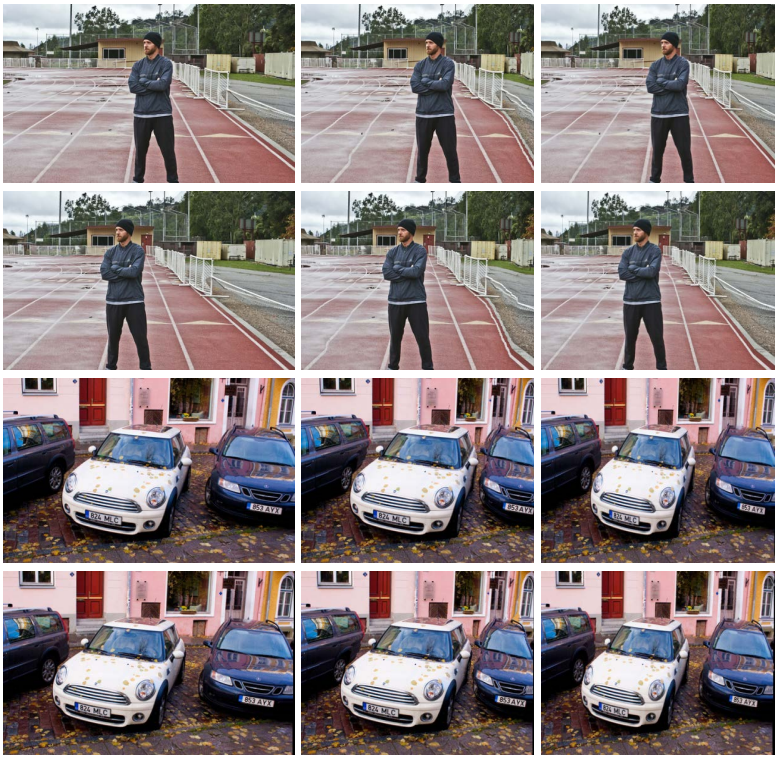


Fig. 4. Correctness of the Proposed Algorithm Columns from left to right: original pair of stereo images, retargeted pair of stereo images, and the disparity map computed before and after retargeting using Libelas algorithm [15]. The size reduction in width is 25%.

Figure 4 shows the correctness of our algorithm for regular stereo image retargeting problem (without disparity adjustment). We note that the face is kept intact with minimal amount of distortion. Furthermore, more detailed features such as the shape of eyebrows and hairs are also well-preserved. In our algorithm, the disparity values are kept approximately exact, as the algorithm manipulates the positions of each pair of corresponding pixels to force them to assume a particular disparity value. The third column of Figure 4 shows the disparity maps computed by ELAS algorithm [15] that verifies the agreement between the disparity values before and after retargeting. In [2] they already proved that it is not sufficient to preserve the depth(disparity) by uniform scaling, thus we do not incorporate uniform scaling into the comparisons.



Original Image Basha et.al. [2] Our Result

Fig. 5. Comparison Between Our Method and [2] Image width is decreased by 20% in this example. **Left to Right**: Original image, results from[2], our results. Note the straight track lines produced by the proposed algorithm in the first example.

We have also compared our algorithm with the current state-of-art method [2]. Results shown in Figure 5 indicate that our method can provide results that are visually more appealing with better preservation of image structures. In the 'man' example (first two rows), the track lines on the ground are clearly not straight in their result because non-horizontal/vertical lines are difficult to preserve using seam carving, which can be better handled using an interpolation-based method such as ours. The 'car' example (third and forth row) further demonstrates the difference between our algorithm and [2]. Seam carving can remove more pixels in small and twisted areas, like the gap between the white Polo in the center and the blue car on the right. However, the structural integrality of an object cannot be guaranteed once a seam has cut through an object. Importance filtering given in Equation (4)([1]) can make the neighboring pixels in the same column to have similar shift-map values, and it helps to decrease the amount of unnecessary distortion. In the 'car' example, the back end of the black wagon on the right suffers serious and visible distortion after seam carving, but it is essentially intact in our result.

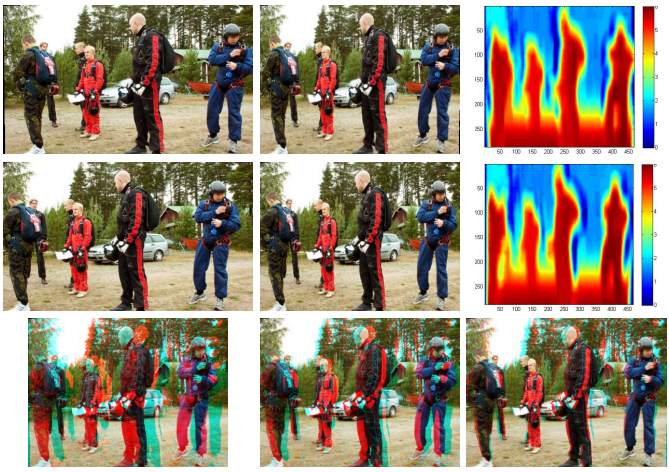


Fig. 6. Effectiveness of Disparity Adjustment Top two rows (left to right): original stereo images, retargeted stereo images, adjustment-maps. Bottom row (left to right): Anaglyph with no disparity adjustment, Anaglyph with trimming only, Anaglyph with trimming and disparity adjustment. (Best viewed in color and red-cyan 3D glasses.)

4.2 Disparity Adjustment

The term *comfort zone* discussed in Section 3.1 and [4] implies that not all the stereo images are display-ready. In our experiment, we have observed that for a 23-inch LED screen, reasonable disparity values for display is between -15 (behind the screen) and +5 (in front of the screen). For the disparity values outside this range, the objects will appear too near or too far away, which cause unpleasant physiological responses such as dizziness. Moreover, it is reasonable to place the important foreground objects at the screen level (with disparity value close to 0) for better visual effect.

Two examples of our disparity adjustment algorithm are shown in Figure 6 and 7. The original images are not ready for display because the disparity values are too large. People in the anaglyph made by direct combination of the retargeted stereo images (without adjusting disparity values) appear too near to the viewer. After trimming, the entire scene is uniformly moved away from the viewer; however, it is still not optimal since the depth difference between the foreground and background is too large for comfortable viewing. By adjusting the disparity values, we compress the disparity range by moving people behind the screen, while keeping the background disparity unchanged. The detail and pattern of pixel shifts are shown in the adjustment-map, which indicates that a large amount of pixel movement occurs in the foreground region while the background mostly stays in the original position.

Inevitably, for the areas where abrupt changes occur in the adjustment-map (e.g., the region around the man's head), visible distortion appears because of the

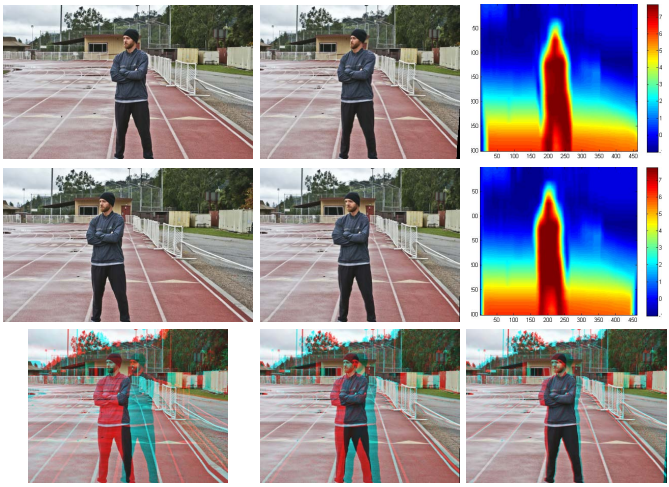


Fig. 7. Another Example of Disparity Adjustment For the region where abrupt differences occur in the non-smooth adjustment-map, distortion becomes visible due to smoothing. However, this visual defect is not particularly noticeable by human while viewing the images in 3D. By comparing the three anaglyphs in the bottom row, the one with trimming and nonlinear disparity adjustment clearly provides the best viewing result. (Best viewed in color with red-cyan 3D glasses.)

inconsistent pixel movement due to smoothing. Similar problem has also been reported in [4] and it was considered to be unavoidable without any high-level feature descriptors or further constraints. However, it is interesting to see that this distortion is not particularly noticeable when viewed in 3D. For evaluation of pleasing viewing in 3D we actually hold a user study among a small group of 7 people, and 6 of them support the idea that the retargeted 3D images with trimming+nonlinear disparity adjustment provide the best viewing experience in 3D among the three alternatives shown in the figures.

5 Conclusions

We have presented a novel stereo image retargeting algorithm that simultaneously computes the retargeted images and adjusts the disparity values. We design the algorithm based on the novel principle that, besides preserving image saliency, 3D image retargeting should also consider adjusting the disparity values for maximizing viewing experience. This later requirement is a novel notion peculiar to 3D images and it has hitherto not been considered in previous image retargeting work. We have successfully extended shift-map and importance filtering to stereo images, and the proposed algorithm integrating both image retargeting and disparity adjustment in one single algorithmic framework. We have evaluated the proposed method with several well-known stereo image pairs and the results compared favorably with the state-of-the-art method [2] that uses seam carving, both in terms of output image quality and running time.

References

1. Ding, Y., Xiao, J., Yu, J.: Importance filtering for image retargeting. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 89–96. IEEE (2011)
2. Basha, T., Moses, Y., Avidan, S.: Geometrically consistent stereo seam carving. In: ICCV, pp. 1816–1823 (2011)
3. Howard, I.: Seeing in depth: vol. 1, Basic mechanisms. University of Toronto Press (2002)
4. Lang, M., Hornung, A., Wang, O., Poulakos, S., Smolic, A., Gross, M.: Nonlinear disparity mapping for stereoscopic 3d. ACM Transactions on Graphics (TOG) 29, 75 (2010)
5. Pritch, Y., Kav-Venaki, E., Peleg, S.: Shift-map image editing. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 151–158. IEEE (2009)
6. Setlur, V., Takagi, S., Raskar, R., Gleicher, M., Gooch, B.: Automatic image retargeting. In: Proceedings of the 4th International Conference on Mobile and Ubiquitous Multimedia, pp. 59–68. ACM (2005)
7. Santella, A., Agrawala, M., DeCarlo, D., Salesin, D., Cohen, M.: Gaze-based interaction for semi-automatic photo cropping. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 771–780. ACM (2006)
8. Avidan, S., Shamir, A.: Seam carving for content-aware image resizing. ACM Transactions on Graphics (TOG) 26, 10 (2007)
9. Rubinstein, M., Shamir, A., Avidan, S.: Improved seam carving for video retargeting. ACM Transactions on Graphics (TOG) 27, 16 (2008)
10. Rubinstein, M., Shamir, A., Avidan, S.: Multi-operator media retargeting. ACM Transactions on Graphics (TOG) 28, 23 (2009)
11. Wolf, L., Guttmann, M., Cohen-Or, D.: Non-homogeneous content-driven video-retargeting. In: IEEE 11th International Conference on Computer Vision (ICCV 2007) pp. 1–6. IEEE (2007)
12. Wang, Y., Tai, C., Sorkine, O., Lee, T.: Optimized scale-and-stretch for image resizing. ACM Transactions on Graphics (TOG) 27, 118 (2008)
13. Guo, Y., Liu, F., Shi, J., Zhou, Z., Gleicher, M.: Image retargeting using mesh parametrization. IEEE Transactions on Multimedia 11, 856–867 (2009)
14. Kav-Venaki, E., Peleg, S.: Feedback Retargeting. In: Kutulakos, K.N. (ed.) ECCV Workshops 2010, Part II. LNCS, vol. 6554, pp. 145–155. Springer, Heidelberg (2012)
15. Geiger, A., Roser, M., Urtasun, R.: Efficient Large-Scale Stereo Matching. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 25–38. Springer, Heidelberg (2011)

Object Templates for Visual Place Categorization

Hao Yang and Jianxin Wu

School of Computer Engineering, Nanyang Technological University, Singapore
{lancelot365,wujx2001}@gmail.com

Abstract. The Visual Place Categorization (VPC) problem refers to the categorization of the semantic category of a place using only visual information collected from an autonomous robot. Previous works on this problem only made use of the global configurations observation, such as the Bag-of-Words model and spatial pyramid matching. In this paper, we present a novel system solving the problem utilizing both global configurations observation and local objects information. To be specific, we propose a local objects classifier that can automatically and effectively select key local objects of a semantic category from randomly sampled patches by the structural similarity support vector machine; and further classify the test frames with the Local Naive Bayes Nearest Neighbors algorithm. We also improve the global configurations observation with histogram intersection codebook and a noisy codewords removal mechanism. The temporal smoothness of the classification results is ensured by employing a Bayesian filtering framework. Empirically, our system outperforms state-of-the-art methods on two large scale and difficult datasets, demonstrating the superiority of the system.

1 Introduction

Place categorization is described as the problem of assigning a semantic label (bedroom, bathroom, kitchen, etc.) to a specific place. It is different from place recognition, which refers to recognizing a place that has been previously visited. To be specific, a place categorization system should be able to assign label "Kitchen" to a kitchen that has never been visited previously; while a place recognition system should assign label "Kate's Kitchen" to the very same kitchen that has been traveled before but maybe in different conditions (e.g. illumination, weather). Place categorization is an important topic in both computer vision and robotics. In computer vision, the semantic category can exert strong prior on the objects it may contain [1]. Hence, being able to label the semantic category of a place should boost the performance of object recognition and visual search. In the robotics area, successful place categorization will lead to better human-robot interaction and improve location awareness of robots.

Visual Place Categorization (VPC), according to [2], refers to "the identification of the semantic category of a place using visual information collected from an autonomous robot platform". The emphasis on "visual" makes VPC different from place categorization problem addressed by the robotics community [3, 4],

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 470–483, 2013.

which usually make use of other sensory data such as laser. The emphasis on "autonomous" and "image sequences" makes it different from the scene categorization/recognition problem in computer vision that focuses on recognizing the semantic category of a single image captured by a person. Despite the differences, VPC is still closely related to these topics. The breakthrough in VPC will help to improve the performance in place categorization and scene categorization, and vice versa.

Based on the nature of VPC, *extracting useful information to make good prediction for single frame* and *integrating temporal predictions over time* are two main challenges in order to solve the problem. To be specific, the two challenges can be expressed as how can one make prediction for a frame that is representative for certain category, like a frame from living-room with sofa and television; and how can one make prediction for a frame that does not have specific features or objects, like a frame only with a wall, using previous frames.

In [2], these two challenges were addressed by a discriminative approach with the Bag-of-Words model based on the CENTRIST descriptor [5] and a Bayesian Filtering framework. A state-of-the-art work on VPC [6] tackled these two problems by a generative classification system and an online Bayesian change point detection framework.

Both of these works utilizing only the global configurations observation, which refers to extracting information using dense grid features and spatial pyramid without spending efforts on detecting or recognizing the objects of a frame. Although objects are very important in place/scene categorization, locating and recognizing key objects of a semantic category in a single frame may be more difficult than the VPC problem itself. And, manual annotating object in images or frames (for training object models and testing) requires too many resources. Therefore, in this paper, we mainly make these contributions:

- We propose an object templates classification method to utilize the local objects information. This method can automatically select important objects or key features of a semantic category based on training data by structural similarity SVM [7], and use them as templates for future Local Naive Bayes Nearest Neighbors [8] based classification on testing images. Combining this method with the global configurations approach, the accuracy is significantly improved. We show the results in Section 4.
- We further improve the global configurations approach. First, we employ the histogram intersection codebook, instead of linear codebook, since histogram intersection codebook has been proven to achieve higher classification accuracy than Euclidean distance codebook [9]. We also employ a noisy codewords removal mechanism that removes noisy codewords with small difference between intra-category similarity and inter-category similarity .

The paper is organized as follows: We introduce related works of this research topic in Section 2. In Section 3 we demonstrate the details of our system and the principles behind it. Section 4 shows the experimental results on two large and difficult datasets. Finally, Section 5 draws conclusion and discusses future improvements.

2 Relates Works

In this section, we briefly review the most relevant literature of the Visual Place Categorization problem. As we have mentioned, VPC is closely related to research topics like place categorization, scene categorization and place recognition. Hence we also review recent research efforts on these fields.

Place Categorization and Recognition. One of the early works on these two problems is a context-based vision system using "gist" feature by Torralba *et al.* [1]. [1] achieved high accuracy on recognizing places, while for categorizing, [1]'s accuracy was not satisfactory. Pronobis *et al.* [3, 4, 10, 11] extensively studied place recognition and categorization. [10,11] focused on addressing place recognition under various weather and illumination conditions. Later works [3,4] tried to solve the place categorization problem in a hierarchical and multi-modal way. Pronobis *et al.* also published several datasets in order to set up standard benchmark [4, 12]. In [13], Ullah *et al.* used a SVM-based method to address place recognition and categorization on the dataset COsy Localization Database (COLD) [12].

Scene Categorization/Recognition. This problem is quite similar to place categorization besides that it usually concern about recognizing single images captured by a person while the former two problems focus on image sequences or videos taken by a robot. In this case, since all the training and testing images are informative and independent, the problem of integrating temporal predictions is not a concern. There are plenty of works in computer vision dedicated to this problem. In [14], Lazebnik *et al.* employed spatial pyramid matching to recognize scene categories. In [15], Quattoni and Torralba described an indoor scene as a "root" containing the holistic information and movable "regions of interest"(ROIs). Their idea is quite similar to our classification system, which also tried to utilize both global information and local ROIs, except that their work relies on manual annotations while our system discovers key local features automatically. Recently, [16,17] also tried to address the problem with the same global + local fashion. [16] used deformable part-based models (DPM) with latent SVM [15] to automatically select ROIs. [17] used deformable part-based models to explicitly locate and recognize objects, then used Adaboost to merge weak hypotheses from each object to a strong hypothesis of a scene.

Visual Place Categorization. This problem was first described by Wu *et al.* [2]. This paper also published a new dataset on VPC along with a Bag-of-Words model algorithm to solve the problem. In [6, 18], the VPC problem was addressed in a novel way based on a fully probabilistic framework. They used a Bayesian change point detection algorithm to detect abrupt changes in image sequences and a Bag-of-Words model to measure place labels. The system was tested on the VPC dataset and the result matched state-of-the-art. The other work by Ranganathan an Lim [19] addressed the problem of categorizing areas in maps with given labels. This work is a real life application example of VPC.

3 A Novel Classification System for Visual Place Categorization

In this section, we introduce detail configurations of our system and the principles behind it. We divide this section into three subsections. In Section 3.1, we present the framework of the system. We also introduce the Bayesian filtering approach that aims to ensure temporal smoothness in this section. In Section 3.2, we present the global configurations approach by briefly reviewing [2] and describing a few improvements we make on it. In Section 3.3, we demonstrate how we utilize local information by selecting object templates with structure similarity SVM [7] from training images automatically and effectively, and classifying test images with these templates using Local NBNN [8].

We formulate the VPC problem as follows: Given a sequence of images taken by a conventional camera mounted on an autonomous robot, for each image of that sequence, which we call a frame, we need to assign a label to it. Assuming that we have L categories and they are represented as $C_1, C_2, ..., C_L$, we denote the category label of a frame t as X_t, and X_t should be taken from C_1 to C_L. We need to calculate the probability distribution of X_t, given a sequence of observation $Z_1, Z_2, ..., Z_t$. In other words, we estimate $P(X_t|Z_{1:t})$, here $Z_{1:t}$ represents the observations from time step 1 to t. Our system utilizes two kinds of observations: global configurations observation Z_t^g and local object templates observation Z_t^l. Z_t^g and Z_t^l are assumed as independent if the label X_t is known.

3.1 Framework of the Classification System

Our classification system mainly consists of two parts, the observation part perceives the global and local information and the Bayesian filtering part integrates prediction of the current frame with predictions from previous frames. The system framework is shown in Figure 1. Detailed description of the observation part can be found in Section 3.2 and 3.3. We first introduce th Bayesian filtering process here.

Given that we are dealing with image sequences, and we have no knowledge about whether a frame is representative of certain category or not, it is important to integrate information from many frames. Therefore, we employ the Bayesian filtering process to exploit image history in order to effectively integrate information. This process is also used in [2].

We first assume a Markovian property between frames, that is, $P(X_t|X_{1:t-1}) = P(X_t|X_{t-1})$. Thus, the Bayesian filtering process is determined by:

- The prior distribution $P(X_0)$, which is uniform since we assume no knowledge about the environment at the beginning;
- The category transition distribution $P(X_t|X_{t-1})$, which is set to be 0.99 if $X_t = X_{t-1}$ to reflect the fact that consecutive frames are likely to have the same semantic label;
- The observation distribution $P(Z|X)$. Since we assume Z_t^g and Z_t^l are independent when X_t is known, we can get $P(Z_t|X_t) = P(Z_t^g|X_t)P(Z_t^l|X_t)$.

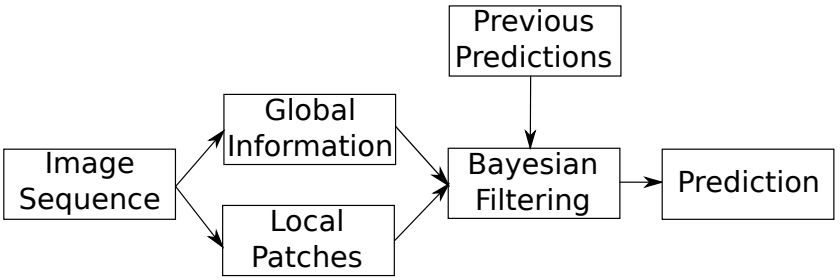


Fig. 1. Framework of the Classification System for VPC

According to [2], we can then evaluate $P(X_t|Z_{1:t})$ by:

$$P(X_t|Z_{1:t}) \propto P(Z_t|X_t)P(X_t|Z_{1:t-1}) \tag{1}$$

and

$$P(X_t|Z_{1:t-1}) = \sum_{i=1}^{L} P(X_t|X_{t-1} = C_i)P(X_{t-1} = C_i|Z_{1:t-1}). \tag{2}$$

We then classify the category of a frame t to be $\arg\max P(X_t|Z_{1:t})$. The missing part of the system is how to get the global configurations observation $P(Z_t^g|X_t)$ and the local object templates observation $P(Z_t^l|X_t)$, for which we propose in detail in the next two subsections.

3.2 Global Configurations Observation

For the global configuration $P(Z_t^g|X_t)$, we inherit the Bag-of-Words model from [2] and make a few improvements. We first divide each image into $4 \times 4 = 16$ sub-windows, and extract a CENTRIST [5] descriptor from each sub-window. Then, we apply k-means clustering to generate one visual codebook for every sub-window, with codewords 1 to K so that each sub-window can be represented as an integer from 1 to K. Hence, the whole image is represented by a 16 dimensional vector:

$$Z_t^g = (z_{t,1}^g, z_{t,2}^g, ..., z_{t,16}^g), \tag{3}$$

and the posterior probability $P(X_t|Z_t)$ is estimated by

$$P(X_t|Z_t^g) \propto P(Z_t^g|X_t)P(X_t) = \prod_{i=1}^{16} P(z_{t,i}^g|X_t)P(X_t). \tag{4}$$

We assume uniform prior class distribution here and $P(z_{t,i}^g|X_t)$ can be easily estimated from the training data.

We make several modification to improve the global configuration observation. First of all, histogram intersection based codebook is used instead of the linear

codebook, since histogram intersection codebook has exhibited higher categorization accuracy than euclidean based linear codebook [9]. Given a histogram $\mathbf{h} = (h_1, \ldots, h_d) \in \mathbb{R}_+^d$, representing a sub-window (CENTRIST descriptor) in this case, the histogram intersection kernel κ_{HI} is defined as:

$$\kappa_{\mathrm{HI}}(\mathbf{h}_1, \mathbf{h}_2) = \sum_{i=1}^{d} \min(h_{1i}, h_{2i}). \tag{5}$$

Since κ_{HI} is a valid positive definite kernel, there exists a mapping ϕ such that $\kappa_{\mathrm{HI}}(\mathbf{h}_1, \mathbf{h}_2) = \phi(\mathbf{h}_1)^T \phi(\mathbf{h}_2)$. The histogram intersection kernel visual codebook evaluates the kernel distance, i.e. the distance between $\phi(\mathbf{h}_1)$ and $\phi(\mathbf{h}_2)$, while linear codebook only make used of the Euclidean distance between $\mathbf{h}_1$ and $\mathbf{h}_2$. Therefore, the HIK codebook gives better classification results for vision problems. The HIK codebook is generated following Algorithm 1 in [9].

Secondly, we employ a noisy codeword removal mechanism. For a visual codebook consisting of many codewords, unavoidably, there exist some codewords that are not useful, even harmful for the classification, since the codebook is generated in an unsupervised manner. Therefore, we want to identify these codewords and remove them for better classification results.

We consider codewords that cannot distinguish intra-class difference and inter-class difference well as noisy. To be specific, for each of the codeword k, we calculate the histogram intersection distance d_{HI} between every two feature descriptors $\mathbf{h}_i^k$ and $\mathbf{h}_j^k$ mapped to the codeword. If the $\mathbf{h}_i^k$ and $\mathbf{h}_j^k$ are from the same category, we accumulate $d_{\mathrm{HI}}(\mathbf{h}_i^k, \mathbf{h}_j^k)$ into sum_{intra}, otherwise we accumulate it into sum_{inter}. Finally, we normalize sum_{intra} and sum_{inter} by dividing them by the intra-category descriptor pairs number and inter-category descriptor pairs respectively to get mean_{intra} and mean_{inter}. If the difference of these two means is small, i.e. $|\mathrm{mean}_{inter} - \mathrm{mean}_{intra}| < \epsilon$, then we consider the expressive power of the codeword to be weak and remove it. Experiments show that these two improvements give better result, *especially for the categories with less frames.*

3.3 Structure Similarity SVM Based Object Templates Classification

It is known that key features and salient objects of a semantic category are vital for place categorization and scene understanding [15]. As human, objects are the main information source for us to judge the semantic category of a place. [2] used a global configurations approach and did not make use of local objects information. Moreover, according to [20], the employment of bag-of-words model hurts the discriminative power of the feature descriptors. Therefore, we believe that adding non-parametric object classifiers will improve the classification accuracy.

However, it is possible that locating and recognizing key objects of a semantic category in a single frame may be more difficult than the VPC problem itself, and manual annotations cost a lot of time and heavy human labors. Hence, we need a way to automatically find out key objects/features of a category using all training frames of that category in a computationally efficient manner.

We make two assumptions: Firstly, since for a category there are usually several thousands of training frames, if we randomly sample some patches from each frame, it is very likely that many of these patches contain the key objects (or part of the key objects) of that category. Secondly, even though the objects of the same kind may have large intra-class variances, there exist some key objects from the same category share similar shape, contour, etc. across different places, e.g. beds and pillows in bedrooms, sofas and televisions in living-rooms, cabinets and sinks in kitchen, tiles in the toilets, and tables and chairs in the dining-room. Based on the latter assumption, it is reasonable to claim that, if a sampled patch is more similar to patches from the same category than patches from other categories, then this patch is more likely to contain a key object or part of a key object of that semantic category.

Usually, the similarity between two patches can be measured by many positive definite kernels such as the linear kernel $\kappa_{\mathrm{LIN}}(x_1, x_2) = x_1^T x_2$, or the histogram intersection kernel $\kappa_{\mathrm{HI}}(x_1, x_2) = \sum_j \min(x_{1j}, x_{2j})$. However, these kernels only consider the pairwise similarity between x_1 and x_2, which may lead to misleading similarity scores, especially when the intra-class variance is large. Unfortunately, for the VPC case, the intra-class variance is usually large since we try to deal with frame from different places. Therefore, a more robust similarity measurement is needed.

If we use examples beyond the pair x_1 and x_2 to assist in similarity computation, the similarity of x_1 and x_2 can be better evaluated, e.g. using background patches [21]. Furthermore, [7] illustrates that the structural similarity can greatly help to remove the ambiguities of pairwise similarity and provide more robust comparison results. Therefore, we use bibliographic coupling strength to measure the structural similarity of patches x_1 and x_2, i.e., counting the number of patches that are nearest neighbors of both x_1 and x_2. We can further encode a local graph as a sparse neighborhood vector, and existing SVM training and testing techniques can be perform directly. [7] show that the structural similarity is more robust for examples with large intra-class variance and for imbalance dataset, making it suitable for our usage.

Suppose we sample p patches from each of the n frames, we have a training set $X = \{x_1, \ldots, x_{pn}\}$ with pn patches. Given two patches x_i and x_j, if x_j is in the k-nearest-neighbors of x_i, we denote it as $x_i \mapsto x_j$. The structural similarity kernel is then defined as:

$$k_{\mathrm{STR}}(x_i, x_j) = \left|\{x : x \in X, x_i \mapsto x \text{ and } x_j \mapsto x\}\right|. \tag{6}$$

Here $|\cdot|$ is the size of a set. [7] has shown that the structural similarity kernel is a valid SVM kernel.

However, existing SVM algorithms cannot be applied directly applied to solve this kind of kernel SVM problem. Alternatively, we can define a neighborhood vector $n(x)$ for each patch x. $n(x)$ is a $pn \times 1$(since we have pn patches) vector where $n(x)_j$ is 1 if x_j is in the k-nearest-neighbors of x, the k here is denoted as k_1, to differ from the nearest neighbors number of the testing phrase. Thus, we have $n(x_1), n(x_2) ... n(x_{pn})$ totally pn neighborhood vectors. $k_{\mathrm{STR}}(x_i, x_j)$ is then $n(x_i)^T n(x_j)$.

By transforming the patch descriptor to a neighbor vector, we can apply existing SVM algorithm (such as LIBLINEAR [22] or PmSVM [23]) to solve the structural similarity kernel problem. This kind of SVM is called the structural similarity SVM.

It is shown in [7] that the decision values of structural similarity SVM can be used to measure how similar a patch compared to other patches from the same category. If we define G as the adjacency matrix of the kNN graph Γ_X built for the training set X, and $s = G^T(\alpha \odot y)$, here α is the Lagrangian dual variables $[\alpha_1, \ldots, \alpha_{pn}]^T$, y is the instance labels $[y_1, \ldots, y_{pn}]^T$ and $\odot$ is the element-wise product, then s_j defined as

$$s_j = \sum_{i=1}^{pn} G_{ij}\alpha_i y_i = \sum_{j:x_i \mapsto x_j} \alpha_i y_i, \tag{7}$$

measures the balance in the local neighborhood $x_i : x_i \mapsto x_j$ related to the patch x_j. A linear SVM using $n(x_i)$ as training examples will lead to a classification boundary:

$$w = \sum_{i=1}^{pn} \alpha_i y_i n(x_i) = G^T(\alpha \odot y) = s. \tag{8}$$

Thus the local balance vector s equals the classification boundary w. In this case, the local balance s contains in effect the signed authority scores learned through the SVM optimization. s_j then reflects the similarity of one patch with other patches in the same category. By our previous assumption, if a sampled patch is more similar to patches from the same category than patches from the other categories, then this patch is more likely to contain a key object or part of a key object of that semantic category. Therefore, the higher s_j, the more likely that this patch contains a key object. For that reason, we select certain number of patches with the highest s_j (or w_j) from each category as the object template for that category.

Fig. 2 shows the training patches with highest s_j in 5 categories of the VPC dataset, which we use as object templates for this dataset. As one can see from the figure, we caught beds and pillows in bedroom, tiles and sinks in bathrooms, cabinets in kitchens, sofa in living-rooms and chairs in dining-rooms. These templates indeed contain the key objects of certain category.

Once we have the templates of each category, classifying one frame is then just measuring the Image-to-Template distances. In other words, we find out the closest template set of the testing frame and assign the templates' label to the frame. The Naive Bayes Nearest Neighbors (NBNN) framework can evaluate this kind of distance effectively. For implementation convenience, we choose to use a recently proposed Local NBNN framework [8], shown in Algorithm 1.

However, the output of Local NBNN algorithm, totals[C], is not a qualified distribution. Thus we use the softmax transformation

$$\frac{\exp(-\text{totals}[C_i])}{\sum_{k=1}^{L} \exp(-\text{totals}[C_k])} \tag{9}$$

Fig. 2. Examples of object templates captured by our system automatically. Row 1 to 5 correspond to templates from bedroom, bathroom, kitchen, living-room and dining-room, respectively.

Algorithm 1. Local NBNN(Q,k_2) [8]

Require: A nearest neighbor indexer for all templates, NN($patch, \#neighbors$).
Require: A lookup table, $Class(patch)$, returning the class of a template patch.
for $i = 1, ..., m$ **do**
  $\{p_1, p_2, \ldots, p_{k_2+1}\} \leftarrow \text{NN}(d_i, k_2 + 1)$
  $dist_B \leftarrow \|d_i - p_{k_2+1}\|$
  for all categories C found in the k_2 nearest neighbors **do**
    $dist_C = \min_{\{p_j | Class(p_j)=C\}} \|d_i - p_j\|$
    totals$[C] \leftarrow$ totals$[C] + dist_C - dist_B$
  end for
end for
Output: totals$[C]$.

to get a normalized distribution, which is $P(X_t|Z_t^l)$. The framework of the object templates classification system is shown in Algorithm 2.

4 Experiments

We tested our system on two large scale and difficult datasets: the VPC dataset [2] and the COsy Localization Database (COLD) [12]. The methodology and parameters are presented here. We implemented our system in C++ with no special optimizations on certain datasets. Since we used FLANN [24] to do the nearest neighbors search and PmSVM [23] to train the neighbor vectors, the training and testing for all six homes in the VPC dataset took about 2 hours, and for 3 sequences of COLD took about 15 minutes, which are acceptable.

Algorithm 2. Object templates classification using structural similarity

Input: n training frames and their corresponding labels, p, t, m, k_1, k_2.
Training:
for each training frame **do**
  Resize it to width 320 and sample p 64×64 patches;
  for each patch **do**
    Generate a visual descriptor for it, denote as x_j, assign the frame label to it as y_j;
  end for
end for
for all training patches x_j **do**
  Create a neighborhood vector with k_1 nearest neighbors among all training patches, denote as $n(x_j)$;
end for
for each category C_i **do**
  Use $(n(x_j), y_j), \forall j$ to train an HIK SVM model with one-vs-all strategy. Each instance is then associated with a decision value w_j;
  Choose t patches with highest decision value w_j and $y_j = C_i$ as object templates
  for this category;
end for
Testing:
for all testing frames **do**
  Sample m patches, generate a descriptor for each patch;
  Get totals[C] using Algorithm 1;
  Use softmax transformation to get the distribution $P(X_t|Z_t^l)$.
end for

The VPC dataset [2] was taken from a conventional camera mounted on a mobile tripod to simulate an autonomous robot. It contains frames from 6 homes and 12 categories. Each homes typically contains several thousands frames with size 1280×720. This dataset is significantly difficult because of the "autonomous" property. There are many frames that are not informative at all. The example image are shown in Figure 3. It reflects the real situation that a robot may come up with.

The COsy Localization Database (COLD) [12] was taken by two robots with a standard camera and an omni-directional camera. It contains frames from 3 labs in 3 different countries and a total number of 12 different categories. Typically, each lab has several sequences in different weather conditions. The image size of each frame is 640×480, which is much smaller that VPC dataset, and it is more diverse than that of the VPC dataset, which was taken from 6 homes all in the Greater Atlanta area (in Georgia, USA).

For the global configurations part, we let codebook centers K to be 50 so that we have 50 visual words for each division. For object templates, there are five parameters: p, t, m, k_1 and k_2. We set $p = 10$, $k_1 = 10$ and $t = 1000$ so that we sample 10 patches from each training frame, find out 10 nearest-neighbors for

Fig. 3. Example images from VPC dataset. As shown here, some of the images are informative (the upper two) while many are not (the lower two).

each patch and select 1000 templates for each category. We then set $m = 24$ and $k_2 = 10$ so that we sample 24 patches from each testing frame, and find out $k_2 + 1 = 11$ nearest-neighbors for each patch among object templates for the Local NBNN classification.

4.1 Testing on Visual Place Categorization *without* Reject Option

In the VPC dataset, there are 5 categories that occur in every home: bedroom, bathroom, kitchen, living-room and dining-room. [2] tested their system on these 5 common categories using the leave-one-out strategy (i.e., they trained on 5 homes and test on the remaining one, and then average the results of 6 homes). We follow their method to test our system, the result is shown in Table 1. As one can see, our improvements on the global approach boost the accuracy by 2.34%. In particular, the accuracy of living-room and dining-room that have less frames are significantly improved. The result reflects that our improvement on codebook generation can effectively handle imbalance classification problem. The employment of object templates leads to a further 2% accuracy improvement, demonstrating the validity of our local object template classification.

4.2 Testing on Visual Place Categorization *with* Reject Option

[6] not only tested on the 5 common categories, this system also can reject the other categories using statistical hypothesis testing and mark them as a special category "transition" that include all frames from these categories. Our system dose not contain a statistical hypothesis testing part, but we can train a model for all the other categories so that we can compare our result with [6]. The result is shown in Table 2. Only using global configurations and Bayesian filtering,

Table 1. Comparing our system with [2] and [25]. [2]'s result is with Bayesian filtering. Here G means global configurations approach, O means object templates classification.

	bed	bath	kitchen	living	dining	average
VPC [2]	64.89	74.77	48.24	20.59	19.61	45.62
HOUP [25]	68.76	73.01	15.46	46.55	25.88	45.94
G+Bayesian	50.46	76.44	48.48	28.83	35.59	47.96
G+O(SIFT)+Bayesian	55.12	80.92	51.13	35.43	27.05	49.95

Table 2. Comparing our system with [6]. [6]'s result is using SIFT descriptor, which is the best in his paper.

	bed	bath	kitchen	living	dining	transition	average
PLISS [6]	61.72	69.61	51.11	29.08	14.94	42.82	44.88
G+Bayesian	54.57	38.34	50.15	32.41	34.04	42.18	41.95
G+O(SIFT)+Bayesian	58.62	44.64	51.52	37.26	38.42	45.47	45.99

Table 3. Comparing our system with [13]. [13]'s result is from cloudy condition, given that we also use sequences from cloudy conditions.

	PA	CR	2PO	BR	average
COLD [13]	11.67	76.17	14.67	8.00	27.63
G+Bayesian	13.04	62.18	25.06	45.75	36.51
G+O(SIFT)+Bayesian	25.37	60.16	33.93	44.60	41.01

the result is worse than [6]. However, if we employ local object templates, our system becomes higher in terms of accuracy, leading [6] by about 1%.[1]

4.3 Testing on COLD

Although COLD is mainly intended for place recognition, we can still test our system on this dataset to show that the global configurations and local object templates classification system are very effective despite the large diversity and smaller image size. Here, we do not care about the weather condition, so use just use three standard sequence from three labs in the cloudy condition and use the leave-one-out strategy to test our system. We only tested 4 categories that are available for all 3 labs: Printer Area (PA), Corridor (CR), 2-Person Office (2PO) and Bathroom (BR). [13] tested their place recognition and categorization system on COLD, their categorization testing method is similar with ours, so we compare our result with them in Table 3. Note that since they only provide

[1] [6] has a recent journal extension [18], which achieved accuracy 46.85% (without reject option) and 48.48% (with reject option), respectively.

histogram of their result with no exact numbers, their accuracy is estimated from the histogram. As shown in Table 3, both the global configurations and local object templates classification system are still very useful despite the diversity of different labs and smaller image size.

5 Conclusion

In this paper, we present a novel system to solve the Visual Place Categorization problem, utilizing both the global configurations information and the object templates information. We propose a novel local objects classifier that can automatically and efficiently select key objects from randomly sampled patches using structural similarity SVM and further classify the test frames by Local NBNN. We further improve the global configurations observation of [2] by employing HIK codebook and a noisy codewords removal mechanism. We ensure the temporal smoothness of the image sequences with the Bayesian filtering process.

Experiments on two large scale and difficult datasets demonstrate the superiority of our system. Our system not only gives overall accuracy that outperforms the state-of-the-art methods, but also provide more balanced and stable results between categories.

The system can be improved by employing a change point detection mechanism similar to [6] so that the image sequences are better segmented. Furthermore, we may enhance the object templates classifier by introducing some kind of noise removal mechanism like what we did in the global configurations approach.

References

1. Torralba, A., Murphy, K.P., Freeman, W.T., Rubin, M.A.: Context-based vision system for place and object recognition. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 273–280 (2003)
2. Wu, J., Christensen, H.I., Rehg, J.M.: Visual Place Categorization: Problem, dataset, and algorithm. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4763–4770 (2009)
3. Pronobis, A., Jensfelt, P.: Hierarchical multi-modal place categorization. In: Proceedings of the 5th European Conference on Mobile Robots (2011)
4. Pronobis, A., Mozos, O.M., Caputo, B., Jensfelt, P.: Multi-modal semantic place classification. The International Journal of Robotics Research, Special Issue on Robotic Vision 29, 298–320 (2010)
5. Wu, J., Rehg, J.M.: CENTRIST: A visual descriptor for scene categorization. IEEE Transactions on Pattern Analysis and Machine Intelligence 33, 1489–1501 (2011)
6. Ranganathan, A.: PLISS: Detecting and labeling places using online change-point detection. In: Proceedings of Robotics: Science and Systems (2010)
7. Wu, J.: Balance Support Vector Machines Locally Using the Structural Similarity Kernel. In: Huang, J.Z., Cao, L., Srivastava, J. (eds.) PAKDD 2011, Part I. LNCS, vol. 6634, pp. 112–123. Springer, Heidelberg (2011)
8. McCann, S., Lowe, D.G.: Local naive bayes nearest neighbor for image classification. CoRR abs/1112.0059 (2011)

9. Wu, J., Rehg, J.M.: Efficient and effective visual codebook generation using additive kernels. Journal of Machine Learning Research 12, 3097–3118 (2011)
10. Pronobis, A., Caputo, B.: Confidence-based cue integration for visual place recognition. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2394–2401 (2007)
11. Pronobis, A., Caputo, B., Jensfelt, P., Christensen, H.: A discriminative approach to robust visual place recognition. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3829–3836 (2006)
12. Pronobis, A., Caputo, B.: COLD: Cosy localization database. International Journal of Robotics Research 28, 588–594 (2009)
13. Ullah, M.M., Pronobis, A., Caputo, B., Luo, J., Jensfelt, P., Christensen, H.I.: Towards robust place recognition for robot localization. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 530–537 (2008)
14. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2169–2178 (2006)
15. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 413–420 (2009)
16. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1307–1314 (2011)
17. Viswanathan, P., Southey, T., Little, J., Mackworth, A.: Place classification using visual object categorization and global information. In: Proceedings of the Canadian Conference on Computer and Robot Vision, pp. 1–7 (2011)
18. Ranganathan, A.: PLISS: labeling places using online changepoint detection. Autonomous Robots 32, 351–368 (2012)
19. Ranganathan, A., Lim, J.: Visual Place Categorization in maps. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3982–3989 (2011)
20. Boiman, O., Shechtman, E., Irani, M.: In defense of Nearest-Neighbor based image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 1–8 (2008)
21. Wolf, L., Hassner, T., Taigman, Y.: Similarity Scores Based on Background Samples. In: Zha, H., Taniguchi, R.-i., Maybank, S. (eds.) ACCV 2009, Part II. LNCS, vol. 5995, pp. 88–97. Springer, Heidelberg (2010)
22. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research 9, 1871–1874 (2008)
23. Wu, J.: Power mean SVM for large scale visual classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2344–2351 (2012)
24. Muja, M., Lowe, D.G.: Fast approximate nearest neighbors with automatic algorithm configuration. In: International Conference on Computer Vision Theory and Application, pp. 331–340 (2009)
25. Fazl-Ersi, E., Tsotsos, J.K.: Histogram of oriented uniform patterns for robust place recognition and categorization. International Journal of Robotics Research 31, 468–483 (2012)

Reconstructing Sequential Patterns without Knowing Image Correspondences

Saba Batool Miyan and Jun Sato

Department of Computer Science and Engineering,
Nagoya Institute of Technology, Nagoya, 466-8555, Japan

Abstract. In this paper, we propose a method for reconstructing 3D sequential patterns from multiple images without knowing image correspondences and without calibrating camera sensitivity parameters on intensity. The sequential pattern is defined as a series of colored 3D points. We assume that the order of the points is obtained in multiple images, but the correspondence of individual points is not known among multiple images. For reconstructing sequential patterns, we consider a camera projection model which combines geometric and photometric information of objects. Furthermore, we consider camera projections in the frequency space. By considering the multi-view relationship on the new projection model, we show that the 3D sequential patterns can be reconstructed without knowing correspondence of individual image points in the sequential patterns, and also the recovered 3D patterns do not suffer from changes in camera sensitivity parameters.

1 Introduction

When we have multiple objects in the 3D scene, we often have the order of objects and their order is visible in images. For example, a necklace shown in Fig. 1 (a) consists of some small 3D objects with various colors, and the order of these small 3D objects is visible in the image. The line of vehicles on a road shown in Fig. 1 (b) is also an example of such objects. In this paper, we consider a method for reconstructing such 3D sequential patterns from multiple images.

Since the sequential patterns consist of many 3D points with various colors, finding the correspondence of these points among multiple images is not easy in general. The intensity pattern of the sequential points is a useful clue for finding correspondences among multiple images. However, the image intensity depends on the intensity sensitivity parameters of camera, which are different in each camera in general. To cope with the problem, people often calibrate camera intensity parameters before using image intensity information. However, the calibration of intensity parameters is not easy and is time consuming. Also, the intensity calibration is not always available. Thus, we in this paper propose a method which enables us to calibrate both geometric and photometric relationship among multiple cameras simultaneously without knowing point correspondences in multiple images. As a result, the 3D sequential patterns can be

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 484–496, 2013.

(a) necklace (b) a line of vehicles

Fig. 1. Example of 3D sequential patterns

reconstructed from multiple images without knowing correspondence of individual points and without knowing intensity parameters of each camera.

For obtaining the geometric relationship between multiple cameras, multilinear relationship is often used [1]. Recently, efforts have been made to expand the scope of traditional multiple view geometry by using extended camera models, for various applications and more general camera configurations [2,3,4,5]. However, the computation of multilinear relationship requires accurate correspondence of image features, such as image points and image lines. Thus, if some wrong correspondences exist in the image data, the computation of multilinear relationship fails, and 3D reconstruction suffers from the wrong correspondences. For obtaining corresponding points in multiple images, image tracking is often used [6]. However, the image tracking is only possible for a single moving camera, and is not applicable to the correspondence problem in general multi camera configurations. For general multi camera configurations, geometric constraints and statistical methods have often used [7,8,9]. However, finding accurate correspondences is still a serious problem in stereo and multiview reconstructions.

To cope with this problem, we in this paper consider multilinear relationship in the frequency space, and show that it enables us to describe multiple images of sequential patterns without knowing exact point correspondences. We also extend the traditional multilinear relationship, so that it can describe not only geometric relationship but also describe photometric relationship among multiple images. The extended multilinear relationship of images can be used for obtaining the geometric and photometric relationship among multiple cameras, and can be used for reconstructing sequential patterns without knowing point correspondences and without calibrating intensity parameters of cameras.

Although the proposed method can be applied only for sequential patterns, it shows the new possibility of relaxing the correspondence problem in stereo and multi camera reconstruction.

2 Camera Projection in Frequency Space

In this research, we define the 4D point in object space to be composed of geometric and photometric property i.e. we define object point to be composed of point's geometrical coordinates in 3D space and point's intensity. For this, we define an extended affine camera projection model from 4D object space to 3D

image space assuming weak perspective projection in geometry and Lambertian projection in photometry. Since we assume Lambertian projection in photometry, the image intensity depends on the relative orientation between light sources and a surface normal, but it is irrelevant to the viewpoint of cameras. Let a 4D point in homogeneous coordinates $\boldsymbol{X} = [X, Y, Z, I, 1]^\top$ be projected to $\boldsymbol{x} = [x, y, i, 1]^\top$ in the 3D image, where I and i represent the intensity of 3D point and intensity of projected 2D image point respectively. Then, this projection can be represented as below:

$$\mathbf{x} = \mathbf{P}\mathbf{X} \tag{1}$$

where, the projection matrix $\mathbf{P}$ in (1) is defined as follows:

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & 0 & p_{15} \\ p_{21} & p_{22} & p_{23} & 0 & p_{25} \\ 0 & 0 & 0 & p_{44} & p_{45} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

In this projection model, as defined in (2), the upper left 2x3 part comprising of $p_{11}, p_{12}, p_{13}, p_{21}, p_{22}$ and p_{23} denotes rotation, and upper right 2x1 vector consisting of p_{15} and p_{25} represents translation. The intensity of the 3D point is projected to the image intensity using the 3rd row of the projection matrix, where p_{34} stands for camera intensity gain and p_{35} stands for camera intensity offset, together they are known as camera sensitivity parameters. In general, these camera sensitivity parameters have different values in different cameras. As it is obvious from (2), the extended camera projection model can not only capture geometric information, but also capture photometric information of the scene.

Suppose we have two cameras $\boldsymbol{C}$ and $\boldsymbol{C'}$, and a series of 3D points $\mathbf{X}(i)$ $(i = 1, \cdots, N)$ is projected into these two cameras. We assume that the point correspondences in these two cameras are unknown, but the order of the points is obtained in each view. Thus, the ith 3D point $\mathbf{X}(i)$ is observed as ith image point $\mathbf{x}(i)$ in image 1 and is observed as i'th image point $\mathbf{x}'(i')$ in image 2 as follows:

$$\mathbf{x}(i) = \mathbf{P}\mathbf{X}(i) \tag{3}$$

$$\mathbf{x}'(i') = \mathbf{P}'\mathbf{X}(i) \tag{4}$$

i and i' are different in general. However, since we assume that the order is preserved in each image, the following relationship holds:

$$i' = i + k \tag{5}$$

where, k is the shift in sampling of image 2 with respect to sampling of image 1. Thus, k is unknown but is constant. Since k is unknown, we do not know the correspondence between $\mathbf{x}(i)$ and $\mathbf{x}'(i')$. To cope with the correspondence problem, we next consider camera projection in the frequency space.

Suppose we have N image points in image 1 and image 2 respectively. Then, by applying Fourier transform to both side of (3), we have following representation for image projection in frequency space:

$$\begin{aligned}\mathbf{z}(n) &= \frac{1}{N}\sum_{i=0}^{N-1}\mathbf{P}\mathbf{X}(i)e^{\frac{-j2\pi ni}{N}} \\ &= \mathbf{P}\mathbf{Z}(n)\end{aligned} \tag{6}$$

where, $\mathbf{z}(n) = [x_f(n), y_f(n), i_f(n), \delta(n)]^\top$ represents the image point in the frequency space, and $\mathbf{Z}(n) = [X_f(n), Y_f(n), Z_f(n), I_f(n), \delta(n)]^\top$ represents the 4D point in the frequency space. $\delta(n)$ is the delta function, whose value is equal to 1 for $n = 0$ and 0 for others. Similarly, applying Fourier transform to (4) we have the projection of camera C' in the frequency space as follows:

$$\begin{aligned}\mathbf{z}'(n) &= \frac{1}{N}\sum_{i'=0}^{N-1}\mathbf{P}'\mathbf{X}(i'-k)e^{\frac{-j2\pi ni'}{N}} \\ &= \mathbf{P}'\mathbf{Z}(n)e^{\frac{-j2\pi nk}{N}}\end{aligned} \tag{7}$$

where $\mathbf{z}'(n) = [x'_f(n), y'_f(n), i'_f(n), \delta(n)]^\top$ is a point in second image in frequency space. We may rewrite (7) as follows:

$$\lambda(n)\mathbf{z}'(n) = \mathbf{P}'\mathbf{Z}(n) \tag{8}$$

where $\lambda(n) = e^{\frac{j2\pi nk}{N}}$ represents the phase shift of sampling in camera C' relative to camera C. From (6) and (8), we find that in the frequency space both cameras project the same 4D point $\mathbf{Z}(n)$ and thus it is possible to consider the correspondence of image data, even if the sampling of image 2 is different from the sampling of image 1.

3 Multiview Relations on Sequential Patterns

Multiview relations in 4D space can be considered as the intersection of 5 hyperplanes meeting at a single point[1]. A necessary and sufficient condition for five hyperplanes to meet in a common point in 4D space is that the determinant of the matrix formed from the vectors representing these hyper planes should vanish. Mathematically this condition is written as follows:

$$\det[\mathbf{P}^\top\mathbf{s}^1, \mathbf{P}^\top\mathbf{s}^2, \mathbf{P}^\top\mathbf{s}^3, \mathbf{P}'^\top\mathbf{s}'^1, \mathbf{P}'^\top\mathbf{s}'^2] = 0 \tag{9}$$

where, $\mathbf{s}^1$, $\mathbf{s}^2$ and $\mathbf{s}^3$ represent planes which go through the corresponding point $\mathbf{z}$ in image 1, and $\mathbf{s}'^1$ and $\mathbf{s}'^2$ represent planes which go through the corresponding point $\mathbf{z}'$ in image 2. Then, (9) can be rewritten in tensor format as follows:

$$\epsilon^{pqrst}s^1_a P^a_p\ s^2_b P^b_q\ s^3_c P^c_r\ s'^1_d P'^d_s\ s'^2_e P'^e_t = 0 \tag{10}$$

Then from (10), following bilinear constraints on $\mathbf{z} = [z^1, z^2, z^3, z^4]^\top$ and $\mathbf{z'} = [z'^1, z'^2, z'^3, z'^4]^\top$ are obtained in the frequency space:

$$z^i z'^j \mathcal{T}_{fij} = 0_f \tag{11}$$

where, the bifocal tensor $\mathcal{T}_{fij}$is defined as follows:

$$\mathcal{T}_{fij} = \epsilon_{abci}\epsilon_{defj}\epsilon^{pqrst} P^a_p P^b_q P^c_r P'^d_s P'^e_t \tag{12}$$

ϵ_{abci} is a $4 \times 4 \times 4 \times 4$ tensor, which takes 1 for even permutation, -1 for odd permutation and 0 for others. Also, ϵ^{pqrst} is a $5 \times 5 \times 5 \times 5 \times 5$ tensor which takes $1, -1$ and 0 depending on its permutation. The tensor $\mathcal{T}_{fij}$ is a $4 \times 4 \times 4$ tensor. It is the algebraic representation of our extended two view geometry and shows the linear relationship between the two 3D images. It represents the relative camera position and orientation.

The affine bifocal tensor $\mathcal{T}_{fij}$ can be computed from a minimum of 5 point correspondences. From (11) we find, one point correspondence pair gives four equations, however only three out of these four equations are linearly independent. Similarly, although the tensor $\mathcal{T}_{fij}$ has 64 elements, only 18 components are non-zero. The nonzero unique elements include $\mathcal{T}_{144}, \mathcal{T}_{244}, \mathcal{T}_{344}, \mathcal{T}_{341}, \mathcal{T}_{342}, \mathcal{T}_{413}, \mathcal{T}_{423}, \mathcal{T}_{431}, \mathcal{T}_{432}$. The other set of non-zero elements can be obtained by taking negative and reversing the indices e.g.$\mathcal{T}_{441} = -\mathcal{T}_{144}$. For the multiview relations in frequency domain, there exist additional conditions, that is the elements $\mathcal{T}_{144}, \mathcal{T}_{244}, \mathcal{T}_{344}, \mathcal{T}_{341}, \mathcal{T}_{342}$ can be computed only from the point correspondence at $n = 0$. Since we have only one point correspondence at $n = 0$ which provides us 3 linear constraints, a single sequential pattern is not sufficient. Therefore, we must use at least two sequential patterns to compute the aforementioned 5 elements. The other four elements $\mathcal{T}_{413}, \mathcal{T}_{423}, \mathcal{T}_{431}, \mathcal{T}_{432}$ can be obtained from point coordinates in the case of $n \neq 0$.

4 Estimation of Phase Shift

Let the upper 3x5 matrix of $\mathbf{P}$ and $\mathbf{P'}$ be represented as $\mathbf{R}$ and $\mathbf{R'}$, and the upper 3x1 part of $\mathbf{z}$ and $\mathbf{z'}$ be denoted as $\mathbf{w}$ and $\mathbf{w'}$ respectively. Then a pair of cameras can be expressed as follows:

$$\begin{bmatrix} \mathbf{R} & \mathbf{w} \\ \mathbf{R'} & \lambda\mathbf{w'} \end{bmatrix} \begin{bmatrix} \mathbf{Z} \\ -1 \end{bmatrix} = \mathbf{0} \tag{13}$$

Since the matrix on the extreme left, say $\mathbf{M}$, has a non-trivial right null space; the vector $[\mathbf{Z}, 1]^\top$, its determinant must be zero. Expanding $\mathbf{M}$ by the last column and separating the phase term we obtain:

$$\lambda(n) = -\frac{w^1 Q_1 + w^2 Q_2 + w^3 Q_3}{w'^1 Q_4 + w'^2 Q_5 + w'^3 Q_6} \tag{14}$$

where, Q_i are determinants of the 5x5 minors of the matrix $\mathbf{M}$ obtained by removing the ith row and last column of $\mathbf{M}$.

To compute the phase shift in image 2 with respect to image 1, we compute the phase terms for two consecutive frequencies, say λ_{n_1} and λ_{n_2}, using (14). Then take their ratio as shown in the following equation.

$$\frac{\lambda_{n_2}}{\lambda_{n_1}} = \frac{e^{jn_2\theta}}{e^{jn_1\theta}} = e^{j(n_2-n_1)\theta} \tag{15}$$

where, $\theta = \frac{2\pi k}{N}$. Then the shift term k can be computed as follows:

$$k = \frac{N}{2\pi j(n_2 - n_1)} log \frac{\lambda_{n_2}}{\lambda_{n_1}} \tag{16}$$

5 Reconstruction

Assuming a canonical camera pair we can define the two camera matrices as follows:

$$\mathbf{P} = [\mathbf{I}|\mathbf{0}] \qquad\qquad \mathbf{P}' = [\mathbf{H}|\mathbf{e}'] \tag{17}$$

where $\mathbf{I}$ is the 4x4 identity matrix, $\mathbf{0}$ stands for null 4-vector, $\mathbf{H}$ represents the homography between a pair of images and $\mathbf{e}'$ is the epipole in the second image. Then, $\mathbf{H}$ and $\mathbf{e}'$ can be computed from $\mathcal{T}_{fij}$ as described in the following sections. Once the homography and the epipole are computed, camera matrices are recovered from (17), and the 4D points $\mathbf{X}(i)$ $(i = 1, \cdots, N)$ can be reconstructed.

5.1 Computation of Epipoles

For a given bifocal tensor $\mathcal{T}_{fij}$, the epipole in image 1, represented as $\mathbf{e}$, can be computed as the left null space of the bifocal tensor $\mathcal{T}_{fij}$. Similarly, the epipole in image 2, represented as $\mathbf{e}'$, can be computed as the right null space of the bifocal tensor $\mathcal{T}_{fij}$, as follows:

$$e_i \mathcal{T}_{fij} = 0_{fj} \qquad\qquad e'_j \mathcal{T}_{fij} = 0_{fi} \tag{18}$$

5.2 Extraction of Homography

Let us consider an arbitrary but fixed plane π, not passing through any of the camera centers. Then, a point $\mathbf{z}$ in image1 is related to a point $\mathbf{z'}_\pi$ in image 2 by the following relationship:

$$z'^l_\pi = H^l_i z^i \tag{19}$$

Then, an epipolar line $\mathbf{l}'$ in image 2 can be defined as the join of epipole $\mathbf{e}'$ and the point $\mathbf{z'}_\pi$ as follows:

$$l'_{jf} = \epsilon_{jflr} e'^r z'^l_\pi \tag{20}$$

However from (11), epipolar line $\mathbf{l}'$ can also be defined as the corresponding line in image 2 due to a point present in image 1.

$$l'_{jf} = \mathcal{T}_{fij} z^i \tag{21}$$

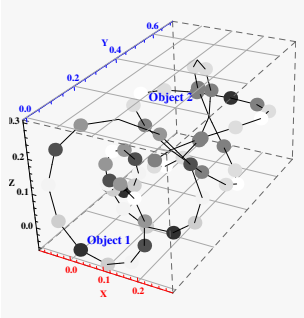
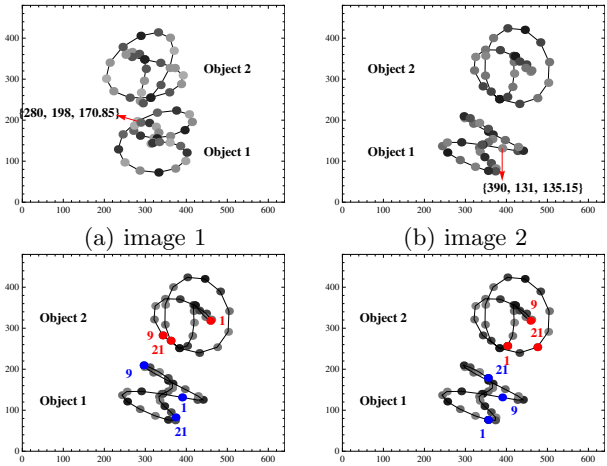


Fig. 2. 3D sequential patterns used in our synthetic image experiment. There exist two series of 3D points, which are connected by lines, i.e. object 1 and object 2.



(a) image 1 (b) image 2

(c) image 2 before phase shift (d) image 2 after phase shift

Fig. 3. 3D Images taken from different viewpoints. Index numbers in red and blue points show the sampling order.

Then from (19), (20) and (21) we obtain the following relationship for the homography:

$$H_i^l = \epsilon^{jflr} e'_r \mathcal{T}_{fij} \tag{22}$$

Thus, we can compute both $\mathbf{H}$ and $\mathbf{e}'$ from the bifocal tensor $\mathcal{T}_{fij}$, and camera matrices $\mathbf{P}$ and $\mathbf{P}'$ can be obtained.

6 Experimental Results

In this section, we show the efficiency of the proposed method from real image experiments and synthetic image experiments. We also show the stability analysis of the proposed method.

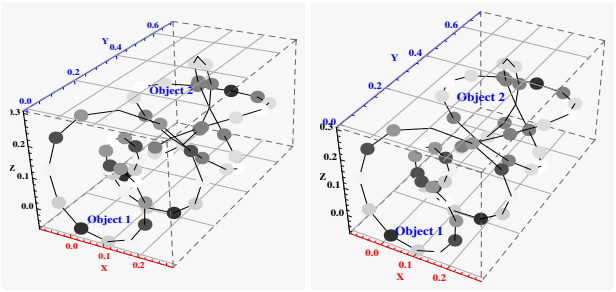
6.1 Synthetic Image Experiments

We first show the results from synthetic image experiments. Fig.2 shows a synthetic scene considered in our experiments. There exist two series of 3D points, which are connected by lines in this figure. As shown in this figure, each 3D point has its own intensity. Since the objects have 3D shape and 1D intensity, we consider them as 4D objects.

Then, we projected the 4D objects into two extended affine cameras at different positions with different camera gains as shown in Fig.3 (a) and Fig.3 (b). As it can be seen, the images also have point position and intensity, and thus we consider them as 3D images. The image 1 shown in Fig.3 (a) is high intensity image, where as the image 2 shown in Fig.3 (b) is a low intensity image. Moreover, we shifted the sampling order of points in image 2 with 9 to demonstrate the correspondence freeness of our method. Fig.3 (c) and (d) show image 2 before and after the shift in sampling order. Then using the proposed method we first computed the bifocal tensor $\mathcal{T}_{fij}$ from the 3D images shown in Fig.3 (a) and (d). Then the estimated bifocal tensor was used for computing camera matrices and for reconstructing 4D object that is 3D shape and intensity.

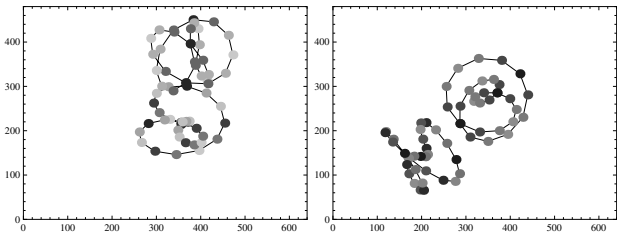
From Fig.4 (a) and Fig.4 (b) we find that, the reconstruction is correct both geometrically and photometrically. This correctness of reconstruction is an indication of two clear distinctions of our proposed method. First, it shows that our method is correspondence free. Unlike the traditional reconstruction method in the spatial domain, the proposed method can reconstruct the 4D objects even in the absence of exact corresponding points. Second, it shows that our method is independent of camera sensitivity parameters, and off line calibration of camera intensity parameters is not necessary.

Once the original object is recovered we can generate the arbitrary views of the reconstructed object by projecting it onto virtual cameras at different locations. Fig.5 (a) and Fig.5 (b) show two arbitrary views of the sequential pattern. From Fig.5 (a) and Fig.5 (b), we conclude that our extended multiple view geometry in frequency space can accurately generate 3D shape and point intensity of a



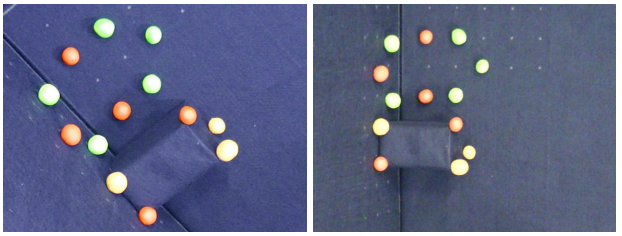
(a) original object (b) reconstructed object

Fig. 4. Original and reconstructed 4D object. The reconstructed object has correct 3D shape and point intensity values.



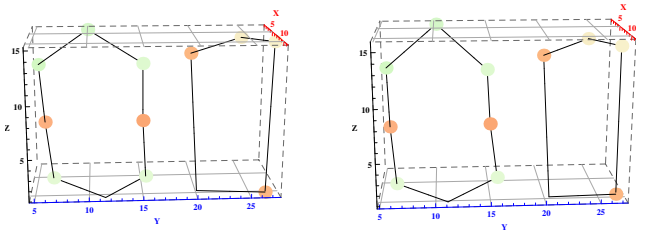
(a) arbitrary view 1 (b) arbitrary view 2

Fig. 5. Arbitrary views of reconstructed sequential pattern



(a) image 1 image 2

Fig. 6. Images of sequential patterns used in our real image experiment 1



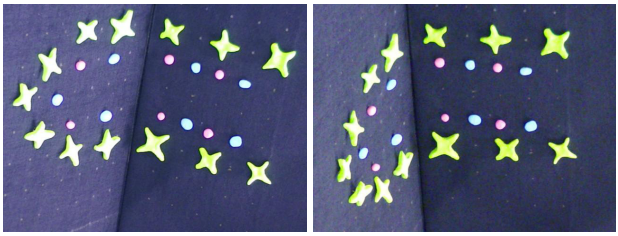
(a) original 4D object (b) reconstructed 4D object

Fig. 7. Reconstruction results in the real image experiment 1

sequential pattern from arbitrary positions of cameras, and thus it is able to transfer the sequential patterns between images via reconstruction.

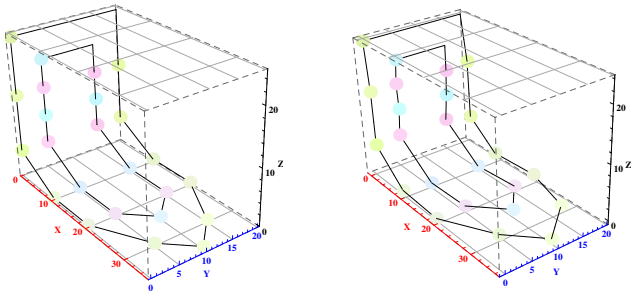
6.2 Real Image Experiments

In this section we show the results of the proposed method tested with two different examples of real images. Fig.6 (a) and Fig.6 (b) show two real images used in the first experiment. These images were taken with Everio GZ-HM450 video camera, which has a gain up function. The image 1, shown in Fig.6 (a) is taken under normal imaging conditions, where as the image 2, shown in Fig.6 (b) is a gain up image, and therefore it can be considered as a high intensity image compared to the image 1. Moreover, the sampling phase of image 2 was shifted



(a) image 1 (b) image 2

Fig. 8. Images of sequential patterns used in our real image experiment 2



(a) original 4D object (b) reconstructed 4D object

Fig. 9. Reconstruction results in the real image experiment 2

with 3. The image size is 640 x 480. The blobs in the image were detected, and the centroid of each blob and average intensity were used for the experiments. We have only one dimension for point intensity whereas usually color image data is three dimensional consisting of red, blue and green channels. Therefore for the reconstruction of colored object, the RGB values were reconstructed separately and combined after reconstruction. However for consistency the reconstruction of shape was done using gray scale image data.

Then, using the 3D image shown in Fig.6 (a) and Fig.6 (b) the bifocal tensor $\mathcal{T}_{fij}$ was computed, and the original objects were reconstructed in spatial domain. Also, the phase shift was computed using the proposed method, and the estimated phase shift was 3. Thus, phase shift was computed correctly. Fig.7 (a) and Fig.7 (b) show the original and reconstructed object space, which confirms following important results (i) correspondence freeness of our multi view relations (ii) camera sensitivity parameter freeness of our camera model. It shows that our method is independent of camera intensity parameters, and we can freely use it for images taken by cameras with different camera gains.

Another real image experiment was conducted with different 3D shape, object colors and more points, which is shown in Fig.8 (a) and Fig.8 (b). Fig.9 (a) and Fig.9 (b) show the original and reconstructed objects, which show that the proposed method works quite well for different settings of real image experiments as well.

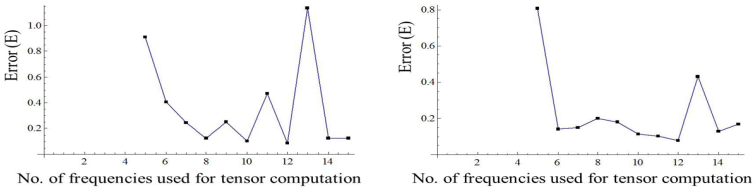
6.3 Stability Analysis

We next show the stability of the bifocal tensor $\mathcal{T}_{fij}$ used for the computation of 3D shape and point intensity to generate the sequential pattern of the object in 4D space, using synthetic images shown in Fig.3 (a) and Fig.3 (d). For stability analysis a Gaussian noise of standard deviation of 1 pixel was added to each image point and a Gaussian noise of standard deviation of 1 was added to each image point intensity. The image size is 640x480 and the image intensity varies from 0 to 255. Then, the following reconstruction error was computed from N reconstructed points:

$$E = \frac{1}{N}\sum_{i=1}^{N} d(\mathbf{X}(i) - \hat{\mathbf{X}}(i))^2 \tag{23}$$

where $d(\mathbf{X}(i) - \hat{\mathbf{X}}(i))^2$ represents the square distance between the true point and a point reconstructed from the proposed method in spatial domain. By varying the experimental conditions, such as 4D configuration, camera position and noise magnitude, we increased the number of corresponding frequencies used for computing the bifocal tensor between a pair of images from 5 to 15 and evaluated the reconstruction errors averaged over 100 iterations in shape and intensity separately, which is shown in Fig.10 (a) and Fig.10 (b) respectively. Fig.10 (a) shows the relationship between the number of corresponding frequencies used for tensor computation and the reconstruction error in 3D shape; the horizontal axis represents the the number of corresponding frequencies used for bifocal tensor computation and the vertical axis shows the reconstruction error in spatial domain. Fig.10 (b) shows the relationship between the number of frequencies used for bifocal tensor computation and reconstruction error in intensity; the horizontal axis shows the the number of points used for tensor computation and the vertical axis represents the reconstruction error in intensity in spatial domain.

From Fig.10 (a) and Fig.10 (b), we find that there exists a tradeoff. As we increase the number of corresponding frequencies, the magnitude of reconstruction error is first reduced. However as we continue to increase the number of corresponding frequencies for tensor computation in frequency domain, higher frequencies are



(a) reconstruction error in 3D shape (b) reconstruction error in intensity

Fig. 10. Reconstruction error in shape and intensity. The horizontal axes represent the number of frequencies used for computing the bifocal tensor. The vertical axes show errors in shape and intensity.

involved which are more sensitive to noise, and they cause instability of bifocal tensor. Consequently the reconstruction error curve follows a see-saw pattern afterwards, which brings us to the conclusion that in frequency space only lower frequencies should be utilized for the computation of bifocal tensor.

7 Conclusion

In this paper, we proposed a method for calibrating cameras and reconstructing 3D sequential patterns from multiple images without knowing image correspondences and without calibrating camera sensitivity parameters on intensity. We assumed that the order of the points is obtained in multiple images, but the correspondence of individual points is not known among multiple images. For reconstructing sequential patterns, we considered a camera projection model which combines geometric and photometric information of objects. Furthermore, we considered camera projections in the frequency space, and derived multi-view relationships in the frequency space. We showed that by using the multi-view relationship in the frequency space, the 3D sequential patterns can be reconstructed without knowing correspondence of individual image points in the sequential patterns, and also the recovered 3D patterns do not suffer from changes in camera sensitivity parameters.

Although the proposed method is limited for sequential patterns, it shows a new possibility of relaxing the correspondence problem in stereo reconstruction.

References

1. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2000)
2. Wolf, L., Shashua, A.: On projetion matrices $\mathbf{P^k} \rightarrow \mathbf{P^2}, k = 3, ..., 6$, and their applications in computer vision. International Journal of Computer Vision 48, 53–67 (2002)
3. Wan, C., Sato, J.: Multiple view geometry under projective projection in space-time. IEICE - Transactions on Information and Systems E91-D, 2353–2359 (2008)
4. Matsumoto, H., Sato, J., Sakaue, F.: Multiview constraints in frequency space and camera calibration from unsynchronized images. In: IEEE Conference on Computer Vision and Pattern Recognition, pp.1601–1608 (2010)
5. Thirthala, S., Pollefeys, M.: Multi-view geometry of 1d radial cameras and its application to omnidirectional camera calibration. In: IEEE International Conference on Computer Vision (2005)
6. Newcombe, R.A., Lovegrove, S.J., Davison, A.J.: Dtam: Dense tracking and mapping in real-time. In: IEEE International Conference on Computer Vision (2011)
7. Agarwal, S., Snavely, N., Simon, I., Seitz, S.M., Szeliski, R.: Building rome in a day. In: International Conference on Computer Vision (2009)

8. Ogale, A.S., Aloimonos, Y.: Shape and the stereo correspondence problem. International Journal of Computer Vision, 147–162 (2005)
9. Li, G., Zucker, S.W.: A differential geometrical model for contour-based stereo correspondence. In: International Conference on Computer Vision (2003)
10. Efros, A., Freeman, W.: Image quilting for texture synthesis and transfer. In: ACM SGGRAPH. Computer Graphics Proceedings, pp. 341–346 (2001)
11. Hertzmann, A., Jacobs, C., Oliver, N., Curless, B., Salesin, D.: Image analogies. In: Proceedings of ACM SGGRAPH. Computer Graphics Proceedings, pp. 327–340 (2001)

Registration of Multi-view Images of Planar Surfaces

Radomír Vávra and Jiří Filip

Institute of Information Theory and Automation of the ASCR, The Czech Republic

Abstract. This paper presents a novel image-based registration method for high-resolution multi-view images of a planar material surface. Contrary to standard registration approaches, this method aligns images based on a true plane of the material's surface and not on a plane defined by registration marks. It combines the camera calibration and the iterative fitting of desired position and slant of the surface plane, image re-registration, and evaluation of the surface alignment. To optimize image compression performance, we use an error of a compression method as a function evaluating the registration quality. The proposed method shows encouraging results on example visualizations of view- and illumination-dependent textures. In addition to a standard multi-view data registration approach, it provides a better alignment of multi-view images and thus allows more detailed visualization using the same compressed parameterization size.

1 Introduction

Acquisition of a multi-view appearance is often used to achieve realistic visualization of textured objects. This paper is focused on visualization techniques which deal with multiple photos of the same planar surface acquired from different positions. This way a photo-realistic appearance of the surface can be captured, but the acquired photos cannot be directly used for rendering. They have to be mutually registered and rectified first.

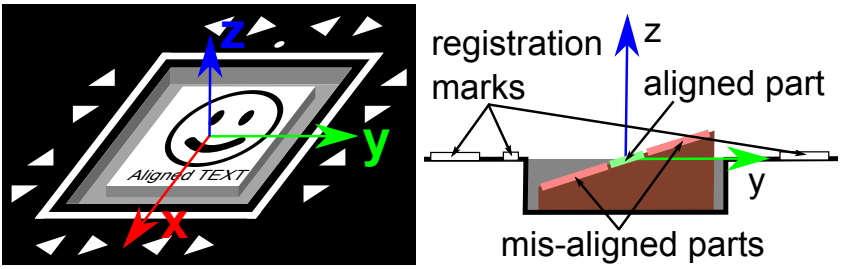
The most general function of multi-view photos of a planar surface is probably the *Bidirectional Texture Function* (BTF) proposed by Dana et al [1]. This seven-dimensional function $BTF(\lambda, x, y, \theta_i, \phi_i, \theta_v, \phi_v)$, describes reflectance properties of a material where λ is a wave length of incoming light or just a color channel; (x, y) are spatial coordinates on a surface of the material, and θ, ϕ are the elevation and azimuthal spherical angles of the vector of illumination- and view-directions (see [2]). A typical size of a BTF dataset containing thousands of images amounts to several gigabytes.

An another example of multi-view data is *Surface Light Field* [3]. It can be defined as a subset of a BTF with a fixed illumination direction.

Processing of acquired multi-view data consists of two steps: data registration and compression. Although the measured materials are planar, their rough structure often shows height variations causing significant variance of their appearance depending on illumination- and view-directions. The final appearance is

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 497–509, 2013.

affected by self-occlusions, shadows, inter-reflections, and subsurface-scattering. This is the reason why the features of the material are non-stationary and cannot be directly used for reliable feature-based registration. Due to this, we use registration marks placed on a reference plane, which allows the measured sample to be easily replaced. However, the sample's orientation and shift with respect to the registration plane is unknown (see Fig. 1). Although one might use tilt/shift mechanical stage to fine-tune this misalignment manually, it is expensive and far less accurate than the proposed approach.



(a) (b)

Fig. 1. (a) A measured sample, the reference plane with the registration marks (white frame and triangles) and the world coordinate system. (b) a cross-section of (a), shows aligned and misaligned parts of the surface of the measured material when the positions of the reference plane and of the measured surface differ.

The registration based on the registration marks properly aligns only those parts of the measured sample which lie close to the plane specified by the marks (see Fig. 1-b). This is not an issue when a distance between the reference and the material's surface planes is small; the slant difference is also negligible. This might be less relevant when registered data are of low resolution or are used directly for a rendering. However, it has a significant impact on the resulting visual quality if compression methods for multi-view data are used (e.g., all the classes of global factorization methods based on PCA [4] or clustering [5], etc.).

The contribution of the paper – The main contribution of this paper is a novel technique for registration of multi-view images of planar surfaces that aligns measured planar surfaces regardless of their slight height and slant differences from the reference registration plane. Such data alignment allows us to better exploit the power of compression techniques and produce an image closer to the original image reconstruction using the same number of parameters.

Organization of the paper – The paper starts with description of related past research in Section 2, the brief description of a standard registration procedure is explained in Section 3. A basic overview of the proposed approach is presented in Section 4, while details of a camera calibration and an iterative registration procedure are given in Sections 5 and 6. Results of the presented method are shown in Section 7. Main conclusions and suggestions for future work are outlined in Section 8.

2 Related Work

The proposed approach is, to a great degree, related to methods for multi-view data registration and methods which reconstruct a 3D surface of a sample.

View-dependent image-based data are generally captured by setups based on gonioreflectometers realizing four mechanical degrees of freedom (Sattler et al [4], Holroyd et al [6]) and setups which reduce measurement times or complexity by using multiple lights or sensors simultaneously (Müller et al [7], Neubeck et al [8]). Sattler et al [4] measured BTF data and registered it using a projective transformation based on registration marks (see Section 3). Such registration is sufficient as long as the resolution of the captured images of the measured material is low enough and the plane which represents the surface of the material is close to the reference registration plane. Additionally, the problem of registration is not so pressing in this case because the authors applied compression to data of individual views separately and thus the multi-view correspondence does not affect performance of compression to the same degree as other compression methods do [9,10] .

Neubeck et al [8] were aware of a problem with BTF alignment. Their work is the most relevant to our paper as they propose to evaluate quality of BTF alignment using a function which computes average Euclidean distance between the intensities of those neighboring views that share the same lighting direction. They tested several plane heights and selected the one for which this function is minimal. In contrast, our technique allows us to not only compensate for height misalignment, but also for mutual rotation of the registration and sample planes, without need for repetitive measurement.

Müller et al [7] used a setup with no moving parts. Therefore, positions of the image sensors are known in advance and registration can be done in sub-pixel accuracy without the need for registration marks. In another paper, Müller et al [11] proposed an approach attempting to align individual BTF pixels based on optimization techniques reducing certain intra-variations in the data. This method rotates individual ABRDFs to achieve better global compression performance, and therefore it requires storage of an additional per-pixel rotation map. Nevertheless, in both cases the accuracy of the measurement or the fit depends on an initial position of a calibration plane and its difference from the plane representing the surface of a material, which can be compensated when our registration technique is used.

Ruiters and Klein [12] published a technique which represents the appearance of a material using a combination of surface depth-map and spatially-varying reflectance. The authors define a dense reference mesh and align its polygons to best fit the original data to estimate a depth-map of near-flat surfaces. This technique can deal with materials having variable surface-height; however, our method is easier to implement and it is computationally less expensive. We do not attempt to interpret surface depth (which might not even be possible for some translucent materials) but to find an alignment that maximizes the quality of registration of multi-view data.

Methods for simultaneous acquisition of shape and reflectance exist; e.g., Müeller's setup can be used for an acquisition of even non-planar objects [13]. Holroyd et al [6] used a system based on a spherical gantry, where each arm is fitted with a camera and a high frequency, spatially-modulated light sharing a common focal point and an optical axis. The proposed measurement method exploits multi-view stereo, phase-based profilometry, and light descattering to avoid 2D-3D data registration problems and leverage a restrictive assumption about BRDF as is often done by related methods. Weinmann et al [14] added multiple projectors into the setup [7] for a detailed 3D acquisition of an object. The projectors emit structured light used for unique identification of points on a surface of the object. Although such approaches allow us to find an exact position of a flat material surface as well, the required hardware would unnecessarily increase the financial cost of a setup with no advantage compared to the technique we have proposed.

Additionally, our method is robust, easy to implement, computationally efficient, and optimal in terms of the compression method used.

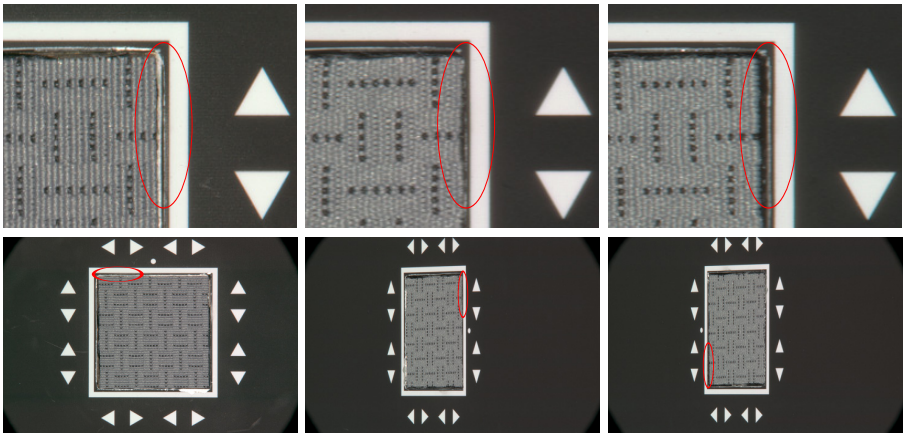
3 A Standard Image Registration Approach

This section outlines a principle of image registration. Given a set of photos of the same planar surface, registration applies a projective transformation to all the photos so that the features of the planar surface are aligned across all of the transformed images. In a standard registration approach depicted in Fig. 4-a, registration marks are placed around the photographed planar surface of a material. First, their 2D coordinates are found in all of the photos. Then, projective transformation matrices projecting these points to the desired target coordinates are computed. Finally, all the photos are transformed using these projective matrices. All the registered images have the same coordinate system.

A projective transformation, also called a *homography*, is a 2D coordinate transformation preserving straight lines (see [15] for a 2D coordinate transformation survey). Given a photo of a planar surface we want to transform together with an orthonormal coordinate system (u, v) of the photo, $\mathbf{m} = [u, v, 1]^T$ denotes an augmented point in this system in the planar surface. An augmented 2D point $\mathbf{m'} = [u', v', 1]^T$ in a new orthonormal coordinate system (u', v') into which we transform can be computed as $s\mathbf{m'} = \mathbf{Hm}$, where $\mathbf{H}$ is the 3×3 homography matrix and s is an arbitrary scalar. The homography can be computed if we know coordinates of at least four corresponding points in the source and target images and it is defined uniquely up to a scale factor. If there are more than four such points and they are not perfectly corresponding, the homography has to be computed in a least-square sense (e.g., [16]).

4 An Overview of the Proposed Registration Method

When a standard registration approach is used, the features which do not lie in the registered plane will not be aligned after application of the projective



$\theta_v = 0°, \phi_v = 0°$ $\theta_v = 60°, \phi_v = 90°$ $\theta_v = 60°, \phi_v = 270°$

Fig. 2. An example of incorrect registration of a material's surface for images taken from different views, while the reference plane (the registration marks) is registered correctly (top row)

transformation as it is shown in Fig. 2. The same pixel will correspond to different physical points on the surface of the material (see Fig. 3-a,c). Therefore, we have to estimate the plane which represents the surface of the registered sample for appropriate registration (Fig. 3-b,d). Unfortunately, this plane (i.e., its offset and orientation with respect to the world coordinate system) cannot be determined accurately enough from the specimen of the measured sample, or directly from the acquired photos.

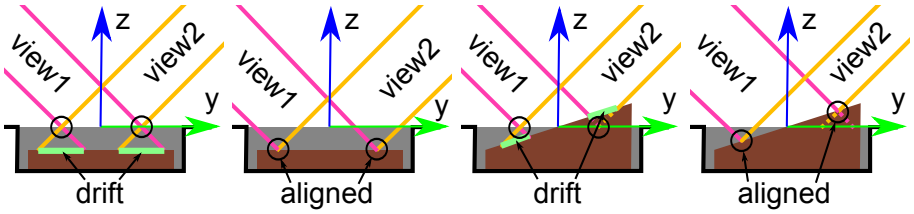
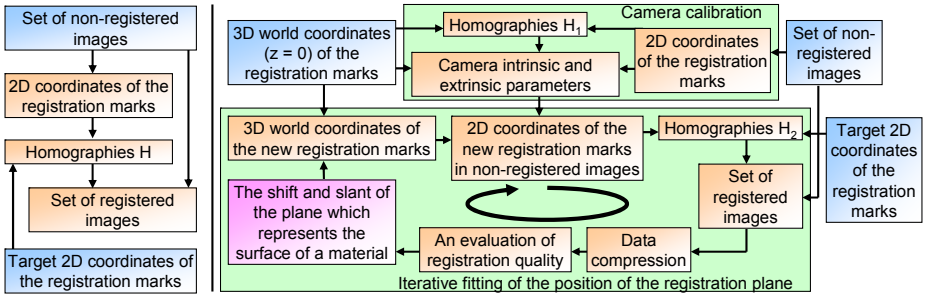


Fig. 3. When the plane defined by the registration marks is chosen for registration (a,c), the same pixel will correspond to different points on the surface of the measured sample, i.e., resulting in a small drift. When the appropriate plane on the surface of the material is chosen, the drift disappears (b,d).

We propose to find the position and slant of an ideal registration plane as follows. First, the reference plane defined by the registration marks is taken. As we expect that the estimated plane which represents the surface of the material is close to the reference plane, a new hypothetical position of the estimated

plane can be generated using a slight modification of the reference plane position by shifting it in a direction of its normal vector and/or by tilting it. Finally, the best estimation of the position and slant of the plane can be found by repeatedly alternating adjustment of the position and slant, registration of photos and evaluation of alignment for the surface features. A principle of the proposed registration method is expanded upon in Fig. 4-b. The method consists of two main parts discussed in more detail in the next two sections. The first one is the camera intrinsic and extrinsic parameters estimation (Section 5) and the second is the iterative fitting of the estimated plane position (Section 6).



(a) (b)

Fig. 4. A standard image registration approach (a) and a scheme of the proposed method (b)

5 Camera Calibration

The calibration of a camera is a process where we look for a 3×3 matrix $\mathbf{A}$ of the camera's intrinsic parameters, and the camera's extrinsic parameters which consist of a 3×3 rotation matrix $\mathbf{R}$ and a translation vector $\mathbf{t}$. While intrinsic parameters $\mathbf{A}$ do not change as long as the internal setup of the camera does not change (e.g., focal length), the extrinsic parameters change when the camera moves, i.e., all the photos of the measured planar sample should have the same camera's intrinsic parameters but the corresponding extrinsic parameters can be different. Using the camera parameters we can project an augmented 3D point $M = [x, y, z, 1]^T$ from the world coordinate system into an image by

$$s\mathbf{m} = \mathbf{A} \left[\mathbf{R}\,\mathbf{t} \right] M, \tag{1}$$

where $\mathbf{m} = [u, v, 1]^T$ denotes an augmented 2D point and s is an arbitrary scalar. The usual pinhole camera model is assumed.

In a case where we work with extensive view- and illumination-dependent data (e.g., BTF), the procedure of calibrating the camera and iteratively fitting the position of the estimated plane should start with selection of their representative subset. Although the registration plane can be determined more precisely if all the photos are used, the estimation process would take a very long time

if there were hundreds or even thousands of images. Therefore, we recommend working with images of one surface light field only; i.e., images where illumination directions are fixed while view angles are changed.

Next, the registration marks are found for all images in the subset. Without loss of generality, we define the world coordinate system so that the reference registration marks plane is on $z = 0$ (see Fig. 1). Spatial coordinates of the marks (x, y) should correspond to their real positions in a natural system of units, i.e., millimeters. Now, projective transformation matrices $\mathbf{H}_1$ projecting points from the reference plane to photos of the plane are computed based on the coordinates of the registration marks. From 1, we have

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{r}_1 \, \mathbf{r}_2 \, \mathbf{r}_3 \, \mathbf{t} \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \\ 1 \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{r}_1 \, \mathbf{r}_2 \, \mathbf{t} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{H}_1 \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \tag{2}$$

where $\mathbf{r}_i$ are column vectors of the rotation matrix $\mathbf{R} = [\mathbf{r}_1 \, \mathbf{r}_2 \, \mathbf{r}_3]$. Using the knowledge that column vectors of $\mathbf{R}$ are orthonormal and the matrix $\mathbf{A}$ is upper triangular, the camera intrinsic and extrinsic parameters can be derived from homographies $\mathbf{H}_1$. For a detailed explanation we refer the reader to Zhang's paper [17]. For camera calibration we have used the Camera Calibration Toolbox for Matlab[1] which implements Zhang's work.

6 Iterative Fitting of the Position of the Registration Plane

We can define an almost arbitrary position of the expected plane which represents the surface of a material in the world coordinate system just by setting new z-coordinates of the registration marks. If we project the estimated 3D coordinates of the new registration marks back to non-registered photos, we obtain new 2D coordinates of the registration marks in the coordinate systems of the photos. We can now register the images by specification of target coordinates of the registration marks and by computation of homography matrices $\mathbf{H}_2$ for registration (see Fig. 4-b). The latter should not be confused with the homography matrices $\mathbf{H}_1$ mentioned above, which project points in the reference registration marks plane of the world coordinate system to the non-registered images. In contrast, these new homography matrices $\mathbf{H}_2$ project points from the non-registered photos to the registered images as homography matrices $\mathbf{H}$ in a standard registration approach (Fig. 4-a).

Therefore, we suggest a novel iterative method for the position and slant of the registration plane estimation, image registration and alignment of surface features evaluation depicted in Fig. 4-b. As we look for optimal vertical shift and slant of the material surface plane, three parameters have to be found: a z-coordinate of an auxiliary point $P = [0, 0, z]^T$ which lies in the surface

[1] `http://www.vision.caltech.edu/bouguetj/calib_doc/`

plane, an elevation θ of the normal vector of the surface, and an azimuthal angle φ of the normal vector. As a search state space is three-dimensional and the image registration, compression and evaluation function execution can be computationally demanding, at least the local minimum can be found quickly by alternating between estimation of individual parameters. As there can be significant variations in height on the material's surface, there may be more than one good surface plane position.

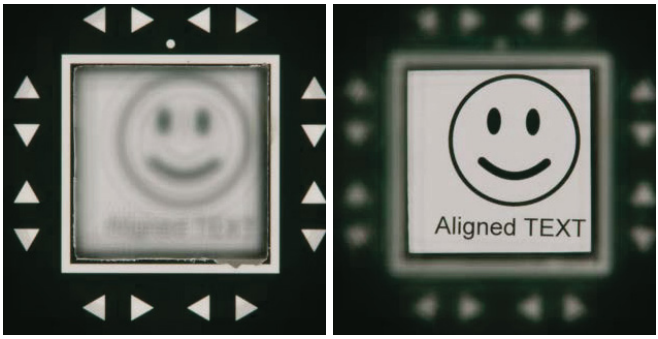
Our goal is to provide as accurate a visualization of the measured material sample as possible. As the visualization quality relies mostly on visual quality after data compression, evaluation functions which estimate alignment of surface features should reflect properties of the selected compression method. Therefore, the error of a compression technique will be used as an objective quality measure. As we work with only a subset of all of the photos, the compression as well as its error evaluation can be done quickly enough to be practical. An ideal position of the plane is the one where the compression (i.e., rendering) error is minimal. One iteration of reference plane modification, data registration, and visual quality evaluation after the compression takes about one second depending on counts of the registered pixels and images. In order to avoid local minima the search space was sampled uniformly, alternating between estimation of the three parameters (height z, plane normal's elevation θ, and azimuth φ) with the following step sizes: z=0.1 mm (range [-2,2]mm), θ=0.1^o (range [0,3]^o), φ=10^o (range [0,360]^o), and then refined near a global minimum (step sizes: z=0.01 mm, θ=0.01^o, φ=1^o). Typically, around 800 iterations are necessary to find a proper orientation and height of the registration plane.

7 Results

This section illustrates performance of the method on two registration experiments using artificial and real data. In the experiments we used the PCA compression of all registered images [9] and applied its data reconstruction error as a registration performance evaluation function in the proposed method. All pixels selected from individual BTF images are ordered into vectors and centered using the mean BTF image vector. All these vectors form a matrix $\mathbf{B}$, whose PCA is computed. The individual eigenvalues from the resulting diagonal matrix weight the importance of the resulting eigenvectors. A limited set k of eigenvectors is used to reconstruct the original n images, where $k << n$. The PCA-based methods are the most common in multi-view data compression; however, any other global BTF data compression technique would also benefit from the proposed algorithm.

In the first experiment a flat paper printout was used, positioned approximately one millimeter below the reference registration plane. We took 80 different views on the plane which uniformly covered a hemisphere of viewing directions. An illumination direction was fixed in a direction opposite to the reference plane's normal vector. When a standard registration approach was used, only the reference plane features were aligned, while the misalignment in individual

images caused the mean image of all the registered images to be blurred in the area of the measured sample (see Fig. 5-a). In contrast, when the proposed approach was applied we obtained the mean image shown in Fig. 5-b, where the desired surface was aligned well but the registration marks were blurred. The estimated surface plane's deviation is 1.24 millimeters bellow the reference plane, its normal vector elevation is 0.29° and its azimuthal angle is 176°.



standard registration proposed approach

Fig. 5. An extreme example of registration based on the reference plane (left) and based on the plane which represents the true surface of the sample found using the proposed method (right) when the heights of the reference plane and the plane of the surface differ considerably. Mean images of the 80 registered images are depicted here.

In the second experiment, five BTF samples were registered using standard and proposed approaches. The samples *wood01*, *fabric01*, *fabric02*, *fabric03*, and *leather01* were taken from the UTIA BTF database[2]. The results of our method are shown in Fig. 7 and mark a considerable improvement against the standard registration approach without alignment. The compression of data registered in a standard way (Fig. 7-b) leads to data visualization that is less sharp in comparison with the non-compressed aligned data (Fig. 7-a) considered the ground-truth. The compression after application of the proposed data registration method leads to considerably sharper images (Fig. 7-c). Note that in both cases the same compressed parametric representation is used (50 PCA components allowing real-time rendering). Registration of such a BTF sample comprising 6561 images typically takes around five hours on Intel Xeon 2.7 GHz using our Matlab implementation using six cores. However, due to the massive size of datasets (415 GB) much of this time is consumed by disk data transfer operations. Note that a smaller visible area of non-aligned datasets (Fig. 7-b) is due to cropping of visual artifacts at borders of individual misaligned images. As the proposed alignment method re-projects original locations of registration marks, the registered images are slightly shifted and scaled. Therefore, their fair pixel-wise comparison (e.g., using RMSE, SSIM) with the original image is impossible.

[2] `http://btf.utia.cas.cz`

Table 1. The estimated values of shift (z) and slant (θ,φ) for the tested samples

sample	height z [mm]	elevation θ	azimuth φ
wood01	-0.49	1.15°	101°
fabric01	-0.20	0.15°	128°
fabric02	0.12	0.20°	260°
fabric03	-0.30	0.43°	323°
leather01	0.10	0.05°	354°

As there is no robust texture-similarity measure available, we performed a psychophysical experiment with 5 naive subjects comparing Fig. 7-a with Fig. 7-b and c in a random order. The (c) was always perceived as more visually similar to the (a) than to the (b).

Tab. 1 shows estimated values of vertical position and orientation of the estimated plane with respect to the reference plane. The images show that the more the sample's plane deviated from the reference plane, the higher visual improvement was achieved as, e.g., for the sample of *wood01* in Fig. 7. From the values shown it is apparent that even when the sample is aligned with the registration plane as much as possible, the estimated differences are still relatively high. Finally, we remark that the visual effects of such misalignment are more pronounced if the resolution of captured images is higher.

Speed of the algorithm depends on the size of user-defined patches on the planar surface that are used for registration quality evaluation, as well as on a number of multi-view images. Fig. 6 shows execution times for a single iteration of the algorithm depending on the number of pixels and images processed. While the speed increases almost linearly with the number of pixels, it depends on the number of images n with $O(n^3)$ due to PCA compression used.

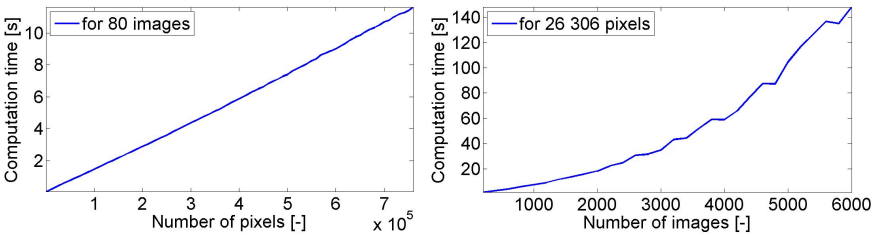


Fig. 6. Computational time of one iteration of the algorithm depends on the number of pixels used for quality evaluation (left), and on the number of processed images (right)

The proposed method is very robust. Its only obvious limitation is that it cannot guarantee a correct alignment for surfaces having wide height variations or several possible alignment heights (see, e.g., material *fabric03* in Fig. 7). However, even in such a case the material will be aligned to minimize the compression/rendering error. Additionally, only pixels which belong to the required height can be selected by a user-defined mask and can be taken into account during the registration to achieve even better alignment for such materials.

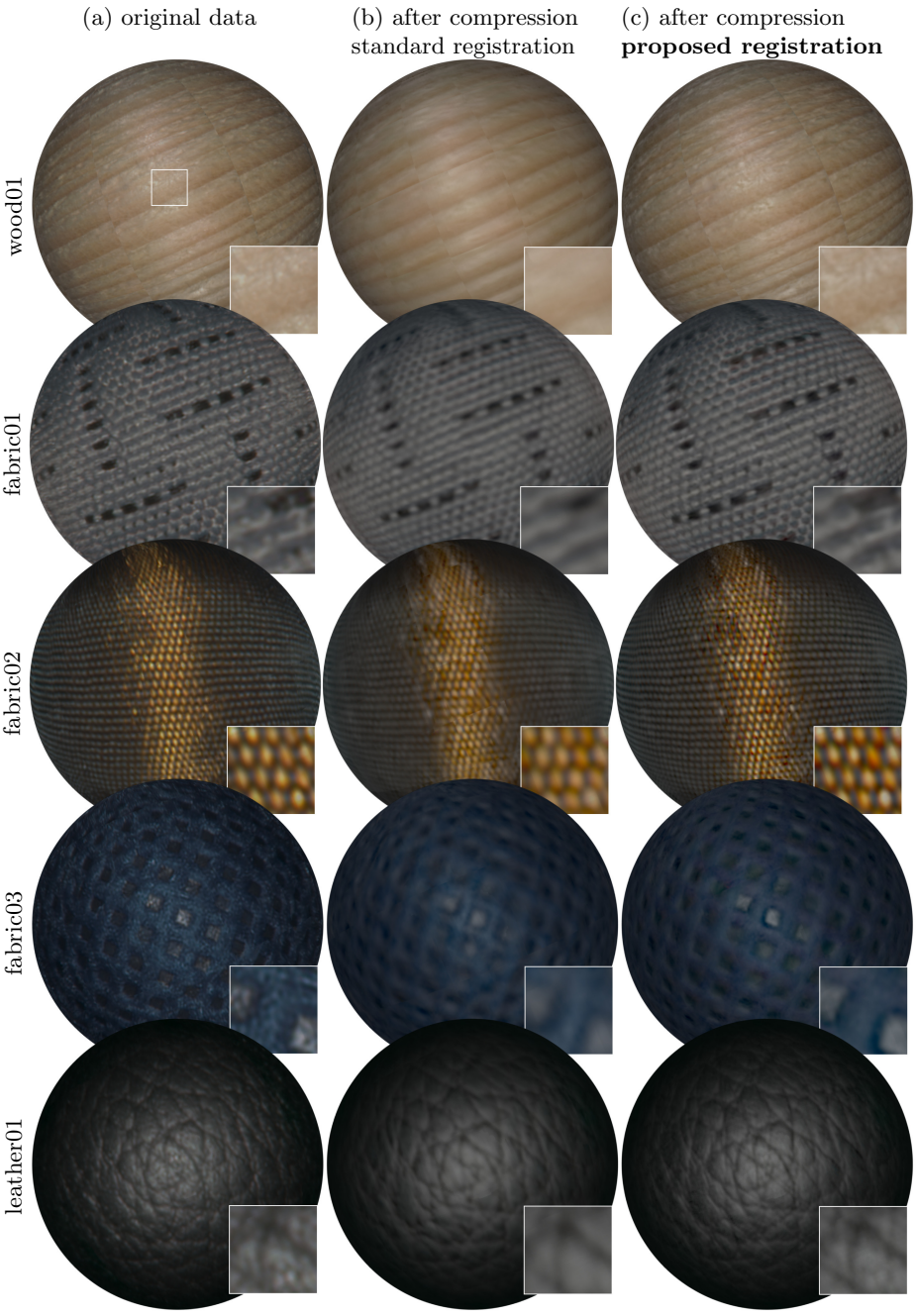


Fig. 7. A comparison of BTF visualization: (a) rendering using all 6561 images, (b) and (c) rendering from a compressed representation using only 50 eigen-images without and with application of the proposed alignment method

8 Conclusions and Future Work

In this paper we focus on the correct registration of multi-view images of planar material surfaces. Our approach exploits the fact that the reference registration plane and measured sample plane may be misaligned. When this misalignment is found and compensated from the measured dataset during the registration stage, a better material features alignment is achieved. Quality of the registration is verified by a reconstruction error of the data compression method. Consequently, the proposed approach allows more efficient application of multi-view data compression approaches, i.e., producing sharper images using the same size of compressed parametric representation. The proposed method is robust, easy to implement, and computationally efficient.

Acknowledgement. This work has been supported by GAČR grants 103/11/0335 and 102/08/0593, EC Marie Curie ERG 239294, and CESNET grant 409. We would like to thank Škoda-Auto a.s. for providing samples for measurement.

References

1. Dana, K., van Ginneken, B., Nayar, S., Koenderink, J.: Reflectance and texture of real-world surfaces. ACM Transactions on Graphics 18, 1–34 (1999)
2. Filip, J., Haindl, M.: Bidirectional texture function modeling: A state of the art survey. IEEE TPAMI 31, 1921–1940 (2009)
3. Gortler, S.J., Grzeszczuk, R., Szeliski, R., Cohen, M.F.: The lumigraph. In: Proceedings of SIGGRAPH 1996, pp. 43–54 (1996)
4. Sattler, M., Sarlette, R., Klein, R.: Efficient and realistic visualization of cloth. In: Proceedings of the 14th Eurographics workshop on Rendering, pp. 167–177 (2003)
5. Havran, V., Filip, J., Myszkowski, K.: Bidirectional texture function compression based on multi-level vector quantization. Computer Graphics Forum 29, 175–190 (2010)
6. Holroyd, M., Lawrence, J., Zickler, T.: A coaxial optical scanner for synchronous acquisition of 3D geometry and surface reflectance. ACM Trans. Graph. 29, 99:1–99:12 (2010)
7. Müller, G., Meseth, J., Sattler, M., Sarlette, R., Klein, R.: Acquisition, synthesis and rendering of bidirectional texture functions. Computer Graphics Forum 24, 83–109 (2005)
8. Neubeck, A., Zalesny, A., Gool, L.V.: Viewpoint consistent texture synthesis. In: Proceedings of the 3D Data Processing, Visualization, and Transmission, 2nd International Symposium, pp. 388–395 (2004)
9. Koudelka, M., Magda, S., Belhumeur, P., Kriegman, D.: Acquisition, compression, and synthesis of bidirectional texture functions. In: Texture 2003, pp. 47–52 (2003)
10. Müller, G., Meseth, J., Klein, R.: Compression and real-time rendering of measured BTFs using local PCA. In: Vision, Modeling and Visualisation, pp. 271–280 (2003)
11. Müller, G., Sarlette, R., Klein, R.: Data-driven local coordinate systems for image-based rendering. Computer Graphics Forum 25 (2006)
12. Ruiters, R., Klein, R.: Heightfield and spatially varying BRDF reconstruction for materials with interreflections. Computer Graphics Forum 28, 513–522 (2009)

13. Müller, G., Bendels, G.H., Klein, R.: Rapid synchronous acquisition of geometry and BTF for cultural heritage artefacts. In: The 6th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST), pp. 13–20 (2005)
14. Weinmann, M., Schwartz, C., Ruiters, R., Klein, R.: A multi-camera, multi-projector super-resolution framework for structured light. In: International Conference on 3DIMPVT, pp. 397–404 (2011)
15. Szeliski, R.: Image alignment and stitching: A tutorial. Found. Trends. Comput. Graph. Vis. 2, 1–104 (2006)
16. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2000) ISBN: 0521623049
17. Zhang, Z.: Flexible camera calibration by viewing a plane from unknown orientations. In: ICCV, vol. 1, pp. 666–673 (1999)

Automatic Stave Discovery for Musical Facsimiles

Radu Timofte[1] and Luc Van Gool[1,2]

[1] ESAT-VISICS /IBBT, Catholic University of Leuven, Belgium
[2] D-ITET, ETH Zurich, Switzerland

Abstract. Lately, there is an increased interest in the analysis of music score facsimiles, aiming at automatic digitization and recognition. Noise, corruption, variations in handwriting, non-standard page layouts and notations are common problems affecting especially the centuries-old manuscripts.

Starting from a facsimile, the current state-of-the-art methods binarize the image, detect and group the staff lines, then remove the staff lines and classify the remaining symbols imposing rules and prior knowledge to obtain the final digital representation. The first steps are critical for the performance of the overall system.

Here we propose to handle binarization, staff detection and noise removal by means of dynamic programming (DP) formulations. Our main insights are: a) the staves (the 5-groups of staff lines) are represented by repetitive line patterns, are more constrained and informative, and thus we propose direct optimization over such patterns instead of first spotting single staff lines, b) the optimal binarization threshold also is the one giving the maximum evidence for the presence of staves, c) the noise, or background, is given by the regions where there is insufficient stave pattern evidence.

We validate our techniques on the CVC-MUSCIMA(2011) staff removal benchmark, achieving the best error rates (1.7%), as well as on various, other handwritten score facsimiles from the Renaissance.

1 Introduction

There is a growing interest in the automated digitization and transcription of handwritten music score facsimiles [1,2]. Of particular interest are old musical manuscripts, to be brought into a modern form as to prevent their music from being forgotten.

Most of the state-of-the-art Optical Music Recognition (OMR) systems share the same processing steps. First, operations such as enhancement, contrast stretching, or color normalization are applied. Second, a binarized image is obtained by means of local content-adaptive or global color thresholding. Third, the staff lines are detected and removed. Fourth, the musical symbols are segmented and classified. The final step typically combines symbol recognition with relative stave positioning, often exploiting music-specific rules. At that point, the transcription into modern formats (which can very well be digital, like MIDI) is

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 510–523, 2013.

possible. Obviously, the first processing steps are critical for the overall OMR performance. As a matter of fact, the current OMR systems have serious difficulties in dealing with handwritten scores and especially those with non-standard page layouts and deviating, century-old notations (*e.g.* Renaissance). Noise, corruption and variations in handwriting are other common issues that one OMR needs to address.

We contribute an automatic process for stave discovery based on the accumulated evidence of staves as 5-groups of staff lines, integrating binarization, stave and staff line detection, staff line and background removal.

We propose to handle binarization, staff line detection and noise removal by means of dynamic programming (DP) formulations. Our main insights are: a) the staves (the 5-group of staff lines) are more constrained and therefore more informative, and thus we propose a direct optimization over such patterns instead of first spotting single staff lines, b) the optimal binarization threshold also is the one giving the maximum evidence of the presence of staves, c) the noise or background, i.e. regions not containing musical score information, corresponds to the regions without sufficient evidence for the presence of stave patterns.

Binarization of musical score facsimiles aims at the accurate separation of the musical information (staves and musical symbols) from the paper support (background). The most employed techniques are Otsu thresholding and local adaptive thresholding over the grayscale image [3]. While there were many attempts at improving these basic thresholding schemes, few explicitly use the musical content for driving the binarization process. We will assume that the best separation is achieved when we have the strongest cumulated evidence for staves as repetitive line patterns. The approach of [4] probably comes closest and uses the distribution of vertical run-lengths of paired subsequent black and white segments to select the best gray-level threshold. Those paired run-lengths provide the most likely staff line thickness and following spacing. Our approach, on the other hand, does not stop at global statistics of run-length codes, but computes the local constrained stave evidence, thus more strongly exploiting the musical score properties. The best cumulated stave evidence for an image is spotting the best parameters for automatic stave discovery: gray-level threshold, staff line thickness, and inter-line spacing. A more detailed description comes later.

Staff detection and removal aims at localizing the staves as major components for musical score documents and at accurately removing the staff lines as to ease musical symbol recognition. The defining characteristics of staffs are their line thickness and the inter-line spacing within the staves. The detection of staff lines is affected by the fact that in most real musical facsimiles they exhibit different kinds of deformations such as bending, interruptions, pixel noise, intersections with other symbols, and deviations from constant line thickness and spacing. Horizontal and vertical projections [5,6] are simple ways to detect staff lines when the deformations are (very) small. More robust approaches are based on line tracking [7,8], staff segments [9,10], Skeleton fitting [1], stable path search [2] and staff line shape determination [11].

The removal techniques assume a prior detection step was performed to then remove the staff lines. They focus on the intersections with other musical symbols, as well as on the local line thickness. The algorithms can be grouped into those that apply line tracking, vector fields, run-lengths and skeletonization [1].

Most staff detection algorithms are based on the 3 steps of spotting the staff lines individually, staff line grouping into staves, and stave validation. Our approach, on the other hand, considers each stave as a whole, i.e. as an equi-distant repetition of staff lines. These patterns are important to impose context, through the power of the entire group of staff lines. The stave patterns are likely to show up in vertical slices through the score image. After detecting such local evidence, a dynamic programming formulation links up such evidence, crossing the page from left to right and also from right to left. The cumulation of the evidence should ideally lead to the same stave twice. Finally, the horizontal extent of the staves is determined, i.e. until where exactly they run, which includes finding possible interruptions breaking up the staves. To that end, a second dynamic programming algorithm makes the distinction between staves and background.

Note that the ***use of DP*** is not novel for staff detection. The stable path [2] and the candidate point matching [10] techniques most resemble our DP formulation for staff detection. The stable paths [2] are in fact iteratively extracted shortest paths (and thus individual staff line candidates) over a built graph representation of the binarized image, after which they are grouped into staves. The candidate point matching technique [10] is optimizing the assignment of staff line candidate points in neighboring vertical scan lines using a penalized edit distance formulation, thus the potential staff lines are propagated across the vertical scan lines. We, on the other hand, impose a pattern (group of lines) for stave (and not staff line!) detection and optimize in a DP formulation the evidence for such stave propagation across the vertical scan lines. Thus, our DP formulation uses patterns instead of points and accumulates the highest possible evidence for complete staves.

In summary, ***our contributions*** are: a stave-based staff detection method and principled ways for automatic binarization and stave parameter discovery.

We validate our techniques on the CVC-MUSCIMA (2011) staff removal benchmark [12], achieving the best error rates (1.7%) and on various handwritten score facsimiles from the Renaissance.

The structure of the remainder of the paper is as follows. Section 2 introduces our proposed methods for binarization and staff detection and removal focusing on the most innovative aspects. Section 3 describes the experimental setup and discusses the obtained results. Finally, the conclusions are drawn in Section 4.

2 Proposed Method

The overall procedure of our proposed binarization and staff line detection and removal technique is depicted in Fig. 1. First, for each threshold and the corresponding binarized image, stave evidence is summed for various staff line thicknesses and staff inter-spacings. The best evidence is considered to represent the

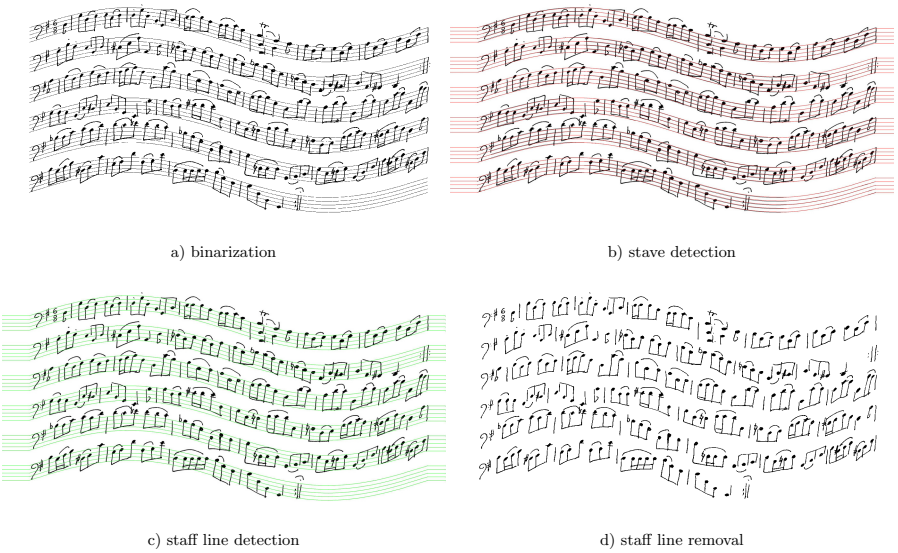


Fig. 1. Processing steps on a CVC-MUSCIMA deformed score

real stave content of the image. Second, for the best global threshold and estimated staff line thickness and staff spacing we trace back the staves. Third, for each traced stave proposal we post-label it into stave information and background, thus segmenting the musical content from the surroundings. At last, the final image with musical symbols is obtained after the removal of the detected staff lines. The accurate staff line detection is necessary since the loose stave model does not fit precisely the staff lines. In Fig. 1 is shown that the red lines predicted by stave detection step are less precise than the green lines from the accurate staff detection step, in following the black staff lines.

In the rest of this section we focus on stave detection using the DP formulation, then on how the summed stave evidence is used for automatic discovery of stave parameters and the grayscale threshold for binarization, and finally we describe the staff line removal part.

2.1 Stave Detection

The input of our stave detection procedure is a binary musical document. The staff lines and musical symbols are marked with '1', while the background is '0' (see Fig. 2a). The staff line thickness (α) and staff line spacing (β) are considered fixed for the moment. Section 2.4 describes how they are determined.

Along each of a number of regularly spaced vertical page sections, corresponding to the image's columns, a pixel-wise stave evidence score is calculated for the black-and-white pattern just below. This score quantifies the degree to which 5 black runlengths of more or less equal length are alternating with white runlengths, also of equal length. Such runlength sequence is illustrated in Fig. 2b.

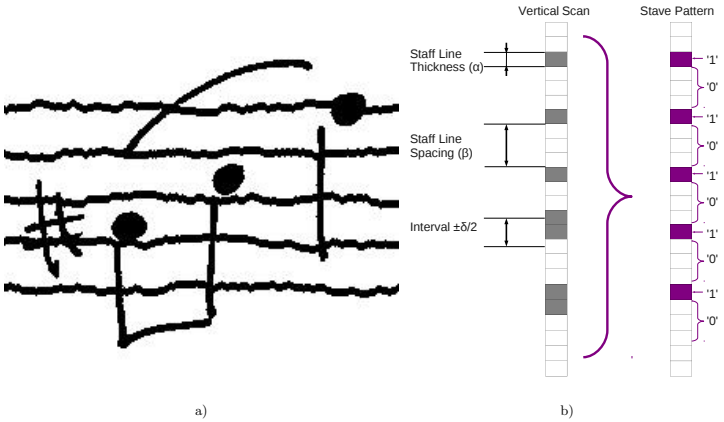


Fig. 2. Stave and pattern example

The ideal situation is shown in the figure on the right, but, of course, in reality one has to allow for some slack on these lengths, as shown on the left and as discussed later when we describe the staff line removal process. In any case, we assume that each staff line is at least one black pixel ('1') thick and, similarly, that each inter-staff spacing is at least one white pixel ('0') thick. As to the allowable tolerance in the relative positions of the staff lines, we assume that there can be an independent jitter on each. The need for this tolerance is illustrated in Fig. 2a. Each of the staff lines are assumed to lie within an interval symmetrical about their nominal position (i.e. when following a strict succession of intervals with width α and β). The width of this jitter interval is given by δ, defined as

$$\delta = \min\{2 \times \alpha, \alpha + \beta - 2\} \tag{1}$$

We subtract 2 to avoid counting 2 neighboring black pixels as evidence for 2 neighboring staff lines. We validate this insight experimentally on several scores. A similar formula is used for the maximum staff line thickness, see equation (7).

For a point at image position (i, j), the local stave evidence score is computed as follows. Let M be the binarized image, then the local stave evidence $v(i, j)$ at position (i, j) of our pattern of K staff lines (always $K = 5$ in our case) is given as:

$$v(i,j) = \sum_{k=0}^{K-1} \max_{t \in \{-\lfloor \delta/2 \rfloor, \cdots, \lfloor \delta/2 \rfloor - 1\}} M(t + i + kL, j) + \sum_{k=0}^{K-1} (1 - \min_{t \in \{\lfloor \alpha/2 \rfloor, \cdots, L - \lfloor \alpha/2 \rfloor - 1\}} M(t + i + kL, j)) \tag{2}$$

where α is the staff line thickness, β is the staff spacing, $L = \alpha + \beta$. This score counts the number of times the nominal positions of staff lines and spacings coincide with the corresponding black and white regions, resp. Each such match is only counted once. When also taking a spacing below the lowest staff line,

the maximum score is $2K$. At first glance $v(i,j)$ is a very permissive pattern evidence score, but later, when using the context information under the form of the stave evidence in pixels from the neighboring columns, extra smoothing constraints will be imposed for propagating the global stave evidence.

In the next step, we integrate the local stave evidence scores across the vertical sections (columns). A dynamic path search algorithm accumulates local stave evidence scores, trying to keep that sum maximal, while moving from left to right. Note that the vertical scan lines impose a strict order, not allowing for the path to meander heavily. Moreover, in this left-to-right propagation a smoothing constraint is applied that penalizes vertical displacements. When selecting a pixel i in the next column j, the recursive DP evidence is obtained as

$$\begin{aligned} c(i,1) &= v(i,1), \\ c(i,j) &= v(i,j) + \max_{k\in\{-2,-1,0,1,2\}} (c(i+k, j-1) - |k|\gamma) \end{aligned} \tag{3}$$

where γ is the penalty set to $\lceil 2K + K/2 \rceil = 13$ and empirically validated on several scores.

Tracing back we obtain the candidate stave paths with the highest evidence. Applying a second pass of the stave detection DP algorithm in reverse order of the scans, allows us to detect the stable paths as in [2]. That is, those paths that regardless of the direction of the DP computation accumulate the same evidence. These stable paths are our stave detections. The procedure can be repeated after masking the already detected staves to extract staves with lower evidence. Note that the number of repetitions needed is bounded by the worst case number, given by the number of rows of M divided by $K(\alpha+\beta)$, where K is the number of staff lines in the stave pattern. In practice, for most of the binarized images we run up to 2 iterations to find all the staves. The technique assumes that one stave path always connect from the first to the last column. If the staves are vertical, or close to, this is detected and the score is rotated such that the staves are horizontal and the assumption holds for applying the technique. Since the technique handles well large rotations, for uncontrolled scores, the best is to run the whole stave discovery process for both original and 90 degrees rotated score and pick the rotation with the largest cumulated stave evidence.

2.2 Stave Segmentation

Once we have detected a stave (see Section 2.1), under the form of a path connecting the first up to the last column, we want to segment the real musical content (stave and musical symbols) from the surrounding background. We remind the reader that a path (stave detection) connects single pixels at positions p_j from each column j in an ordered way.

We reuse the local stave evidence scores ($v(i,j)$) and the smoothness as implemented previously. Moreover, we impose prior knowledge under the form of a minimum stave segment width (λ_1) and a minimum background segment width

(λ_0). In this way we avoid segments that are too short. Thus, we have a two-class labeling problem ('stave'='1' and 'background'='0'). The recursive DP formulation for the stave segmentation is:

$$\begin{aligned}
d('1',1) &= v(p_1,1) - \theta_1,\\
d('0',1) &= \theta_0 - v(p_1,1),\\
d('1',j) &= \max\{v(p_j,j) - \theta_1 + d('1',j-1) - |p_j - p_{j-1}|\gamma,\\
&\qquad \sum_{i=j}^{i=j-\lambda_0+1} (\theta_1 - v(p_i,i)) + d('0',j-\lambda_0)\}\\
d('0',j) &= \max\{\theta_0 - v(p_j,j) + d('0',j-1),\\
&\qquad \sum_{i=j}^{i=j-\lambda_1+1} (v(p_i,i) - \theta_0) + d('1',j-\lambda_1)\}
\end{aligned} \tag{4}$$

where for each label class we empirically set the thresholds deciding where it is more likely to have staves and where not as $\theta_1 = 9.5$ and $\theta_0 = 7$. We are taking $\lambda_0 = 2 \times (\alpha + \beta)$ tolerating background segments no shorter than twice the staff height (thickness plus spacing), and $\lambda_1 = 5 \times (\alpha + \beta)$, thus tolerating stave segments no shorter than the height of the stave (a 5-group of staff lines).

Tracing back for the best label solution gives us the optimal segmentation.

2.3 Accurate Staff Detection

In practice, the currently extracted staff line trajectories and the subsequent musical content (staves) segments are not precise enough (see Fig. 1b) to apply staff line removal. In this section, we refine such data. For this purpose, we use the current stave detections to define corridors where we can search for the accurate positions of the staff lines. The corridors are $\alpha+\beta$ wide and centered on the staff line position predicted by the stave detection. In this step, the different staff lines are refined more individually.

The local evidence w of staff line existence at pixel (i,j) inside the corridors is calculated as follows. A column template corresponding to a group of three staff lines with nominal width α and β spacings in between is considered. This template is positioned with its center within the staff line corridor. Beneath the three staff line intervals (each of width α) the system looks for black pixels. Each of these pixels contributes with their own weight, with emphasis on those in the middle staff interval. Moreover, within an interval of δ high around the middle staff interval, but excluding the latter, white pixels are sought. More precisely, the local evidence is now calculate as:

$$\begin{aligned}
w(i,j) = &\sum_{k=i-\lfloor\alpha/2\rfloor}^{i+\lfloor\alpha/2\rfloor} (\sigma M(k,j) + M(k-(\alpha+\beta),j) + M(k+(\alpha+\beta),j))\\
&+ \sum_{k=1}^{\lfloor\delta/2\rfloor} (2 - M(i-\lfloor\alpha/2\rfloor - k, j) - M(i+\lfloor\alpha/2\rfloor + k, j))
\end{aligned} \tag{5}$$

where δ is given by eq. (1) and we empirically set $\sigma = 3$. This procedure is repeated with the template moved within the corridors of each single staff line. For the pixels (i, j) outside the corridors or at their boundaries $w(i, j) = -\infty$.

The evidence propagation is solved using the same DP formulation from stave detection:

$$\begin{aligned} f(i,1) &= w(i,1), \\ f(i,j) &= w(i,j) + \max_{k\in\{-2,-1,0,1,2\}} (f(i+k, j-1) - |k|\gamma_s) \end{aligned} \tag{6}$$

with a different γ_s as penalization term enforcing smoothness. In our experiments γ_s is set to half from maximum local evidence, $\gamma_s = 0.5(\delta + \alpha(\sigma + 2))$.

Note that we can solve the evidence propagation for all the corridors in one pass through the image. Tracing back the best cumulated evidence path along each corridor provides the accurate staff line detection (see Fig. 1c).

2.4 Stave Parameter Determination

The main characteristics for a stave are the number of staff lines (5 lines are common for Western musical scores), the staff line thickness (α), and the inter-staff spacing (β). We propose to use the overall stave pattern evidence to drive the process of fitting the best thickness and spacing. For this, we start from an interval for L, defined as the sum of these two parameters and we assume that we have stave patterns on a page. Then, we start the search for the optimal values in a crude fashion. First we investigate the possible ($\alpha = \beta$ or $\alpha + 1 = \beta$) and $L = \alpha + \beta$. The L values for which the stave detection returns a sufficiently high overall evidence are then considered in more detail. For those values, we test all combinations $\alpha \in \{1, \cdots, L/2\}$ and $\beta = L - \alpha$ and pick the combination returning the highest overall evidence.

2.5 Threshold Selection for Binarization

At the start of the algorithm, we look for a global threshold, applied to the intensity values (and not to color, similar to [3]). This thresholding operation is supposed to split music symbols and staves from background. In reality, parts of the illustrations and background blotches will end on the music side. Whereas Pinto et al. [3] consider vertical run lengths to drive the search, we once more use the staff evidence as computed using the stave detection procedure, applied with different threshold candidate values (see sections 2.1 and 2.4). It is assumed that the best evidence is achievable using the proper set of parameters (threshold, staff line thickness and staff spacing). For a chosen threshold, we extract staves, then compute the staff line evidence for all the detected staves. Summing up for all staff lines we have a staff evidence at score level. We offer the insight, verified by experiments, that the best such evidence is achieved for the proper binarization threshold. As with α and β not all threshold values need to be tested. An Otsu threshold extracted from the entire image tends to narrow down the range of potential threshold values quite effectively.

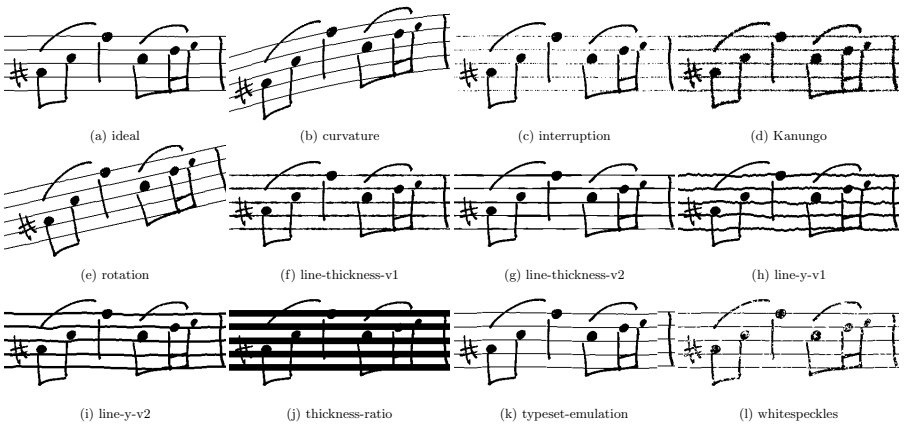


Fig. 3. Deformation types in CVC-MUSCIMA benchmark

2.6 Staff Line Removal

After stave discovery and accurate staff line detection, we remove the staff line pixels to serve a possibly following music symbol classification with a cleaner set of symbols. The problems come from intersections between the staff lines and those symbols, and from the fact that the detected staff lines do not match the real lines exactly, due to small errors or interpolation issues. The interpolation issues are caused by the fact that DP picks the shortest path for interpolation across gaps in the stave/staff evidence, which may not be the one with the curvature of the flanking portions. However, if not all staff lines have a gap simultaneously, the complete ones will drive the course of the interpolated ones.

Here we use one of the simplest ways to handle the staff line removal. It follows the idea from [11]. The vertical segments intersecting the detected staff line are considered to belong to the line if the segment length is less than the maximum staff line thickness. Otherwise, they are considered part of the music symbols. The maximum staff line thickness is empirically set to:

$$\delta = \min\{2 \times \alpha, \alpha + \beta\} \tag{7}$$

3 Experiments

3.1 Parameter Setting

The presented automatic stave discovery method encompasses several distinct problems with specific algorithmic solutions. One can note that we have a large number of parameters. The parameters can be categorized into fixed (K), automatically determined (α,β,L), derived from those (δ,γ,λ_0,λ_1), and manually tuned (γ_s,θ_0,θ_1,σ).

Thus, the number of staff lines per stave is set to $K = 5$ for the musical scores considered here. The binarization threshold, staff line thickness α, spacing β and $L = \alpha + \beta$ are automatically determined. The maximum bound (δ), the penalty

(γ), the minimum allowed stave segment width (λ_1) and background segment width (λ_0), depend solely on the fixed and automatically determined parameters. However, the user only needs to set the penalty γ_s, the likeliness thresholds (θ_0, θ_1) and the weight σ which depend on the noise and conditions from the score images used as input. While we have tried automatic methods for adaptively set these parameters to different input conditions, we do not have yet a general solution.

3.2 Staff Detection and Removal

The proposed method is qualitatively evaluated on real Renaissance facsimiles[1] and quantitatively evaluated for staff removal on the CVC-MUSCIMA benchmark [12] as used during the competition held at ICDAR 2011 [13]. The dataset consists of 1,000 handwritten music score images from 50 writers and the images are (synthetically) distorted using 11 deformation models (see Fig. 3). Each deformation model generates 1,000 images. In total, the dataset contains 12,000 images equally divided into training data and testing data. These images are already binarized. The raw images are not available. For each of the binarized images the dataset also provides the ground truth binary image with the staff lines removed. To the best of our knowledge, it is the largest benchmark for staff line removal evaluation.

Our results are evaluated using the pixel based metric as used for the competition, where the error rate is computed as described in [1,13]:

$$E.R. = 100 \times \frac{\#misclassified\ sp + \#misclassified\ non\ sp}{\#all\ sp + \#all\ non\ sp} \tag{8}$$

where # means "number of" and sp = stands for "staff line pixels". Only the musical content pixels are considered for classification ('black'/'1' pixels in our paper, see Fig. 1a).

First, in Table 1, we show quantitative results of our method on the competition training dataset. Thus, we compare with the results reported by [11] for this training (!) set. Our method performs very well for each kind of deformation, which is not the case for the reference method [11], which performs weakly on the 'large thickness ratio' images. The average error rate is 1.5%, which compares favorable even when the other method does not account for the failure cases.

Second, in Table 1 we provide the more relevant results for the competition testing dataset and compare also with the top 3 performing methods (out of 8 entries) [13,11]. ISI01-HA is the winning method of the competition [13], NUG04-LTr is the line tracking method [1] and INP02-SP is the stable path method of [2]. Our proposed method performs the best overall, works for all kinds of deformations (like ISI01-HA does), and clearly improves over the stable path method (INP02-SP). The stable path method follows a DP formulation based on iteratively computing shortest paths (thus potential staff lines) and post grouping these lines into staves. We, on the other hand, directly compute the

[1] Qualitative results are included in the supplementary material.

Table 1. Results comparison on CVC-MUSCIMA staff removal dataset

	Error Rates (%)						
	Testing data					Training data	
Deformation Type	[11]	NUG04-LTr	ISI01-HA	INP02-SP	Ours	[11]	Ours
Ideal		2.08	1.50	1.50	**1.34**	1.33	
Curvature		100	1.66	1.80	**1.34**	1.43	
Interrupted		100	0.91	6.1	**0.84**	1.02	
Kanungo		4.33	2.84	2.86	**2.59**	2.84	
Rotation		100	1.76	2.03	**1.65**	1.65	
Line Thickness-variation-v1		3.74	2.17	2.70	**1.85**	3.62	
Line Thickness-variation-v2		3.74	2.15	3.01	**1.90**	2.89	
Staff Line-y-variation-v1		5.94	1.89	2.43	**1.51**	4.58	
Staff Line-y-variation-v2		3.73	1.83	2.27	**1.40**	3.64	
Thickness ratio	N/A	10.78	**2.86**	6.89	4.05	N/A	
Typeset-emulation		4.83	1.60	1.60	**1.34**	2.09	
White speckles		1.76	1.48	1.73	**1.39**	1.37	
Overall Error Rate	1.95	28.41	1.89	2.91	**1.76**	2.41	**1.57**

staves as entities, using a DP formulation, rather than individual staff lines. The results show that this is beneficial. If we do not employ the accurate staff detection step in our pipeline the performance significantly degrades and the overall error rate increases from 1.7% to 3.1% on the testing set.

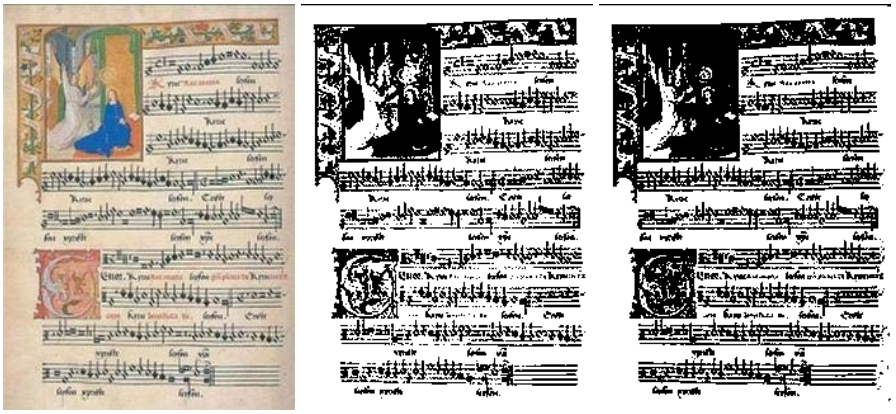
3.3 Binarization

We qualitatively assess our proposed stave evidence driven binarization method on facsimiles from the Renaissance. In Fig. 4 we depict a difficult case with very low resolution (230x320), blur, and an increased amount of non-musical score content (pictures). Our method picks a grayscale global threshold level such that the stave evidence is maximized. On the other hand, Otsu's method finds a good separation of the image pixels into two classes, thereby not guaranteeing that this global threshold is reasonable to separate music symbols and staves from the non-uniform background. However, while our method is able to provide an optimal global threshold for stave evidence, it is not clear if the same threshold is optimal for musical symbol classification after staff line removal. The best solution for automated music transcription may therefore come out to first segment the musical content using the staves (as in Fig. 5d) and then to get the optimal threshold using also the musical symbols, not only the staff lines, for those parts surviving the first threshold. The global thresholding methods (such ours and Otsu's) were proved to perform worse than the local adaptive thresholding methods [3]. However, we did not quantitatively evaluate our global method against local methods.

3.4 Renaissance Facsimiles

The proposed automatic stave discovery method is qualitatively evaluated on hundreds of handwritten facsimiles from the Renaissance.[2]

[2] Qualitative results are included in the supplementary material.



a) facsimile (230 × 320 pixels) b) Otsu method c) Ours

Fig. 4. Binarization example

Fig. 5 shows partial results during the automated stave discovery process. This particular facsimile has low resolution (858 × 600), quite some distortions due to the non-planar paper support, residual noise transpiring from the other side of the paper, the omnipresent pictures and text, and the variations in the handwriting. Moreover, the musical content region is relatively small *w.r.t.* the whole image space and there are two columns of staves with different lengths.

The stave discovery led to a visually meaningful binarization (Fig. 5b) with an excellent (and correct) stave detection as well (see Fig. 5c), the blue-green lines). The segmentation into musical content (green colored segments in Fig. 5c) and non-musical content (blue colored segments in Fig. 5c) is able to cope with large gaps between the staves and with the presence of noise. Moreover, the final content segmentation keeps most of the musical information (see Fig. 5d).

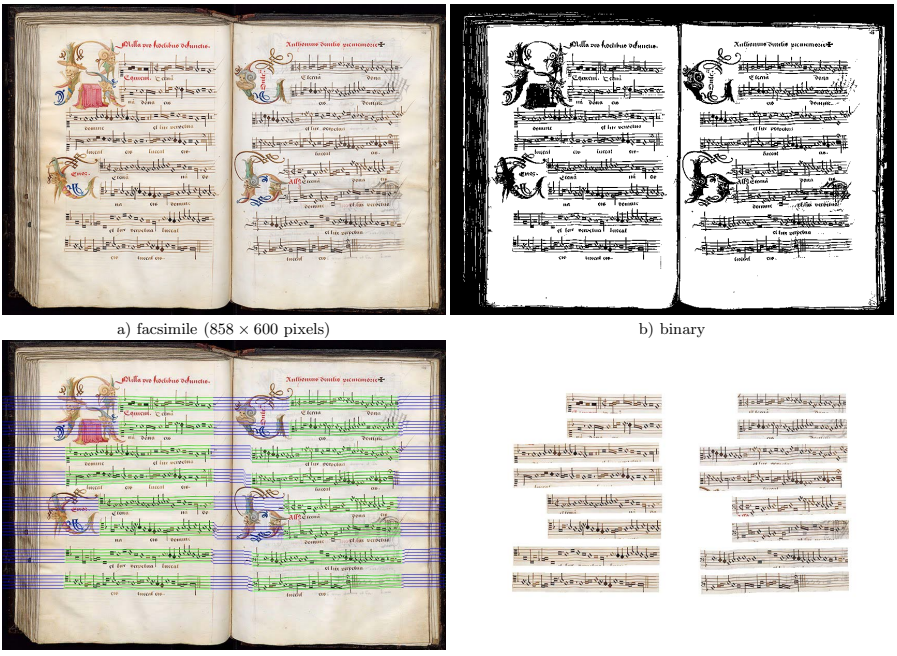
3.5 Limitations and Further Research

The proposed staff line detection and removal method is robust and effective for different kinds of deformed musical facsimiles, but it has some limitations.

First, the method does not interpolate the missing evidence (interruptions) in the facsimile, but rather tends to find the shortest path in an unnatural way. Using the path line extrapolation as extra smoothness term, shape line priors, or post-smoothing by line interpolation could mitigate this problem.

Second, the staff line removal criteria use only the staff line thickness, but context and crossings information should improve the performance.

Third, while we made a large step forward into automated binarization, parameter selection, and staff line/noise removal, the running time is linearly dependent on the number of pixels, which can be very large. A multi-scale strategy is desirable, starting with the DP optimization at coarser levels and only adding details to the finer levels. The running time of our single-thread CPU C++ unoptimized code for stave detection, accurate staff detection and staff line removal



a) facsimile (858 × 600 pixels) b) binary

c) stave detection d) musical content segmentation

Fig. 5. Renaissance facsimile example

is, on a Core 2 Quad 2009 desktop, on average less than 1 second per binarized image in the CVC-MUSCIMA dataset, where the average image is $\sim 2000 \times 3500$ pixels.

While relatively few work on this automatic stave discovery problems, this goes for many other applications. But exposing the community to applications is inspiring. Moreover, the problems dealt with are akin to wider issues, like text page layout analysis. One can think of an extension to handwritten documents with similar ruled but irregular structure.

4 Conclusions

A novel DP-based music staff line detection and removal technique for musical documents has been proposed. It uses the fact that the staff lines are grouped into staves following constrained patterns. Vertical slices of the document can reveal the stave patterns. Using DP we can propagate in one pass the stave evidence from one side of the document to another. The best staves are generated by tracing back the solutions. Another DP formulation optimally solves the labeling of such tracked solution into background and stave segments. This leads to accurate stave / background segmentations and thus noise removal. Also, we propose the maximum evidence criterion for binarizing the images. That is, we made the successful assumption that the staves will accumulate the largest

evidence for the best binarized musical score image. The parameters of a stave such as staff line thickness and spacing were derived in the same way.

The proposed technique is the top performer on the latest musical staff line removal benchmark under various deformations. Also, the techniques exhibit robustness and effectiveness on low quality facsimiles with Renaissance handwritten musical scores.

Acknowledgement. This work was supported by the Flemish IWT/SBO project ALAMIRE.

References

1. Dalitz, C., Droettboom, M., Pranzas, B., Fujinaga, I.: A comparative study of staff removal algorithms. IEEE Trans. Pattern Anal. Mach. Intell. 30, 753–766 (2008)
2. dos Santos Cardoso, J., Capela, A., Rebelo, A., Guedes, C., da Costa, J.P.: Staff detection with stable paths. IEEE Trans. Pattern Anal. Mach. Intell. 31, 1134–1139 (2009)
3. Pinto, T., Rebelo, A., Giraldi, G.A., Cardoso, J.S.: Music score binarization based on domain knowledge. In: Pattern Recognition and Image Analysis - 5th Iberian Conference (IbPRIA), pp. 700–708 (2011)
4. Cardoso, J.S., Rebelo, A.: Robust staffline thickness and distance estimation in binary and gray-level music scores. In: 20th International Conference on Pattern Recognition, pp. 1856–1859 (2010)
5. Blostein, D., Baird, H.S.: A critical survey of music image analysis. In: Structured Document Image Analysis (1992)
6. Fujinaga, I.: Staff detection and removal. In: George, S.E. (ed.) Visual Perception of Music Notation: On-Line and Off Line Recognition, pp. 1–39. IGI Global, Hershey (2004)
7. Bainbridge, D., Bell, T.C.: Dealing with superimposed objects in optical music recognition. In: 6th International Conference on Image Processing and its Applications, pp. 756–760 (1997)
8. Randriamahefa, R., Cocquerez, J., Fluhr, C., Pepin, F., Philipp, S.: Printed music recognition. In: International Conference on Document Analysis and Recognition, pp. 898–901 (1993)
9. Dutta, A., Pal, U., Fornés, A., Lladós, J.: An efficient staff removal approach from printed musical documents. In: 20th International Conference on Pattern Recognition, pp. 1965–1968 (2010)
10. Miyao, H.: Stave Extraction for Printed Music Scores. In: Yin, H., Allinson, N.M., Freeman, R., Keane, J.A., Hubbard, S. (eds.) IDEAL 2002. LNCS, vol. 2412, pp. 562–634. Springer, Heidelberg (2002)
11. Su, B., Lu, S., Pal, U., Tan, C.L.: An effective staff detection and removal technique for musical documents. In: 10th IAPR International Workshop on Document Analysis Systems (2012)
12. Fornes, A., Dutta, A., Gordo, A., Llados, J.: CVC-MUSCIMA: A ground-truth of handwritten music score images for writer identifcation and staff removal. International Journal on Document Analysis and Recognition, 1–9 (2011)
13. Fornés, A., Dutta, A., Gordo, A., Lladós, J.: The ICDAR 2011 music scores competition: Staff removal and writer identification. In: ICDAR, pp. 1511–1515 (2011)

Unsupervised Language Learning for Discovered Visual Concepts

Prithwijit Guha[1] and Amitabha Mukerjee[2]

[1] Department of Electronics & Electrical Engineering, IIT Guwahati
pguha@iitg.ernet.in
[2] Department of Computer Science & Engineering, IIT Kanpur
amit@cse.iitk.ac.in

Abstract. Computational models of grounded language learning have been based on the premise that words and concepts are learned simultaneously. Given the mounting cognitive evidence for concept formation in infants, we argue that the availability of pre-lexical concepts (learned from image sequences) leads to considerable computational efficiency in word acquisition. Key to the process is a model of bottom-up visual attention in dynamic scenes. We have used existing work in background-foreground segmentation, multiple object tracking, object discovery and trajectory clustering to form object category and action concepts. The set of acquired concepts under visual attentive focus are then correlated with contemporaneous commentary to learn the grounded semantics of words and multi-word phrasal concatenations from the narrative. We demonstrate that even based on mere 5 minutes of video segments, a number of rudimentary visual concepts can be discovered. When these concepts are associated with unedited English commentary, we observe that several words emerge - more than 60% of the concepts discovered from the video are associated with correct language labels. Thus, the computational model imitates the beginning of language comprehension, based on attentional parsing of the visual data. Finally, the emergence of multi-word phrasal concatenations, a precursor to syntax, is observed where there are more salient referents than single words.

1 Introduction

It is now widely accepted that infants acquire some visual concepts before they acquire language. While a large body of work in Computer Vision deals with associations between images and language, it is surprising that the acquisition of visual concepts has not been used to drive the learning of associations from language.

We can classify work that considers both language and image data into three groups. The first *keyword* group considers simple linguistic inputs such as keywords or small phrases, and attempts to relate these to segmented scene regions in mostly static images [1,2]. Recent extensions of this line of work have considered text fragments and word sequences [3,4]. The second, *meta-data* group uses complex scenes and videos, but uses the linguistic component only as meta-data to be able to better classify the visual input [5,6]. In both these bodies of

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 524–537, 2013.

research, the linguistic input is secondary to the visual. In the third, *semantic* class, the emphasis is more on language, and the images are relatively simple. Early work in this genre made extensive use of prior knowledge about the domain - e.g. in terms of predicates of contact as in [7], or with simple words or phrases as in [8]. Others used predicate representations of sentential meaning to learn language structures in several languages [9]. Recently, the latter has been impressively extended to video of simple scenes involving contact of a few objects [10]. In this third class of work however, the set of images are relatively simple, and the situations can only have a small variation, so that the linguistic input provided by user is focused on the visual topic.

In none of this work is there an attempt to discover the visual category of objects through unsupervised analyses of the video. Another surprising omission is that none of this body of work makes attempts to restrict the visual focus to a part of the scene - i.e. to use attention models. The use of attention for language learning in a visual situation was elegantly demonstrated in the cognitive computation context in the work of Yu and Ballard [11]. Here, the objects in focus are identified by actually tracking the speaker's gaze, which is not an option readily available to a computational vision system.

There are three main contributions in this work. First, we first perform an unsupervised object category learning on surveillance videos. Here we cluster object appearances and trajectory features in a completely unsupervised framework. This then constitutes an internal semantics or concept lexicon for language learning. Our second contribution is to use prior attentional biases to restrict the choices in language learning. Thirdly, in terms of language, we use no prior knowledge of grammar or syntax, despite a wide diversity of linguistic descriptions. Thus, we have multiple naive users describe the action in the video in unconstrained linguistic narrative. Very often, the narratives address the same scene focusing on different actions or agents. Nonetheless, we show that we are able to associate linguistic terms or phrases with object categories in the video across a wide range of visual domains.

While our results may not be as specific [8], or we may not learn phrasal structures [3] or grammar [9], we use far fewer priors than any of these works. Partly, we feel that this is possible owing to the use of visual attention.

A side benefit of the first step in this work is that the generated visual model makes explicit the characterizations of perceptual conceptualizations. Thus, the first two aspects (visual category discovery, and absence of linguistic priors) make it easy to scale up, this third aspect results in applicability in a range of visual situations, as demonstrated in the videos used in this work.

The work also has a possible cognitive fallout. The traditional view in developmental psychology has been due to Piaget [12] who suggests that concepts underlying language do not arise until the end of one and a half years, roughly the same time as language itself. However, mounting evidence for infant skills in categorization and event structuring has challenged this position [13], leading to what may be called the *Perceptual-conceptualization* view: that processes of perceptual abstraction, arising much earlier, leads directly to perceptual structures;

these then form the scaffolding of meaning for learning associations for linguistic units and constructions. In contrast to all of the earlier work cited, the approach presented here considers a computational model where early perception acts as a kind of pre-lexical concept, and guides the process of associating language labels with these concepts. However, our cognitive ambitions are limited; we do not work with speech, using word-separated language commentary instead. However, the visual concepts are learned from complex real-life image sequences in this pre-lexical stage. Second, we also suggest that the availability of such concepts may make it any easier to acquire language based on contemporaneous image sequences and word-segmented narratives.

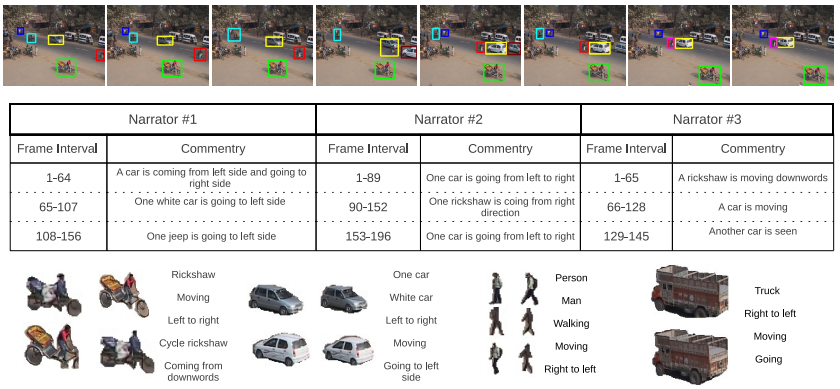


Fig. 1. *Sample input Scene* (first row); A complex scene with a variety of objects – pedestrians, cycle, motorbikes, cars, trucks, rickshaw, buses etc – as many as 20 moving objects in some frames. Object appearances and trajectories are analyzed to abstract the respective object-category and scene-event concepts, which are then associated with diverse sets of un-edited and un-parsed textual narratives based on attentive focus. Note the wide divergence in the commentaries (second row). Discovered visual categories and associations with linguistic phrases are shown at bottom.

2 Role of Attention in Concept Development

The main difficulty in this process - which is also one of the traditional objections to perceptual symbols - is how to identify which part of a scene is relevant to the concept [14] - e.g. in the event of "a person coming out of a car", who is more relevant – person or the car? We posit bottom-up visual attention as a mechanism for determining visual saliency, and show how this results in significant pruning of the possible concepts that can be associated with language labels. We use a computational model of visual attention [15] to compute the saliency distribution over the image space.

Consider the traffic scene of figure 1, say, with the complex interactions between vehicles, pedestrians, animals, bicycles, etc. How is the system to make sense of this complex domain? We feel that a developmentally motivated approach, focusing on the capabilities that an infant brings to bear on such a task, may be relevant. Around the age of six months [16], infants are seen to observe the background for some time before beginning to pay attention to figure objects (foreground). This corresponds to well-known techniques in computer vision for learning a background model in order to identify and track the foreground objects. A key component of this process is a computational model of visual attention [15]. This model is the key to identifying the objects and actions in a scene, and eventually, in associating them with linguistic labels [11]. An overview of our proposition for language label learning of the discovered visual concepts is depicted in figure 2.

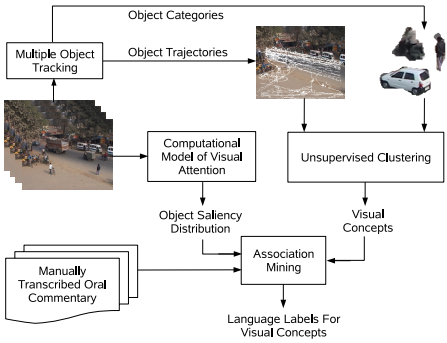


Fig. 2. *System overview.* Multiple targets are tracked in the input image sequence. The appearance features and the trajectories are used to obtain object and action categories. These constitute the class of visual concepts. Oral commentaries acquired synchronously with the image sequence are now associated with the images. The association of a visual concept (concerning a certain object) to a language label (single or multi-word phrase) is computed as a function of the probability that the object is in attentive focus.

3 Visual Concept Model Acquisition

In the first phase of the work, concept models are built from monocular image sequences (acquired with static camera) through prior free bottom-up computations. The moving objects (*figure*) are detected as connected regions of change (foreground blobs) with respect to a background model learned from the static regions of the scene (*ground*) [17]. We have used existing methods from the literature to track multiple moving objects and cluster their appearances and trajectories to from object category and action concepts. Brief descriptions of the adopted methodologies from existing work are presented next.

Multiple Object Tracking [18] – An object is represented by its image pixel set support and corresponding *RGB* pixel color values which collectively define its appearance model. A second order motion model (of object bounding box centroid) initialized mean shift iterations are used to localize the object in subsequent frames. A reasoning scheme is used to handle multiple objects while detecting them in the states of isolation and static/dynamic occlusions along with entry/exit or disappearance/reappearance events. The object features are only updated when the object is detected unoccluded (isolation) by other objects or scene structures. In case of occlusions, the object is tracked using multiple patches for better localization and only the motion model is updated. Multiple object tracking provides us with an object-centric description of the video input in terms of the extracted appearance models and image plane trajectories of different objects during their scene presence. These sets of appearances and trajectories are processed next to construct object category and action concepts. Results of multiple object tracking on one Indian traffic video are shown in Figure 1

Object Discovery [19] – The object category concepts are obtained through a two stage clustering process. First, the appearance modes are learned in shape and Haar feature space using a GMM variant and subsequently the object models are constructed as joint distributions of the learned appearance modes. In the second phase, we categorize the dominant modes associated with an object by DBSCAN based on a Bhattacharya distance metric between joint distributions in the appearance mode space.

Trajectory Clustering [20] – The object trajectories are represented as a time indexed sequence of motion direction states where the directions are quantized in accordance with the 8 compass directions (e.g. *eastward*, *north-eastward* and so on till *south-eastward*) along with a state of rest. Assuming that the actions are manifested by state changes, the trajectories are modeled by "*Compressed Suffix Trees*" (CST) learned from the transition sequences of the object motion direction states. The CST is similar to the variable length Markov model (VLMM) or the probabilistic suffix tree (PST) except that only the transitions are learned leading to a compressed representation. A weighted Bhattacharya distance based semi-metric defined between the CSTs is used to group these trajectory models into action concepts by DBSCAN.

The experiments are performed on eight videos; Three standard datasets – PETS2000 (1451 frames), PETS2001 Dataset-1 (View 1, 2695 frames), PETS2001 Dataset-2 (View 2, 2823 frames) and five of our own videos shot on Indian roads (3 on service roads, 1 in a parking lot and 1 on a highway). The object/activity discovery algorithms classify entire tracks into clusters. The use of DBSCAN enables us to automatically identify outliers. For ground truth labeling purposes, we consider the cluster to be of a particular category based on a majority voting. Figure 3 shows the distribution of each discovered object category in the 8 experimental videos. In case of trajectories, we observe the formation of two strong clusters forming two distinct groups of LEFT TO RIGHT and RIGHT TO LEFT. Rare instances of other kinds

of trajectories like TURNING AROUND, U-TURN, WALKING ACROSS are observed but are not discovered as they disappeared into the strong clusters of LEFT TO RIGHT and RIGHT TO LEFT.

Object Category / Video Name	CAR	PERSON	BUS	TRUCK	TRACTOR	RICKSHAW	CYCLE	TEMPO	BIKE	COW	BIRD	MISC
PETS2000	3	4	×	×	x	×	×	×	×	×	1	3
PETS2001 Dataset 1 View 1	7	11	x	×	x	×	×	×	×	×	x	6
PETS2001 Dataset 2 View 2	2	14	x	x	x	x	2	x	x	x	x	x
Indian Traffic Video 1 (Service Road)	11	72	3	2	2	14	19	12	25	9	x	40
Indian Parking Lot	13	46	x	x	x	2	50	x	24	x	5	38
Indian Traffic Video 2 (Service Road)	14	90	4	7	x	3	31	21	36	x	x	35
Indian Highway Video	33	x	x	1	x	x	x	1	4	x	x	82
Indian traffic video 3 (Service Road)	20	19	5	10	x	6	52	23	45	x	x	83

Fig. 3. Distribution of each discovered category in the 8 experimental data sets. "MISC" represents the ones arising out of either spurious foreground blobs or tracking failure. The symbol × indicate the absence of a certain category in a video.

3.1 Visual Attention and Perceptual Theory of Mind

Language Learning is largely a social activity, reflected in the *Theory of Mind* hypothesis [21] - that the learner has a model for aspects of the speaker's mind, including a sensitivity to the object being attending to, intentions, belief structures, etc. When the learner is presented with only the visual stream and is not in the presence of the speaker, attention is mediated by visual saliency alone, and not by cues received from the speaker's gaze. In many learning situations where both speaker and viewer are looking at the same scene, this appears to be the case, and we call this the *Perceptual Theory of Mind* – i.e., we assume that the speaker would have attended to those parts of the scene that the learner also finds salient.

Models of Visual Attention involve both bottom-up and top-down processes. While top-down processes are task-dependent, bottom-up processes capture those features of the scene that have the highest payoff in terms of generating conceptual abstractions in most relevant domains. Top-down processes require a conceptual sophistication which is still not available to our pre-lexical learner, and even bottom-up visual attention processes are in the formational stage. Nonetheless, we assume a degree of perceptual saliency measures are available to our language learner.

The computational model of visual attention proposed in [15] (Figure 4(a)) is encoded based on multi-scale extraction of intensity, color and orientation contrast feature maps. The pixel-wise saliency values so obtained are further normalized over the image so that they sum up to unity. In this work, we use this model of visual attention to compute the saliency distribution (Figure 4) and the summation of the normalized pixel saliency values over a pixel-set indicate the probability of that region being attended.

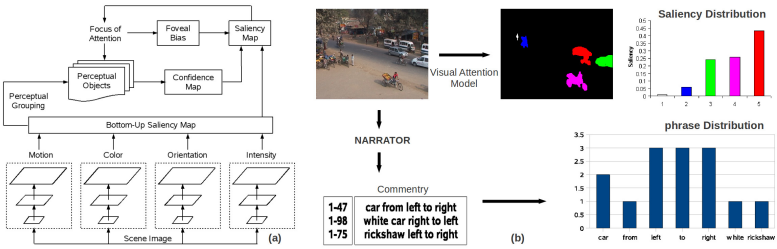


Fig. 4. (a) Bottom-Up Dynamic Visual Attention Model [15]. (b) Saliency distribution over the 5 foreground blobs formed by 6 moving objects in the traffic scene input (frame 20). *Constructing frame-wise phrase distributions from commentaries* – The first sentences from three different sample narrations are used to compute the single word phrase frequencies. For a certain image in the sequence, all the words in the commentary sentences are relevant as this frame lies within the description interval of all the sample narrations. The frequencies of the associated words are computed accordingly. Similar constructions can be formed for multi-word phrases as well. These frequency distributions are normalized further to indicate the probabilities of association of the phrases to the image contents.

Let $\Gamma_c = \{\gamma(r); r = 1, \ldots R_c\}$ be the set of visual concepts constituting both the discovered object categories and trajectories. The probability of a certain visual concept γ being attended in the t^{th} frame is given by $P(\gamma|t)$ and is computed by summing up the normalized pixel-saliency values within the supports of the objects associated to the same concept category. For example, for the objects in the 20^{th} frame of the traffic scene of Figure 4(b), the probability of the concept category "CAR" given by $P(\gamma = \text{CAR}\ |t = 20)$ is computed by summing up the normalized pixel-saliency values of the pixel-set supports of the 2 cars. These probability values ($P(\gamma|t)$) are normalized so that they sum up to unity over the set of the visual concepts. In our case, the concept categories are the clusters which are inspected post-association to validate the language learning.

4 Learning from Textual Narratives

A group of 20 student volunteers (Indian English speakers, ages $18 - 31$, 16 males, 4 females) were shown the videos and instructed to "describe the scene as it happens" without any further cues about the experimental objectives. Each sentence in the resulting oral narrative (manually transcribed) was synchronized with the image sequences, and each word in the sentence correlated with the objects under attentive focus in that time span.

The learning task then becomes one of associating conceptual image-schema (object categories or trajectory classes) from the set of acquired concepts Γ, with k-word phrasal concatenations from the narrative constituting the lexicon $\Sigma_k = \{\Sigma_k(l); l = 1, \ldots L_k\}$. In order to retain generality, we consider the k-word

concatenations $\Sigma_k(l)$ appearing in the narrative; so that Σ_1 consists of single words. Thus, from a sentence such as "*bus moves from left to right*", we would have the set of single word phrases as $\Sigma_1 = \{$"*bus*", "*moves*", "*from*", "*left*", "*to*", "*right*"$\}$; the set of 2-word phrases as $\Sigma_2 = \{$ "*bus moves*" , "*moves from*" , "*from left*" , "*left to*" , "*to right*" $\}$ and so on.

While a commentary sentence is uttered during a certain frame interval, we assume that the single/multiple word phrases derived from the sentence (in word segmented form) is particularly relevant to the visual concepts associated with each image in that time interval. Considering multiple such sentences obtained from different commentators, we construct phrase distributions for the images which provide us with the extents of the association of the phrases to the attended objects/events observed in the images. The probability of association of the l^{th} k-word phrase $\Sigma_k(l)$ with the t^{th} frame is given by $P(\Sigma_k(l)|t)$ and is computed by normalizing the frequency distribution of the phrases for the image. Figure 4(b) illustrates the process of constructing frame-wise phrase distributions from commentaries.

For a certain t^{th} frame, we thus have two distributions - first, the probabilities $P(\gamma|t)$ of the visual concepts (object categories or actions) being attended and second, the probabilities $P(\Sigma_k(l)|t)$ of the phrases being associated to the frame. Using these probabilities, we construct the "*Concept-Phrase Association Matrix*" ($CPAM$) whose elements are given by $CPAM(l,r) = \sum_{t=1}^{T} P(\Sigma_k(l)|t)P(\gamma(r)|t)$.
where T is the total number of frames. The phrasal labels of the acquired visual concepts are finally learned from these concept-phrase association matrices. We next describe the measures used for the task of language label learning and illustrate with the results of single word language labels for object categories acquired from the PETS2000 dataset. From the PETS2000 dataset, we have acquired 4 different object categories, viz. "MAN", "CAR", "BIRD" and the "MISC" consisting of spurious foreground blobs etc. The results of the language label learning for the first three categories for three different measures are shown in Figure 6.

Joint Association Measure – The simplest criterion to associate the phrasal labels for co-attentive visual concepts is to use the joint association measures directly as obtained in the $CPAM$. Thus, the joint association measure is given by $\mathcal{J}_c(\Sigma_k(l), \gamma(r)) = CPAM(l,r)$. For example, using this measure, we can list the top 4 single word language labels corresponding to the categories "MAN", "CAR" and "BIRD" (acquired from the PETS2000 dataset). The results, in the descending order of association are shown in Figure 6.

Conditionally Weighted Joint Association Measure – The absence of sufficient data (most concept-phrase combinations appear too infrequently to compute joint probabilities of association) motivated the work in [22] to propose a modification by weighing the joint association measure with the conditional probability of a visual concept given a certain phrase. Thus, the *conditionally weighted joint association measure* of the visual concept $\gamma(r) \in \Gamma_c$ with the

phrase $\Sigma_k(l)$ is given by $\mathcal{CJ}_c(\Sigma_k(l), \gamma(r)) = P(\gamma(r)|\Sigma_k(l))\mathcal{J}_c(\Sigma_k(l), \gamma(r))$. The work presented in [22] dealt with language label learning from synthetic videos consisting of simple geometrical shapes. By using this measure, we list the top 4 single word language labels corresponding to the categories "MAN", "CAR", and "BIRD" (acquired from the PETS2000 dataset) in the descending order of conditionally weighted joint association in Figure 6.

Dominance Weighted Joint Association Measure – The measures of (conditionally weighted) joint association are largely focused on the association distributions of the concerned visual concept and the phrasal label only. These measures do not consider the nature of the distribution of the joint associations of a certain language label over the different categories. The problem of language acquisition often encounters the cases of multiple associations and the previous measures clearly do not address this issue. For example, for the different categories of moving objects like MAN, CAR, MOTORBIKE, RICKSHAW etc. verbs relevant to motion events like "moving", "coming" or "going" will have high associations. On the other hand, the nouns which actually correspond to the labels of the categories will have high association with only the relevant class of objects. Thus, it is important to differentiate the cases of unimodal joint association from the multi-modal ones (Figure 5).

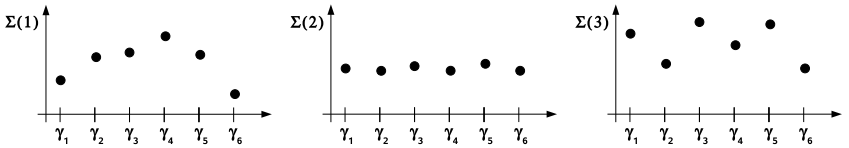


Fig. 5. Joint association distributions – (a) Unimodality – The joint association of the phrasal label $\Sigma(1)$ is peaky around the visual concept γ_4; (b) Uniform distribution - the joint association of the phrasal label $\Sigma(2)$ is almost the same (or flat) with all the visual concepts γ_1–γ_6; (c) The joint association distribution of the phrasal label $\Sigma(3)$ is a multi-modal one, i.e. the same label may be relevant to multiple visual concepts

Consider the case of associating three phrasal labels $\Sigma(1)$–$\Sigma(3)$ with six visual concepts γ_1–γ_6. Figure 5 illustrates three different cases of joint association distributions over a set of visual concepts. The label $\Sigma(1)$ has the strongest association with γ_4 as shown in figure 5(a). On the other hand, $\Sigma(2)$ has almost similar association with all of γ_1–γ_6 (Figure 5(b)) and $\Sigma(3)$ has strong association with some from the 6 visual concepts, i.e. exhibit a multi-modal joint association distribution (Figure 5(c)). In such cases, the phrasal label $\Sigma(1)$ should earn the precedence over the other labels to get associated with the visual concept γ_4. Thus, we need to assign priorities to the labels exhibiting a unimodal joint association distribution over categories and at the same time penalize the flat or multi-modal distributions.

To compute the joint association distributions for the l^{th} phrasal label, we normalize its joint associations with the category/action concepts as $nCPAM(l,r) = \frac{CPAM(l,r)}{\sum_{r=1}^{R_c} CPAM(l,r)}$.

The dominance of the association distributions of the l^{th} phrasal label around the r^{th} concept is next computed as $w_c(l,r) = \frac{1}{R_c - 1} \sum_{i \neq r} (nCPAM(l,r) - nCPAM(l,i))$.

These weights may also be negative if the distribution does not peak around the r^{th} concept and thus the weights w_c are rescaled to ${}^s w_c$ in the interval $[0,1]$ as ${}^s w_c(l,r) = \frac{w_c(l,r) - min_r\{w_c(l,r)\}}{max_r\{w_c(l,r)\} - min_r\{w_c(l,r)\}}$

Thus, the *dominance weighted joint association measure* of the visual concept $\gamma(r) \in \Gamma_c$ with the phrasal label $\Sigma_k(l)$ is computed as $\mathcal{DJ}_c(\Sigma_k(l), \gamma(r)) = {}^s w_c(l,r) \mathcal{J}_c(\Sigma_k(l), \gamma(r))$.

By using this criterion, we list the top 4 single word language labels corresponding to the categories "MAN", "CAR" and "BIRD" (acquired from the PETS2000 dataset) in the descending order of peakiness weighted joint association in Figure 6.

Joint association Measure						Conditionally Weighted Joint Association Measure						Dominance Weighted Joint Association Measure					
MAN		CAR		BIRD		MAN		CAR		BIRD		MAN		CAR		BIRD	
towards	11.725	**car**	82.156	right	0.036	**he**	62.805	**car**	556.24	white	0.00061	**he**	6.620	**car**	72.780	white	0.0115
person	10.693	towards	63.089	white	0.036	wearing	23.239	towards	291.87	started	0.00021	**person**	5.129	towards	52.906	passed	0.0040
car	10.183	left	45.582	car	0.030	center	22.478	coming	291.87	right	0.00016	**man**	4.027	left	42.848	right	0.0032
he	8.758	coming	44.916	towards	0.010	person	21.894	black	288.83	passed	0.00011	towards	3.303	coming	38.954	started	0.0028

Fig. 6. The top 4 single word language labels (in descending order of association value) learned for object category concepts acquired from the PETS2000 dataset. Note the emergence of the synonymous labels "he", "person" and "man" for the category MAN in case of the dominance weighted joint association measure, which clearly shows the superiority of the proposed measure over the earlier ones.

Note the improvement in the results, specially in fetching proper language labels for the category MAN, where the single word language labels "*he*", "*person*" and "*man*" top the list. The proposed criteria of *dominance weighted joint association* is used further to learn the phrasal labels from the traffic scene which is much richer in object category/action content and is presented next.

5 Association Results

The measure of joint association and its conditionally/dominance weighted variants are applied to the object category and trajectory concept classes extracted from the 8 video data sets (Figure 3). The results of single word language label learning for the object categories acquired from these video data sets are presented in Figure 7.

Table 1 present the results of (multi-word) phrasal label associations to different trajectories acquired from the standard data sets (PETS) and the Indian

Video Name \ Object Category	CAR	PERSON	BUS	TRUCK	TRACTOR	RICKSHAW	CYCLE	TEMPO	BIKE	COW	BIRD
PETS2000	**car** towards left	**he person man**	×	×	x	×	×	×	×	×	white passed right
PETS2001 Dataset 1 View 1	**car** white van	**person** student walking	x	×	x	×	×	×	×	×	x
PETS2001 Dataset 2 View 2	**car** blue passing	walking road person	x	x	x	x	going bike blue	x	x	x	x
Indian Traffic Video 1 (Service Road)	**car** silver going	road going man	**bus** moving starts	**lorry** truck cement	grasses vegetables way	going car silver	going side previous	**tempo** standing going	slow going pedestrian	motorcycle bike coming	x
Indian Parking Lot	**car** started maruti	shirt going student	x	x	x	road bike shirt	**cycle** moving bike	x	**bike** road moving	x	**peacock** green road
Indian Traffic Video 2 (Service Road)	passing red bench	white shirt going	**bus** road slow	**truck** cement heavy	x	road person going	**cycle** bike moving	**vikram** waiting white	**bike** road fast	x	x
Indian Highway Video	**maruti** vehicle straight	x	x	**lorry** straight slow	x	x	x	**auto** going yellow	crossing auto fast	x	x
Indian traffic video 3 (Service Road)	**car** black road	cycle left pedestrian	**bus** left big	**truck** big right	x	road tempo going	**cycle** bike right	**vikram** tempo starting	**bike** cycle speed	x	x

Fig. 7. The top 3 single word language labels (in descending order of association value) learned for object category concepts acquired from the 8 experimental video datasets. The top meaningful associations are rendered in bold font. Note the emergence of the synonymous labels "he", "person" and "man" for the category MAN, "vikram", "auto" for the category TEMPO etc. The association values are omitted due to space constraints.

traffic videos. In case of PETS 2000 data set, we failed to learn any associations for the trajectories LEFT TO RIGHT and RIGHT TO LEFT and hence the corresponding association results are not presented in table 1. In the results from both the data sets, hardly any meaningful language labels for the object categories top the lists for multiple word language labels. However, in cases of the trajectories we see the emergence of the multi-word phrasal labels as opposed to the single word ones. This tallies with the fact that multiple words are required to form the directionality concept in the linguistic descriptors of the trajectories.

Some labels are easier to learn compared to others for several reasons. First, there are instances of *synonymy*, e.g. a concept like MAN can have labels *people, sardarji, person, guys, guy* etc., diluting the effect of any particular label (we neither remove plurals nor do any kind of morphological processing on the text). This is true also for CAR and for TEMPO. Secondly, our computational model of *visual saliency* may not have selected the objects mentioned in the narrative. This is particularly true of people, who are preponderant in the scene but are not selected either in the narrative nor by the visual focus. When they do appear in the narrative, they are sometimes not in attentive focus, and we see that for the category MAN, no relevant label appears in the top four. On the contrary, MOTORBIKEs are mentioned quite frequently, but are not as frequently in attentive focus, and given the preponderance of objects (varying between five and twenty at any time), MOTORBIKE emerges as one of the high contenders for several concept categories. On the other hand, large objects like truck, which appeared only twice, despite two equal synonyms (*truck*, *lorry*), have both these labels at the top of the list. This is due to the high visual saliency of this large moving

Table 1. Phrasal label (up to 3 words) association results for object trajectories acquired from PETS data sets and Indian traffic videos

Trajectory	Phrase Size = 1		Phrase Size = 2		Phrase Size = 3	
PETS 2000						
DOWNWARD AND RETURN	starts	431.72	**going back**	481.0	**is going back**	481
	look	429.87	**man turns**	442.46	coat is going	481
	takes	429.87	**turns back**	436.80	man turns back	442.46
PETS 2001 (Dataset-1, View-1 and Dataset-2, View-2) and Indian Traffic Videos						
LEFT TO RIGHT	**left**	27.22	**from left**	13.54	**left to right**	11.15
	right	25.47	from right	12.54	from left to	11.14
	going	13.66	to left	10.99	from right to	11.11
RIGHT TO LEFT	**left**	25.09	**from right**	17.33	**from right to**	13.63
	right	21.89	**to left**	14.75	**right to left**	13.35
	going	16.01	right to	12.84	left to right	4.4

region; the same may also hold for BUS. Finally, there are issues related to the *Categorization Level*, i.e., the narratives may refer to objects at a subordinate (or superordinate) level. Thus, the concept CAR is referred to by model names such as *maruti, Sumo, Zen* as well as *taxi, van, car, cars* etc. There are also eight instances of the superordinate "vehicle" being used. Clearly, a much richer characterization of objects and their subcategories would need to be learned before these distinctions can be mastered [23].

To reiterate the main results - this work represents a completely unsupervised process relying on visual attention to parse the visual input. Place the camera at the scene and have some adults comment on what is happening, and even with very primitive statistical association measures, our infant learner is able to build mappings between discovered concepts and new words/phrases – MAN, CAR, BUS, TRUCK, TRACTOR, CYCLE, TEMPO, BIKE, DOWNWARD AND RETURN, LEFT TO RIGHT and RIGHT TO LEFT from the 8 experimental videos. Compared to the enormous prior knowledge deployed in many computational systems, this is a respectable start for the infant learner. Here, we have only shown the results where meaningful associations are established between visual concepts and phrasal labels.

6 Conclusion

In this work, we have attempted an ambitious approach for associating multi-word language labels to concepts of object appearances and actions acquired from complex multi-object videos. We have used an attentional consistency hypothesis, i.e. "the commentator's gaze tallies with the computational model of attention". A bottom-up approach to computational model of attention is then used to associate the phrases from the commentary to the scene object appearance/action concepts. Language labels are learned by computing the joint associations between concepts and phrasal labels. We have explored the performances of (conditionally/dominance weighted) joint association measures. The

proposed dominance weighted joint association measure was found to outperform the other two while associating proper phrasal labels. Experimental results on 8 data sets show partial discovery of the language labels (using dominance weighted joint association measure) to the corresponding object categories of MAN, CAR, BUS, TRUCK, TRACTOR, CYCLE, TEMPO, BIKE and object action (trajectory) DOWNWARD AND RETURN, LEFT TO RIGHT, RIGHT TO LEFT.

To our knowledge, this is the first work that takes a complex scene, separately identifies perceptual concepts in a completely unsupervised manner, and then associates these with unedited text inputs, to obtain a few phonetic to perceptual schema mappings. The main burden of computation in this task is in the visual processing - i.e. the visual concepts may be harder to learn than (at least some) of the linguistic mappings. While our approach is rich in terms of perception, the learner is not an active participant in the scene. Thus crucial aspects such as intentionality, purposive action, and social interaction have been ignored in the present study. This corresponds to the intuition that the very initial steps in language learning may involve passive inputs, but clearly contingent interaction is a powerful force that would be important to explore in future work. While the specific appearance models are indexed upon the specific view, the concept classes (by appearance or actions) are more general and can be applied to novel situations. It would be important to consider the correlations between multiple views in constructing the appearance models, so that all canonical views can be covered.

Finally, while we have used attentive focus to associate visual concepts with words, we have not used attention at all for the task of forming conceptual clusters. The use of attention for learning concepts is significant since the learned concepts can then act as top-down mediators and bring in elements of intentionality into the system. On the whole, such associative maps for word meanings are clearly just the first step - the vast majority of adult vocabularies are acquired by extrapolation from a few grounded words, primarily by reading [21]. However, these first grounded words constitute the foundation on which these other meanings can be anchored.

References

1. Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D.M., Jordan, M.I.: Matching words and pictures. Journal of Machine Learning Research 3, 1107–1135 (2003)
2. Feng, S., Manmatha, R., Lavrenko, V.: Multiple bernoulli relevance models for image and video annotation. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1002–1009 (2004)
3. Kulkarni, G., Premraj, V., Dhar, S., Li, S., Choi, Y., Berg, A.C., Berg, T.L.: Baby talk: Understanding and generating simple image descriptions. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1601–1608 (2011)
4. Siddiquie, B., Gupta, A.: Beyond active noun tagging: Modeling contextual interactions for multi-class active learning. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) (2010)

5. Quattoni, A., Collins, M., Darrell, T.: Learning visual representations using images with captions. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) (2007)
6. Cour, T., Jordan, C., Miltsakaki, E., Taskar, B.: Movie/Script: Alignment and Parsing of Video and Text Transcription. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 158–171. Springer, Heidelberg (2008)
7. Siskind, J.M.: Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic. Journal of Artificial Intelligence Research 15, 31–90 (2001)
8. Roy, D.K., Pentland, A.P.: Learning words from sights and sounds: a computational model. Cognitive Science 26, 113–146 (2002)
9. Dominey, P., Boucher, J.: Learning to talk about events from narrated video in the construction grammar framework. Artificial Intelligence 167, 31–61 (2005)
10. Madden, C., Hoen, M., Dominey, P.: A cognitive neuroscience perspective on embodied language for human-robot cooperation. Brain and Language 112, 180–188 (2010)
11. Yu, C., Ballard, D.H.: A multimodal learning interface for grounding spoken language in sensory perceptions. ACM Transactions on Applied Perception (2004)
12. Piaget, J.: The Construction of Reality in the Child. Basic Books (1994)
13. Mandler, J.M.: Foundations of Mind. Oxford University Press, New York (2004)
14. Quine, W.V.O.: Word and Object. John Wiley and Sons, New York (1960)
15. Itti, L., Koch, C.: Computational modeling of visual attention. Nature Reviews Neuroscience 2, 194–203 (2001)
16. Coldren, J.T., Haaf, R.A.: Priority of processing components of visual stimuli by 6-month-old infants. Infant Behavior and Development 22, 131–135 (1999)
17. Zivkovic, Z.: Improved adaptive gaussian mixture model for background subtraction. In: Proceedings of the 17th International Conference on Pattern Recognition, vol. 2, pp. 28–31 (2004)
18. Guha, P., Mukerjee, A., Subramanian, V.K.: Formulation, detection and application of occlusion states (oc-7) in the context of multiple object tracking. In: 8th IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS), pp. 1–6 (2011)
19. Nandi, S., Guha, P., Venkatesh, K.: Objects from animacy: Discovery in joint shape and haar feature space. In: Indian Conference on Vision, Graphics and Image Processing (2008)
20. Guha, P., Mukerjee, A., Venkatesh, K.S.: Activity Discovery Using Compressed Suffix Trees. In: Maino, G., Foresti, G.L. (eds.) ICIAP 2011, Part II. LNCS, vol. 6979, pp. 69–78. Springer, Heidelberg (2011)
21. Bloom, P.: How Children Learn the Meanings of Words. MIT Press, Cambridge (2000)
22. Sarkar, M., Mukerjee, A.: Perceptual theory of mind: An intermediary between visual salience and noun/verb acquisition. In: International Conference on Developmental Learning (ICDL 2006) (2006)
23. Mukerjee, A., Joshi, N., Mudgal, P., Srinath, S.: Bootstrapping word learning: A perception driven semantics first approach. In: IEEE International Conference on Development and Learning, vol. 2, pp. 1–6 (2011)

Parameterized Variety Based View Synthesis Scheme for Multi-view 3DTV

Mansi Sharma, Santanu Chaudhury, and Brejesh Lall

Department of Electrical Engineering, Indian Institute of Technology, Delhi
mansisharmaiitd@gmail.com, {santanuc,brejesh}@ee.iitd.ac.in

Abstract. This paper presents a novel parameterized variety based view synthesis scheme for 3DTV and multi-view systems. We have generalized the parameterized image variety approach to image based rendering proposed in [1] to handle full perspective cameras. An algebraic geometry framework is proposed for the parameterization of the variety associated with full perspective images, by image positions of three reference scene points. A complete parameterization of the 3D scene is constructed. This allows to generate realistic novel views from arbitrary viewpoints without explicit 3D reconstruction, taking few multi-view images as input from uncalibrated cameras.

Another contribution of this paper is to provide a generalised and flexible architecture based on this variety model for multi-view 3DTV. The novelty of the architecture lies in merging this variety based approach with standard depth image based view synthesis pipeline, without explicitly obtaining sparse or dense 3D points. This integrated framework subsequently overcomes the problems associated with existing depth based representations. The key aspects of this joint framework are: 1) Synthesis of artifacts free novel views from arbitrary camera positions for wide angle viewing. 2) Generation of signal representation compatible with standard multi-view systems. 3) Extraction of reliable view dependent depth maps from arbitrary virtual viewpoints without recovering exact 3D points. 4) Intuitive interface for virtual view specification based on scene content. Experimental results on standard multi-view sequences are presented to demonstrate the effectiveness of the proposed scheme.

1 Introduction

Over the intervening years, 3DTV technology has matured significantly to provide a realistic 3D impression of the scene. Multi-view systems (e.g. multi-view autostereoscopic displays) emerged as a core technology for 3DTV. The foremost requirement of these systems is the generation of high quality multi-view images. A variety of different 3D video representations exist to support these advanced 3D systems, with their own features and limitations. Multi-view video provides high quality 3D content and support wide angle viewing, but requires large amount of data to be processed. This needs sophisticated coding and bandwidth efficient transmission schemes. Video-plus-depth representation is quite popular for rendering of 3D views. It consists of monoscopic color video accompanied

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 538–551, 2013.

with per-pixel depth data. As it explicitly contains 3D geometry information, virtual views can be rendered by depth image based rendering (DIBR) technique. This format is widely accepted as it is easily adapted to different 2D/3D display systems but does not support wide angle viewing. This is because DIBR falls into the category of point based rendering algorithms, and thus suffers from resampling problem, which possibly cause ghosting artifacts to appear in the rendered views. Moreover, the annoying visual artifacts (like holes, cracks) are present in the synthesized views due to inherent visibility and disocclusion problems. To support wide range multiview 3D displays, multi-view video-plus-depth is more appropriate. Rendered view quality is better as the representation uses more than one texture (color) and depth data. It avoids high complexity and maintain moderate size of the data. However, artifacts still occur in the synthesized views due to complex error prone processing steps and depth based rendering. Although DIBR based systems greatly reduce the bandwidth requirement as only two streams are needed to generate multi-view images, they are not suitable for high quality view generation from potentially arbitrary viewpoints.

For addressing these issues, a novel parameterized variety based representation and rendering scheme for multiview 3DTV systems is presented. The method construct a minimal parameterization of 3D space using a relatively small number of captured scene views. The scene is assumed to be captured by multiple uncalibrated cameras located at arbitrary positions. It has been shown earlier [1] that the set V of all views of n 3D points is a six dimensional variety of vector space R^{2n} for weak perspective, paraperspective and full perspective cameras. The parameterization of the variety in weak perspective and paraperspective cases were proposed earlier[1]. Our major contribution lies in the generalization of this approach to full perspective cameras. Euclidean constraints associated with the perspective cameras are explicitly taken into account. This yields a system of five quadratic multivariate polynomial equations, termed as parameterized image variety or PIV associated with the scene. This extension of variety based approach to full perspective cameras has a major advantage. It constructs a complete parameterization of 3D space (in terms of structure coefficients) which is not the case in weak and paraperspective cases as explained in [1]. The coefficients defining the PIV, allows to render novel views from arbitrary viewpoints without explicit 3D reconstruction. The technique produces photorealistic novel images without explicit depth recovery, therefore overcomes the most common problems associated with depth based methods. Moreover, using relatively less input views, large number of views can be synthesized from arbitrary viewpoints. These facts give the primary motivation to use this variety based approach for 3DTV view generation instead of depth based methods.

This variety model is used to build a new flexible multi-view 3DTV system that allows to render high quality virtual views of a 3D scene from arbitrary camera positions. Typical application of the methodology is in 3D viewing of wide range of indoor and outdoor urban scenes. The proposed system integrates two different view synthesis pipelines (transfer-based and depth-based) into one common framework. For merging the two different approaches without explicitly

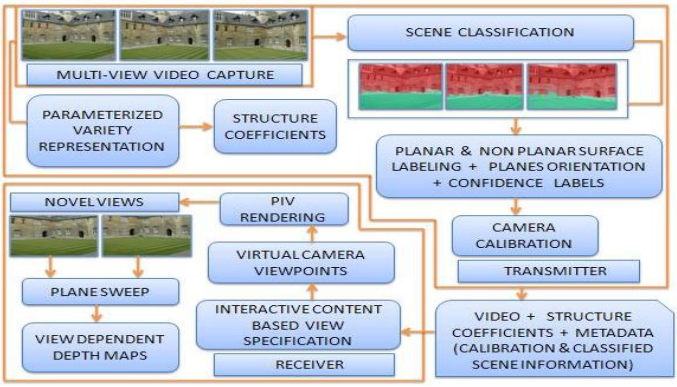


Fig. 1. A flexible variety based multi-view 3DTV system

obtaining sparse or dense set of 3D points, a statistical learning method is used[2]. Multiple input views are classified to detect planar and non planar regions along with their orientations and associated confidence measures. These are used to calibrate the input cameras. The calibration and orientation information of the classified input views are used to automatically define novel viewpoints with respect to the existing viewpoints. Novel views are synthesized using computed structure coefficients and PIV rendering. The realistic virtual views are obtained without using any calibration and depth information. Recovered orientation information of detected planes in classified input views are used to obtain dense depth maps (for all input and novel synthesized views) by using a plane sweep algorithm[3]. The orientations give the directions for sweeping planes, and thus avoid the need to reconstruct sparse or dense 3D points for depth estimation. This integrated framework supports standard video-plus-depth and multi-view video-plus-depth workflows, by generating reliable depth maps for all input and arbitrary virtual camera views. Extracted high quality view dependent depth maps can be used for content creation and 3D post production applications for multi-view displays. The architecture also provides important system features, such as an intuitive way of specifying virtual viewpoints based on content of the scene without complicated user intervention, compatibility with existing multi-view coding standards and adaptability to different 2D/3D displays.

2 Parameterized Variety Based Multi-view 3DTV System

Fig. 1 gives an overview of the proposed system. The proposed methodology is applicable for interactive 3D viewing of wide range of multiplanar environments like indoor and outdoor urban scenes. The input to the system are a collection of multi-view images taken from arbitrary, uncalibrated cameras. There are two stages of processing for signal generation. The first stage involves the construction of parameterized variety representation of the 3D scene (in terms

of structure coefficients). The system automatically establishes the sparse point correspondences across the multiple input views using scale invariant feature transform (SIFT) detector. Using the establish correspondences, parameterized variety is constructed. The coefficients defining the variety are computed and stored. These structure coefficients are the representative of the geometry information of the scene. The second stage is to use the classifier of [2] to classify all the input views, and identify all planar (vertical and horizontal) and nonplanar regions along with their orientations and associated confidence labels. The classification does not rely on any calibration or 3D scene information. Input cameras are self calibrated using the inter image homographies obtained from the located set of coplanar points across the views and applying the method presented in [4].

The outcome of these two stages are computed structure parameters, calibration and scene classification information (orientations, confidence labels etc.) of the input views. Thus, the structure coefficients along with the video forms the signal representation. Calibration and scene classification information are embedded as metadata part of the signal. The signal generation is an offline process. The generated signal is encoded and transmitted. At the receiver end, the user interactively selects certain part (e.g. a wall) of the scene in one of the input image. The system automatically specifies the virtual viewpoints using the plane orientation information of that part of the image. The virtual viewpoints are defined as such that the selected part (i.e. wall) is best viewed. A series of high quality virtual views are generated using the transmitted structure coefficients and PIV rendering, without using any calibration and explicit depth information. The calibration information of input views is used only in automatic viewpoint specification. Although PIV requires no explicit depth data to render virtual views, it is possible to extract the dense depth maps of novel synthesized images without obtaining dense 3D points, using the decoded classification information of the input views. A plane sweep approach is basically followed [3] for extracting view dependent depth maps. Instead of identifying the surface normals by analyses of dense 3D points through structure from motion, orientation information of the classified planes is used to identify the directions for sweeping. This gives additional flexibility to the architecture to support existing multi-view systems that relies on depth based representations. The other advantages of this signal representation are:

1. The representation is bandwidth efficient as one needs to transmit relatively small number of multiple views. The structure parameters and metadata can be efficiently encoded and transmitted with a less overhead. It is even compatible with existing multi-view coding and compression schemes.
2. In DIBR based systems, coding/transmission artifacts generally occur in the depth maps (blocking effects, ringing artifacts around the edges etc.). In our approach, depth maps of input and virtual views are obtained at the receiver end using the signal representation only, which ensures its good quality.

The details of each component are presented in the following sections.

2.1 Signal Generation

To generate the required signal, two stage of processing is involved 1) Given multi-view images, construct the parameterized representation of the 3D scene, and estimate the corresponding structure coefficients. 2) Obtain the scene classification and calibration information of the input views.

2.1.1 Parameterized Variety Representation of 3D Scene

Suppose we observe three scene points Q_0, Q_1, Q_2 whose images $q_0 = (u_0, v_0)^T$, $q_1 = (u_1, v_1)^T$, $q_2 = (u_2, v_2)^T$ are not collinear. Define the coordinate vectors of these points in a Euclidean coordinate system as $Q_0 = (0, 0, 0)^T$, $Q_1 = (1, 0, 0)^T$ and $Q_2 = (p', q', 0)^T$. The values of p' and q' are nonzero but (a priori) unknown. Point PIV is parameterize using these three scene points. Consider a point $Q = (x', y', z')^T$ and its projection $\boldsymbol{q} = (u, v)^T$ in the image plane. The values of (x', y', z') are unknown. The image (x_i, y_i) of any scene point X_i=$[X, Y, Z]^T$ under perspective camera model[12] can be written as

$$\lambda_i \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} = \begin{bmatrix} m_1^T & T_x \\ m_2^T & T_y \\ m_3^T & T_z \end{bmatrix} \begin{bmatrix} X_i \\ 1 \end{bmatrix}, \tag{1}$$

where, λ_i is the projective depth of point X_i. In normalized image coordinates m_1, m_2, m_3 represent the rows of the rotation matrix. The Euclidean constraints associated with the full perspective cameras are:

$$\begin{cases} \mid m_1 \mid^2 = \mid m_2 \mid^2, \mid m_2 \mid^2 = \mid m_3 \mid^2, \\ m_1 \cdot m_2 = 0, m_1 \cdot m_3 = 0, m_2 \cdot m_3 = 0. \end{cases} \tag{2}$$

Projecting Q_0, Q_1, Q_2 and Q under perspective camera model Eq. (1), yields

$$m_1 = BU, m_2 = BV, m_3 = BL, \tag{3}$$

where, $B = \begin{pmatrix} 1 & 0 & 0 \\ \vartheta & \mu & 0 \\ \varsigma_4/z' & \varsigma_5/z' & 1/z' \end{pmatrix}$, $U \stackrel{def}{=} \begin{pmatrix} \lambda_1 u_1 - \lambda_0 u_0 \\ \lambda_2 u_2 - \lambda_0 u_0 \\ \lambda u - \lambda_0 u_0 \end{pmatrix}$, $V \stackrel{def}{=} \begin{pmatrix} \lambda_1 v_1 - \lambda_0 v_0 \\ \lambda_2 v_2 - \lambda_0 v_0 \\ \lambda v - \lambda_0 v_0 \end{pmatrix}$, $L \stackrel{def}{=} \begin{pmatrix} \lambda_1 - \lambda_0 \\ \lambda_2 - \lambda_0 \\ \lambda - \lambda_0 \end{pmatrix}$ and $\vartheta = -p'/q', \mu = 1/q', \varsigma_4 = -(x' + \vartheta y'), \varsigma_5 = -\mu y'$. The λ_0, λ_1, λ_2, λ are the projective depth associated with points Q_0, Q_1, Q_2 and Q. Using Eq. (3) and letting $C_s \stackrel{def}{=} z'^2 B^T B$, full perspective constraints Eq. (2) can be written as

$$\begin{cases} U^T C_s U - V^T C_s V = 0, V^T C_s V - L^T C_s L = 0, \\ U^T C_s V = 0, U^T C_s L = 0, V^T C_s L = 0, \end{cases} \tag{4}$$

with

$$C_s = \begin{pmatrix} \varsigma_1 & \varsigma_2 & \varsigma_4 \\ \varsigma_2 & \varsigma_3 & \varsigma_5 \\ \varsigma_4 & \varsigma_5 & 1 \end{pmatrix}, and \begin{cases} \varsigma_1 = (1 + \vartheta^2) z'^2 + \varsigma_4^2 \\ \varsigma_2 = \vartheta \mu z'^2 + \varsigma_4 \varsigma_5 \\ \varsigma_3 = \mu^2 z'^2 + \varsigma_5^2. \end{cases} \tag{5}$$

Substituting U, V, L, C_s in Eq. (4) and defining the variables $g_1 = \frac{\lambda_1}{\lambda_0}$, $g_2 = \frac{\lambda_2}{\lambda_0}$, $g_3 = \frac{\lambda}{\lambda_0}$, we get a system of five quadratic equations $\{f_1, f_2, f_3, f_4, f_5\}$ in eight unknown variables $\varsigma_1, \varsigma_2, \varsigma_3, \varsigma_4, \varsigma_5, g_1, g_2, g_3$. Five structure parameters $\varsigma_1, \varsigma_2, \varsigma_3, \varsigma_4, \varsigma_5$ remain fixed, when four scene points Q_0, Q_1, Q_2 and Q are rigidly attached to each other. Choosing three points Q_0, Q_1, Q_2 as a reference triangle from n points and writing Eq. (4) for the remaining ones yield a set of $5n - 15$ quadratic equations in $8n - 24$ unknowns. This is the PIV. The structure parameters construct the parameterization of the set of all perspective images of the scene. The parameters are calculated by eliminating three variables g_1, g_2 and g_3 from five quadratic equations $\{f_1, f_2, f_3, f_4, f_5\}$. To eliminate the three variables g_1, g_2 and g_3, we follow closely the approach adopted in [5] to solve recognition problem for six lines. Elimination is performed in stages by applying Cayley-Dixon-Kapur-Saxena-Yang method (KSY method) [13].

Algorithm A
Input:
1) A collection of N input images and n point correspondences.
2) Three points $q_0 = (u_0, v_0)$, $q_1 = (u_1, v_1)$, $q_2 = (u_2, v_2)$ out of n points are chosen as reference points.

For $i = 1..N$ and $s = 1...n - 3$ {
Step 1: Substitute the known values of the eight parameters u_{0i}, v_{0i}, u_{1i}, v_{1i}, u_{2i}, v_{2i}, u_{is}, v_{is} (rational or integral) to quadratic polynomials. This reduces the size and complexity of the polynomials.
Step 2: Choose to work over a finite field like $Z_p\,[g_1, g_2, g_3]\,/(g_1^2 - 3, g_2^2 - 5, g_3^2 - 7)$, where p is a large prime and Z_p is a finite field of order p. This eliminated higher degree terms in g_1, g_2, g_3 occurring at intermediate steps and greatly speed up the computation.
Step 3: Apply KSY to eliminate two variables g_1 and g_2 from three equations $\{f_{1i}, f_{2i}, f_{3i}\}$ obtaining a polynomial q_1 in variables $\{\varsigma_1, \varsigma_2, \varsigma_3, \varsigma_4, \varsigma_5, g_3\}$.
Step 4: Apply KSY to eliminate two variables g_1 and g_2 from three equations $\{f_{3i}, f_{4i}, f_{5i}\}$ obtaining a polynomial q_2 in variables $\{\varsigma_1, \varsigma_2, \varsigma_3, \varsigma_4, \varsigma_5, g_3\}$.
Step 5: Apply KSY to eliminate g_3 from $\{q_1, q_2\}$ to get the final resultant Res.
Step 6: Subsequent higher orders (greater than one) in any of the variables ς_1, ς_2, ς_3, ς_4, ς_5 occuring in Res can be mod out. Choose to mod out by quadratic irreducible polynomial (like $\varsigma_j^2 - 11$ $(j = 1...5)$).
Step 7: Apply numerical techniques (like Jenkins-Traub method [6]) to solve Res. Stored estimated parameters in D_{is}. }
Step 8: Perform singular value decomposition of matrix D_{is} to refine the parameters and store them as a vector $\varsigma = (\varsigma_1, \varsigma_2, \varsigma_3, \varsigma_4, \varsigma_5, 1)^T$.

2.1.2 Classification and Surface Labeling

Scene classification and surface labeling of input views is performed by using the methodology of [2]. The authors perform the labeling of the different regions of an image into different geometric classes based upon their 3D orientation with respect to the capturing camera. This machine learning approach model the

appearance of geometric classes from a set of training images. No calibration and 3D geometry information is required. Using that the likelihood of each of the possible classes for each pixel is estimated. Regions are mainly categorized as ground (horizontal), sky and vertical (left, right or center) and non planar surfaces either porous or solid. The signal components are the orientations, labels and associated confidence measures of each classified region.

2.1.3 Camera Self Calibration Using Scene Planes

Cameras are self calibrated using one or several planar regions obtained from the classified scenes. Point correspondences are establish and detected features are matched across the views. Outliers are removed by robustly fitting fundamental matrix between pairs of views. From the remaining inliers, points belonging to different planes are separated out. For this purpose, any one image can be used as the camera placement is arbitrary and corresponding points may have different labels across the views. From the located coplanar points across the views, inter image homographies have been estimated using the normalized DLT algorithm [7]. Image of absolute conic ω is determined using plane homographies by applying the method similar to [4].

The generated signal (i.e. structure parameters, classified scene and calibration data) is encoded and transmitted. At the receiver end, virtual viewpoints are specified and novel views are synthesized using decoded signal information.

2.2 Viewpoint Specification and Synthesis

We propose an intuitive and practical way for virtual view specification based on the content of the scene. In general, a viewpoint can be specified by performing a translation and rotation with respect to any input view to determine its position and direction. But it is impractical to ask a TV viewer to do this. A more practical way is to start with a given input view and let the user to choose the viewpoint based on scene content. This allows to see the novel 3D views of the chosen part as well as arbitrary virtual views of the entire scene. This content based relative viewpoint moving, in an interactive manner, is much more convenient. Typical application of it is in 3D viewing of indoor and outdoor scenes like building, shopping malls etc. Typical characteristics of such scenes (i.e. extracted planar and nonplanar patches) facilitate in automatic detection of positions and orientations. Our system is designed to synthesize continuum of virtual views from one viewpoint to some other (arbitrary) viewpoint.

2.2.1 View Specification Based on Scene Content

The user interactively selects a part of the scene in any given input view, through an external interface. For instance, if a wall of a monument in input scene is selected, a new viewpoint is defined automatically such that the wall is fronto parallel. We make use of the fact that the best view of a plane is fronto parallel. The orientation information of this part (i.e. wall), obtained from the decoded signal is used to determine the direction in which reference camera

($P^{'}_{ref} = K^{'}_{ref} R^{'}_{ref}[I| - C^{'}_{ref}]$) has to be rotated. Once the direction is specified, the virtual camera matrix ($P^{'}_{final} = K^{'}_{final} R^{'}_{final}[I| - C^{'}_{final}]$) is chosen as:

1. For plane corresponding to right part of the scene, a rotation matrix R_y is defined for rotation about the positive Y axis by an angle φ confined within the angle formed by plane normal and principal axis of the camera. The final virtual camera matrix is chosen as:

$$K^{'}_{final} = K^{'}_{ref},\ R^{'}_{final} = R_Y * R^{'}_{ref},\ C^{'}_{final} = C^{'}_{ref} + [0; 0; t]$$

   A small translation step t is required to keep the intermediate virtual camera view within the image bound. The factor t also provides a zoom in effect since effectively the camera is moving into the image.
2. Similarly for left and ground plane, rotation matrix is calculated for rotation about the negative "Y" and positive "X" axis respectively. For the center plane no rotation matrix is calculated. A gradual interpolation of camera matrices is performed from $P^{'}_{ref}$ to $P^{'}_{final}$ using varying interpolation factor $\alpha \in [0..1]$. Sperical linear interpolation *"slerp"* is applied to each row of the camera matrix.

$$K_\alpha = K^{'}_{ref},\ R_\alpha = slerp(R^{'}_{ref}, R^{'}_{final}, \alpha),\ C_\alpha = C^{'}_{ref} * \alpha + C^{'}_{final} * (1 - \alpha)$$

   These intermediate camera matrices are used to synthesize a continuum of virtual views, and thus we get a feeling of the wall turning towards us. Novel occlusion free views are synthesized using decoded structure parameters and PIV rendering.

2.2.2 Novel View Synthesis Using PIV Rendering

Novel views can be rendered by specifying image positions q_0, q_1 and q_2 for three reference points Q_0, Q_1 and Q_2 at the virtual viewpoint and computing the corresponding image positions of all other points. The algorithm for synthesis of a novel view I_{nv} is summarized as:

1. Define a new view by specifying image positions $q_0 = (u_0, v_0)$, $q_1 = (u_1, v_1)$, $q_2 = (u_2, v_2)$ of three reference points at virtual viewpoint. Let it be $q'_0 = (u'_0, v'_0)$, $q'_1 = (u'_1, v'_1)$, $q'_2 = (u'_2, v'_2)$.
2. Substitute q'_0, q'_1, q'_2 in Eq. (4) in place of q_0, q_1, q_2. Using computed structure coefficients, render the image positions $(u, v)'s$ of all other corresponding points in the new view by solving quadratic equations Eq. (4). Any visibility issue can be resolved using obtained $g'_3 s$ (scaled depth value as $g_3 = \frac{\lambda}{\lambda_0}$) for each corresponding point as z-coordinate values.
3. Triangulate the new view I_{nv} using the rendered points as vertices [11]. Assign a depth to each triangle by taking the mean depth of its three vertices. Sort the triangles in descending order of depth. Texture map the triangles from the given input views in decreasing order of depth. For each pixel p_{nv}, in the current triangle t_c of the novel view, compute the barycentric coordinates

of the pixels in I_{nv}. Find the pixels corresponding to p_{nv} in given input views $I_1, ..., I_N$ by computing the affine combination of the barycentric coordinates and the vertices of the same triangle t_c in $I_i(i = 1..N)$. Find the front-most triangles, the corresponding pixel lies in $I_i(i = 1..N)$. If any of the front-most triangle is the same as the triangle t_c , use the intensity from that triangle. If not, color the pixel black.

2.3 Depth Map Estimation

Depth maps for each of the input images and novel synthesized images are obtained by performing plane sweeping. In our approach, we follow closely to [3]. The basic steps involved are:

1. *Sweeping directions estimation*: Scene classification outputs a labeled image, where each pixel is assigned the label of the geometric class which most likely represents it and also the confidence measures associated with each geometric label. Pixels grouped after the classification are collected and planes are robustly fitted. Let Λ_{kl} denote M family of parallel depth planes, denoted as $\Lambda_{kl} = [n_k^T \;\; d_{kl}]$, $\{k = 1, ..., M\}$. The subscript l indices over number of planes corresponding to k^{th} family and n_k denotes unit length normal of the k^{th} family planes. The depth range $[d_{near}^k \;\; d_{far}^k]$ for each family is obtained empirically.
2. *Obtaining the sweeping planes*: Once the sweeping directions n_k are determined, the actual planes used in sweeping are obtained by varying d_{kl} obtained from the previous step.
3. *Warping*: Homography H_{Λ_{kl},P_i} induced by each of the planes Λ_{kl} is determined between two images (obtained at different camera positions). Let $P_{ref} = K_{ref}R_{ref}[I| - C_{ref}]$ and $P_i = K_iR_i[I| - C_i]$ be the camera projection matrices for the reference view I_{ref} and the other camera view I_i. The homography H_{Λ_{kl},P_i} is used to warp image I_i to obtain I_i^*. Missing pixels are interpolated using bilinear interpolation. For the warped image I_i^*, cost metric is defined as a function of pixel (x, y) in the reference view I_{ref} and for each of the plane Λ_{kl} as:

$$C(x, y, \Lambda_{kl}) = \sum_{(\delta_x,\delta_y)\in W} |I_{ref}(x-\delta_x, y-\delta_y) - I_i^*(x-\delta_x, y-\delta_y)| - \sigma * log(F_{\Lambda_{kl}}(x, y))$$

   where, σ is weight factor which is learned by experiments, W is 3×3 neighbourhood of the pixel and $F_{\Lambda_{kl}}(x, y)$ is the probability that the pixel (x, y) belongs to the plane Λ_{kl}. It is obtained in terms of confidence measures.
4. *Best plane selection*: The simplest possible technique is to choose the plane of minimum cost as $\widehat{\Lambda_{kl}}(x, y) = \arg\min_{\Lambda_{kl}} C(x, y, \Lambda_{kl})$. However, noise is still observed due to incorrectly assigned planes. The solution is formulated in an energy minimization framework similar to [3] and minimize it using techniques like graph cuts [8]. Once the plane label is correctly identified for each pixel (x, y), depth map of image I_{ref} is estimated.

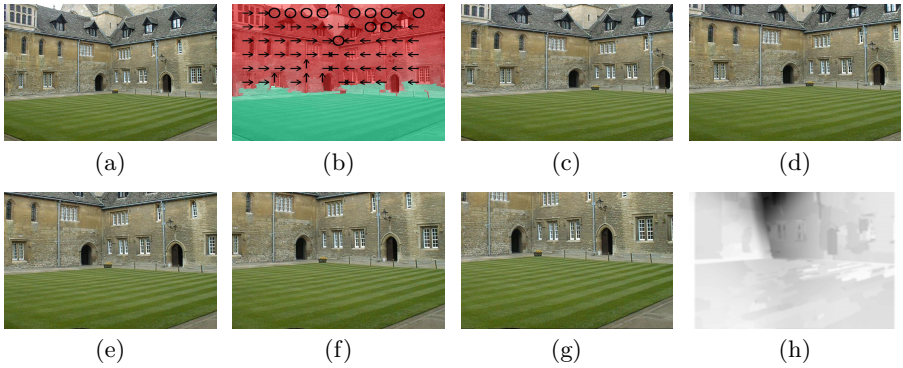


Fig. 2. (a) One of the input image of Merton. (b) Its scene classification and planes orientation. (c,g) Novel synthesized sequence when viewer selected the lower right part of the building. (h) Estimated dense depth map of novel view (g).

3 Implementation Details and Results

The scheme is implemented using MATLAB (R2009a) and MATHEMATICA. Mathematica implementation of KSY Dixon resultant algorithm is used [9] to perform the elimination and finding structure coefficients. Reduction over finite field is performed by interfacing with Sage version 5.0.1. The performance is extensively tested on various standard multi-view dataset and video sequences of indoor and outdoor scenes: 3D video[1], Visual geometry group[2], Kitchen[3] dataset. Test conditions consider both cases of simple and complex camera motion and also taken into account the scenes containing high detail and complex depth structures. Results with only static scenes are presented as it is difficult to perform comparative analysis with dynamic scenes in respect they are unrepeatable. The proposed scheme is workable for dynamic scenes also, by constructing parameterization of each temporal aspect independently.

3.1 View Synthesis Results Using PIV Rendering

The various steps of the proposed scheme are illustrated with Merton[2] dataset. All three images of Merton are used for the estimation of structure coefficients. Fig. (2(a),2(b)) shows the input view and its classified scene planes (green (horizontal), red (vertical)) and orientations (arrows). Fig. (2(c),2(g)) shows the novel synthesized views, when the user is intended to view the lower right wall of the scene closely. No rendering artifacts occur even if the camera is taking a steep turn towards the right part of the scene. Fig. 2(h) shows the estimated depth of novel view obtained from the procedure described in section 2.3.

[1] `http://www.cad.zju.edu.cn/home/gfzhang/projects/videodepth/data/`
[2] `http://www.robots.ox.ac.uk/~vgg/data/data-mview.html`

3.2 Comparative Analysis

The proposed rendering scheme has been compared with state of art DIBR technique [10]. An experiment with Kitchen[3] dataset is performed. Out of eight given camera views, five views (C_4 to C_8) are used for the estimation of structure coefficients (*Algorithm A*). A novel view is resynthesized from viewpoint corresponding to C_2 using PIV rendering. Another experiment is conducted using avaliable ground truth depth maps. Nearest camera views C_1 and C_3 are chosen as reference, and virtual view at camera C_2 are resynthesized using standard DIBR based view synthesis pipeline [10]. Resynthesized views from both methods are compared with the original one, to assess the quality. Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) is measured for objective quality assessment. Rendering artifacts are clearly visible in resynthesized view obtained using DIBR, even after contour correction and hole filling Fig. 3(f) (see marked areas). Artifacts are visible where depth values are erroneous Fig. 3(e). The quality of the synthesized view using PIV rendering is comparatively much better Fig. 3(g). This experiment have revealed another important fact about full perspective PIV representation. The camera views C_1, C_2 and C_3 are not used in estimating the PIV coefficients, yet the rendered view quality at viewpoint C_2 is quite good. The occlusion is correctly handled. This means PIV can be used to extrapolate the views outside the camera basis from arbitrary viewpoints and even using small number of input views. Fig. (3(h),3(j)) shows the novel synthesized PIV views of Kitchen from arbitrary viewpoints.

Quality of the depth map obtained using classified scene data of input views is also accessed. Depth map of PIV resynthesized view Fig. 3(g) is determined using scene classification information of input views (C_4 to C_8). From dense correspondences between resynthesized PIV view and input views, points belonging to different planes are separated out. Fig. 3(k) shows its classification into different planar regions. The regions are divided into left (pink), right (red) or center (cream), ground (green) and ceiling (blue). Labels and associated confidence measures are shown in Fig. (3(l),3(q)). Fig. 3(r) shows the plane family labels obtained after sweeping and graph cut minimisation. Final determined depth map Fig. 3(s) is compared with ground truth. The PSNR value obtained is much better as compared to final depth map obtained using DIBR Fig. 3(e).

Fig. 4 shows the results on scenes containing complex planar and non-planar geometries. Annoying artifacts predominate the DIBR rendered view quality Fig. 4(a) as compared to the proposed method Fig. 4(b), when virtual viewpoint is far away from the original camera position. Fig. 4(c) highlight the shortcoming of DIBR with respect to zoom-in effects. The image quality degrades (holes, cracks) as one move more into the image because of the inherent sampling problem. Comparatively, rendered PIV vitual view Fig. 4(d) are quite realistic and superior in quality, even when the camera is zoomed more into the image. Subjective quality assesment (Tab. 1) has been carried out on a group of 17 human subjects, expressed by a 10 point continuous scale ranging from 1 (severe annoying artifacts) to 10 (imperceptible artifacts).

[3] `http://lttm.dei.unipd.it/downloads/kitchen/`

Table 1. Average mean opinion scores (MOS) and standard deviations (SD)

DIBR		PIV	
MOS (5.364)	SD (1.152)	MOS (8.975)	SD (1.143)

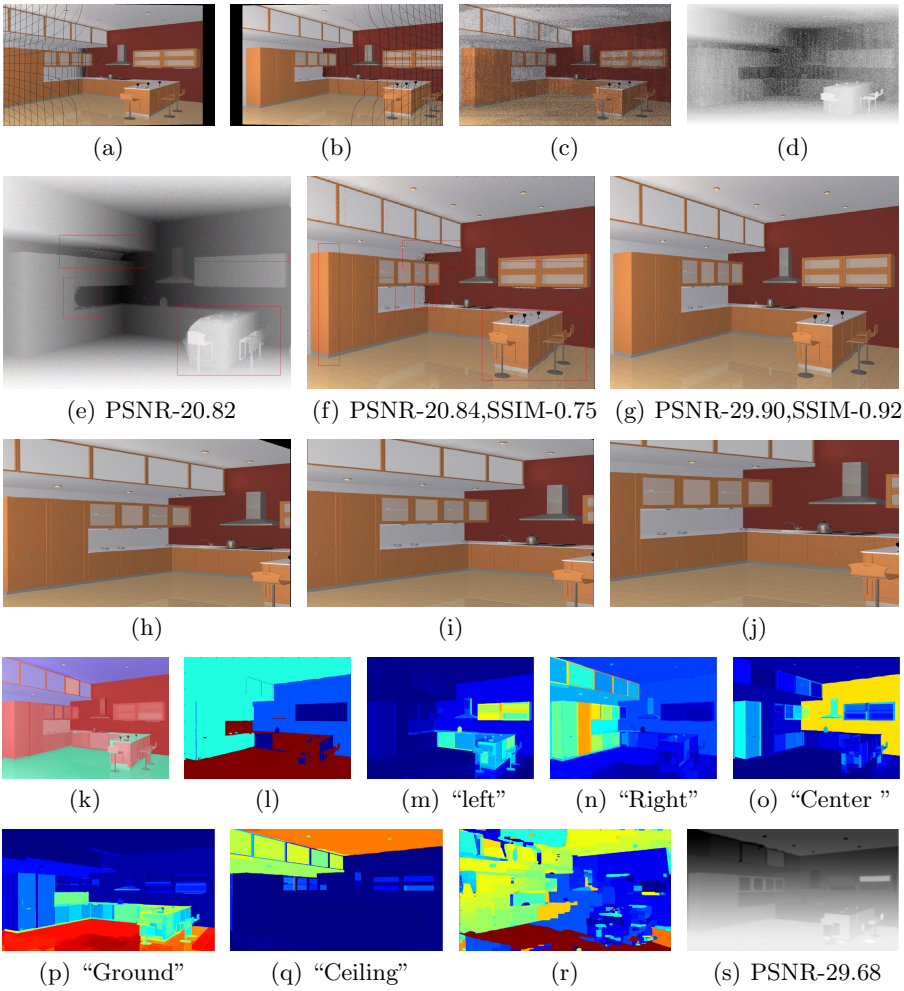
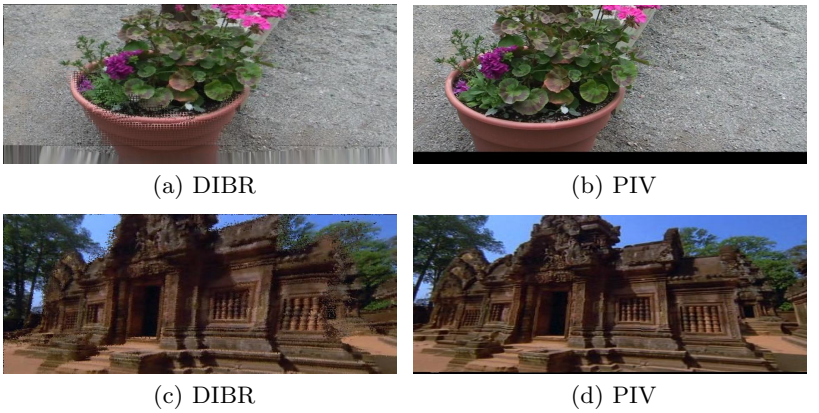


Fig. 3. (a,b) Warped virtual views from left (C_1) and right (C_3) reference camera at viewpoint C_2. (c) View obtained after contour correction and merging (a) and (b). (d) Depth map associated with (c). (e) Final depth map and virtual view (f) obtained after median filtering and hole filling [10] at C_2. (g) PIV resynthesized view at C_2 (h,j) PIV rendered virtual views from arbitrary viewpoints. (k) Classified PIV novel view (g). (l) Label associated with each geometric class. (m,q) Confidence with each label. (r) Graph cut minimized planes family labels. (s) Final depth map at C_2.



(a) DIBR (b) PIV

(c) DIBR (d) PIV

Fig. 4. (a,c) DIBR [10] rendered views of Flower and Temple sequence (holes, cracks). (b,d) Proposed PIV rendered views (realistic, fine texture details are preserved).

3.3 Performance Evaluation and Camera Calibration Results

Tab. 2 shows average CPU time for performing the elimination of variables using KSY method[9], finding structure parameters (*Algorithm A*) and rendering a novel view using estimated parameters. These timing are noted on Intel(R) Core(TM) i3 2.13 GHz PC with 3 GB of RAM with unoptimized matlab code.

Table 2. Computation time (sec) for estimation of structure coefficients (ESC) and rendering a novel view (RNV)

	Merton	Kitchen	Temple
No. of points	40	36	35
No. of input images	3	5	10
Time(sec)	10.32(ESC) 6.224(RNV)	15.48(ESC) 5.602(RNV)	30.01(ESC) 5.446(RNV)

3.4 Camera Calibration

To evaluate the performance of camera self calibration using classified scene data, results are compared with ground truth calibration data available with Temple[1] and Kitchen sequence. The number of cameras varies from 2 to 8 (Tab. 3).

4 Discussion and Conclusions

We present a flexible architecture for multi-view 3DTV build on a novel parameterized variety based representation and rendering scheme. The scheme allows to render a continuum of virtual views from arbitrary viewpoints using few sample

Table 3. Percentage error (%) in focal length estimation

No. of camera views	2	3	4	5	6	7	8
Temple	1.36	1.04	0.76	0.77	0.91	0.99	0.68
Kitchen	1.31	1.33	1.47	0.72	0.79	0.79	0.71

images. It provides a parameterization of all possible views and overcome the shortcomings of depth based methods. The signal representation is bandwidth efficient, compatible with standard multiview coding schemes and adaptable with 2D/3D displays. It duly supports the existing multi-view 3D systems based on depth based representations, by generating high quality views and per-view depth maps from arbitrary camera viewpoints. Looking at these advantages, rendering time is not a critical issue. It can be substantially reduced with GPU implementation of this scheme, which is our next target.

References

1. Genc, Y., Ponce, J.: Image Based rendering using parameterized image varieties. International Journal of Computer Vision 41, 143–170 (2001)
2. Hoeim, D., Efros, A.A., Hebert, M.: Geometrical context from a single image. In: International Conference on Computer Vision, vol. 1, pp. 654–661 (2005)
3. Gallup, D., Frahm, J.M., Mordohai, P., Yang, Q., Pollefeys, M.: Real-Time Plane Sweeping Stereo with Multiple Sweeping Directions. In: CVPR, pp. 1–8 (2007)
4. Sturm, P.F., Maybank, S.J.: On Plane Based Camera Calibration: A General Algorithm, Singularities, Applications. In: CVPR (1999)
5. Lewis, R.H., Stiller, P.F.: Solving the recognition problem for six lines using the Dixon resultant. IMACS 49 (1999)
6. Jenkins, M.A., Traub, J.F.: A three-stage variable-shift iteration for polynomial zeros and its relation to generalized rayleigh iteration. Number. Math. (1970)
7. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision (2002)
8. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. TPAMI 23, 1222–1239 (2001)
9. Nakos, G., Williams, R.: Elimination with the Dixon Resultant. Mathematica in education and research 6, 11–21 (1997)
10. Zinger, S., Doa, L., de With, P.H.N.: Free-viewpoint depth image based rendering. Visual Communication and Image Representation 21, 533–541 (2010)
11. Lhuillier, M., Quan, L.: Edge-constrained joint view triangulation for image interpolation. In: CVPR, vol. 2, pp. 218–224 (2000)
12. Horaud, R., Dornaika, F., Lamiroy, B., Christy, S.: Object pose: The link between weak perspective, paraperspective and full perspective. IJCV (1997)
13. Kapur, D., Saxena, T., Yang, L.: Algebraic and geometric reasoning using the Dixon resultants. In: ACM ISSAC, Oxford, England, pp. 99–107 (1994)

Quasi-regular Facade Structure Extraction

Tian Han, Chun Liu, Chiew Lan Tai, and Long Quan

Hong Kong University of Science and Technology

Abstract. In this paper we present a novel two-stage framework for extracting what we define as a quasi-regular structure in facade images. A quasi-regular structure is an irregular rectangular grid representing the placements of repetitive structural architecture objects, e.g., windows, in a facade. Such a structure generalizes a perfect lattice structure generated by the 2D symmetry groups, studied by the previous work. First, we propose to formulate the quasi-regular structure detection in an object-oriented Marked Point Process framework by treating the architectural elements as objects. This leads to an initial quasi-regular structure map which serves as an indicator map of potential object locations. Then, we propose a regularization scheme to recover the complete quasi-regular structures from the initial incomplete structure. This stage takes advantage of the intrinsic low rank constraint of the quasi-regular structure representing a regularized facade. By applying such a regularization, the complete quasi-regular facade structure is obtained. We have extensively tested our method on a large variety of facade images, and demonstrated both the effectiveness and the robustness of our two-stage framework.

1 Introduction

Facade image analysis has become a very active field in the Computer Vision community over the last few years. A crucial step of this analysis is to extract repetitive structures in facade images, as they exhibit the potential layout of the building facade. Such receptive structures can assist structural elements detection with precise geometries and texture information for detailed 3D facade modeling in large scan urban reconstruction [7].

Although numerous methods have been proposed [1,2,4,6,8,13,16,22,28], it is still challenging to detect facade structures due to several limitations: (1) Facade structures exist in a large range, from a simple lattice defined by symmetry groups to various specific layouts belonging to particular building styles. Less work has been done to tackle the structure extraction in unified way; (2) Some methods utilize directly the facade segmentation to analyze the facade structure. However, since these segmentations are often highly corrupted by severe occlusions, significant noise and large illumination variations, the results are not robust; (3) In some methods, in order to analyse the facades, windows are modeled with too specific assumptions, which limit their uses. For example, edge models are efficient in modern buildings yet may not be suitable for facades with large occlusions from overhanging structures such as balconies. Such limitations prevent the use of facade structure analysis in real urban modeling applications.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 552–564, 2013.

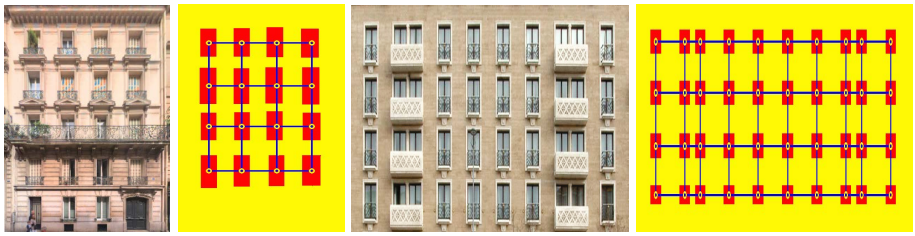
To address the above-mentioned limitations, we present a two-stage facade structure extraction framework. Our contributions are:

- A quasi-regular structure that generalizes the perfect lattice structures, and yet pertains the intrinsic low rank constraints, nicely subject to efficient optimization for the structure recovery.
- A window detection method using a novel object-oriented marked point process model. By treating the windows as geometric objects in the structure detection stage, our method is more stable against pixel-wise noise.
- An efficient regularization that enforces the intrinsic low rank constraints of the quasi-regular structures to recover the complete facade structures.

Related Work. There are two classes of methods that address the repetitive structure extraction problem: low-level detection based on various features and high-level parsing based on grammars and matrix. The low-level approaches mainly focus on discovering repetitive patterns along vertical and horizontal directions based on various features, thus obtain a regular layout of the facade image. On the contrary, high-level approaches impose strong prior constraints on the structural regularity in facade images and then validate the proposed layout through the low-level image cues. Note that the high-level methods actually indirectly address the structure extraction problem by simply connect the neighboring parsed objects. These two types of methods usually differ in terms of their expressive power and scalability.

Low-level approaches measure similarity of different elements in the facade images by local features, e.g edge features [1] and line feature [2] (refer to [27] for a systematic survey). Park et al. [6] extracted three types of features and then made use of their complementarity to group them and obtained a complete lattice. To further improve the robustness of detection, Zhao and Quan [8] used transform space voting technique to extract lattice structures from facade images. These methods all rely heavily on local features, thus they are sensitive to occlusion and noise. In contrast, our method is robust enough to handle relatively large occlusions by using the high-level matrix rank constraint.

There are two classes of high-level approach: grammar-based parsing and matrix-based parsing. On the one hand, grammar-based methods represent facades by using a set of basic shapes and user-defined rules. Alegre et al. [4] proposed the first solution to facade structure interpretation based on probabilistic context-free grammar. Similar to [4] Ripperda et al [20] used reversible jump Markov Chain Monte Carlo (rjMCMC) to optimize the defined probabilistic context-free grammar. Recently, Teboul et al. [3] developed a facade parsing system based on the 2D split grammar and obtained excellent results on Haussmannian buildings. The major limitation of these grammar-based methods is that they need users to design the domain specific rules. Moreover these methods are often computationally heavy. Comparably, our method only uses the grid pattern assumption to extract the repetitive structures in different building styles without user-defined grammars and is straightforward. On the other hand, the emerging matrix-based method [22] provided a novel perspective to



(a) (b) (c) (d)

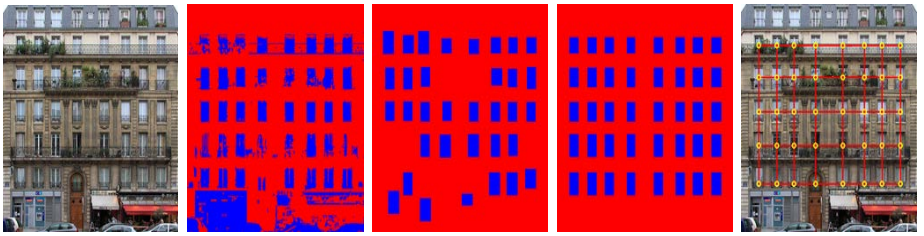
Fig. 1. Examples of regular structure and quasi-regular structure in facade images; (a) and (b), regular facade and its structure; (c) and (d), quasi-regular facade and its structure

view the facade parsing problem and formulated the parsing problem into a matrix approximation problem. Our method presented in structure regularization stage follows closely with [22] but is superior to it in two ways: (1) We impose the rank one constraint in initial structure map (see Figure 2(c)) instead of the wall/non-wall segmentation map (see Figure 2 (b)). The reason is that directly use of rank constraint in high noisy and inconsistent segmentation map may lead to inpredicable results. (2) [22] cannot handle non-rectangular windows since the segmentation map, when treated as a matrix, is not rank one anymore. As we use rectangles to approximate windows in structure detection stage, we can extract structures from windows with arbitrary shapes.

2 Quasi-regular Structure Extraction

In this section, we first define the quasi-regular structure extraction problem and then give an overview of our extraction method. *Quasi-Regular Structure* is a pattern of straight lines that cross each other and form different sized rectangules, see Figure 1 (c) and (d). It is more general than a lattice (formed by same sized rectangles), see Figure 1 (a) and (b). Therefore, such general grid structure can represent large varieties of window patterns in building facades. Our goal is to find a set of rectangular window nodes to construct underlying quasi-regular structure.

Often, the facade data, images or 3D scans, suffer from significant noise, severe occlusion and distortion. In order to extract the structure from highly corrupted data, we propose a framework that consists of two stages: structure detection stage and structure regularization stage. In the structure detection stage, we try to find the potential window rectangles based on facade data. Then in the subsequent structure regularization stage, we enforce a global regularity constraint on the detected window rectangles to get the final structure. A complete example illustrating our framework is shown in Figure 2.



(a) (b) (c) (d) (e)

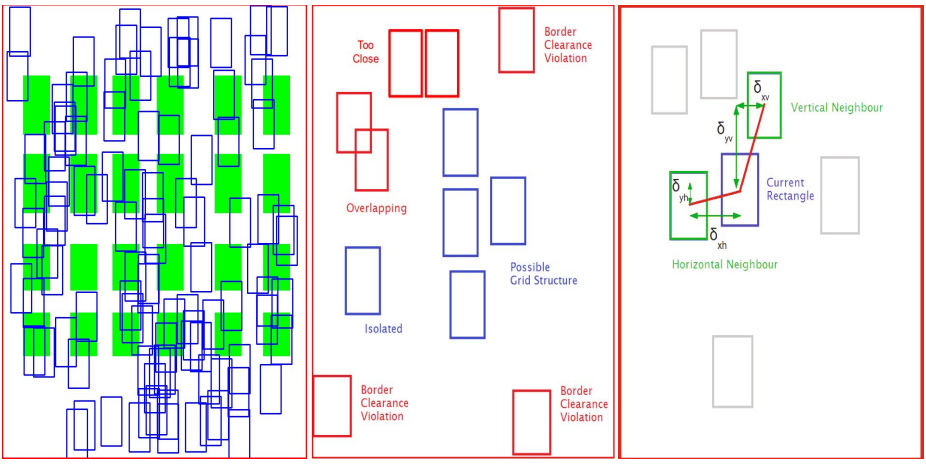
Fig. 2. Our Method consists of two steps: structure detection and structure regularization. (a) the original facade image; (b) wall/non-wall segmentation map input; (c) initial structure map obtained by detecting potential window rectangles with marked point process model; (d) complete structure map after structure regularization by using low rank constraint; (e) final extracted quasi-regular structure on the original image.

3 Structure Detection

Recognizing windows from facade data is a hard inverse problem in computer vision because the input data is often highly corrupted. We thus take a probabilistic approach to solve such problem.

Inspired by the work of [29,5], we use the marked point process model to extract window rectangles from the segmentation map by fitting rectangles. The marked point process model gives an object-oriented approach by allowing to model both geometric object (shape) and object interactions (object layout) at the same time. In our context, we need to model rectangular windows in 2D facades respecting the object evidence in facade data and the spatial topology. The window spatial arrangement relationship can be considered object interactions. Such interactions can include horizontal and vertical alignment, as well as non-overlapping and not being too close to facade borders requirements. The window positions are sampled from the facade space and a window rectangle proposal can be set up by placing a rectangle centered at these positions. Then such a proposal is evaluated by considering both the window likelihood computed from the segmentation map and the spatial relationship. Thus the structure detection problem is transformed into the problem of finding the optimal window rectangle configuration X over 2D facade as a set of rectangles $\{x_0, \ldots, x_{n-1}\}$ (see Figure 3 (a)).

By using marked point process model, we now define structural window rectangle detection problem and give the solution in an energy minimization framework. Let $\{x_0, \ldots, x_{n-1}\}$ denote n rectangles configuration X in a 2D image, we can define the energy function $U(X)$ for penalizing improper configurations which violate data observation and rectangles layout (grid in our context) with large values. Thus this energy function includes the data term $U_{data}(X)$ representing data likelihood and the interaction term $U_{inter}(X)$ enforcing the object grid layout and non-overlapping constraints (see Equation 1). The data term is the sum of individual data term U_D among all rectangles. The interaction term



(a) (b) (c)

Fig. 3. Modeling the window rectangle configurations, please zoom in for details; (a) marked point process for structure extraction; the optimal configuration is in green. (b) window rectangle interactions; (c) definition of window rectangle horizontal and vertical neighbours.

consists of four components including the attraction term U_A, repulsion term U_R, overlapping penalization U_O and image border clearance term U_B.

$$\begin{aligned} U(X) &= U_{data}(X) + U_{inter}(X) \\ &= \textstyle\sum_{x_i \in X} U_D(x_i) + U_O(X) + \sum_{x_i \in X} \left(U_A(x_i) + U_R(x_i) + U_B(x_i)\right) \end{aligned} \tag{1}$$

Next we describe more details on each of the items.

Data Term U_{data} We use the facade wall/non-wall segmentation map to compute the data term. It is the sum of individual window likelihood energy U_D among all window rectangles. Given a window rectangle x_i parametrized by position $(x,\ y)$ and size $(w,\ h)$, we sum up all the pixel values inside the rectangle over the segmentation map. Then we divide the sum by the rectangle size to obtain a window likelihood ratio r_i. With this ratio and a threshold T, we define the individual window likelihood energy U_D as follows:

$$U_D(x_i) = \begin{cases} 1 - \theta \times (r_i/T)^2 & \text{if } r_i < \mathrm{T} \\ -(r_i/T)^2 & \text{if } r_i \geq \mathrm{T} \end{cases} \tag{2}$$

Interaction Term U_{inter} The interaction term consists of four components based on different considerations (see Figure 3 (b)). The attraction term U_A favors neighbouring rectangles according to horizontal and vertical alignment. For each rectangle x_i parametrized by size (*width* for width, *height* for height), the attraction term is defined with its closest horizontal neighbour x_h and its closest vertical neighbour x_v.

$$\begin{aligned} U_A(x_i) &= U_{ah}(x_i, x_h) + U_{av}(x_i, x_v) \\ &= -\alpha e^{(-\|\delta_{yh}/height\|)} - \beta e^{(\|\delta_{xv}/width\|)} \end{aligned} \tag{3}$$

δ_{yh} is the vertical distance between horizontal neighbours, δ_{xv} is the horizontal distance between vertical neighbours (see Figure 3 (c) for details), α and β are the weighting coefficients.

The repulsion term U_R prevents rectangles from being too close. We set two variable thresholds for horizontal distance (t_x) and vertical distance (t_y) between rectangles. These two thresholds are designed for handling different window spacings in various facade styles. And the repulsion item is defined on the closest neighbour of x_i.

$$U_R(x_i) = \begin{cases} 0 & \text{if } \delta_x > t_x \text{ and } \delta_y > t_y \ . \\ E_0 & \text{otherwise} \end{cases} \tag{4}$$

δ_x and δ_y are horizontal and vertical distances between neighbours correspondingly. E_0 is a positive number.

The overlapping term U_O (see Equation 5) is computed on the overlapping count number N. E_1 is a positive number for penalty. The border clearance term U_B prohibits rectangles being too close to the facade border.

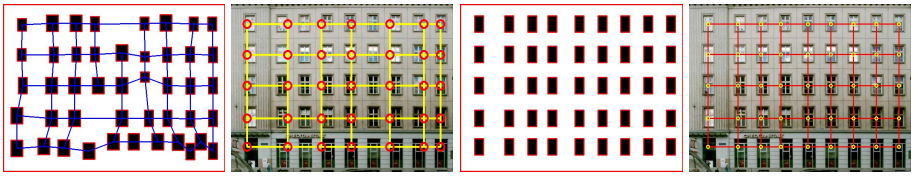
$$U_O(X) = E_1 N. \tag{5}$$

$$U_B(x_i) = \begin{cases} 0 & \text{if (x,y)}\in[w,W-w]\times[h,H-h]. \\ E_2 & \text{otherwise} \end{cases} \tag{6}$$

w and h are minimal horizontal and vertical border distances to windows respectively. W and H are facade size in width and height respectively. E_2 is a positive number for penalty.

By minimizing the energy function, we obtain the optimal set of rectangles, $X^* = \underset{X}{\text{argmax}}\ U(X)$, which constitute the desired initial structure.

Optimization. We use the Monte Carlo sampler for the structure optimization. At each iteration, we perturb the current configuration by adding rectangles or removing rectangles. The new configuration is accepted or denied using the Hasting-Metropolis algorithm. Because the speed of the algorithm is critical in facade structure analysis (it should converge in a reasonable time.), we do several customizations. In the birth move, we modify the rectangle in a small neighbourhood and change its size to become a locally optimized rectangle. Near the end of the optimization process, we apply add-birth-in-the-grid-neighbourhood move to add missing rectangles. In addition, we set the sampler at very low temperature to further speed up. The extracted rectangle configuration may be sub optimal (see Figure 2(c)). We will apply a structure regularization process to fully recover the irregular structure from the initial structure map formed by the extracted rectangles (see Figure 2(c)) in section 4.



(a) (b) (c) (d)

Fig. 4. Structure completion by direct grid completion (a-b) and by rank one regularization (c-d). black rectangles represent window rectangles; (a) grid completion by direct rectangle connection on the structure map; (b) grid completion result on original facade; (c) rank one regularization from initial structure map; (d) structure regularization result on original facade.

4 Structure Regularization

In this section, our goal is to correct and complete the initial structure. A straightforward structure completion solution can be connecting neighboring rectangles and counting numbers of window rectangle in each row and column respectively. Then the full structure is recovered by aligning rectangle positions and averaging the rectangle sizes. However, this method is prone to failure if the number of windows is largely inconsistent between different floors (See Figure 4 (a) and (b) for example). Thus a more powerful regularization method is needed. We advocate using the rank one constraint (see Figure 4 (c) and (d)).

Rank-One Constraint for Structure Completion. It is observed that the quasi-regular structure (see Figure 1 (b)) has an intrinsic low-rank structure [9,22]. Moreover, if we use a bounding box to represent arbitrary shaped windows, the resulting pattern (see Figure 1 (c)), when treated as a matrix, is in fact rank one. Based on this observation, we can add a rank one constraint to the recovered structure map, i.e if we treat the initial structure map as a 0-1 matrix $D \in R^{m\times n}$, where the entry is equal to 1 if it belongs to the window region in the initial structure map and 0 otherwise, we can find the rank one matrix $A \in R^{m\times n}$, such that the discrepancy between A and D, denoted by $E \in R^{m\times n}$, is as sparse as possible, leading to the following constrained optimization:

$$\min_{E} \parallel E \parallel_1, \text{ subject to } \quad D = A + E, \text{rank}(A) = 1, \tag{7}$$

where $\| \cdot \|_1$ denotes the l^1-norm of a matrix (i.e., the sum of the absolute values of matrix entries). The partial augmented Lagrangian function of (7) can then be defined as:

$$L(A, E, Y, \mu) = \parallel E \parallel_* + < Y, D - A - E > \\ + \frac{\mu}{2} \parallel D - A - E \parallel_F^2$$

where μ is a positive scalar, and Y is the Langrange Multiplier. In order to solve this problem efficiently, we apply the Inexact ALM approach [10] to alternately

update E and A. In each iteration, we use the corresponding soft thresholding operator [21] to update E, and enforce the rank one constraint every time we update A. To implement the rank one constraint, we use the algorithm presented in [22]. To be more specific, we first obtain the Singular Value Decomposition of A, i.e $A = USV^{'}$, then select the largest singular value σ_1, and its corresponding column vectors u_1, v_1 of U and V, update A as $A = u_1\sigma_1 v_1^{'}$ in each iteration; see [10,11,22] for more algorithmic details.

Final Structure Extraction. The output from the structure regularization is a clean complete structure map with a consistent grid pattern window rectangle configuration and can be treated as a rough facade parsing result (see Figure 2 (d)). By extracting and connecting the centers of window rectangles, we obtain the final quasi-regular structure (see Figure 2 (e)).

5 Experiment

We evaluated our method in terms of detection efficiency and robustness on various facade images with different window grid structures. These images are acquired by ourselves in Paris or downloaded from the Internet and rectified by using the method in [9]. Some representative results are shown in Figure 5. In addition, we show comparison results on images from [3] and [8] in Figure 6.

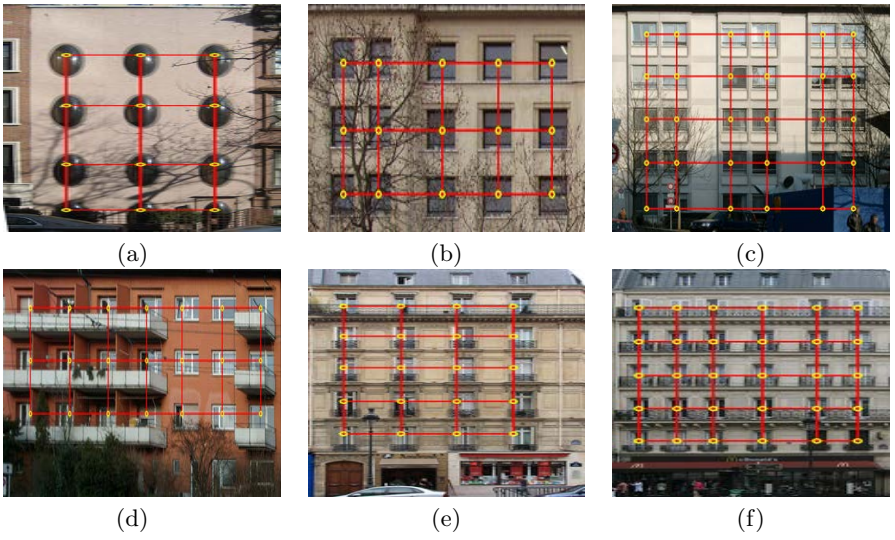


Fig. 5. Various quasi-regular structures extracted by our method. (a) Elliptical Window Facade; (b), (c) and (d) Facades under occlusion and with illumination variations; (e) and (f) Haussmannian facades with significant balcony occlusions over windows. The dormer windows on the roof are rather arbitrary, see (f) for example, we therefore treat them separately in our current system.

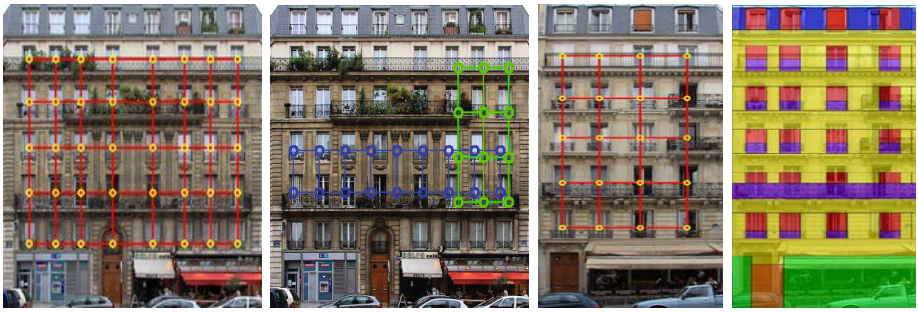
Detection Efficiency. The main computation bottleneck is the structure detection stage since the rank one regularization can be done very fast (i.e < 3 seconds). The running time of the MCMC optimization in the structure detection stage depends on three factors: the window size, the number of windows and the image size. The window size is specified as the width and height range parameters in our system. The larger the range, the longer the computation time. The second factor, the window count number, is related to the facade type and determined in run time. The third factor greatly affects the detection efficiency because it defines the window rectangle searching space. Assume that the rectangle width is in the range of (L_{min}, L_{max}) and the rectangle height is in the range of (H_{min}, H_{max}), and 2D facade image is M by N, the searching space for the rectangles is denoted by $S = [0, M-1]\times[0, N-1]\times[L_{min}, L_{max}]\times[H_{min}, H_{max}]$ if the window rectangles are uniformly sampled on the facade. To make the proposed method more scalable and computationally efficient, we utilize small number of feature points located around windows to guide the rectangle position sampling in the structure detection stage. In our implementation, we use clustered FAST feature points [14] by Normalized Cross Correlation (NCC) because the are scalable to the image size. For a typical 500×500 image with window width in $[40, 60]$ range and window height in $[60, 100]$ range, the overall running time is less than 1 minute for 1000 iterations, allowing interactive facade modeling in a reasonable time.

Robustness. In our framework, we use object-oriented detection based on both low-level features and high-level matrix constraint. Thus our method can handle significant occlusions and illumination changes. Notice that the image in Figure 5 (c), the bottom row of windows are severely occluded by a billboard; windows in Figure 5 (b) and (d) are occluded by vegetation, and windows in Figure 5 (c) and (d) have different light condition from windows in the bottom two rows etc. The results show that our method is robust and able to handle such challenges and recovers the right structure.

Comparative Results. We also performed extensive comparison between the proposed method and the other state-of-art algorithms. Only some of them are shown in Figure 6.

When compared with symmetry detection based methods, our method can detect more general and complete structures (see Figure 6 (a)), while the method proposed by Zhao et al.[8] only extracts a partial structure due to the severe occlusion and tends to separate the irregular structures into small pieces of lattice (Figure 6 (b)).

In comparison with grammar based facade parsing methods (see Figure 6 (c) and (d)), our method can extract the same structure without user-defined specific grammars (Please note that the roofs on Parisian buildings are treated separately outside the structure extraction in our system with a roof detector by using color and edge characteristics). We consider that the grid pattern used in our framework is equivalent to split grammar based facade typology used in [3].



(a) (b) (c) (d)

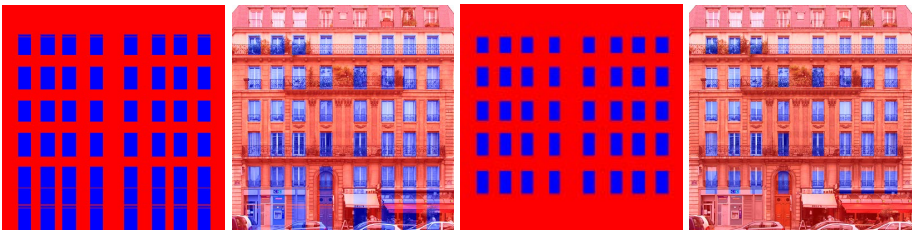
Fig. 6. Comparison. (a),(c) Our results; (b) results of Zhao et al.[8]; (d) results of grammar-based facade parsing method [3]. Note that the roofs on the Parisian buildings are treated separately from structure extraction in our system.

Facades with different number of windows in different floors are not treated as quasi-regular structures, but incomplete structures. Processing such structures using our method may give missing or false positive windows. Grammar-based methods [24] can be helpful in addressing such a problem but involve more specific rules and much higher computation time.

Since the complete structure map obtained after the regularization stage can be treated as the facade parsing result (See Figure 2 (d)), our method can be compared with facade parsing works. Figure 7 shows the comparison between the method of Yang et al. [22] and ours. Both methods use rank-one contraint. However, because we apply the rank-one constraint on the structure map instead of the segmentation map directly, we achieve better result (see Figure 7 (c) and (d)). In terms of true positive window pixel rate, our method achieves 91.2%, while their method achieves only 65.1%. For facade images with significant corruption, the result from [22] are unacceptable (see Figure 8(a)). Further, we can exploit the extracted structure information to segment the repetitive objects by using techniques like [23] and [25], then further refine our initial parsing result. In this way, we can even handle the non-rectangular repetitive objects parsing problem as shown in Figure 8.

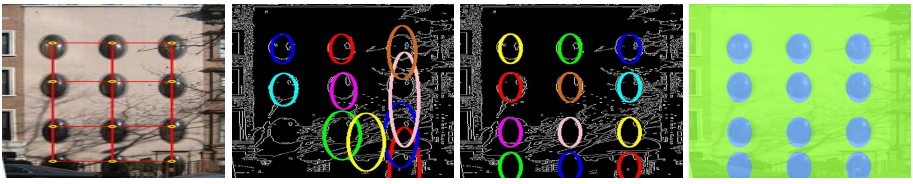
Applications in Facade Modeling. Our structure extraction framework is very promising for complete facade modeling. By knowing the window structure and window shape prior, we can use the color information to estimate the rough sizes of windows. We can then explore the edge information to refine the window geometries so that windows can be detected perfectly in facade image by using similar techniques as in [23]. Figure 9 (b) and (c) show an example of a simplified 3D facade model with window structure extracted by using our method.

Limitations. Our method relies on the wall/non-wall segmentation. If the segmentation is corrupted completely in a floor or column in the facade image, it is not possible to recover the windows unless the higher level building typology information is reinforced.



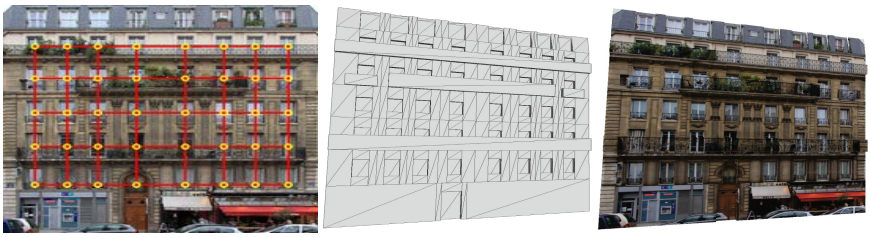
(a) (b) (c) (d)

Fig. 7. Facade parsing comparison with [22]. Blue rectangles represent windows. (a) and (b) show the parsing result of [22] and the result projected on the original image respectively. (c) and (d) present our results. Note that [22] applied rank one constraint on wall/non-wall segmentation map (Figure 2 (b)), while our method use rank one constraint on initial structure map (Figure 2 (c)).



(a) (b) (c) (d)

Fig. 8. Facade parsing with non-rectangular windows. The parsing result (d) is obtained in 2 steps: (1) Given the extracted structure (a), we crop image patches around each node of this structure and apply contour-based elliptical detector [32] to obtain the result in (b). (2) Average all detected ellipses with the median displacements and minimal sizes, and propagate the averaged ellipse over the structure, and the result is shown in (c). Note that ellipse detections at some positions on the structure are not valid so the number of ellipses does not correspond to number of windows shown on (b).



(a) (b) (c)

Fig. 9. Applications in Facade Modeling: (a) Extracted structure using our method; (b) 3D mesh model (dormer windows and doors are added separately); (c) Textured facade model

6 Conclusion and Future Work

We have presented a novel two-stage framework to extract quasi-regular structures from building facade images. In the structure detection stage, we treat the window detection problem at the object level and formulate it under the marked point process model, thus we can effectively locate potential windows. By further integrating the high-level rank one constraint in the regularization stage, more general structures, rather than lattices, are extracted even under severe occlusion. Extensive experiments and comparisons show that the proposed method can efficiently and accurately extract the quasi-regular structures.

In our current framework, we only use low-level hue feature. It is interesting to exploit other low-level cues, such as depth information, edge information, into our model for further boosting performance. Moreover, we shall further investigate multiple-grid window pattern detection in building facades. We leave these as our future work.

References

1. Korah, T., Rasmussen, C.: 2D lattice extraction from structured environments. In: ICIP 2007. Image Processing, vol. 2, pp. 61–64 (2007)
2. Tyleček, R., Šára, R.: A Weak Structure Model for Regular Pattern Recognition Applied to Facade Images. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 450–463. Springer, Heidelberg (2011)
3. Teboul, O., Kokkinos, I., Simon, L., Koutsourakis, P., Paragios, N.: Shape grammar parsing via Reinforcement Learning. In: Shape grammar parsing via Reinforcement Learning. Computer Vision and Pattern Recognition, CVPR 2011, pp. 2273–2280 (2011)
4. Alegre, F., Dellaert, F.: A Probabilistic Approach to the Semantic Interpretation of Building Facades. In: International Workshop on Vision Techniques Applied to the Rehabilitation of City Centres (2004)
5. Ortner, M., Descombes, X., Zerubia, J.: Building extraction from digital elevation models. In: Acoustics, Speech, and Signal Processing, ICASSP 2003, vol. 3, pp. III-337–III-340 (2003)
6. Park, M., Brocklehurst, K., Collins, R., Liu, Y.: Translation-Symmetry-Based Perceptual Grouping with Applications to Urban Scenes. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part III. LNCS, vol. 6494, pp. 329–342. Springer, Heidelberg (2011)
7. Xiao, J., Fang, T., Tan, P., Zhao, P., Ofek, E., Quan, L.: Image-based façade modeling. ACM Transactions on Graphics, TOG 27, 161 (2008)
8. Zhao, P., Quan, L.: Translation symmetry detection in a fronto-parallel view. In: Computer Vision and Pattern Recognition, CVPR 2011, pp. 1009–1016 (2011)
9. Zhang, Z., Liang, X., Ganesh, A., Ma, Y.: TILT: Transform Invariant Low-Rank Textures. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part III. LNCS, vol. 6494, pp. 314–328. Springer, Heidelberg (2011)
10. Lin, Z., Chen, M., Ma, Y.: The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. UIUC Technical Report (2009)
11. Bertsekas, D.P.: Constrained optimization and Lagrange multiplier methods. Computer Science and Applied Mathematics 1 (1982)

12. Shi, J., Malik, J.: Normalized cuts and image segmentation. Pattern Analysis and Machine Intelligence, PAMI 22(8), 888–905 (2000)
13. Schindler, G., Krishnamurthy, P., Lublinerman, R., Liu, Y., Dellaert, F.: Detecting and matching repeated patterns for automatic geo-tagging in urban environments. In: Computer Vision and Pattern Recognition, CVPR, pp. 1–7 (2008)
14. Rosten, E., Porter, R., Drummond, T.: Faster and better: a machine learning approach to corner detection. Pattern Analysis and Machine Intelligence, PAMI 32(1), 105–119 (2010)
15. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision 60(2), 91–110 (2004)
16. Müller, P., Zeng, G., Wonka, P., Van Gool, L.: Image-based procedural modeling of facades. ACM Transactions on Graphics 26(3), 85 (2007)
17. Shi, J., Tomasi, C.: Good features to track. In: Computer Vision and Pattern Recognition, CVPR 1994, pp. 593–600 (1994)
18. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. Image and Vision Computing 22(10), 761–767 (2004)
19. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
20. Ripperda, N.: Grammar based facade reconstruction using rjMCMC. Photogrammetrie Fernerkundung Geoinformation 2, 83 (2008)
21. Cai, J.-F., Candès, E.J., Shen, Z.: A Singular Value Thresholding Algorithm for Matrix Completion. SIAM J. on Optimization 20(4), 1956–1982 (2010)
22. Yang, C., Han, T., Quan, L., Tai, C.L.: Parsing Façade with Rank-One Approximation. In: Conference on Computer Vision and Pattern Recognition, pp. 1720–1727 (2012)
23. Chun, L., Gagalowicz, A.: Image-based Modeling of Haussmannian Facades. The International Journal of Virtual Reality, 13–18 (2010)
24. Riemenschneider, H., Krispel, U., Thaller, W., Donoser, M., Havemann, S., Fellner, D., Bischof, H.: Irregular lattices for complex shape grammar facade parsing. In: Conference on Computer Vision and Pattern Recognition (2012)
25. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. ACM Transactions on Graphics, TOG 23(3), 309–314 (2004)
26. Descombes, X.: Stochastic Geometry for Image Analysis. Wiley (2011) ISBN 978-1-84821-240-4
27. Liu, Y., Hel-Or, H., Kaplan, C.S.: Computational symmetry in computer vision and computer graphics, vol. 1. Now Publishers (2010)
28. Teboul, O., Simon, L., Koutsourakis, P., Paragios, N.: Segmentation of building facades using procedural shape priors. In: Computer Vision and Pattern Recognition, CVPR 2010, pp. 3105–3112 (2010)
29. Ortner, M., Descombes, X., Zerubia, J.: Building outline extraction from digital elevation models using marked point processes. International Journal of Computer Vision 72(2), 107–132 (2007)
30. Tyleček, R., Šára, R.: Modeling symmetries for stochastic structural recognition. In: Computer Vision Workshops (ICCV Workshops), pp. 632–639 (2011)
31. Bokeloh, M., Berner, A., Wand, M., Seidel, H.P., Schilling, A.: Symmetry detection using feature lines. Computer Graphics Forum 28(2), 697–706 (2009)
32. Fitzgibbon, A.W., Pilu, M., Fisher, R.B.: Direct Least Squares Fitting of Ellipses. IEEE Trans. PAMI 21, 476–480 (1999)

Multi-view Synthesis Based on Single View Reference Layer

Yang-Ho Cho, Ho-Young Lee, and Du-Sik Park

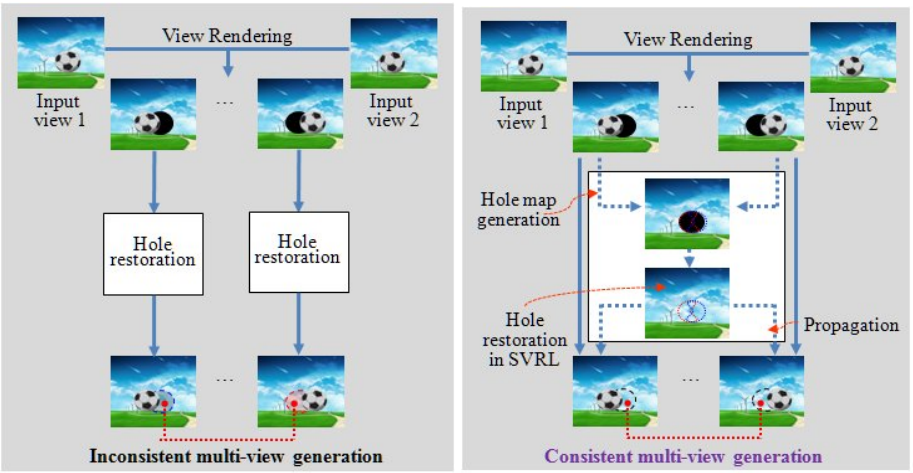
Samsung Advanced Institute of Technology, Republic of Korea

Abstract. We propose a virtual view synthesis method based on depth image-based rendering (DIBR) to realize wide multi-view 3D displays. The proposed multi-view rendering method focuses on reducing the repetitive hole restoration process and generating spatiotemporally consistent multi-views. First, we determine a single view reference layer (SVRL) and set the maximum hole area in this SVRL to cover the maximum hole occurrence in the synthesized views. The hole in the SVRL is also restored by referencing the non-hole region of the current SVRL and the accumulated background data of the previous frame. If the newly uncovered background region exists in the restored SVRL, we continuously accumulate the background region and use it to restore the hole of the next SVRL to achieve temporal consistency of the synthesized views. Finally, the restored hole in the SVRL is propagated to the hole in each synthesized view, thereby preserving the spatial consistency of the synthesized views because the hole region in each synthesized view is restored by using the common SVRL. The experimental results showed that the proposed method generates spatiotemporally consistent multi-view images and decreases the complexity of the hole restoration process by reducing the number of repetitive hole restoration process.

1 Introduction

Recently, various 3D displays have been introduced in the consumer electronics market. These 3D displays are being continuously developed to realize a real 3D environment, for instance, the development of multi-view autostereoscopic displays from stereoscopic displays. In order to generate natural 3D views in a 3DTV system, a number of views are required to cover a wide viewing angle without the use of 3D glasses. However, it is not easy to capture, store, and transmit multi-view images because of the physical limitations of a capturing system, the quantity of data, the bandwidth for broadcasting, and other factors. Thus, the 3D reproduction system needs to synthesize virtual viewpoint images from a small number of captured input views in order to provide a wider viewing angle than that offered by the input views[1,2]. Many virtual viewpoint images need to be synthesized to enhance the viewing angle of a multi-view autostereoscopic display. We use a DIBR to synthesize multi-views. The intermediate views are interpolated and the outside views are extrapolated on the basis of the input views and disparities[3,4]. In this paper, we propose a view rendering framework

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 565–575, 2013.



(a) (b)

Fig. 1. Comparison of view rendering framework: (a) conventional view rendering and (b) proposed view rendering

for generating spatiotemporally consistent views. Fig. 1 shows the block diagram of conventional view rendering and the proposed view rendering. Conventional view rendering needs multiple hole restoration processes, the number of which is determined by the number of newly synthesized views; this increases the complexity of view rendering. Additionally, conventional view rendering methods focus on generating each target viewpoint image, rather than considering the consistency of the whole synthesized view[5,6,7].

Although the quality of each synthesized view is acceptable, if the coherence of each synthesized view is not sufficient, the quality of 3D perception may be poor because 3D perception is achieved by displaying two adjacent synthesized views to the left eye and the right eye. Therefore, we propose a novel view-rendering framework based on an SVRL, which reduces the number of hole restoration repetitions to one and preserves the spatiotemporal consistency of each synthesized view. The proposed method sets the maximum hole area in the SVRL and restores the hole by using the temporally accumulated background region and non-hole region in the current SVRL. Finally, the restored hole is propagated to the hole of each synthesized view. Accordingly, all synthesized views have spatiotemporal consistency because they are generated by a common SVRL.

2 Proposed View Rendering

The block diagram of the proposed view rendering is shown in Fig. 2. If the disparities of input views are not available, they should be estimated using a multi-view disparity estimation algorithm[4]. Then, we determine the center view of

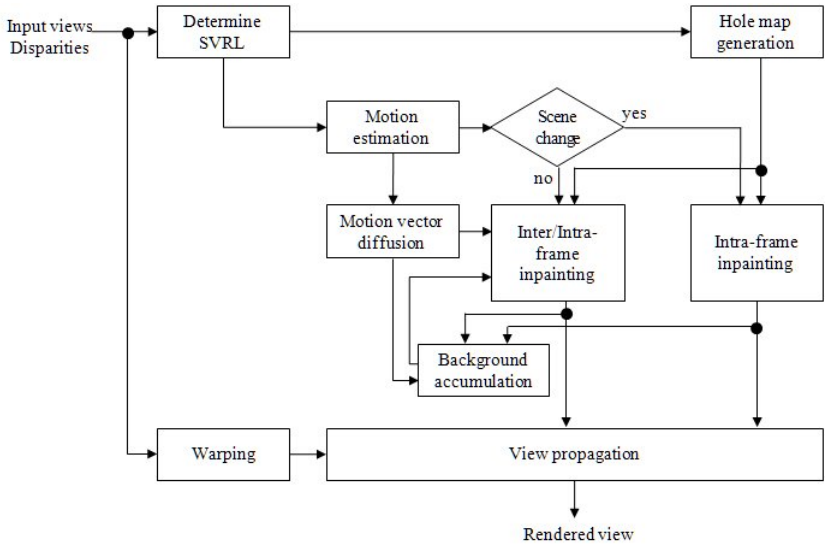


Fig. 2. Block diagram of proposed view rendering

the input views as an SVRL. If the number of input views is even, the interpolated view at the center of the input views is set as the SVRL. Next, a hole map is defined in the SVRL to cover the maximum hole area of all synthesized multi-views. These hole areas are restored by combining the results of inter/intra frame inpainting. In order to reference the inter-frame information, we conduct motion estimation and motion vector (MV) diffusion. The estimated MV is used to restore the hole in the SVRL and accumulate the background region continuously, until no scene change occurs in the sequence. Finally, the input views are warped to the target viewpoint and we propagate the restored hole in the SVRL to the hole of the target multi-views. Therefore, the proposed method can generate spatiotemporally consistent multi-views with a single hole restoration process in the SVRL.

2.1 Hole Map Generation

Most of the hole area is a hidden background area that is uncovered when we change the viewpoint in a real 3D environment, as shown in Fig. 3. Thus a hole map can be defined by using the disparity difference and the output/input baseline ratio. If the disparity difference is positive, the hole can be defined at the right side of current pixel position, when we synthesize multi-views. Otherwise the hole can be defined at the left side of current pixel position. The magnitude of the hole is set as

$$[x, x + \alpha \cdot \Delta d] \in \Psi \qquad \text{if } \Delta d > 0 \tag{1}$$

$$[x - 1 - \alpha \cdot \Delta d, x - 1] \in \Psi \qquad \text{if } \Delta d < 0 \tag{2}$$

where Ψ is the hole area in the SVRL, $\Delta d = d(x, y) - d(x - 1, y)$ is the disparity difference between neighborhood pixels, and α is the output/input baseline length ratio. If the output baseline is larger than the input baseline, we need to scale the disparity difference according to the max hole occurrence in the synthesized views.

2.2 Background Motion Estimation and Diffusion

In order to increase the referencing data for hole restoration in the SVRL, the proposed method continuously accumulates the background data of previous frames because most of the hole region is a background region that was covered by a foreground object in an input view. If we can accumulate the background region of previous frames, the accuracy of hole restoration will increase. In order to accumulate the background region, it is necessary to estimate the background MV occluded by foreground objects. First, we estimate the MVs of successive frames. Then, the MV of the foreground object is iteratively replaced as the MV of the background region to accumulate the background data of previous frames.

In order to diffuse the background MV to the foreground objects, we compare the magnitude of disparity among 4-neighborhood pixels. If the disparity of one of the neighborhood pixels is smaller than that of a current pixel, the MV of the current pixel is replaced with that of a neighborhood pixel. Fig. 4 shows the result of background MV diffusion. Fig. 4(a) is a disparity, Fig. 4(b) is an initially estimated MV (red MV), and Fig. 4(c) is a result of background MV

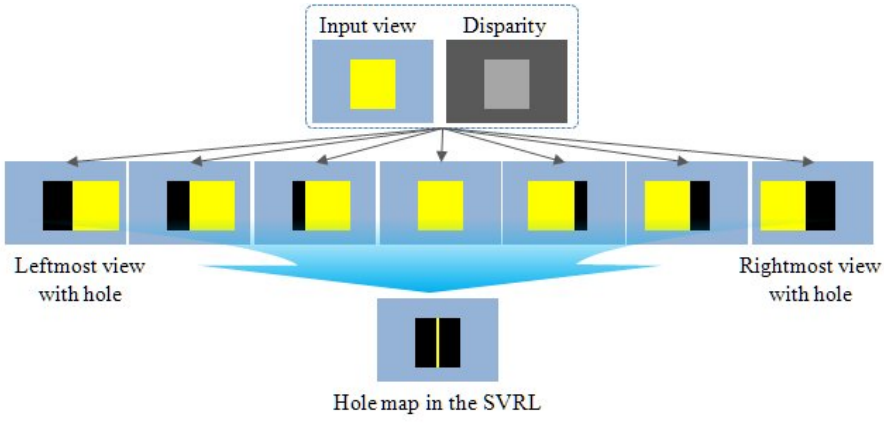
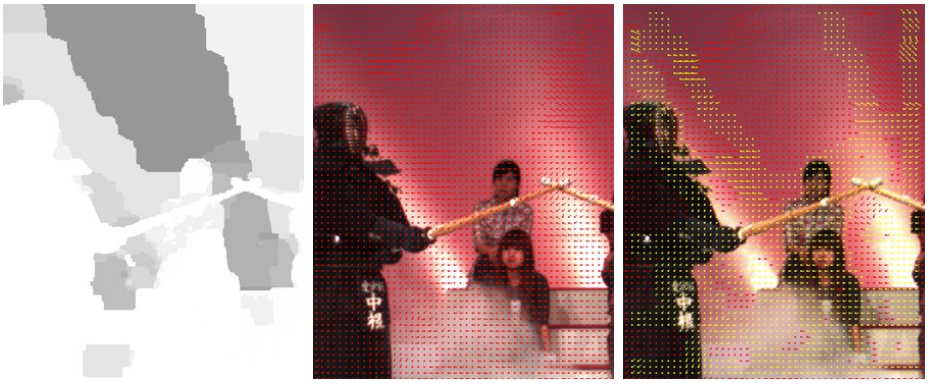


Fig. 3. Multi-view synthesis based on SVRL; The yellow square is a foreground object that has a higher gray value than backgound in the disparity



(a) (b) (c)

Fig. 4. Kendo sequence; (a) disparity, (b) initial MV, and (c) result of background MV diffusion

diffusion (yellow MV) at the boundary of the foreground and background. The background MVs are iteratively diffused to the foreground object for background accumulation and inter-frame inpainting.

2.3 Background Accumulation

According to the diffused background MVs, the background data is accumulated for hole restoration of current SVRL. We compare the disparity of the current SVRL and that of the previous accumulated background data. Accordingly, the current background data is updated as the current SVRL or previously accumulated background data, depending on which has the minimum disparity.

$$C_t^{BG}(x,y) = \begin{cases} C_{t-1}^{BG}(x+i, y+j), & \text{if } d_t^{SVRL}(x,y) > d_{t-1}^{SVRL}(x+i, y+j) \\ C_t^{SVRL}(x,y), & \text{otherwise} \end{cases} \quad (3)$$

where $C_t^{BG}(x,y)$ is the current background data and $d_t^{SVRL}(x,y)$ is the disparity of the current SVRL at (x,y). (i,j) is the background MV. Fig. 5 shows the result of accumulating the background data after 10, 30, and 50 frames. We can observe that the foreground objects are gradually removed from the background accumulation.

2.4 Inter/Intra Frame Inpainting

The proposed method restores the hole of the current SVRL by referencing the accumulated background data(inter-frame inpainting) and the non-hole region of the current SVRL(intra-frame inpainting). We combine the results of inter- and intra-frame inpainting as

(a) (b) (c)

Fig. 5. Accumulated background data; (a) 10 frames, (b) 30 frames, and (c) 50 frames

$$C_{hole}(x,y) = (1-\beta)\cdot C_{inter}(x,y) + \beta \cdot C_{intra}(x,y) \tag{4}$$

where $C_{inter}(x,y)$ and $C_{intra}(x,y)$ are the results of inter- and intra-frame inpainting, and β is a weight factor for combining the results of inter- and intra-frame inpainting. We use an exemplar-based inpainting method that replicates the texture and structure of the known image areas in the holes by using exemplar patches[8]. In multi-view rendering, it is important to fill the holes with background data in order to prevent the propagation of foreground data to the hole area. Accordingly, the proposed method gives high priority to the background region based on the disparity. In addition, we use two methods to exclude foreground pixels when finding the optimal patch fetching process. First, if the foreground pixels and background pixels are mixed in a target patch, we exclude the foreground pixels and use only the background pixels to find the optimal patch. Second, we search for the optimal patch only in a search range that has a disparity smaller than that of the target patch. Both methods are applied to prevent the propagation of the foreground pixels. The patch priority for exemplar-based inpainting[8] is defined by multiplying the structure sparsity term(ρ_p), the patch confidence term(C_p), and the inverse disparity term(ID_p). ρ_p is defined to measure the confidence of a patch located at structure instead of texture. C_p is defined as the number of non-hole pixels in the target patch. ID_p is an additional priority measure for restoring the hole region with a background region rather than a foreground region. However, the minimum disparity of the neighborhood pixel is regarded as the disparity of the hole pixel, because the hole pixel(p) does not have a disparity.

$$ID_p = \frac{1}{\underset{q\in N(p)}{min}\ d(q)} \tag{5}$$

where $d(q)$ is a disparity within a neighborhood pixel of the hole pixel p. The patch with the highest priority is the first selected for restoration. The target

patch is filled by fetching the best match patch from the known region. The best match patch for inter- and intra-frame inpainting is determined by

$$C_{Inter} = \underset{B_S \in \Phi_{inter}}{\operatorname{argmin}} \left(|B_q - B_S| + \mathrm{CENSUS}(B_q, B_S) + |\mathbf{q} - \mathbf{S}| \right) \tag{6}$$

$$C_{Intra} = \underset{B_T \in \Phi_{intra}}{\operatorname{argmin}} \left(|B_q - B_T| + \mathrm{CENSUS}(B_q, B_T) + |\mathbf{q} - \mathbf{T}| \right) \tag{7}$$

where the target patch B_q is an $N \times M$ block centered at the hole pixel q. B_S and B_T are reference blocks that do not have a hole pixel in the search range (Φ_{inter} and Φ_{intra}), whose textural information can be copied to the hole pixel. The search range for inter-frame inpainting is shifted according to the background MV at p. CENSUS(.) is the costs calculated using the Hamming distance of two CENSUS transform pixels[9]. In order to restore the hole with the background region, we exclude the foreground pixel in the target patch and search range. First, we calculate the median disparity of a target patch and classify the target patch into foreground or background pixels according to the median disparity. The pixel classified as a foreground pixel is not used to find the best match patch. Second, each source patch in the search range is also classified as a foreground or background region on the basis of the median disparity of the target patch, and the foreground region is excluded to restore the hole. The weight factor for combining inter- and intra-frame inpainting is determined by how many uncovered background pixels exist within the search range. The proposed method assumes that the hole has to constitute background rather than foreground data.

$$\Phi(p) = \frac{\sum_{q \in \Phi_p} \Phi(q)}{|\Phi_p|} \tag{8}$$

where $|\Phi_p|$ is the area of search range. $\Phi(q)$ is set to as follows;

$$\Phi_q = \begin{cases} 1, & \text{if } d(q) < median(d(B_q)) \\ 0, & \text{otherise} \end{cases} \tag{9}$$

Then, the weight factor for combining inter- and intra inpainting is determined by comparing $\Phi_{inter}(p)$ and $\Phi_{intra}(p)$. If $\Phi_{inter}(p)$ is larger than $\Phi_{intra}(p)$, β is set to 0, otherwise β is set to 1. If $\Phi_{inter}(p)$ is equal to $\Phi_{intra}(p)$, is set to 0.5.

2.5 View Propagation

The restored hole in the SVRL is propagated to the hole in the synthesized views, as shown in Fig. 6. To synthesize the virtual view, given a pixel located at (x_r, y_r) in the image coordinate system of the reference view, we can derive its correspondent location (x_v, y_v) in the virtual view using a warping process. Assuming that the array arrangement of the cameras is 1D parallel and that they have the same focal length and rotation matrix, the 1D shifting method can be considered instead of 3D warping because the disparity only occurs along

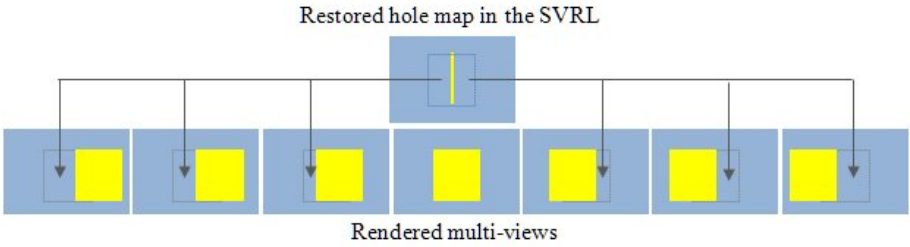


Fig. 6. View propagation

x-axis. Thus, the formulation of the corresponding pixel position is represented by a disparity d:

$$x_v = x_r + (\frac{f \cdot l}{z} + dx) = x_r + d \tag{10}$$

where f is focal length, l is the baseline spacing, z is the depth value and dx is the difference in principal point offset.

3 Experimental Results and Discussion

We assumed that there were three input views and that the maximum extrapolated viewpoint was located at an inter-pupillary distance (IPD) of 2 from the outer input view. MPEG 3DV test sequences (5 sets, 100 frames) and Middlebury test images (30 sets) were used to evaluate the performance of the view synthesis. The peak signal-to-noise ratio (PSNR) between the original input views and the synthesized views were calculated for each test sequence.

Fig. 7 shows the test configuration. Two interpolated (IP) views and two extrapolated (EP) views were used to evaluate the performance of the proposed method. However, in case of EP views, we used only two input views to generate

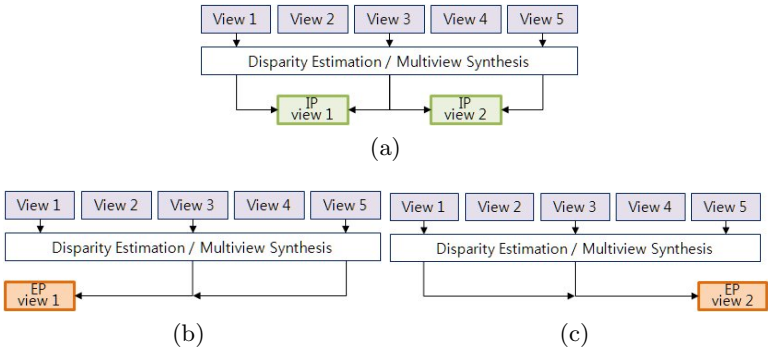


Fig. 7. Test configuration.; (a) interpolated view generation, (b) left extrapolated view generation, and (c) right extrapolated view generation

(a) (b) (c)

Fig. 8. Result of inter- and intra-frame inpainting; (a) inter-frame inpainting, (b) intra-frame inpainting, and (c) combined result of inter- and intra-frame inpainting

Table 1. PSNR result of the proposed method

Test sequence (view no.)	IP view	EP view	PSNR-IE
Bookarrival(12,10,8)	37.45	34.12	35.79
Newspaper(1,3,5)	35.27	30.80	33.03
Balloon(1,3,5)	36.61	34.11	35.36
Kendo(1,3,5)	36.16	32.87	34.52
Pantomime(48,50,52)	35.35	33.98	34.66
Middlebury(1,3,5)	40.58	34.41	37.50
Average	**36.90**	**33.38**	**35.14**

the EP views because the number of input views was limited to evaluate the PSNR for view extrapolation. The average PSNR of the generated multi-views was higher than 35 dB, which is equal to the reference PSNR for the current HD broadcasting of images in the image compression area. Fig 8(a) and 8(b) show the results of inter- and intra-frame inpainting. The combined result of inter- and intra-frame inpainting is shown in Fig 8(c). The left side of man is hole region to be restored. In this sequence, the result of inter-frame inpainting is better than that of intra-frame inpainting because the hole region is uncovered at the previous frame. Spatiotemporally consistency was checked by subjective quality evaluation. Fig. 9 shows the results of the conventional view synthesis method and the proposed method. Fig. 9(a) shows the initially warped views, which are spatially successive frames. Figs. 9(b) and 9(c) are the results of individual hole restoration based on view synthesis reference software 3.5 (VSRS 3.5)[5] and exemplar-based inpainting[8]. Fig. 9(d) is the result of using the proposed

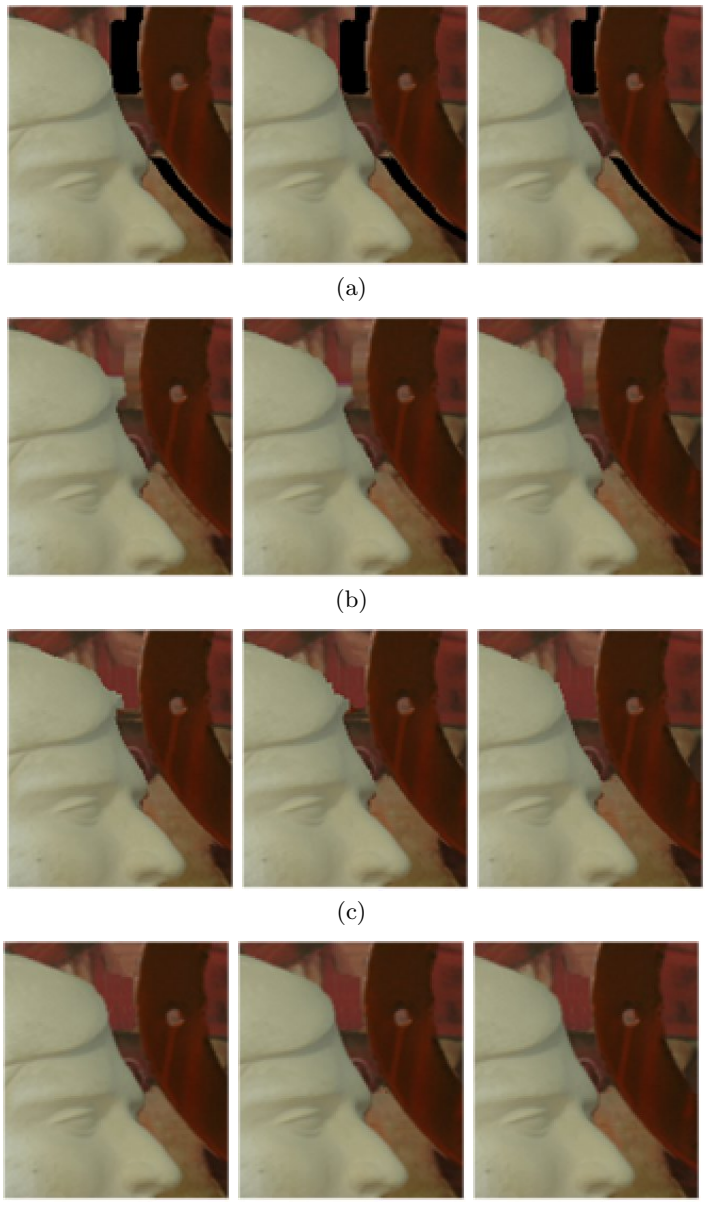


Fig. 9. Result of view synthesis; (a) initially warped views, individual hole restoration based on (b) VSRS 3.5[5], (c) exemplar-based inpainting[8], and hole restoration based on (d) the proposed SVRL

SVRL method. It can be seen that with conventional view-rendering methods, the same background regions are restored differently, whereas the synthesized views of the proposed method give view consistency. The proposed method uses only one SVRL to generate the multi-views, which enhances visual quality for 3D displays.

4 Conclusion

We proposed a multi-view rendering framework based on the SVRL to reduce the repetitive hole restoration process and to generate spatiotemporally consistent multi-view images. In order to restore the hole in the SVRL, the results of inter- and intra-frame inpainting are combined and the background data are accumulated continuously to achieve temporal consistency. Finally, the restored hole region in the SVRL is propagated to the hole of each synthesized view to preserve the spatiotemporal consistency of each synthesized view.

References

1. Tanimoto, M., Tehrani, M.P., Fujii, T., Yendo, T.: Free-viewpoint TV. IEEE Signal Processing Magazine 28, 67–76 (2011)
2. Bartczak, B., Vandewalle, P., Grau, O., Briand, G., Fournier, J., Kerbiriou, P., Murdoch, M., Muller, M., Goris, R., Koch, R., van der Vleuten, R.: Display-independent 3D-TV production and delivery using the layered depth video format. IEEE Transactions on Broadcasting 57, 477–490 (2011)
3. Muller, K., Merkle, P., Wiegand, T.: 3-D video representation using depth maps. Proceedings of the IEEE 99, 643–656 (2011)
4. Kim, J.H., Kolmogorov, V., Zabih, R.: Visual correspondence using energy minimization and mutual information. In: International Conference on Computer Vision, pp. 1–8 (2003)
5. Tanimoto, M., Fujii, T., Suzuki, K.: View synthesis algorithm in view synthesis reference software 2.0(VSRS 2.0). ISO/IEC JTC1/SC29/WG11 M16090, 1–8 (2008)
6. Tian, D., Lai, P., Lopez, P., Gomila, C.: View synthesis techniques for 3D video. Proceedings of SPIE 7443, 643–656 (2009)
7. Koppel, M., Ndjiki-Nya, P., Doshkov, D., Lakshman, H., Merkle, P., Wiegand, T.: Temporal Consistent Handling of Disocclusions with Texture Synthesis for Depth-Image-Based Rendering. In: Proceedings of International Conference on Image Processing, pp. 1809–1812 (2010)
8. Xu, Z., Sun, J.: Image inpainting by patch propagation using patch sparsity. IEEE Transactions on Image Processing 19, 1153–1165 (2010)
9. Humenberger, M., Engelke, T., Kubinger, W.: A census-based stereo vision algorithm using modified semi-global matching and plane fitting to improve matching quality. In: CVPRW, pp. 77–84 (2010)

Hand-Eye Calibration without Hand Orientation Measurement Using Minimal Solution

Zuzana Kukelova[1], Jan Heller[1], and Tomas Pajdla[2]

[1] Center for Machine Perception, Department of Cybernetics
Faculty of Elec. Eng., Czech Technical University in Prague
166 27 Prague 6, Technicka 2, Czech Republic
[2] Neovision, s.r.o., Barrandova 409
143 00 Prague 4, Czech Republic

Abstract. In this paper we solve the problem of estimating the relative pose between a robot's gripper and a camera mounted rigidly on the gripper in situations where the rotation of the gripper w.r.t. the robot global coordinate system is not known. It is a variation of the so called hand-eye calibration problem. We formulate it as a problem of seven equations in seven unknowns and solve it using the Gröbner basis method for solving systems of polynomial equations. This enables us to calibrate from the minimal number of two relative movements and to provide the first exact algebraic solution to the problem. Further, we describe a method for selecting the geometrically correct solution among the algebraically correct ones computed by the solver. In contrast to the previous iterative methods, our solution works without any initial estimate and has no problems with error accumulation. Finally, by evaluating our algorithm on both synthetic and real scene data we demonstrate that it is fast, noise resistant, and numerically stable.

1 Introduction

The problem of estimating the relative position and orientation of a robot gripper and a camera mounted rigidly on the gripper, known as *hand-eye calibration problem*, has been studied extensively in the past [23,24,17,2,25,9]. This problem arises in wide range of applications not only in robotics but also in automotive or medical industries.

The standard formulation of this problem leads to solving a system of equations of the form

$$\mathtt{AX} = \mathtt{XB}, \tag{1}$$

where known $\mathtt{A}$ and $\mathtt{B}$ and unknown $\mathtt{X}$ are homogeneous transformation matrices of the form

$$\begin{bmatrix} \mathtt{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}, \tag{2}$$

with 3×3 rotation matrix $\mathtt{R} \in SO(3)$ and 3×1 translation vector $\mathbf{t} \in \mathbb{R}^3$. It has been shown in [24] that at least two motions with non-parallel rotation axes are

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 576–589, 2013.

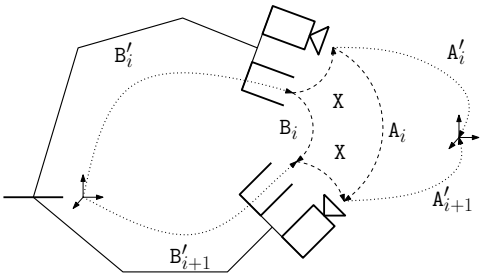


Fig. 1. The relative movement of the camera-gripper rig

required to solve hand-eye calibration problem. In practice, several motion are performed and the overconstrained system

$$\mathtt{A}_i\mathtt{X} = \mathtt{X}\mathtt{B}_i,\ i = 1, \ldots, n \tag{3}$$

is solved as a minimization problem, with every method trying to minimize a different error criteria. The existing methods can be divided into three groups.

The first group of methods [23,24,17,2,15] solves System 3 by decomposing it into matrix equations depending only on rotations and vector equations depending both on rotations and translations. In this way the methods decouple the rotation from the translation and solve for them separately, *i.e.*, first for the rotation and then for the translation. The drawback of such an approach is that the rotation estimation errors propagate to the translation errors. To address the problem of error propagation several methods for simultaneous estimation of rotation and translation appeared [25,9,5,21]. These methods search for the the unknown transformation $\mathtt{X}$ by solving the overconstrained System 3 using different linear or non-linear minimization methods. The methods mostly differ in the error function which is minimized and in the used minimization method. Methods that use iterative optimization techniques suffer from the inherent problems of iterative algorithms, *i.e.*, problems with convergence and the necessity of a good initial estimate of $\mathtt{X}$. In [19] authors proposed a method that uses tracked image points rather than matrices $\mathtt{A}_i$. Recently, another group of methods appeared [8,7,16]. These methods use image correspondences instead of matrices $\mathtt{A}_i$ and employ global optimization to minimize different error functions in L_∞-norm.

In this paper we are concerned with a variation of hand-eye calibration problem that has been scarcely addressed in the literature so far—hand-eye calibration with unknown hand rotation. This problem arises when the robot is not calibrated or the information from the robot is not available. In these situation one has to measure the robot's pose by an external measurement device. In many cases such a measurement device is not able to measure the whole pose, but only the translational part of it, since translation is much easier to measure than rotation. Without the hand rotation measurements none of the previously discussed methods can be used. A method presented in [25] addresses this

problem by nonlinear optimization and estimates simultaneously both rotational and translational parts. However, it requires a good initial estimate of $\mathtt{X}$.

In case of two relative motions, we solve this problem by formulating it as a system of seven equations in seven unknowns and solving it using the Gröbner basis method for solving systems of polynomial equations. This provides an exact algebraic solution and has none of the problems of the former numerical minimization methods, *i.e.*, problems with convergence or the necessity of having a good initial estimate. In case of three of more motions, we use a residual function to select an initial solution among the candidates provided by the Gröbner basis method to initialize the method of [25]. By evaluating our solution on both synthetic and real scene data, we demonstrate that it is efficient, fast, and numerically stable. Further, we show that in case of more than two motions it provides a good estimate for nonlinear optimization.

2 Problem Formulation

First, let us consider the classical hand-eye calibration problem. The goal is to estimate the relative pose, *i.e.*, the rotation and the translation of the camera w.r.t. the gripper, see Figure 1. We will describe this transformation by the homogeneous transformation matrix

$$\mathtt{X} = \begin{bmatrix} \mathtt{R}_\mathtt{X} & \mathbf{t}_\mathtt{X} \\ \mathbf{0}^\top & 1 \end{bmatrix}, \tag{4}$$

where $\mathtt{R}_\mathtt{X} \in SO(3)$ is the unknown rotation from the camera to the gripper and $\mathbf{t}_\mathtt{X} \in \mathbb{R}^3$ the unknown translation.

Let us consider the i^{th} pose of the robot and denote the transformation matrix from the camera to the world coordinate system by $\mathtt{A}'_i$ and the transformation matrix from another coordinate system in the world—usually placed in the robot's base—to the robot's gripper by $\mathtt{B}'_i$, see Figure 1. Camera's transformations $\mathtt{A}'_i$ can be obtained using the well known absolute pose solvers [6,14] and transformations $\mathtt{B}'_i$ from the robot's positioning software.

Figure 1 shows that by knowing two poses of the robot we can get $\mathtt{X}$ from the following equation

$$\mathtt{A}_i\mathtt{X} = \mathtt{X}\mathtt{B}_i, \tag{5}$$

where $\mathtt{A}_i = \mathtt{A}'^{-1}_i \mathtt{A}'_{i+1}$ and $\mathtt{B}_i = \mathtt{B}'_{i+1}\mathtt{B}'^{-1}_i$ are homogeneous transformation matrices representing the respective relative movements. Equation 5 can be decomposed into a matrix and a vector equation

$$\mathtt{R}_{\mathtt{A}_i}\mathtt{R}_\mathtt{X} = \mathtt{R}_\mathtt{X}\mathtt{R}_{\mathtt{B}_i}, \tag{6}$$

$$\mathtt{R}_{\mathtt{A}_i}\mathbf{t}_\mathtt{X} + \mathbf{t}_\mathtt{A} = \mathtt{R}_\mathtt{X}\mathbf{t}_{\mathtt{B}_i} + \mathbf{t}_\mathtt{X}. \tag{7}$$

At least two motions with non-parallel rotation axes are required to solve this system of equations. With two or more motions known, we obtain an overconstrained system of polynomial equations.

In situations where one does not have the information from the robot's positioning software or the robot is not precisely calibrated transformations $\mathtt{B}'_i$ are not readily known. To recover them, one has to use some external measurement equipment. In this paper we are interested in situations where such a measurement device does not allow to recover the whole pose of the robotic gripper, but only its translational part.

Typically, the external measurement devices are able to recover absolute gripper's positions $\mathbf{t}'_\mathtt{B}$ w.r.t. robot's base. However, in Equation 7 relative translations $\mathbf{t}_\mathtt{B}$ appear. In order to compute the relative translations $\mathbf{t}_\mathtt{B}$ there has to be at least one position where the full pose of the robot can be recovered, *i.e.*, where the rotation $\mathtt{R}'_\mathtt{B}$ is known as well. Even for an uncalibrated robot, the robot's home position can be used as such *a priori* known pose. By constructing the relative movements in such a way as to always end in a position with a known rotation $\mathtt{R}'_\mathtt{B}$, relative translations $\mathbf{t}_\mathtt{B}$ can be recovered. Since the positions with *a priori* known poses are usually hard to come by, it is advantageous for a method to be able to calibrate from a minimal number of movements possible.

3 Minimal Problem

First, let us suppose that we can measure two gripper's relative translations $\mathbf{t}_{\mathtt{B}_i}$ and $\mathbf{t}_{\mathtt{B}_j}$ and two respective relative camera motions $\mathtt{A}_i$ and $\mathtt{A}_j$. Now, let us note that the vector Equation 7 does not contain the unknown gripper's rotations $\mathtt{R}_{\mathtt{B}_i}$. By parametrizing the rotation $\mathtt{R}_\mathtt{X}$ by the unit quaternion $q = a + bi + cj + dk$ as

$$\mathtt{R}_\mathtt{X} \equiv \mathtt{R}^q_\mathtt{X} = \begin{bmatrix} a^2+b^2-c^2-d^2 & 2bc-2ad & 2ac+2bd \\ 2ad+2bc & a^2-b^2+c^2-d^2 & 2cd-2ab \\ 2bd-2ac & 2ab+2cd & a^2-b^2-c^2+d^2 \end{bmatrix} \tag{8}$$

and substituting it into the vector Equation 7 we get three polynomial equations in seven unknowns, *i.e.*, three translation parameters for $\mathbf{t}_\mathtt{X}$ and four rotation parameters a, b, c, and d. Now we can apply this substitution to the two motions i and j and by adding the equation defining the unit quaternion q we obtain the following system of equations:

Problem 1 (Minimal Hand-Eye Calibration)

$$\begin{aligned} \textit{Given} \quad & \mathtt{R}_{\mathtt{A}_i}, \mathtt{R}_{\mathtt{A}_j}, \mathbf{t}_{\mathtt{A}_i}, \mathbf{t}_{\mathtt{A}_j}, \mathbf{t}_{\mathtt{B}_i}, \mathbf{t}_{\mathtt{B}_j} \\ \textit{find} \quad & \mathtt{R}_\mathtt{X} \in SO(3), \mathbf{t}_\mathtt{X} \in \mathbb{R}^3 \\ \textit{subject to} \quad & \mathtt{R}_{\mathtt{A}_i}\mathbf{t}_\mathtt{X} + \mathbf{t}_{\mathtt{A}_i} = \mathtt{R}^q_\mathtt{X}\mathbf{t}_{\mathtt{B}_i} + \mathbf{t}_\mathtt{X}, \\ & \mathtt{R}_{\mathtt{A}_j}\mathbf{t}_\mathtt{X} + \mathbf{t}_{\mathtt{A}_j} = \mathtt{R}^q_\mathtt{X}\mathbf{t}_{\mathtt{B}_j} + \mathbf{t}_\mathtt{X}, \\ & a^2+b^2+c^2+d^2 = 1. \end{aligned}$$

Problem 1 is a well-constrained system of seven equations in seven unknowns. To solve it for the unknown hand-eye calibration $\mathtt{X}$, the Gröbner basis method can be readily used. This leads to a fast and non-iterative solution with no need for an initial solution estimate. Note that the minimal number of two relative movements without rotations $\mathtt{R}_{\mathtt{B}_i}$ and $\mathtt{R}_{\mathtt{B}_j}$ is needed to construct the system.

In case rotations $\mathtt{R}_{\mathtt{B}_i}$ and $\mathtt{R}_{\mathtt{B}_j}$ need to be recovered as well, by substituting the solutions for the rotation $\mathtt{R}_{\mathtt{X}}$ into the Equation 6 we get the rotations as

$$\mathtt{R}_{\mathtt{B}_i} = \mathtt{R}_{\mathtt{X}}^{-1}\mathtt{R}_{\mathtt{A}_i}\mathtt{R}_{\mathtt{X}}, \tag{9}$$

$$\mathtt{R}_{\mathtt{B}_j} = \mathtt{R}_{\mathtt{X}}^{-1}\mathtt{R}_{\mathtt{A}_j}\mathtt{R}_{\mathtt{X}}. \tag{10}$$

3.1 Gröbner Basis Method

The Gröbner basis method for solving systems of polynomial equations has recently became popular in computer vision and it has been used to create very fast, efficient and numerically stable solvers to many difficult problems. The method is based on polynomial ideal theory and is concerned with special bases of these ideals called Gröbner bases [3]. Gröbner bases have the same solutions as the initial system of polynomial equations defining the ideal but are often easier to solve. Gröbner bases are usually used to construct special multiplication (action) matrices [18], which can be viewed as a generalization of the companion matrix used in solving one polynomial equation in one unknown. The solutions to the system of polynomial equations is then obtained from the eigenvalues and eigenvectors of such action matrices. See [3,4] for more on Gröbner basis methods and [20,10,1] for their applications in computer vision.

Since general algorithms [3] for computing Gröbner basis are not very efficient for solving problems which appear for example in computer vision, an automatic generator of specific polynomial equations solvers based on the Gröbner basis method has been proposed in [11]. These specific solvers often provide very efficient solutions to a class of systems of polynomial equations consisting of the same monomials and differing only in the coefficients.

Computer vision problems—like the hand-eye calibration problem presented in this paper—share the convenient property that the monomials appearing in the set of initial polynomials are always the same irrespective of the concrete coefficients arising from non-degenerate measurements. Therefore it is possible to use efficient specific solvers instead of less efficient general algorithms [3] for constructing the Gröbner bases.

The process of creating the specific solvers consists of two phases. In the first "offline" phase, the so-called "elimination templates" are found. These templates decide the elimination sequence in order to obtain all polynomials from the Gröbner basis or at least all polynomials necessary for the construction of the action matrix. This phase is performed only once for a given problem. In the second "online" phase, the elimination templates are used with coefficients arising from the specific measurements to construct the action matrix. Then, eigenvalues and eigenvectors of the action matrix provide solutions to the original polynomial equations. The automatic generator presented in [11] performs

the offline phase automatically and for an input system of polynomial equations outputs an efficient online solver.

3.2 Gröbner Basis Solver

To create an efficient solver for Problem 1 we used the automatic generator proposed in [11]. The Gröbner basis solver of the proposed hand-eye calibration problem starts with seven equations in seven unknowns, *i.e.*, three translation parameters for $\mathbf{t}_\mathtt{X}$ and four rotation parameters a, b, c, and d.

From the generator we obtained an elimination template which encodes how to multiply the seven input polynomials by the monomials and then how to eliminate the polynomias using the Gauss-Jordan (G-J) elimination process to obtain all polynomials necessary for the construction of the action matrix. In our case the automatic generator created the action matrix $\mathtt{M}_a$ for multiplication by a.

To get the elimination template the generator first generated all monomial multiples of the initial seven polynomial equations up to the total degree of four. This resulted in 252 polynomials in 330 monomials. Then the generator removed all unnecessary polynomials and monomials, *i.e.*, polynomials and monomials that do not influence the resulting action matrix. This resulted in matrix a 182×203 $\mathtt{Q}$ representing the polynomials for the construction of the action matrix $\mathtt{M}_a$, *i.e.*, the elimination template.

The online solver then only performs one G-J elimination of matrix $\mathtt{Q}$ from the elimination template identified in the offline stage. This matrix contains coefficients which arise from specific measurements, *i.e.*, rotations $\mathtt{R}_{\mathtt{A}_i}$ and $\mathtt{R}_{\mathtt{A}_j}$ and translations $\mathbf{t}_{\mathtt{A}_i}$,$\mathbf{t}_{\mathtt{A}_j}$,$\mathbf{t}_{\mathtt{B}_i}$, and $\mathbf{t}_{\mathtt{B}_j}$. After G-J elimination of matrix $\mathtt{Q}$, action matrix $\mathtt{M}_a$ can be created from its rows. The solutions to all seven unknowns can be found from the eigenvectors of the action matrix $\mathtt{M}_a$. The online stage takes about 1 ms to finish in case of Problem 1.

This gives us a set $\mathcal{X}_{ij}$ of up to 16 real solutions of $\mathtt{X}$. However each of these solutions appears twice, *i.e.*, there are double roots. Therefore we have only up to 8 different real solutions. Usually only one to four of them are *geometrically feasible*, *i.e.*, are real and of a reasonable length of the translation. The correct one can be chosen from the feasible solutions manually using some prior knowledge about the transformation $\mathtt{X}$ or automatically using an additional set of solutions for different relative movements. The next section describes an automatic procedure for selecting the correct transformation.

4 Automatic Solution Selection

In order to automatically select the geometrically correct solution among the algebraically correct ones in $\mathcal{X}_{ij}$, at least one more set of solutions to Problem 1 for a different combination of relative movements is needed. Let $\mathcal{X}_{k\ell}$ be such a set for two additional movements k and ℓ. Supposing that the movements i, j and k, ℓ form a geometrically non-degenerate configuration, we will find the

geometrically correct solution as $\mathcal{X}_{ij} \cap \mathcal{X}_{k\ell}$. In the presence of noise however, the intersection $\mathcal{X}_{ij} \cap \mathcal{X}_{k\ell}$ will most likely be an empty set. In this case we have to select a solution from the union $\mathcal{X}_{ij} \cup \mathcal{X}_{k\ell}$ that best fits the equations of Problem 1 for different motions. We will measure the fitness of a solution $\mathtt{X}$ by the residual error of Equation 7

$$\mathbf{e}_i(\mathtt{X}) = \mathtt{R}_{\mathtt{A}_i}\mathbf{t}_{\mathtt{X}} + \mathbf{t}_{\mathtt{A}_i} - \mathtt{R}_{\mathtt{X}}\mathbf{t}_{\mathtt{B}_i} - \mathbf{t}_{\mathtt{X}}. \tag{11}$$

Now let us formalize the idea of selecting the best solution and to extend it to the case of more that two solution sets. Let n be the number of available relative movements and let I be a set of pairs of indexes of the relative movements

$$I \subset \{\{i,j\} \colon i,j \leq n\}, \quad |I| \geq 2. \tag{12}$$

Let $\mathcal{X}$ be a set of solutions to Problem 1 for the pairs from the index set I,

$$\mathcal{X} = \bigcup_{\{i,j\} \in I} \mathcal{X}_{ij}. \tag{13}$$

We select the geometrically correct solution among the solutions in $\mathcal{X}$ by solving the following problem:

Problem 2 (Minimal Hand-Eye Calibration for n Movements)

$$\begin{aligned} \textit{Given} \quad & \mathtt{R}_{\mathtt{A}_i}, \mathbf{t}_{\mathtt{A}_i}, \mathbf{t}_{\mathtt{B}_i}, I, i = 1, \ldots, n \\ \textit{and} \quad & \textit{a set of solutions } \mathcal{X} = \textstyle\bigcup_{\{i,j\} \in I} \mathcal{X}_{ij} \\ \textit{find} \quad & \mathtt{X}^\star = \arg\min_{\mathtt{X} \in \mathcal{X}} \sum_{i=1}^{n} \mathbf{e}_i(\mathtt{X})^\top \mathbf{e}_i(\mathtt{X}) \end{aligned}$$

As we can see from the above formulation, solving Problem 2 amounts to selecting a minimum from a set of $|\mathcal{X}|$ real numbers.

In the presence of noise and in case $n > 2$, we can further refine the solution by an optimization method. For our experiments, we chose the method of Zhuang and Shiu [25] which requires a good initial estimate $\mathtt{X}^0$. By setting $\mathtt{X}^0 \equiv \mathtt{X}^\star$, we can refine the solution by solving the following minimization problem:

Problem 3 (Zhuang [25])

$$\begin{aligned} \textit{Given} \quad & \mathtt{R}_{\mathtt{A}_i}, \mathbf{t}_{\mathtt{A}_i}, \mathbf{t}_{\mathtt{B}_i}, i = 1, \ldots, n \\ \textit{and} \quad & \textit{an initial solution estimate } \mathtt{X}^0 \\ \textit{find} \quad & \mathtt{X}^\star_{\text{opt}} = \arg\min \sum_{i=1}^{n} \mathbf{e}_i(\mathtt{X})^\top \mathbf{e}_i(\mathtt{X}) \\ \textit{subject to} \quad & \mathtt{R}_{\mathtt{X}} \in SO(3) \end{aligned}$$

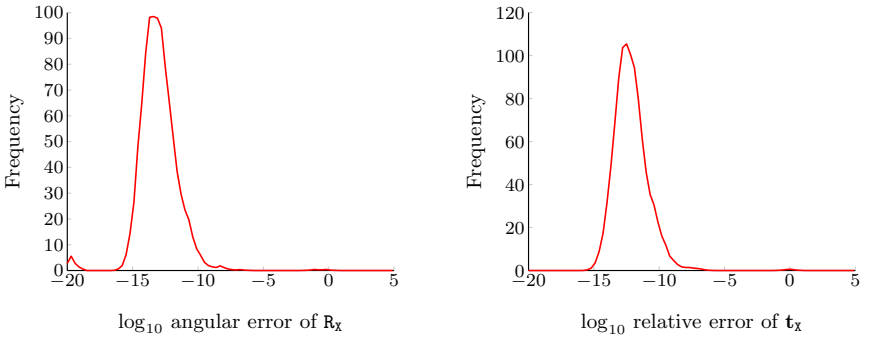


Fig. 2. $\log_{10}$ angular error of the estimated rotation $\mathtt{R}_\mathtt{X}$ (Left) and $\log_{10}$ translation error of $\mathbf{t}_\mathtt{X}$ (Right) for noise free data

5 Experiments

To experimentally validate the proposed solutions, we use both synthetically generated and real word calibration scenarios. First, we use synthetically generated ground truth scenes to study the numerical stability of the proposed solution to Problem 1. Next, we study the behavior of the solutions to Problem 2 and Problem 3 on synthetic scenes consisting of 4 non-degenerate poses. Finally, we show the viability of the minimal solution in a real life experiment with a Mitsubishi MELFA-RV-6S serial manipulator with four draw-wire encoders attached to its end effector to recover the translations $\mathbf{t}_{\mathtt{B}_i}$.

In all of the experiments we scaled the lengths of the input translation vectors $\mathbf{t}_{\mathtt{B}_i}$ and $\mathbf{t}_{\mathtt{A}_i}$ by the length of the largest one of them prior to running the Gröbner basis solver. We observe that this scaling improves the numerical stability of the solution.

The experiments were run on a 3GHz Intel Core i7 based desktop computer running 64-bit Linux. The Matlab implementation of the proposed method used in the experiments is available at `http://cmp.felk.cvut.cz/minimal/handeye.php`.

5.1 Experiments with Synthetic Data

Numerical Stability Experiment. First, we studied the behavior of the proposed Gröbner basis solver of Problem 1 to check its numerical stability. We generated 1000 random scenes with 100 points $\mathbf{P}^k$, $k = 1, \ldots, 100$, evenly distributed in the unit ball. Each scene consisted of 3 random absolute camera poses $\mathtt{A}'_i$. The cameras were positioned to (i) be facing the center of the scene, (ii) see the scene points from the field of view (FOV) ranging from 40° to 80°. For every scene ground truth transformation $\mathtt{X}_{\mathrm{gt}}$ was generated so that the angle and the axis of $\mathtt{R}_{\mathtt{X}_{\mathrm{gt}}}$ were random and uniformly distributed and that $\left\|\mathbf{t}_{\mathtt{X}_{\mathrm{gt}}}\right\| \approx 0.1$.

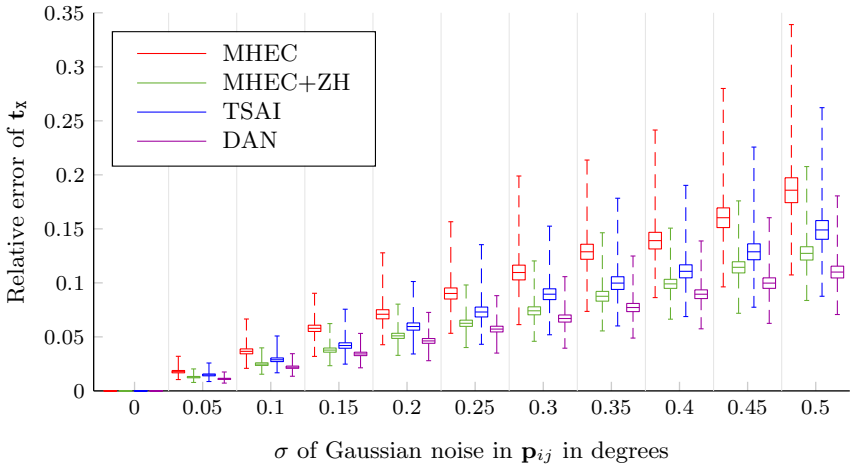


Fig. 3. Relative error of recovered translation $\mathbf{t}_\mathtt{X}$ for different levels of Gaussian noise

Absolute robot poses $\mathtt{B}'_i$ were determined by chaining $\mathtt{X}_{\text{gt}}^{-1}$ and the generated absolute camera positions. For every combination of ground truth $\mathtt{R}_{\mathtt{X}_{\text{gt}}}$, $\mathbf{t}_{\mathtt{X}_{\text{gt}}}$ and the recovered $\mathtt{R}_\mathtt{X}$, $\mathbf{t}_\mathtt{X}$ we measured the error of the rotation as the angle θ of the rotation $\mathtt{R}_\mathtt{X}^\top \mathtt{R}_{\mathtt{X}_{\text{gt}}}$, such that $0 \leq \theta \leq \pi$ and the error of translation as the relative error $\left\|\mathbf{t}_\mathtt{X} - \mathbf{t}_{\mathtt{X}_{\text{gt}}}\right\| / \left\|\mathbf{t}_{\mathtt{X}_{\text{gt}}}\right\|$. Figure 2 shows the histograms of the respective errors, certifying the numerical stability of the solver.

Calibration Experiment. In this experiment we analyzed the performance with respect to image noise. We used the same scheme to generate random scenes as in Numerical Stability Experiment. This time, we generated four absolute robot poses in each scene and recovered the absolute camera positions by P3P algorithm [14].

We started by computing $\mathbf{P}_i^k$—the positions of the 100 random points $\mathbf{P}^k$ with respect to the coordinate systems of the cameras $\mathtt{A}'_i$, $i = 1, \ldots, 4$. Further, we normalized $\mathbf{P}_i^k$ to get only the directional vectors $\mathbf{p}_i^k$ that were progressively corrupted with angular Gaussian noise. Finally, we used P3P in RANSAC loop to obtain noise corrupted absolute camera poses $\mathtt{A}'_i$, $i = 1, \ldots 4$.

We experimented with 11 levels of angular Gaussian noise with the standard deviation σ ranging from 0 to 0.5 degrees, with the highest noise level translating to σ of ca. 20–40 pixels for a 8MP camera with 40°–80° field of view. We generated and recovered camera poses for 1000 random scenes for every noise level.

We recovered hand-eye calibrations $\mathtt{X}$ by four different methods. The first method MHEC identifies the results obtained by the Gröbner basis solver with the solution selected according to Problem 2. The second method MHEC+ZH stands for the results obtained by the method [25] (Problem 3) when initialized

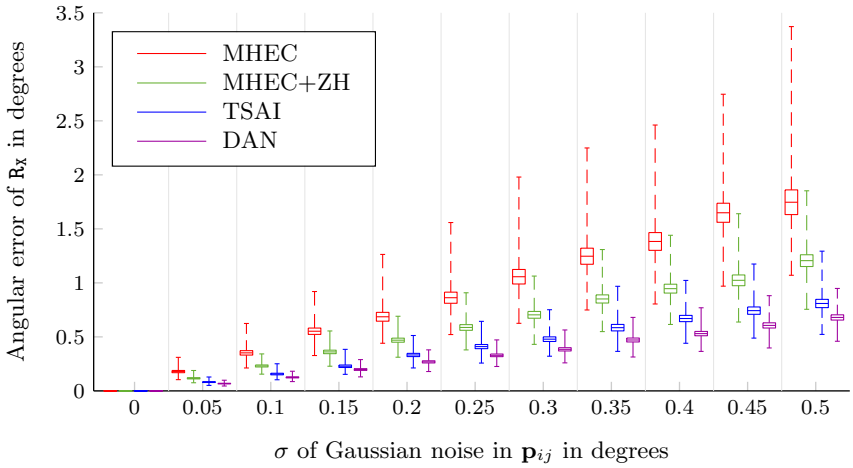


Fig. 4. Angular error of recovered rotation $\mathtt{R}_\mathtt{X}$ for different levels of Gaussian noise

by the results of MHEC. For completeness sake, we include results obtained by the methods [24] labeled as TSAI and [5] labeled in the figures as DAN. These methods are not the direct competitors, since they require known robot rotations $\mathtt{R}_\mathtt{B}$. However, they can be used to gauge the accuracy of the results obtained by MHEC and MHEC+ZH.

Figures 3, 4, and 5 show the statistics of the obtained solutions using the Matlab `boxplot` function depicting values 25% to 75% quantile as a box with horizontal line at median. Figures 3 and 4 show the respective errors of $\mathbf{t}_\mathtt{X}$ and $\mathtt{R}_\mathtt{X}$ using the same measures as described in Numerical Stability Experiment. Figure 5 shows the mean distance between the points $\mathbf{P}_i^k$ transformed into the coordinate system of the gripper using the ground truth hand-eye transformation and the same points transformed into the coordinate system of the gripper using the estimated $\mathtt{X}$. Note that the points were generated into the unit ball, *i.e.*, considering the diameter of this ball to be one meter means that the errors in Figure 5 are in meters.

5.2 Real Scene Data Experiment

In order to acquire a real scene calibration data, four draw-wire encoders were connected to the gripper of a Mitsubishi MELFA-RV-6S serial manipulator. A Canon 350D digital SLR camera with a Sigma 8 mm lens (cca. 130° field of view) was also attached to the gripper to form a hand-eye system.

The robot was instructed to move the gripper to (i) the home position with the known rotation w.r.t. the robot base, (ii) the four positions (backward, forward, left, right) distant approximately 400 mm at 10 degree pitch, (iii) the same four positions at 20 degree pitch, (iv) the position approximately 250 mm under the home position, and (v) the four positions at this height at 10 and 20 degree

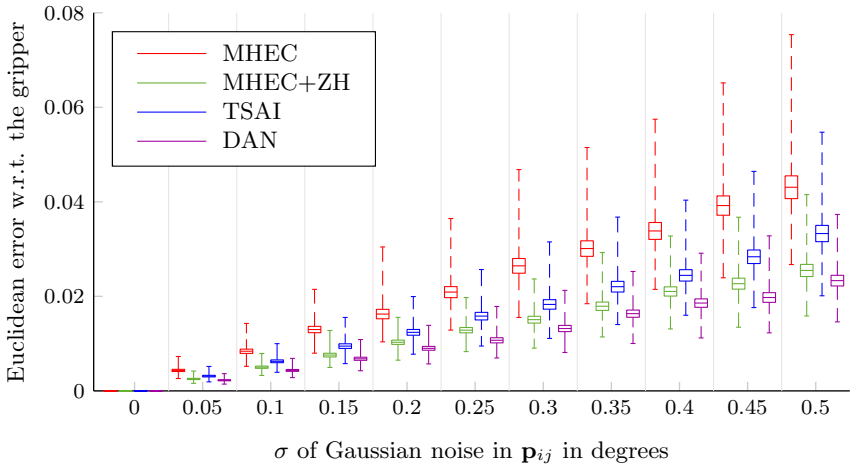


Fig. 5. Euclidean error of recovered calibration $\mathtt{X}$ for different levels of Gaussian noise

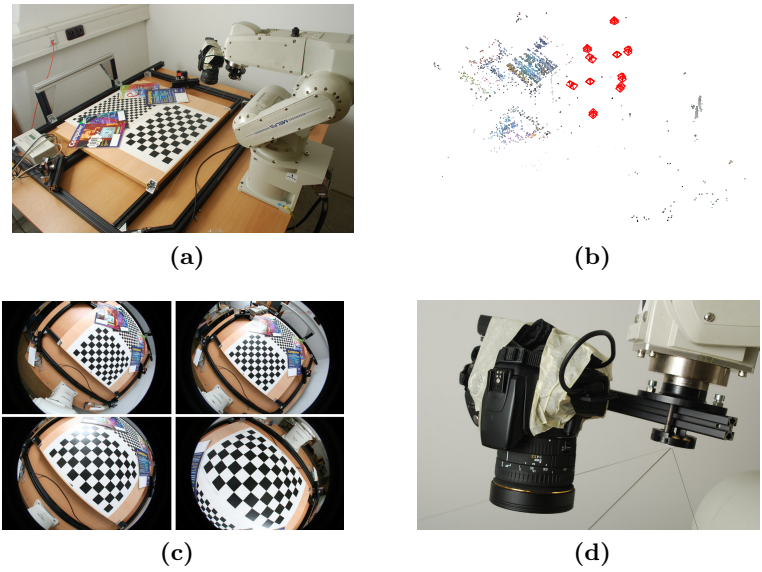
pitch again. While the robot was moving, the camera was remotely triggered to acquire 2,592×1,728 pixels large images of a circular view field with 1,040 pixels radius.

The internal calibration of the camera in the form of a 2-parameter equiangular model [12] was obtained using an image of a checkerboard with manually labeled corners. Then, a state-of-the-art sequential structure-from-motion pipeline [22] was used to automatically generate MSER, SIFT, and SURF feature points, perform approximate nearest neighbor matching in the descriptor space, verify the matches by pairwise epipolar geometries estimated by the 5-point algorithm [13] in a RANSAC loop, and create tracks and triangulated 3D points from verified matches spanning several images. The reconstructed 3D model was scaled to millimeter units by knowing the real dimensions of the checkerboard and measuring the distance of the corresponding 3D points in the model.

We used the system of four draw-wire encoders to determine the absolute positions of the gripper w.r.t. the robot base. For the experiment we chose 2 motions ending in the robots home position. Since the rotation of the robot in the home position is known, it is possible to transform the positions provided by the draw-wire encoders into the home position coordinate system and obtain translations $\mathbf{t}_{\mathtt{B}_1}$ and $\mathbf{t}_{\mathtt{B}_2}$. We used $\mathbf{t}_{\mathtt{B}_1}$ and $\mathbf{t}_{\mathtt{B}_2}$ in combination with $\mathtt{A}_1$, $\mathtt{A}_2$ obtained from structure-from-motion to compute the hand-eye transformation $\mathtt{X}$ and the relative gripper rotations $\mathtt{R}_{\mathtt{B}_1}$ and $\mathtt{R}_{\mathtt{B}_2}$.

For comparison, we also used $\mathbf{t}_{\mathtt{B}_{\mathrm{gt2}}}$ and $\mathbf{t}_{\mathtt{B}_{\mathrm{gt2}}}$ from robots positioning software with the same camera motions $\mathtt{A}_1$ and $\mathtt{A}_2$ to compute hand-eye transformation $\bar{\mathtt{X}}$, $\bar{\mathtt{R}}_{\mathtt{B}_1}$, and $\bar{\mathtt{R}}_{\mathtt{B}_2}$.

Since the robot was calibrated, we can also compare the computed gripper rotations $\mathtt{R}_{\mathtt{B}_1}$, $\mathtt{R}_{\mathtt{B}_2}$, $\bar{\mathtt{R}}_{\mathtt{B}_1}$, and $\bar{\mathtt{R}}_{\mathtt{B}_2}$ with the rotations $\mathtt{R}_{\mathtt{B}_{\mathrm{gt1}}}$ and $\mathtt{R}_{\mathtt{B}_{\mathrm{gt2}}}$ from the robots positioning software, see Table 1.



(a) **(b)** **(c)** **(d)**

Fig. 6. Real data experiment. (a) A Mitsubishi MELFA-RV 6S serial manupulator used to acquire the data for the experiment. (b) The 3D model obtained from SfM. (c) Sample images of our scene taken by the camera mounted on the gripper of the robot. (d) Close up of the camera-gripper rig with draw-wire encoders.

Table 1. Angular rotation errors of estimated gripper rotations in degrees

	$\mathbf{R}_{\mathbf{B}_1}$	$\mathbf{R}_{\mathbf{B}_2}$	$\bar{\mathbf{R}}_{\mathbf{B}_1}$	$\bar{\mathbf{R}}_{\mathbf{B}_2}$
$\mathbf{R}_{\mathbf{B}_{gt1}}$	0.84	—	0.89	—
$\mathbf{R}_{\mathbf{B}_{gt2}}$	—	0.61	—	1.09

Finally, let us express the obtained translations from the gripper to the camera center using the translation from the draw-wire encoders $-\mathbf{R}_{\mathbf{X}}^{\top}\mathbf{t}_{\mathbf{X}} = (110.2, 26.2, 47.9)$, and using the translation from the robot, $-\mathbf{R}_{\bar{\mathbf{X}}}^{\top}\mathbf{t}_{\bar{\mathbf{X}}} = (126.5, 28.7, 51.1)$.

These result are consistent with each other as well as with the rough physical measurement of the mechanical reduction and show the validity of the obtained results.

6 Conclusion

We presented the first minimal problem of hand-eye calibration for the situations where the gripper's rotations are not known. We formulated the problem as a system of seven equations in seven unknowns and solved it using the Gröbner

basis method for solving systems of polynomial equations providing the first exact algebraic solution to the problem. This solution uses the minimal number of two relative movements. Further, we showed how to select the geometrically correct solution using additional relative movements. Finally, our experiments showed that the proposed solver is numerically stable, fast and—since it can handle noisy inputs—that its results can be successfully used as initialization of subsequent minimization methods.

Acknowledgment. The authors were supported by the EC under projects FP7-SME-2011-285839 De-Montes and FP7-288553 CloPeMa and by Grant Agency of the CTU Prague project SGS12/191/OHK3/3T/13. The authors would also like to thank Michal Havlena and Martin Meloun for their help with the real-data experiment.

References

1. Bujnak, M., Kukelova, Z., Pajdla, T.: A general solution to the P4P problem for camera with unknown focal length. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
2. Chou, J.C.K., Kamel, M.: Finding the position and orientation of a sensor on a robot manipulator using quaternions. International Journal of Robotics Research 10(3), 240–254 (1991)
3. Cox, D.A., Little, J.B., O'Shea, D.: Using Algebraic Geometry. Graduate Texts in Mathematics. Springer (2005)
4. Cox, D.A., Little, J.B., O'Shea, D.: Ideals, Varieties, And Algorithms: An Introduction to Computational Algebraic Geometry And Commutative Algebra, Number v. 10. Undergraduate Texts in Mathematics. Springer (2007)
5. Daniilidis, K.: Hand-eye calibration using dual quaternions. International Journal of Robotics Research 18, 286–298 (1998)
6. Fischler, M., Bolles, R.: 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. Heller, J., Havlena, M., Pajdla, T.: A branch-and-bound algorithm for globally optimal hand-eye calibration. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
8. Heller, J., Havlena, M., Sugimoto, A., Pajdla, T.: Structure-from-motion based hand-eye calibration using l_∞ minimization. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3497–3503 (2011)
9. Horaud, R., Dornaika, F.: Hand-eye calibration. The International Journal of Robotics Research 14(3), 195–210 (1995)
10. Kukelova, Z., Pajdla, T.: A minimal solution to the autocalibration of radial distortion. In: IEEE Conference on Computer Vision and Pattern Recognition (2007)
11. Kukelova, Z., Bujnak, M., Pajdla, T.: Automatic Generator of Minimal Problem Solvers. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 302–315. Springer, Heidelberg (2008)
12. Micusik, B., Pajdla, T.: Structure from motion with wide circular field of view cameras. IEEE Transactions on Pattern Analysis and Machine Intelligence (2006)

13. Nistér, D.: An efficient solution to the five-point relative pose problem. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(6), 756–770 (2004)
14. Nistér, D., Stewénius, H.: A minimal solution to the generalised 3-point pose problem. Journal of Mathematical Imaging and Vision 27(1), 67–79 (2007)
15. Park, F.C., Martin, B.J.: Robot sensor calibration: solving AX=XB on the euclidean group. IEEE Transactions on Robotics and Automation 10(5), 717–721 (1994)
16. Ruland, T., Pajdla, T., Kruger, L.: Globally optimal hand-eye calibration. In: IEEE Conference on Computer Vision and Pattern Recognition (2012)
17. Shiu, Y.C., Ahmad, S.: Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form AX=XB. IEEE Transactions on Robotics and Automation 5(1), 16–29 (1989)
18. Hans, J.: Stetter. Numerical polynomial algebra. SIAM (2004)
19. Stewenius, H., Aström, K.: Hand-eye calibration using multilinear constraints. In: Proceedings of the Asian Conference on Computer Vision (2004)
20. Stewenius, H., Schaffalitzky, F., Nister, D.: How hard is 3-view triangulation really? In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 686–693 (2005)
21. Strobl, K.H., Hirzinger, G.: Optimal Hand-Eye Calibration. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, pp. 4647–4653 (2006)
22. Torii, A., Havlena, M., Pajdla, T.: Omnidirectional image stabilization for visual object recognition. International Journal of Computer Vision 91(2), 157–174 (2011)
23. Tsai, R.Y., Lenz, R.K.: Real time versatile robotics hand/eye calibration using 3D machine vision. In: IEEE International Conference on Robotics and Automation, vol. 1, pp. 554–561 (1988)
24. Tsai, R.Y., Lenz, R.K.: A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. IEEE Transactions on Robotics and Automation 5(3), 345–358 (1989)
25. Zhuang, H., Shiu, Y.C.: A noise tolerant algorithm for wrist-mounted robotic sensor calibration with or without sensor orientation measurement. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, pp. 1095–1100 (1992)

Detecting Changes in Images of Street Scenes

Jana Košecka

George Mason University, Fairfax, VA, USA

Abstract. In this paper we propose an novel algorithm for detecting changes in street scenes when the vehicle revisits sections of the street at different times. The proposed algorithm detects structural geometric changes, changes due to dynamically moving objects and as well as changes in the street appearance (e.g. posters put up) between two traversal times. We exploit geometric, appearance and semantic information to determine which areas have changed and formulate the problem as an optimal image labeling problem in the Markov Random Field framework. The approach is evaluated on street sequences from 3 different locations which were visited multiple times by the vehicle. The proposed method is applicable to monitoring and updating models and images of urban environments.

1 Introduction

Services like Google StreetView and GoogleEarth are becoming great resource for navigation and search of the constantly growing number of street locations. From the research standpoint these large image (video) datasets continue to pose novel computer vision challenges. In the context of this domain several techniques have been developed for vision based pose estimation, localization and loop closure detection using stereo, monocular or omnidirectional views. Development of robust solutions to these problems tackled the challenges related to the large scale of these datasets and as well as difficulty of lighting conditions due to often low resolution of images and uncontrolled image acquisition environments. The existing solutions exploited the advancements in structure and motion estimation techniques, dense multi-view 3D reconstruction and wide baseline matching and efficient indexing for large scale location recognition. Examples of these can be found in [1], [2], [3], [4], [5], [6], [7] and references therein.

With the success of these services maintenance of 3D city models and associated image panoramas is of importance. At the scale of the city many structural geometric changes (e.g. structures are raised and put down) and appearance changes (e.g. new posters are raised or facades of the buildings modified) happen over larger periods of time. Due to the scale of these datasets the development of automated methods for updating such models or monitoring and reporting the change is of importance. This work focuses on detecting changes in street scenes from images acquired by a moving vehicle. To quantify the amount of change at the level of images we formulate the change detection as optimal labeling problem in Markov Random Field framework, where regions of newly acquired images are labelled into two categories: changed or unchanged.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 590–601, 2013.

(a) (b) result and ground truth

Fig. 1. Left: (a) and (b) are images of a single location visited at different times; changes are present due to moved cars. Right: is the change detection results obtained by our method and the ground truth. We crop the boundaries for the visualization of the results to visualize only parts which are common in both views.

Contribution. The proposed algorithm for change detection in Street View™ images exploits geometric, appearance and semantic information to determine which areas in the image have changed. In the first stage of our approach we recover a coarse 3D geometry of the scene and register the novel views with the previously acquired reference images of the location. The coarse geometric registration is followed by an appearance transfer stage, where the image regions of a novel view are reprojected to the closest view captured at previous time and their appearance consistency is quantified. In the last stage we exploit semantic content of both previous and current views to gather additional evidence about the change hypotheses. These sources of evidence and integrated in the final energy minimization framework. Depending on whether the changes are structural (building went down), appearance (billboards) or just temporary presence of dynamically moving objects (pedestrians, cars) additional processing steps can be invoked to update 3D geometric models, or Street View images. The example results of the proposed approach can be found in Figure 1.

2 Related Work

The problem of mapping and maintaing models of environments is of fundamental importance for continuous operation in urban environments. Depending on the application domain various instances of this problem have been considered in the autonomous robot localization and mapping communities and surveillance communities. In the surveillance setting the change detection problem is often formulated as 2D-2D image comparison and typically assumes static cameras focusing on the problem of background subtraction [8]. Review of different approaches can be found in 2D images [9]. The methods based on purely 2D information have been found sensitive of changes in illumination and weather conditions. In the work of [10] authors proposed to learn a probabilistic appearance model for a 3D scene and formulated the change detection problem in 3D using voxel based representation of the world. The proposed per voxel appearance model was an extension of mixture of Gaussians estimated from reprojected

pixel intensities. In more recent work of [11] authors focus on geometric changes only. They assume the availability of an accurate 3D model of the scene and use the images and their reprojections to new views to generate hypotheses about consistency of the new images with the 3D model. The final inference was formulated in the Markov Random Field framework, where the graph was induced by a 3D voxel grid and the evidence about the voxel change was computed by counting inconsistently projected regions.

Earlier works in the robotics community considered issues of dynamically changing environments in the context of simultaneous localization and mapping problem. These methods typically rely purely on 3D geometry or 2D occupancy maps. [12] addressed the problem of localization in dynamic environments in an on-line manner using occupancy grid based representation, where both static and dynamic parts of the environment were represented in terms of separate occupancy grids. In the work of [13] the issue of dynamic changes have been tackled at the level of entire map using map differencing techniques and Expectation Maximization Algorithm; [14] proposed a method for on-line detection and identification of moving objects assuming ideal localization. The proposed work is the closest to [10,11] approaches to change detection. We also exploit information about 3D geometry and relative poses between the views, but formulate the final inference problem in 2D space of the new image instead of 3D voxel grid. In addition to geometric geometric changes, we consider capturing changes in environment appearance, such as posters or billboards put up or removed.

Instead of considering freely moving camera, we tackle the change detection problem using Street View image panoramas acquired by moving vehicle. The problem of change detection in this context is relevant for navigation and loop closing, where areas of the city are revisited by the vehicle. These omnidirectional views make the problem of image registration better conditioned despite their lower resolution, but also pose some challenges due to dramatic appearance variations and presence of large repetitive structures. The change detection algorithms are applied only to the side views of the panorama, oriented 90^o from heading direction of the vehicle.

Outline. In Section 3 we discuss the techniques for pose estimation used to register the views of a location acquired at different times. Section 4 describes our algorithm for change detection, detailing the geometric, appearance and semantic cues. We formulate the problem as optimal image labeling in Markov Random Field framework, followed by the results and conclusions in Section 5.

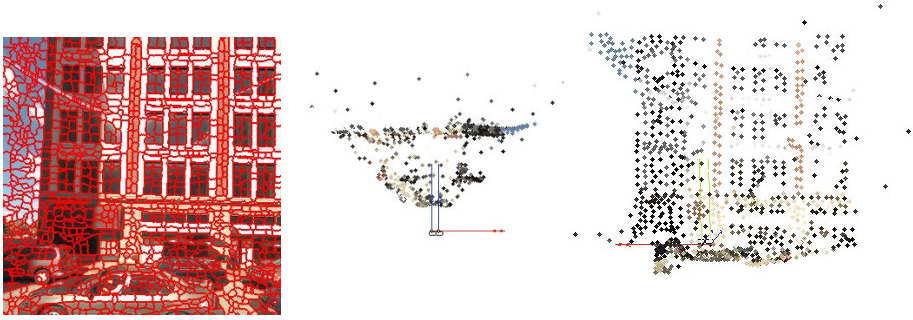
3 Preliminaries

The Street View images have been acquired by standard perspective cameras aligned in a circle. Our panorama is composed of four perspective images covering 360^o horizontally and 127^o vertically. We have multiple frames of each location available. Examples of images from 3 different locations at different times and changes we consider are in Figure 2.

Fig. 2. Images of example locations and the same locations revisited at different time of the day

Given a reference sequences of images $I_i^r, ..., I_j^r$ and a sequence acquired at later time $I_k^q, ..., I_l^q$, the first stage of our algorithm recovers the relative pose between the views in the reference sequence and recovered 3D structure. We employ standard visual odometry pipeline to recover relative poses and one single global scale of these views from the images. We use the wide baseline matching using SIFT features between each consecutive image pair along the sequence. The prismatic representation of the omnidirectional image allows us to construct corresponding 3D rays $\mathbf{p}, \mathbf{p}'$ for established tentative point matches $\mathbf{x}_t^q \leftrightarrow \mathbf{x}_{t+1}^q$. The tentative matches are validated through RANSAC-based epipolar geometry estimation formulated on their 3D rays, $\mathbf{p}'^{\top} \mathtt{E}\, \mathbf{p} = 0$, yielding thus the essential matrix $\mathtt{E}$ [15]. Improved convergence of RANSAC can be achieved if rays are sampled uniformly from each of four subparts of the panorama. It has been shown in the past that this yields more accurate estimates of pose [16] even in the absence of bundle adjustment. We denote the two consecutive novel views I_t^q and I_{t+1}^q and the nearest reference view I_k^r. We establish correspondences $\mathbf{x}^q \leftrightarrow \mathbf{x}^r$ between the novel view I_t^q and the closest reference view I_k^r and compute the pose from the essential matrix between the views. For solving the scales of translations between consecutive pairs of images and the reference view we set the norm of the translation for the first novel pair to be 1. Scale of the translation is estimated by a linear closed-form 1-point algorithm on corresponding 3D points triangulated from the query image pair and the reference view.

Given the registered set of novel views, we compute a coarse 3D structure of the scene. Instead of employing the full 3D dense reconstruction pipeline, we segment the image into small superpixels and establish correspondences between each centroid of the superpixel and it's consecutive view in the query sequence. Due to the fact that these frames are relatively close in time and the displacements are small, we used dense optical flow method [17] to establish the correspondences and using the median flow of pixels in the superpixel as displacement. 3D position of the superpixel centroid is then triangulated yielding a coarse 3D model. The quality of the model can be substantially improved using more advanced multi-view stereo reconstruction techniques. An example of 3D reconstruction at the superpixel level can be seen in Figure 3.



(a) (b) (c)

Fig. 3. (a) Image segmented into small watershed superpixels; (b) Bird's eye view of 3D reconstruction of elementary superpixels; (c) Side view of the same 3D structure

4 Change Detection

Previous section discussed the components of our system for the image alignment and a coarse 3D reconstruction. We propose to formulate the change detection problem as an optimal labeling problem in MRF framework, where we will seek an optimal label assignment 0 or 1 to each superpixel signifying whether the region changed (1) or remained the same (0). We seek to maximize the posterior probability of the labels $\mathbf{L} = \{0, 1\}$ given image observations. The label likelihoods and joint prior are expressed as unary and binary functions used in a second-order MRF framework. This maximization problem is equivalent to the energy minimization re-written in a log-space and has the following form

$$\underset{\mathbf{L}}{\operatorname{argmin}} \Big(\sum_{s_i \in S} \phi^U(s_i) + \lambda_s \sum_{(i,j) \in \mathcal{G}} \phi^P(s_i, s_j) \Big). \tag{1}$$

where the terms $\phi^U(s_i)$ are unary potentials quantifying the amount of change in a superpixel and $\phi^P(s_i, s_j)$ measure the pairwise consistency between the neighboring superpixels. The structure of MRF is induced by image superpixels s_i. These in our case are computed by watershed segmentation on Laplacian of Gaussian (LoG) interest points as seeds and can be seen in Figure 3(a). LoG interest points are selected as extrema of 4 level Laplacian of Gaussian pyramid described in more details [18]. This method of seed selection places interest points densely yielding small regions when followed by watershed segmentation. These elementary regions typically do not straddle boundaries between different classes and naturally contain semantically meaningful object or scene primitives. Furthermore, they dramatically reduce computational complexity of 3D reconstruction and an MRF inference. We describe the form of unary and binary potentials next.

4.1 Unary Term

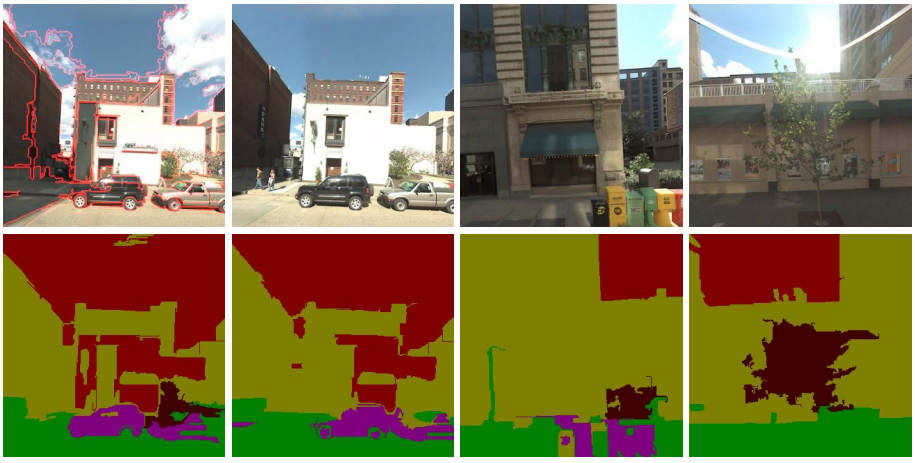
Geometry and Appearance. One component of the unary term quantifies the geometric and appearance change for each superpixel. To capture the appearance of superpixel s_i^q in the query view I_t^q, each superpixel is characterized by SIFT descriptor d_i computed at the superpixel's center. We use the 3D reconstruction of the superpixel s_i^q and the pose between the novel view and the closest reference view to find the corresponding superpixel in the reference view s_j^r and its associated descriptor d_j. As a measure of similarity $dist(s_i, s_j)$ we use the cosine of the angle between the descriptor of the query superpixel s_i and the superpixel s_j which is nearest to the location of the reprojected centroid of s_i in the reference view I^r.

$$\phi_{SIFT}(s_i) = \begin{cases} \exp\Big(-\frac{(1-d(s_i,s_j))^2}{2\sigma^2}\Big), & \text{if } r_{err}(s_i) < \tau \\ 0.5 & \text{otherwise} \end{cases} \tag{2}$$

where $r_{err}(s_i)$ is the reprojection error of the 3D reconstruction of i^{th} superpixel, from the two consecutive views of the novel sequence. In our experiments we use $\sigma = 0.25$ and $\tau = 1$ pixels. This strategy for appearance transfer is similar to the methods used for semantic labeling explored by SIFT flow [19], but the process of finding correspondences is eased by the availability of a coarse 3D geometry. Figure 5c shows an example visualizing different confidence values of appearance changes. Note that darker areas of lower confidence are due to either dramatic lighting changes or large reprojection errors errors caused by dynamically moving objects (e.g. cars).

Semantic Labeling. In order to gather additional evidence to support the final inference process, we propose to incorporate evidence about different semantic labels associated with image regions. In the next section we describe our approach to semantic labeling and describe how to incorporate the evidence about semantic labels into the final inference stage. Various approaches to semantic labeling with the focus on street scenes include works of [20], [21], [22] and [23]. In the context of our domain we consider the problem of assigning semantic labels *ground, sky, building, car, tree* to different regions of the image. We choose the superpixels obtained by color based over segmentation scheme proposed in [24].

The choice of features has been adopted from [25] where each superpixel is characterized by location and shape (position of the centroid, relative position, number of pixels and area in the image), color (color histograms of RGB, HSV values and saturation value), texture (mean absolute response of the filter bank of 15 filters and histogram of maximum responses) and perspective cues computed from long linear segments and lines aligned with different vanishing points. The entire feature vector is of 194 dimensions. In order to compute the likelihood of individual superpixels, we use boosting [26]. In our implementation, each strong boosting classifier has 15 decision trees and each of the decision trees has 6 nodes. The classifier was trained using randomly selected half of the 320 side view dataset similar to [27] and [28]. The other half of the dataset is used for



(a) (b) (c) (d)

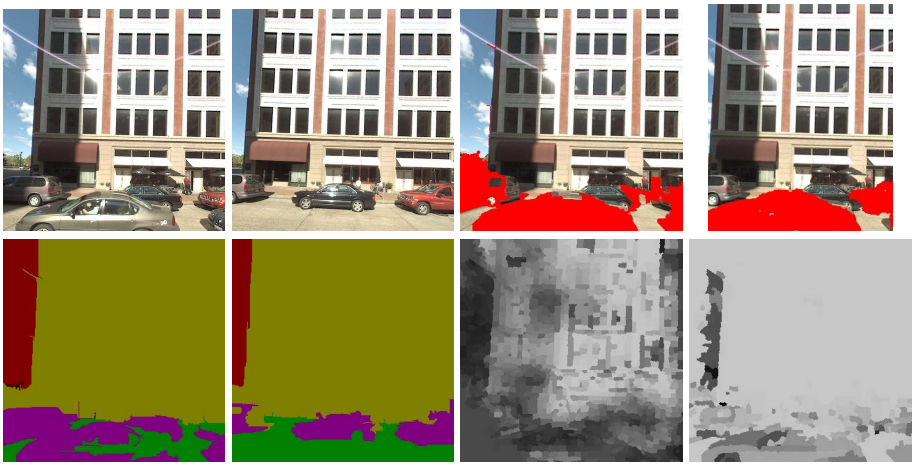
Fig. 4. Top row: (a) Example of the color-based over segmentation using method of [24] superpixel boundaries are marked by red color; (b)-(d) example street views. Bottom row: Semantic labeling result for the given over segmentation and boosting classifier, only data term is visualized. Note that due to the crude initial segmentation, several image regions are misclassified. (e.g. shaded are of the building in (a) is misclassified as sky (due to the same color). (c) mailboxes are classified as car. The color coding is the following: building: yellow, car: purple, ground: green, sky: red, tree: brown.

Table 1. Category wise accuracy of boosting classifier; global and average accuracy in % correct

System	build.	car	ground	sky	tree	glob.	aver.
[28]	89.1	56.4	89.6	97.1	69.7	88.4	80.4
[27]	95.3	40.5	96	92.5	41.4	93.2	73.1
our	96.4	68.3	94.4	97.2	48.9	94.4	81

testing. Each pixel of an image was assigned one of the five classes or *void* if it does not fall into any of the categories. Although the semantic labeling is not the final goal of this work, we have compared the performance of the boosting classifier and with the state of the art systems in Supervised Label Transfer [28] and Non-parametric scene parsing [27] in Table 1. Note that despite the fact that we do not use any MRF regularization stage, our approach outperforms the previously proposed methods for the categories of interest. Some examples of the results of semantic segmentation are in Figure 4.

While for the chosen categories the approach performs quite well due to rich features and large regions of support, there are still many cases where the label assignments are incorrect, see Figure 4 or 5. One source of errors is the local ambiguity of the region as described by the features and another is the errors of initial over segmentation into superpixels.



(a) (b) (c) (d)

Fig. 5. Change detection example and ingredients. Top row: query view, reference view, result and ground truth information. Bottom row: (a), (b) semantic labels of the two views, (c)confidence map: distance between descriptor of a superpixel and its reprojected counterpart in the previous view, (d) KL divergence between semantic layout of each superpixel and its reprojected counterpart.

To quantify the amount of semantic change between two views, we use the entire label distribution obtained for each large superpixel for both the query view and the reference view. The output of the boosting classifier returns confidence values $f^k(s_i)$ for each superpixel belonging to a particular class k, which can be interpreted as probability by passing it through a sigmoid function

$$p_k = P(l = k|f(s_i)) = \frac{1}{1 - \exp(-f(s_i))}.$$

This gives a probability distribution of labels for each superpixel $p_q = [p_1, \ldots p_k]$ in the query view and reference view $p_r = [p_1, ...p_k]$. The amount of change can then be related to the difference between the two distributions. Commonly used difference is the Kullback-Leibler Divergence of p_r and p_q defined as

$$\phi_{KL}(s_i) = \frac{1}{k} \sum_{i=1}^{k} p_q(i) \log \frac{p_q(i)}{p_r(i)}.$$

This difference is computed for each registered small superpixel s_i and its reprojected counterpart s_j in the reference view.

The final form of the unary term then becomes weighted combination of the semantic and the appearance information

$$\phi^U(s_i) = \alpha\phi_{SIFT}(s_i) + (1 - \alpha)\phi_{KL}(s_i) \ . \tag{3}$$

In our experiments the we find the optimal α by validation with respect to the ground truth data as $\alpha = 0.7$. We have a small dataset of 10 ground truth views, from 3 different locations, where we manually annotated the regions in the novel query views, which do not appear in the closest reference view. Ideally this term should be determined in a data driven way as the confidence in semantic segmentation can vary dramatically for different query views.

4.2 Pairwise Term

In our case we choose simple data driven prior based on color differences. The joint prior or the smoothness term, is approximated by pairwise potentials as

$$\phi_{smooth}(s_i, s_j) = \exp\Big(\sum_{(i,j)\in\mathcal{E}} g(i,j)\Big), \tag{4}$$

where the pairwise affinity function g is defined as

$$g(i,j) = \begin{cases} 1-e, & \text{iff } l_i = l_j \\ \delta + e, & \text{otherwise,} \end{cases} \tag{5}$$

with $e = \exp(-\|\mathbf{c}_i - \mathbf{c}_j\|^2/2\sigma^2)$, where $\mathbf{c}_i$ and $\mathbf{c}_j$ are 3-element vectors of mean colors expressed in the Lab color space for i-th and j-th superpixel, respectively, and σ is a parameter set to 0.1. The set $\mathcal{E}$ contains all neighboring superpixel pairs. The smoothness term is a combination of the Potts model penalizing different pairwise labels by the parameter δ and a color similarity based term. The aim is on one side to keep the same labels for neighboring superpixels, and on the other, to penalize same labels if they have different color. The scalars λ_s and δ weigh the importance of the terms (set to 1 and 0.2 in our experiments).

We perform the inference in the MRF by efficient and fast publicly available MAX-SUM solver [29] based on linear programming relaxation and its Lagrangian dual. Figure 6 shows some examples of the proposed change detection algorithm. We achieved 73.5% average accuracy of the change detection, averaged over 3 different locations.

There are two sources of inaccuracies in our method. As mentioned at the beginning we rely on a coarse 3D reconstruction, where correspondences are established using optical flow techniques. While the small baseline makes the problem of establishing correspondences easier there are still errors in the areas of uniform intensities and occlusions. These errors are further propagated to the reconstruction stage. Due to the fact that we use simple linear triangulation without additional regularization stage, 3D coordinates of superpixels have errors. These errors are propagated to novel views causing incorrect confidences in the appearance change. Some of these issues can be tackled by more robust motion estimation methods which explicitly model occlusion phenomena [30] or more advanced stereo reconstruction techniques. Availability of accurate 3D model would improve the accuracy of the reprojection stage [6]. Note also that we do not explicitly handle dynamically moving objects in the query view pair. In case

(a) (b) (c) (d) (e)

Fig. 6. Examples results of the change detection. The top row are the new query views; middle row are the closest views from the reference databased taken at earlier period of time and bottom row are the results of the change detection algorithm. Columns (d) and (e) show mistakes of the algorithm, which are due to differences in semantic labeling shown in 4.

the extent of moving objects and their motion is small their effect on pose estimation and 3D reconstruction is negligent. Additional challenge comes from the fidelity of the semantic segmentation. While the proposed method is comparable with the state of the art methods, it often produces incorrect labels. These unreliable label distributions are further propagated to the final optimization stage. More advanced methods for semantic segmentation would further improve the estimated label confidences.

5 Conclusions

We have presented a novel algorithm for change detection which combines geometric, appearance and semantic information. Street View images are acquired by a moving vehicle and densely sampled making the viewpoint changes between the new and old views constrained. This makes the use of patch based descriptors and their invariance properties feasible. In order to tackle the difficult appearance variations due to illumination changes, reflections and inter-reflections we use the hypotheses generated by semantic segmentation algorithm. This algorithm uses over-segmentation in to larger superpixels and exploits statistics (features) computed over larger spatial regions. In the current approach the evidence is integrated in a single global MRF inference. Further improvements

can be achieved by using more advanced 3D reconstruction methods as well better semantic segmentation strategies which exploit geometry and temporal continuity.

Acknowledgement. The author would like to thank the anonymous reviewers for their valuable comments. This project has been supported by Army Research Office Grant W911NF-1110476 and NGA Initiation Grant.

References

1. Anati, R., Daniilidis, K.: Constructing topological maps using Markov Random Fields and Loop Closure Detection. In: NIPS, pp. 37–45 (2009)
2. Cummins, M., Newman, P.: Highly scalable appearance-only slam - FAB-MAP 2.0. In: Robotics Science and Systems, RSS, Seattle, USA (2009)
3. Kumar, A., Tardif, J.P., Anati, R., Daniilidis, K.: Experiments on visual loop closing using vocabulary trees. In: CVPR Workshop, pp. 1–8 (2008)
4. Jones, E., Soatto, S.: Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. International Journal of Robotics Research (2011)
5. Micusik, B., Košecka, J.: Piecewise planar city modeling from street view panoramic sequences. In: IEEE Conference on Computer Vision and Pattern Recognition (2009)
6. Pollefeys, M., Nister, D., Frahm, J.M.: Detailed realtime urban 3D reconstruction from video. Int. Journal on Computer Vision (2008)
7. Schindler, G., Brown, M., Szeliski, R.: City-scale location recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–7 (2007)
8. Stauffer, C., Grimson, E.: Adaptive background mixture models for real-time tracking. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 246–252 (1999)
9. Radke, R., Andra, S., Al-Kofani, O., Roysam, B.: Image change detection algorithms. IEEE Transactions of Image Processing (2005)
10. Pollard, T., Mundy, J.: Change detection in 3D world. In: IEEE Conference on Computer Vision and Pattern Recognition (2011)
11. Taneja, A., Ballan, L., Pollefeys, M.: Image based detection of geometric changes in urban environments. In: IEEE Conference on Computer Vision and Pattern Recognition (2011)
12. Wolf, D., Sukhatme, G.: Online simultaneous localization and mapping in dynamic environments. In: IEEE Conference on Robotics and Automation (2004)
13. Hahnel, D., Triebel, R., Burghard, W., Thrun, S.: Map building with mobile robots in dynamic environments. In: IEEE Conference on Robotics and Automation (2002)
14. Biswas, R., Limetkai, B., Sanner, B., Thrun, S.: Towards object mapping in non-stationary environments with mobile robots. In: International Conference on Intelligent Robots and Systems (2002)
15. Ma, Y., Soatto, S., Košecka, J., Sastry, S.: Invitation to 3D vision: From Images to Geometric Models. Springer (2002)
16. Tardif, J.P., Pavlidis, Y., Daniilidis, K.: Monocular visual odometry in urban environments using an omnidirectional camera. In: Proc. of IEEE Int. Conf. on Intelligent Robots and Systems, IROS (2008)

17. Brox, T., Malik, J.: Large displacement optical flow: descriptor matching in variation motion estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence 33, 500–513 (2011)
18. Lowe, D.: Distinctive image features from scale-invariant keypoints. Int. Journal on Computer Vision 60, 91–110 (2004)
19. Liu, C., Yuen, J., Torralba, A., Sivic, J., Freeman, W.T.: SIFT Flow: Dense Correspondence across Different Scenes. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 28–42. Springer, Heidelberg (2008)
20. 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)
21. Huang, Q., Han, M., Wu, B., Ioffe, S.: A hierarchical conditional random field model for labeling and segmenting images of street scenes. In: IEEE Conference on Computer Vision and Pattern Recognition (2011)
22. Ladicky, L., Sturgess, P., Russell, C., Sengupta, S., Bastanlar, Y., Clocksin, W., Torr, P.H.: Joint optimisation for object class segmentation and dense stereo reconstruction. In: Proc. of British Machine Vision Conference (2010)
23. Xiao, J., Quan, L.: Multiple view semantic segmentation for street view images. In: Proc. of Int. Conference on Computer Vision, pp. 1–8 (2009)
24. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. Int. Journal on Computer Vision 59, 167–181 (2004)
25. Hoeim, D., Efros, A., Hebert, M.: Recovering surface layout from an image. Int. Journal on Computer Vision, 151–172 (2007)
26. Schapire, R.E., Singer, Y.: Improved boosting using confidence-rated predictions. Machine Learning 37, 297–336 (1999)
27. Zhang, H., Fang, T., Chen, X., Zhao, Q., Quan, L.: Partial similarity based nonparametric scene parsing in certain environment. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2241–2248 (2011)
28. Zhang, H., Xiao, J., Quan, L.: Supervised label transfer for semantic segmentation of street scenes. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 561–574. Springer, Heidelberg (2010)
29. Werner, T.: A linear programming approach to Max-sum problem: A review. PAMI 29, 1165–1179 (2007)
30. Ayvaci, A., Raptis, M., Soatto, S.: Sparse occlusion detection with optical flow. International Journal of Computer Vision 97 (2012)

Adaptive Background Defogging with Foreground Decremental Preconditioned Conjugate Gradient

Jacky Shun-Cho Yuk and Kwan-Yee Kenneth Wong

Dept. of Computer Science
The University of Hong Kong, Hong Kong

Abstract. The quality of outdoor surveillance videos are always degraded by bad weathers, such as fog, haze, and snowing. The degraded videos not only provide poor visualizations, but also increase the difficulty of vision-based analysis such as foreground/background segmentation. However, haze/fog removal has never been an easy task, and is often very time consuming. Most of the existing methods only consider a single image, and no temporal information of a video is used. In this paper, a novel adaptive background defogging method is presented. It is observed that most of the background regions between two consecutive video frames do not vary too much. Based on this observation, each video frame is firstly defogged by a background transmission map which is generated adaptively by the proposed foreground decremental preconditioned conjugate gradient (FDPCG). It is shown that foreground/background segmentation can be improved dramatically with such background-defogged video frames. With the help of a foreground map, the defogging of foreground regions is then completed by 1) foreground transmission estimation by fusion, and 2) transmission refinement by the proposed foreground incremental preconditioned conjugate gradient (FIPCG). Experimental results show that the proposed method can effectively improve the visualization quality of surveillance videos under heavy fog and snowing weather. Comparing with the state-of-the-art image defogging methods, the proposed method is much more efficient.

1 Introduction

Outdoor surveillance videos are always degraded by challenging bad weathers, e.g., haze, fog, raining, snowing, etc. Some degradations, like under the haze and fog weathers, are mainly due to light absorption and scattering by atmospheric particles. The light from the viewing objects is being partly absorbed before it reaches the camera. The farther the objects from the camera, the more the light is being absorbed. The degraded videos always have low contrast and bad color fidelity. These degraded videos not only produce poor visualizations, but also make further vision-based analysis, such as foreground/background segmentation, more difficult. There are desires to improve the visual qualities of surveillance videos under hazy or foggy weathers. The goal is not only for better

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 602–614, 2013.

visualizations, but also improve the correctness of the further higher level video analysis.

Image haze/fog removal techniques have been researched for more than a decade. The defogging problem on a single image is under-constrained due to lack of depth information. Early researches required multiple images of the same scene under different exposures (e.g., under different weather conditions [1,2], or different degree of polarization [3]) to recover a foggy scene. Although these methods can significantly improve the visual quality, manual works are always required to prepare suitable images under different conditions for defogging.

Later on, single image defogging [4,5,6,7,8] got great progress and success. Based on the assumption that non-foggy image patches usually have a high contrast, Tan [4] proposed to recover a foggy image by maximizing the local contrast. Tan's method produces nice defogging results, but his assumption may not be physically correct. Fattal [5] assumed the transmission and image shading were locally uncorrelated. He estimated the albedo values and inferred the medium transmission by MRF. Fattal's approach, however, may fail under heavy fog scenarios. He *et al.* [6,9] proposed the state-of-the-art dark channel prior for estimating image transmissions which are refined by soft matting. Although single image defogging is now pretty mature, existing methods are seldom applied to defog video sequences. Most of the methods only target at defogging a single image, and no temporal information of the video sequences is considered. Without temporal information, each video frame has to be processed individually, and this makes the defogging procedure very time consuming (Tan's method [4] required 5 minutes to process a frame, while the methods of Fattal [5] and He *et al.* [6,9] require about 20 to 30 seconds per frame).

Recently, Dong *et al.* [10] proposed to locate the foreground regions on foggy video frames by comparing the foreground and background transmission maps. Their method, however, requires manually selecting 2 foreground-free scenes under different weather conditions for calculating the background transmission map. Once the background transmission map has been calculated, there will not be any further update on the map, and therefore, the method is not able to tolerate any background change.

This paper proposes a novel adaptive fog removal method for foggy surveillance video scenes. Based on the observation that most of the background regions between consecutive video frames will not vary too much, a video frame is firstly defogged by a background transmission map (fig. 1) which is generated and updated adaptively by the proposed foreground decremental preconditioned conjugate gradient(FDPCG). FDPCG targets at reducing the influence of foreground regions during the estimation of the transmission map. The background-defogged frame is then processed by foreground/background segmentation to generate the foreground map. The foreground regions in a background-defogged frame could vary from nearly fog-free, when foreground and background are nearly at the same depth (fig. 1 (a)), to extremely dark, when the depth between foreground and background is large (fig. 1 (b)). Both cases can make the further foreground/background segmentations easier.

The rest of the paper is organized as follows. Section 2 briefly describes the transmission model and the state-of-the-art dark channel prior defogging. Section 3 introduces the adaptive background defogging by the proposed FDPCG. Section 4 describes the foreground transmission estimation. Experimental results are presented in Section 5, followed by conclusions in Section 6.

(a) Background-defogging when the foreground and background are nearly at the same depth.

(b) Background-defogging when the depth between foreground and background is large.

Fig. 1. Examples of background defogging. The first column shows the original foggy video frames, followed by the corresponding background transmission maps and the background-defogged frames in column 2 and 3, respectively.

2 Dark Channel Prior Defogging

2.1 Transmission Model

In computer vision and graphics, a haze/fog image is widely formulated by the following transmission model [4,5,6,9],

$$I(x) = J(x)t(x) + A(1 - t(x)), \tag{1}$$

where $I(x)$ and $J(x)$ are the observed intensity and the fog-free scene radiance at pixel x, respectively. A is the global air light, and t is the transmission value which describes the portion of the light finally reaching the camera. The target of fog removal is to recover the fog-free scene radiance, J, according to the estimated t, and the observed I and A,

$$J(x) = \frac{I(x) - A}{t(x)} + A. \tag{2}$$

As suggested in [6,9], $t(x)$ is lower bounded by $t_0 = 0.1$, so that the recovered image will not be too dim,

$$J(x) = \frac{I(x) - A}{\mathrm{MAX}(t(x), t_0)} + A. \tag{3}$$

2.2 Dark Channel Prior

Dark channel prior was firstly proposed in [6]. It was shown to be able to effectively predict the transmission map based on the observation that in most of the non-foggy image patches, at least one color channel has very low intensity at some pixels. The dark channel, J^{dark}, at pixel x is defined as

$$J^{dark}(x) = \text{MIN}_{c\in\{r,g,b\}}(\text{MIN}_{\{y\in\Omega(x)\}}(J^c(y))), \tag{4}$$

where $J^c(y)$ is one of the RGB color channels at pixel y and $\Omega(x)$ is the local image patch centered at x.

The air light, A, is assumed to be a non-zero constant, and A^c, J^c and I^c of a particular color channel, c, are coplaner [6]. By applying the dark channels to the transmission model (1), the estimated transmission at pixel x, $\tilde{t}(x)$, can be derived as

$$\tilde{t}(x) = \frac{J^{dark}(x)}{A^{dark}}\tilde{t}(x) + 1 - \frac{I^{dark}(x)}{A^{dark}}. \tag{5}$$

Since $\tilde{t}(x)$ is ranging between [0, 1], and the non-foggy dark channel $J^{dark}(x)$ should have very low intensity, $J^{dark}(x) \to 0$, so $\tilde{t}(x)$ can then be directly calculated as

$$\tilde{t}(x) = 1 - \text{MIN}_c(\text{MIN}_{y\in\Omega(x)}\frac{I^c(y)}{A^c}). \tag{6}$$

2.3 Soft Matting

The transmission model (1) is similar to the alpha matting problem. This allows using soft matting [11] to refine the transmission map by treating the transmission map as an alpha map. This refinement can be done by minimizing the following cost function,

$$E(\mathbf{t}) = \mathbf{t}^T\mathbf{L}\mathbf{t} + \lambda(\mathbf{t} - \tilde{\mathbf{t}})^T(\mathbf{t} - \tilde{\mathbf{t}}). \tag{7}$$

where $\mathbf{t}$ and $\tilde{\mathbf{t}}$ are the refined and predicted transmission map, respectively. Both $\mathbf{t}$ and $\tilde{\mathbf{t}}$ are in l = width $\times$ height dimensions. λ is for regularization. $\mathbf{L}$ is a $l \times l$ dimensional Matting Laplacian matrix [11]. The (i,j)-th element of L is defined as

$$L_{i,j} = \sum_{k|(i,j)\in\omega_k} (\delta_{ij} - \frac{1}{|\omega_k|}(1 + (\mathbf{I}(i) - \mu_k)^T(\mathbf{\Sigma}_k + \frac{\varepsilon}{|\omega_k|}\mathbf{U}_3)^{-1}(\mathbf{I}(j) - \mu_k))), \tag{8}$$

where $\mathbf{I}(i)$ is the 3-dimensional RGB color at pixel i. δ_{ij} is the Kronecker delta. μ_k and $\mathbf{\Sigma}_k$ are the mean and covariance matrix of the colors in window ω_k. $\mathbf{U}_3$ is a 3x3 identity matrix. ε is for regularization, and $|\omega_k|$ is the number of pixel in ω_k.

3 Adaptive Background Defogging

3.1 Preconditioned Conjugate Gradient

The optimal $\mathbf{t}$ in (7) can be obtained by solving

$$(\mathbf{L} + \lambda \mathbf{U})\mathbf{t} = \lambda \tilde{\mathbf{t}}, \tag{9}$$

where $\mathbf{U}$ is $l \times l$ identity matrix, and $\lambda = 10^{-4}$ is a small constant so that $\mathbf{t}$ is softly constrained by $\tilde{\mathbf{t}}$.

Equation (9) can be solved by preconditioned conjugate gradient (PCG) method,

$$\mathbf{M}^{-1}(\mathbf{L} + \lambda \mathbf{U})\mathbf{t} = \mathbf{M}^{-1}(\lambda \tilde{\mathbf{t}}), \tag{10}$$

where $\mathbf{M}$ is $l \times l$ preconditioning matrix. The main purpose of the preconditioning matrix is to help the PCG converge faster. In this paper, $\mathbf{M}$ is chosen to be a diagonal Jacobi preconditioner [12]. The element, $m_{i,j}$, of the Jacobi preconditioner is defined as

$$m_{i,j} = \begin{cases} a_{i,i} \text{ if } i = j \\ 0 \quad i \neq j \end{cases}, \tag{11}$$

where $a_{i,i}$ is the i-th diagonal element of $(\mathbf{L} + \lambda \mathbf{U})$. Jacobi precontioner simply normalizes the i-th row of the matrix in (10) by its i-th coefficient value. The process of iterative PCG is presented in algorithm 1.

Algorithm 1. Preconditioned Conjugate Gradient (PCG). The initial transmissions $\mathbf{t}_0$ is initialized to $\mathbf{t}_{\text{init}}$, where each element of $\mathbf{t}_{\text{init}} = 0.2$.

Initialization:
$k = 0$, $\mathbf{t}_0 = \mathbf{t}_{\text{init}}$,
$\mathbf{r}_0 = \lambda \tilde{\mathbf{t}} - (\mathbf{L} + \lambda \mathbf{U})\mathbf{t}_0$, $\mathbf{z}_0 = \mathbf{M}^{-1}\mathbf{r}_0$, $\mathbf{d}_0 = \mathbf{z}_0$
$err_0 = \mathbf{z}_0^T \mathbf{r}_0$

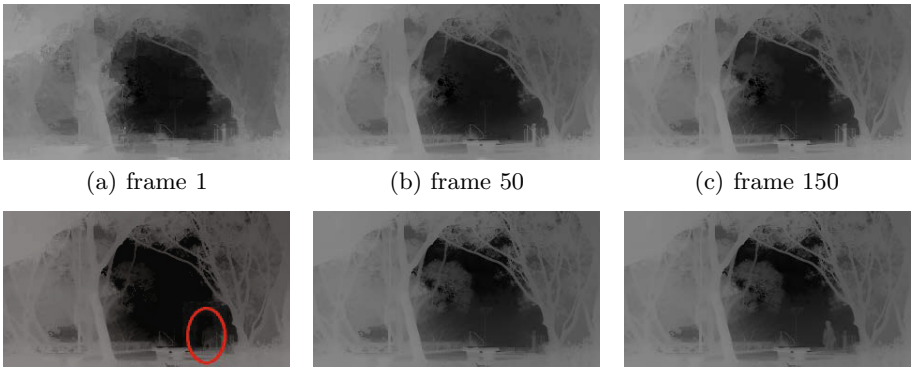
Iteration:
while $err_k > \epsilon$ and $k < K$ {

$$\begin{aligned}
\alpha_k &= \frac{\mathbf{z}_k^T \mathbf{r}_k}{\mathbf{d}_k^T (\mathbf{L} + \lambda \mathbf{U})\mathbf{d}_k} \\
\mathbf{t}_{k+1} &= \mathbf{t}_k + \alpha_k \mathbf{d}_k \\
\mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k (\mathbf{L} + \lambda \mathbf{U})\mathbf{d}_k \\
\mathbf{z}_{k+1} &= \mathbf{M}^{-1}\mathbf{r}_{k+1} \\
\beta_{k+1} &= \frac{\mathbf{z}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{z}_k^T \mathbf{r}_k} \\
\mathbf{d}_{k+1} &= \mathbf{z}_{k+1} + \beta_{k+1} \mathbf{d}_k \\
err_{k+1} &= \mathbf{z}_{k+1}^T \mathbf{r}_{k+1} \\
k &= k + 1
\end{aligned}$$

}

3.2 Adaptive Foreground Decremental Preconditioned Conjugate Gradient

The iterative PCG (alg. 1) will stop either when the error, err_k, is smaller than ϵ or the algorithm reaches maximum number of iterations, K. In our experiment, when K was not limited and ϵ was set to $\text{MAX}(err_0 \times 0.001, 10^{-7})$, the PCG algorithm converged after around 700 to 850 iterations for each frame.



(a) frame 1 (b) frame 50 (c) frame 150

(d) frame 700 (e) background transmission map of frame 700 (f) complete transmission map of frame 700

Fig. 2. (a)-(c) The iterative PCG converges over frames. More and more details at background regions are recovered in later frames. (d) Defects at the foreground regions (red circled) are caused by non-completed PCG. (e), (f) The corresponding FDPCG background transmission map and final transmission map, respectively.

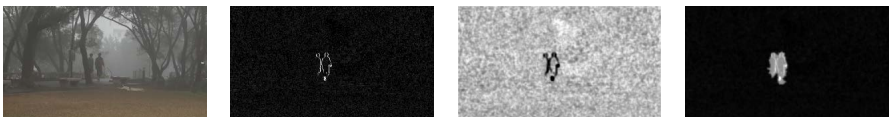
As the characteristics of surveillance video using a static camera, the background between consecutive frames does not change a lot. This suggests that the iterative part of PCG algorithm can be applied over frames. To achieve this, we slightly modify the initialization of $\mathbf{t}_0$ in PCG (alg. 1). We set $\mathbf{t}_0 = \mathbf{t}^{fn-1}$, where $\mathbf{t}^{fn-1}$ is the refined transmission map of the previous frame. The maximum iterations K is also limited to 50 in our experiments. This approach is able to adaptively refine the transmissions of the background regions from blocky (fig. 2(a)) to detail (fig. 2(c)). Every frame makes contributions to the transmission refinement, and therefore, the refined transmission maps can also tolerate continuous background environmental changes. However, adaptive PCG may also generate some rare defects at foreground regions as shown in figure 2(d). To overcome these defects, we propose a novel Foreground Decremental Preconditioned Conjugate Gradient (FDPCG). In addition to the Jacobi preconditioner, a foreground decremental preconditioning matrix $\mathbf{F_D}$ is introduced to reduce the influence of foreground pixels in PCG.

$$\mathbf{F_D}^{-1}\mathbf{M}^{-1}(\mathbf{L} + \lambda\mathbf{U})\mathbf{t}_{\text{bg}} = \mathbf{F_D}^{-1}\mathbf{M}^{-1}(\lambda\tilde{\mathbf{t}}). \tag{12}$$

$\mathbf{F_D}$ is also chosen to be a diagonal matrix, and is constructed based on the difference between two consecutive frames. Since $(\mathbf{L} + \lambda\mathbf{U})$ is a sparse matrix, the i-th diagonal element, $f^D_{i,i}$, of $\mathbf{F_D}$ will only affect the transmission results of the pixels in the neighborhood, $\mathbf{N}(i)$, of pixel i. Based on (8), $\mathbf{N}(i)$ is chosen to be a 5×5 windows centered at pixel i. The element of $\mathbf{F_D}$ is then defined as

$$f^D_{i,j} = \begin{cases} \left(\sum_{x \in \mathbf{N}(i)} G(x, \sigma_s) N(d, \sigma_d)\right)^{-1} & \text{if } i = j \\ 0 & i \neq j \end{cases}, \tag{13}$$

where $G_s(x, \sigma_s)$ is a spatial Gaussian function centered at pixel i. $N(d, \sigma_d)$ is a normal distribution function, and $d = d_r + d_g + d_b$ is the per pixel RGB-color difference between previous and current frames at pixel x. In the implementation, both $G(x, \sigma_s)$ and $N(d, \sigma_d)$ are pre-calculated for efficiency, and σ_s and σ_d are set to 1 and $0.1 \times d_{max}$, respectively. This formulation decreases the weights of the neighboring equations when the pixel difference increases. Figure 3(c) visualizes the diagonal element values of ${\mathbf{F_D}}^{-1}$.



(a) original frame (b) frame difference (c) ${\mathbf{F_D}}^{-1}$ (d) ${\mathbf{F_I}}^{-1}$

Fig. 3. (a) Original frame, and the visualizations of (b) frame difference, (c) ${\mathbf{F_D}}^{-1}$ in (12), and (d) ${\mathbf{F_I}}^{-1}$ in (18), respectively

4 Foreground Transmission Recovery

4.1 Foreground/Background Segmentation

We need foreground/background segmentation algorithm for recovering the foreground transmissions. Applying traditional background modeling on foggy videos may not generate good foreground results, especially those texture-based background modeling. Instead, we apply background modeling on the background-defogged video frames. The foreground regions in the background-defogged frames could vary from nearly fog-free, when background and foreground are almost at the same depth, to extremely dark, when the depth difference between foreground and background is large. Any of these cases enhances the foreground to be more distinctive from the background, and therefore, improves the foreground/background segmentation results.

In this paper, a texture-based PLPM background modeling [13] was used for illustration. The main reason to choose PLPM is that texture-based background modeling is usually more tolerant to outdoor scenes, and PLPM can perform the foreground/background segmentation in a very efficient manner.

4.2 Foreground Transmission Estimation by Fusion

With the help of the foreground map, the transmissions of each foreground region is estimated by 1) temporal transmission prediction, and 2) environmental transmission prediction. These two predictions are then fused together for generating the final estimated foreground region transmissions.

The temporal transmission prediction, $p_t^{\mathbf{R}}$, predicts the transmissions of current foreground region, $\mathbf{R}$, from previous frame. $p_t^{\mathbf{R}}$ is defined as

$$p_t^{\mathbf{R}} = \frac{1}{N} \sum_{x \in \hat{\mathbf{R}}} \hat{t}(x), \tag{14}$$

where $\hat{\mathbf{R}}$ is the corresponding region of $\mathbf{R}$ in previous frame, N is the number of pixel in $\hat{\mathbf{R}}$, and $\hat{t}(x)$ is the resulting transmission value at pixel x of previous frame.

The environmental transmission prediction, $p_e^{\mathbf{R}}$, predicts the transmissions of foreground region, $\mathbf{R}$, from the current background transmission map, $\mathbf{t}_{\text{bg}}$ (12). Base on the observation that the foreground regions usually have larger transmission values than background (since foreground objects are usually closer to the camera) , $p_e^{\mathbf{R}}$ is defined as

$$p_e^{\mathbf{R}} = \mu_{t_{\text{bg}}} + \omega \sigma_{t_{\text{bg}}}, \tag{15}$$

where $\mu_{t_{\text{bg}}}$ and $\sigma_{t_{\text{bg}}}$ are the mean and standard deviation of transmission values of $\mathbf{t}_{\text{bg}}$, respectively. ω is a configurable parameter which was set to 1.5 in the experiments.

The final estimated foreground transmission value, $t_{\text{fg}}(x)$, at foreground pixel x is then fused as

$$t_{\text{fg}}(x) = \beta p_t^{\mathbf{R}} + (1 - \beta) p_e^{\mathbf{R}}, \tag{16}$$

where $\beta = e^{-\frac{1}{2}(\frac{d}{\sigma})^2}$. d is the per pixel RGB color difference between current and previous frames, and σ here is a control parameter which is set to $0.05 \times d_{max}$ in our experiments. When $d \to 0$, temporal transmission prediction, $p_t^{\mathbf{R}}$, is preferred. Otherwise, environmental transmission prediction , $p_e^{\mathbf{R}}$, is preferred.

4.3 Foreground Transmission Refinement

The resultant transmission map, $\mathbf{t}_r$, is then constructed by combining foreground and background transmission maps,

$$t_r(x) = \begin{cases} t_{\text{fg}}(x) & \text{if pixel } x \text{ is on foreground.} \\ t_{\text{bg}}(x) & \text{otherwise} \end{cases} \tag{17}$$

We further refine the transmission map by foreground incremental preconditioned conjugate gradient (FIPCG). FIPCG is similar to FDPCG but targeting to increase the foreground effect during PCG,

$$\mathbf{F_I}^{-1}\mathbf{M}^{-1}(\mathbf{L} + \lambda \mathbf{U})\mathbf{t} = \mathbf{F_I}^{-1}\mathbf{M}^{-1}(\lambda \tilde{\mathbf{t}}), \tag{18}$$

in which, the final transmission map, $\mathbf{t}$, is initialized to $\mathbf{t}_r$ in the PCG (alg. 1). $\mathbf{M}$ and $\tilde{\mathbf{t}}$ are Jacobi preconditioner [12] and the dark channel transmission map (6), respectively. Similar to $\mathbf{F_D}$ (13), $\mathbf{F_I}$ is also chosen to be a diagonal matrix, and its elements are defined as

$$f^I_{i,j} = \begin{cases} \left(\sum_{x \in \mathbf{N}(i)} G(x, \sigma_s)(\delta(x) + \frac{1}{N(d,\sigma_d)+\varepsilon}) \right)^{-1} & \text{if } i = j, \\ 0 & i \neq j \end{cases}, \tag{19}$$

where $G(x, \sigma_s)$ is the same spatial Guassisn function in (13). $\delta(x)$ is a delta function, $\delta(x) = 1$ when pixel x is on foreground region, and $\varepsilon = 0.01$ for regularization. $\mathbf{F_I}$ increases the importance of pixels with large frame difference and/or pixels at foreground regions in PCG. Figure 3(d) visualizes the pixel map of $\mathbf{F_I}$.

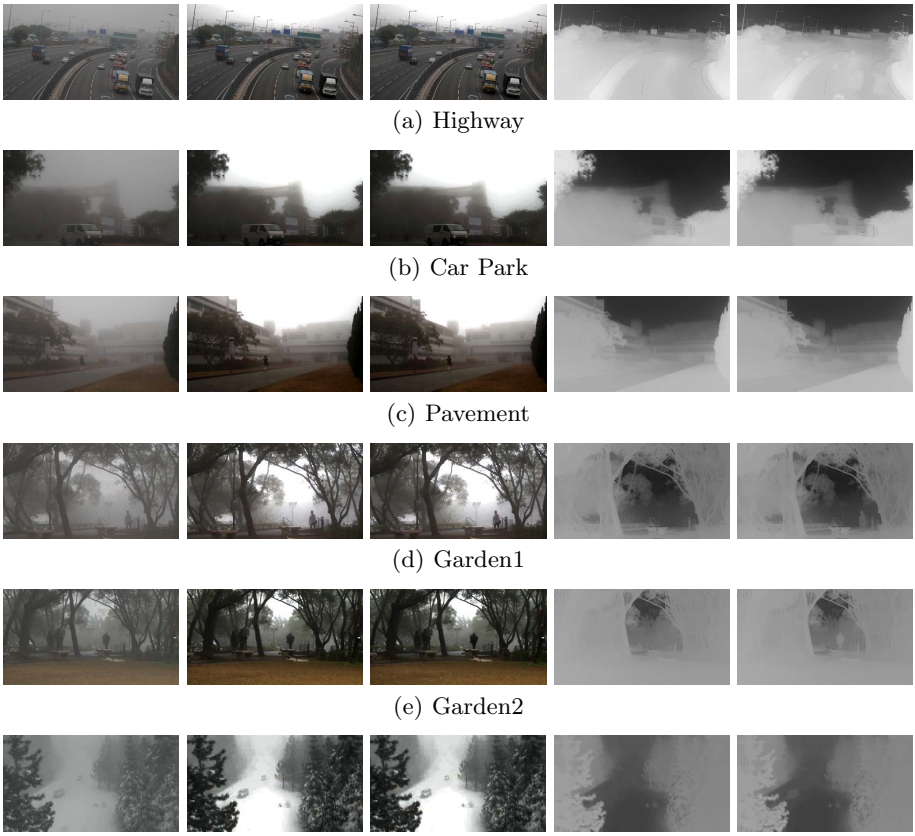
5 Experimental Results

Six challenging real-life surveillance video sequences[1] (see fig. 4) were used to evaluate the proposed methods. Five of the videos are foggy scenes, including highway, car park, and garden scenarios. The remaining one is a heavily snowing scene. The experiments were performed on a computer with an Intel Core 2 CPU 6300 @ 1.86GHz. In our implementation, without any optimization, the proposed background defogging requires about 1.5 to 2 seconds for a 320 × 180 frame, and an additional 1 second for PLPM background modeling and foreground defogging. The detailed time measurement is listed in table 1. Comparing to the state-of-the-art dark channel prior defogging modules [6,9], which requires about 25 to 30 seconds to completely defog a frame, the proposed method is much more efficient, and has a high potential to be optimized on GPU in order to fulfill the real-time requirement.

Table 1. The number of PCG converging iterations and processing time per frame

	He *et. al.* [6,9]		Proposed Method			
	avg. PCG iterations	avg. time(sec)	avg. FDPCG iterations	avg. FIPCG iterations	1st frame time (sec)	avg. time(sec)
highway	791	28.24	17	18	3.11	2.77
car park	805	28.73	16	19	3.03	2.71
pavement	779	27.87	16	20	3.03	2.72
garden1	710	25.50	16	9	3.05	2.27
garden2	773	27.70	19	24	3.05	3.15
snowing	809	35.35	32	30	3.64	4.44

[1] The testing video sequences are available for download at `http://www.cs.hku.hk/~scyuk/downloads.htm`.



(a) Highway

(b) Car Park

(c) Pavement

(d) Garden1

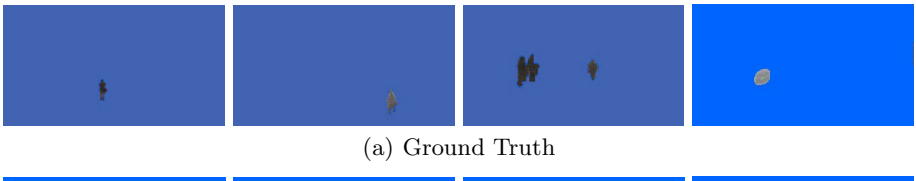
(e) Garden2

(f) Snowing

Fig. 4. Testing video sequences: (a) highway, (b) car park, (c) pavement, (d) garden1, (e) garden2, and (f) snowing. The 1st column shows the original frames. The 2nd and 3rd columns are the background and final defogged frames, respectively, and the 4th and 5th columns show the FDPCG background transmission maps and final transmission maps, respectively.

Table 2. *F-Score* foreground/background segmentation measurement of PLPM [13] running on original foggy frames, final defogged frames and background defogged frames, respectively

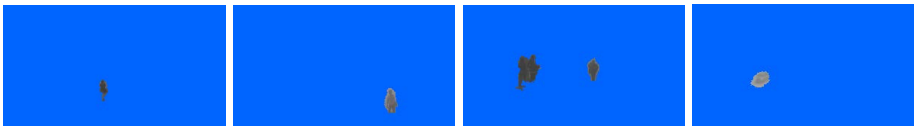
	highway	car park	pavement	garden1	garden2	snowing	avg.
orig. foggy	0.82	0.90	0.05	0.61	0.71	0.38	0.58
final defogged	0.82	0.92	0.83	0.79	0.87	0.75	0.83
bg. defogged	0.81	0.93	0.82	0.83	0.86	0.75	0.83



(a) Ground Truth

(b) PLPM[13] running on original foggy frames

(c) PLPM[13] running on background-defogged frames



(d) PLPM[13] running on final defogged frames

Fig. 5. Foreground/background segmentation results: (a) ground truth, and PLPM[13] running on (b) original foggy frames, (c) background-defogged frames, and (d) final defogged frames

(a) Defogging results of the proposed method.

(b) Defogging results of He *et al.* [6,9]

Fig. 6. Defogging results compare with He *et al.* [6,9]]

Figure 4 shows the defogging results by the proposed method. The proposed FDPCG is shown to be able to effectively remove the foreground effects when calculating the background transmission maps. As discussed in the earlier sections, when the foreground objects are nearly at the same depth as the background such as the highway and snowing sequences, the resultant quality of background-defogged video is already very good. On the other hand, if the depths of the

foreground objects are largely different from the background, the foreground objects on the background-defogged frame could be extremely dark. In both cases or the cases in between, the foreground regions became more distinguishable from the background, and this improves the foreground/background segmentation results.

Figure 5 shows the foreground/background segmentation results, and table 2 lists the *F-Score* [13] of the results. The *F-Score* is defined as $\frac{2TP}{2TP+FP+FN}$, where TP, FP and FN are true positive, false positive and false negative, respectively. Results show that the PLPM background modeling [13] running on the background-defogged videos is almost the same, or even better than (car park and garden1) running on the completely defogged frame. In most of the scenarios, the completely defogged videos as well as the background-defogged videos got better PLPM results then the original foggy videos.

Figure 6 shows the comparisons between the proposed method and the state-of-the-art dark channel prior [6,9]. The results of [6,9] are supposed to be the best results that the proposed method can achieve as the results of [6,9] (fig. 6 (b)) converged completely for each frame. As shown in figure 6 (a), the proposed method performed almost the same as He *et al*. [6,9] in garden1 sequence. For the car park sequence, the proposed method can only recover the transmissions on background regions in details, but is not able to recover the transmission details on the foreground objects. This is because the proposed method did not perform enough iterations for converging the transmissions on foreground regions. The complete converges on the foreground regions could be very time consuming. Such detailed transmissions [6,9] on foreground regions, however, may not be necessary. Instead, it is reasonable to assume that the whole area of each foreground object should be at nearly the same depth from the shooting camera. Therefore, the transmissions within each object was assumed to be nearly the same. Results also show that the defogged results of the proposed method are not degraded much comparing with He *et al*. [6,9].

6 Conclusions

This paper proposes a novel Foreground Decremental Preconditioned Conjugate Gradient (FDPCG) for adaptive background defogging of surveillance videos. Each background-defogged frame is then processed by foreground/background segmentation algorithm, and the transmissions on foreground regions are recovered by the proposed fusion technique. Afterward, the final transmissions of each frame are refined by Foreground Incremental Preconditioned Conjugate Gradient (FIPCG). Unlike the previous state-of-the-art algorithms [4,5,6], which completely defog an image without using any temporal information, the proposed method defogs the video scenes adaptively. Hence, the proposed method is able to tolerate any background change in the scenes. Experimental results show that the proposed method can produce high quality defogged videos. The foreground/background segmentation results based on the background-defogged frames are also improved dramatically. Comparing to the previous

state-of-the-art defogging techniques [4,5,6], the proposed method is much more efficient, and therefore, retains a high capability to be implemented and optimized on GPU which can fulfill the real-time purpose.

References

1. Narasimhan, S., Nayar, S.: Chromatic framework for vision in bad weather. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR, Hilton Head, SC, USA, pp. 598–605 (2000)
2. Nayar, S., Narasimhan, S.: Vision in bad weather. In: Proc. International Conference on Computer Vision, ICCV, Kerkyra, Corfu, Greece, pp. 820–827 (1999)
3. Shwartz, S., Namer, E., Schechner, Y.: Blind haze separation. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR, New York, NY, USA, pp. 1984–1991 (2006)
4. Tan, R.: Visibility in bad weather from a single image. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR, Anchorage, Alaska, USA (2008)
5. Fattal, R.: Single image dehazing. In: Proc. SIGGRAPH, Los Angeles, California, USA (2008)
6. He, K., Sun, J., Tang, X.: Single image haze removal using dark channel prior. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR, Miami, Florida, USA, pp. 1956–1963 (2009)
7. Ancuti, C., Ancuti, C., Bekaert, P.: Effective single image dehazing by fusion. In: Proc. IEEE Conf. International Conference on Image Processing, ICIP, Hong Kong, China (2010)
8. Ancuti, C.O., Ancuti, C., Hermans, C., Bekaert, P.: A Fast Semi-inverse Approach to Detect and Remove the Haze from a Single Image. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part II. LNCS, vol. 6493, pp. 501–514. Springer, Heidelberg (2011)
9. He, K., Sun, J., Tang, X.: Single image haze removal using dark channel prior. IEEE Transactions on Pattern Analysis and Machine Intelligence, TPAMI 33, 2341–2353 (2011)
10. Dong, W., Jia, Z., Shao, J., Li, Z., Liu, F., Zhao, J., Peng, P.Y.: Adaptive object detection and visibility improvement in foggy image. Journal of Multimedia 6 (2011)
11. Levin, A., Lischinski, D., Weiss, Y.: A closed form solution to natural image matting. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR, New York, NY, USA, pp. 61–68 (2006)
12. Smith, I., Wong, S.: Pcg methods in transient fe analysis. part i: First order problems. International Journal for Numerical Methods in Engineering 28, 1557–1566 (1989)
13. Yuk, J.S.C., Wong, K.Y.K.: An efficient pattern-less background modeling based on scale invariant local states. In: Proc. IEEE International Conference on Advanced Video and Signal-Based Surveillance, AVSS, Klagenfurt, Austria, pp. 285–290 (2011)

A Shadow Repair Approach for Kinect Depth Maps

Yu Yu, Yonghong Song, Yuanlin Zhang, and Shu Wen

Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University

Abstract. The depth data provided by Kinect is incomplete because of no-measured depth (NMD for short) pixels, so a preprocessing approach for depth map is necessary. In this paper, a depth map repair approach is proposed for one specific NMD pixels' (shadow) removal. Firstly, the NMD pixels are divided into three types. Then a mathematical model based on the depth measurement of Kinect is built to explain the cause of shadow. A shadow discriminant based on the model is also designed. Finally, the repair approach is proposed for shadow regions detection and removal. Experimental results show that our method is both time saving and accurate.

1 Introduction

Kinect is a motion sensing input device by Microsoft for the XBOX 360 video game console. This device enables users to interact with the XBOX 360 through a natural user interface, which revolutionizes the way people play games. The key technology lies in human pose recognition, an area that has always been the research focus of computer vision. But it has been proven difficult with normal videos. Kinect, capturing depth data of the environment directly, makes this issue much easier. So this device is drawing attention from both researchers and application developers.

However, the depth data provided by Kinect is not perfect. Noises such as no-measured depth pixels(NMD pixels: pixels whose depth value is zero), noisy object boundaries and depth measurements fluctuation [1] make the depth map both incomplete and inaccurate. As shown in Fig. 1(a), the black regions represent NMD pixels. At present, the noisy depth data is used directly by most applications and affects the accuracy of these applications greatly. So a preprocessing for depth map is necessary.

Several methods have been developed to improve the quality of depth maps. Some of the literatures are specially for the post-processing of automatically generated depth maps. A joint bilateral filter to smoothen depth map is presented in [2], where the depth map is unsampled to be aligned with the original image. Another approach using adaptive cross-trilateral median filtering to improve generated depth maps is proposed in [3]. Some of the works are designed specially for depth maps captured by Time-of-flight sensors. Qingxiong Yang [4] enhances the resolution of depth map by using one or two registered and potentially high-resolution color images as reference. In [5], a method combining results from two

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 615–626, 2013.

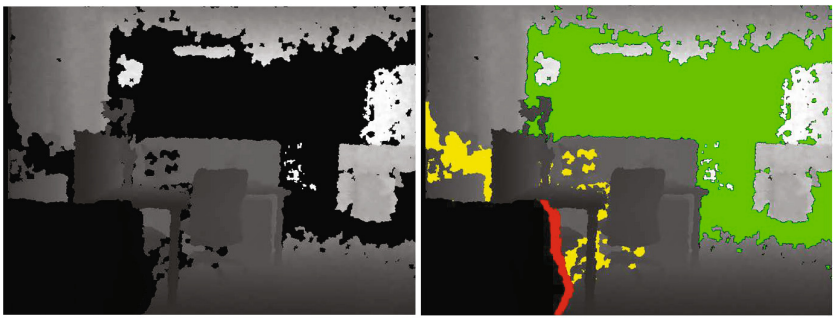


Fig. 1. Kinect Depth Map (a) noises in depth map, black pixels represent NMD regions. (b) three types of NMD regions, Green : out-of-range regions. Yellow: mirror-like regions. Red : shadow regions.

stereo methods is introduced to enhance depth measurements, where a depth probability distribution function is calculated and global methods such as belief propagation and graph cuts are applied. Nonetheless, few work deals with depth maps generated by structure sensor such as Kinect. An adaptive spatio-temporal filter [1] is presented to improve the accuracy and stability of Kinect depth, where an interpolation algorithm is also used to remove NMD regions and obtain a more complete depth map. Massimo Camplani [6] presents a joint-bilateral filtering framework to inpaint the depth maps. In [7], a method of filtering is proposed to fill NMD regions and improve the temporal stability of Kinect depth data.

In fact, NMD regions, which make the depth data incomplete, are caused by various factors. In this paper, they are divided into three types.

- out-of-range regions;
- mirror-like regions;
- shadow regions.

The three types above are illustrated in Fig. 1(b) by green, yellow and red pixels respectively. They are presented with different features, like shadow region usually presented on the right side of object, as is shown in Fig. 2(a). Furthermore, Fig. 2(b) shows that the shape of shadow is almost the same as the object. Therefore, there is a relationship between the object and the shadow. Actually, the occurrence of shadow is related with the depth measurement of Kinect. In contrast, out-of-range regions and mirror-like regions appear randomly, so filling them is meaningless and unreliable.

Approaches in [1] [6] [7] are partially meant for NMD regions removal, but they fail to distinguish these regions further. So the results may be inaccurate.

In this paper, a repair method is proposed specially for shadow removal. The objective of our approach is to fill the shadow regions and make the depth data more complete. We firstly build a mathematic model explaining the cause of

shadow. Then a discriminant based on this model is proposed to distinguish shadow from other NMD regions. Then, a horizontal scanning method and an exhaustion method is applied to detect shadow regions. Finally, we remove the shadow regions by a simple filling strategy. Experimental results show our method is both time saving and reliable.

Fig. 2. Shadow Regions in Depth Map (a) shadow presented on the right side of person. (b) shadow shape similar with object.

This paper is organized as follows: the mathematic model of shadow is introduced in section 2, then the shadow repair approach is given in section 3. In section 4, the experimental result is presented. Finally, the conclusion and future works are discussed in section 5.

2 Shadow Modeling

According to [8], the Kinect is a structured sensor consists of one infrared laser emitter, one infrared camera and one RGB camera. The measurement of depth is the so-called triangulation process: the laser source firstly emits a constant pattern of speckles into the scene, then the speckles are reflected and get captured by infrared camera. By comparing the captured speckle pattern with a reference one, the sensor acquires depth of pixels in the scene. Based on this principle, a model is built to present the cause of shadow in this section and a mathematical expression for shadow offset is also derived.

2.1 Cause of Shadow

Fig. 3 presents the cause of shadow. It illustrates a simple scene consisting of one object and one background. Their distances to the sensor are z_o and z_b, respectively. L denotes the laser emitter and I denotes the infrared camera.

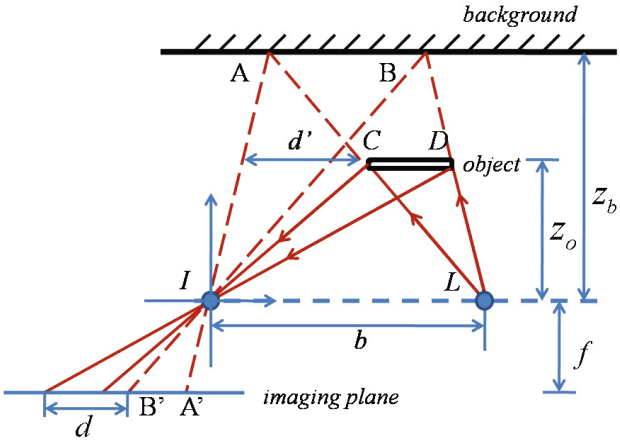


Fig. 3. Cause of Shadow

Suppose a number of speckles are projected onto the scene, as is shown in Fig. 3 by red solid lines. Some of the speckles hit the background directly while some are blocked by the object. We extend the line LC and LD to background and get a region marked as AB. Obviously, region AB is not reached by any speckles. As a result, $A^{'}B^{'}$, its corresponding region on imaging plane, receives no speckles from the scene. In other word, depth in region $A^{'}B^{'}$ is not measured and shadow is formed.

From the projection model above, we conclude that shadow is an area on background where the speckles from the laser emitter cannot reach due to obstruction by an object. In other word, shadow is the projection of object on background. This model also explains why shadow is always presented on the right side of object.

2.2 Shadow Offset

In practice, the distance between object and shadow changes if the object gets close to or draws away from the sensor. This feature differs shadow from the other two NMD regions. We define the distance between object and shadow as the shadow offset. The offset is horizontal because the laser emitter and infrared camera are fixed on a horizontal bar. The length of offset can be measured by the distance between two corresponding edge pixels, as is illustrated in Fig. 4.

The derivation of mathematical expression for shadow offset is as follows. From the similarity of triangles in Fig. 3, we have:

$$\frac{d^{'}}{d} = \frac{z_o}{f} \tag{1}$$

$$\frac{d^{'}}{b} = \frac{z_b - z_o}{z_b} \tag{2}$$

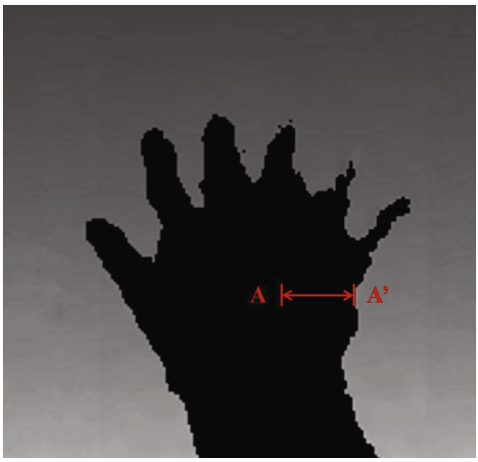


Fig. 4. Offset of Shadow. A indicates the object-edge pixel while $A^{'}$ indicates its corresponding shadow-edge pixel.

Where d represents the shadow offset and $d^{'}$ is an intermediate variable marked in Fig. 3, b denotes the base length between laser emitter and infrared camera, f is the focal length of infrared camera. Substituting $d^{'}$ from Eq.1 into Eq.2 we have Eq.3

$$d = bf\left(\frac{1}{z_o} - \frac{1}{z_b}\right) \tag{3}$$

Eq. 3 is the mathematical expression for shadow offset. It is concerned with object depth and background depth. The parameters b and f in Eq. 3 can be determined from the depth map.

3 Shadow Repair Approach

In this section, we introduce the approach of shadow repair. The proposed approach consists of three procedures: Edge Pixel Searching, Shadow Detection and Shadow Filling. The flow chart is shown in Fig. 5

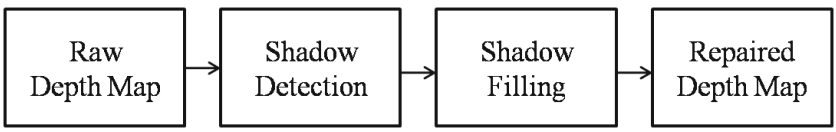


Fig. 5. Flowchart of Shadow Repair

3.1 Shadow Discriminant

Eq. 3 describes the theoretical shadow offset. This offset can be further modified as a shadow discriminant, as is shown in Eq. 4

$$D = \left(Edge_{NMD} - Edge_{object}\right) - bf\left(\frac{1}{Depth_{object}} - \frac{1}{Depth_{background}}\right) \quad (4)$$

Where $Edge_{NMD}$ denotes the edge position of an NMD region while $Edge_{object}$ the edge position of an object. Their difference describes the actual offset between the NMD region and the object region. The rest of Eq. 4 is migrated from Eq. 3, which describes the theoretical offset. Theoretically, the two parts are equal and the discriminant should give a value zero if the NMD region is a shadow. But errors of depth and positions occur frequently in depth map, so the discriminant value may have some slight fluctuations.

3.2 Shadow Detection

To compute the discriminant D, edge pixels are supposed to be searched first. As the shadow offset is horizontal, a horizontal scanning method is applied in edge pixel searching.

As a matter of fact, the position of edge pixel searched may not be accurate if the shadow of one object is covered by another one. This problem occurs when two separate objects are close to each other. To make it clear, we divide the problem of covered shadow into four categories, as is shown in Fig. 6. The white and gray ellipse represent object and shadow respectively.

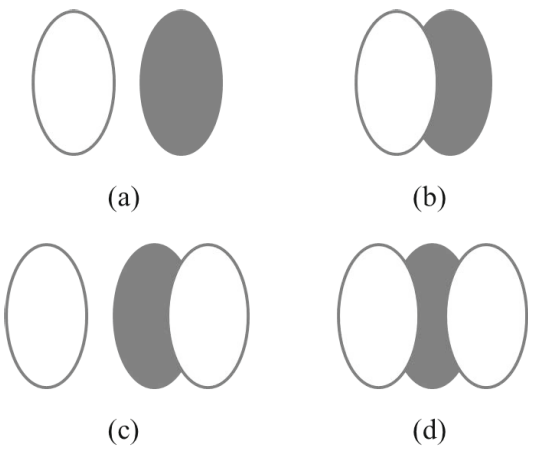


Fig. 6. Problem of Covered Shadow.(a)complete shadow.(b)left part of shadow covered.(c)right part of shadow covered.(d)both sides of shadow covered.

The shadow in Fig. 6(a) is presented complete; In Fig. 6(b), the left part of shadow is covered and only the right edge is available; Fig. 6(c) shows that the right part of shadow is covered and only the left edge is available; In Fig. 6(d), shadow edge of both sides is covered.

The problem of covered shadow affects the accuracy of edge pixel searching. However, we find the cases in Fig. 6(c) and Fig. 6(d) occur only when objects are densely placed in the scene, which is not common in reality. Therefore, these two cases are not supposed to be considered in our method. As for the other two cases, we just need to find out their right edge pixels. In addition, the depth of object and background should be recorded during edge pixel searching.

After the edge pixels in one row is searched out, an exhaustion method is applied to detect the shadow regions. In our method, the shadow regions are detected row by row. Firstly, we put edge pixels of object and NMD regions in the same row into two separate groups. Then all possible pixel pair between the two group are extracted to compute the discriminant above. If the discriminant we compute is less than a threshold we set, the NMD pixels between the pixel pair is supposed to be a shadow. After one row is detected, we clear the two groups and repeat the operation for next row.

3.3 Shadow Filling

Once a shadow region is found out, a depth-filling strategy should be applied to make the depth data more complete. Although the exact depth value of a shadow region can never be acquired, proper estimation could make the data as reliable as possible. In this paper, we assume the background the shadow lies in is flat, thus the depth value of background can be used to fill the shadow region, shown in Eq. 5.

$$Depth_{shadow} = Depth_{background} \tag{5}$$

The depth value of background can be computed as Eq. 6

$$Depth_{background} = \frac{1}{4}\left[\sum_{1\le i\le 6} Depth\,(X+i,Y) - Depth_{max} - Depth_{min}\right] \tag{6}$$

Eq. 6 tries to eliminate the instability of depth map, where X and Y denote the position of a shadow-edge pixel, and $Depth_{max}$ and $Depth_{min}$ denote

$$\begin{aligned} Depth_{max} &= Max\,(Depth\,(X+i,Y))\; 1 \le i \le 6 \\ Depth_{min} &= Min\,(Depth\,(X+i,Y))\; 1 \le i \le 6 \end{aligned} \tag{7}$$

Results in the following section shows our filling strategy is both reliable and smoothing.

4 Experimental Results

In this section, the performance of our method is given. The experiment is carried out concerning two aspects, the shadow detection performance and the time cost.

Three depth video clips generated by Kinect are employed as our test set. These videos record a person with various poses in the scene. To evaluate the shadow detection performance when depth changes, we set the distance(Dist) between the person and sensor in three videos to 1.4m, 1.8m and 2.2m, respectively. Each video consists of 200 depth frames whose resolution is 640×480 and the FPS is 30Hz.

To make a quantitative evaluation, three measurements are proposed to evaluate the performance of shadow detection. Their definitions are as follows.

$$Recall(R) = \frac{DSR}{GSR} \tag{8}$$

$$Precision(P) = \frac{DSR}{DR} \tag{9}$$

$$F = \frac{2 \times R \times P}{R + P} \tag{10}$$

where the abbreviations in Eq.(8)(9)(10) are defined as follows:

- DSR: pixel number of correctly detected shadow regions;
- GSR: pixel number of ground-truth shadow regions;
- DR: pixel number of all detected regions.

among which, GSR is labelled manually.

The Kinect SDK we use is OpenNI [9] and the alignment function for depth map and RGB map is not applied.

The Experimental Environment is listed in Table 1

Table 1. Experimental Environment

Item	Configuration
CPU	Intel Core i5-2400@3.10GHz
RAM	2GB, 1333MHz
OS	Windows 7
IDE	Microsoft Visual Studio 2010

4.1 Experiment on Shadow Detection

Table 2 shows the test result on shadow detection. We find all the three measurements are fairly high, which proves our approach effective. Besides, the three measurements vary slightly as distance changes, so the detection performance is robust against depth changing. However, some regions fail to be detected or error detected. These errors are mostly resulted from the instability of depth value.

Some of the results are shown in Fig. 7, 8, 9. The three rows demonstrate raw depth maps, shadow detection results and the final results, respectively.

Table 2. Test Result on Shadow Detection

Video	Frame Count	Dist (m)	R (%)	P (%)	F
1	200	1.4	95.87	98.39	0.9678
2	200	1.8	95.97	98.21	0.9708
3	200	2.2	96.27	97.29	0.9711
Total	600	-	95.99	98.08	0.9703

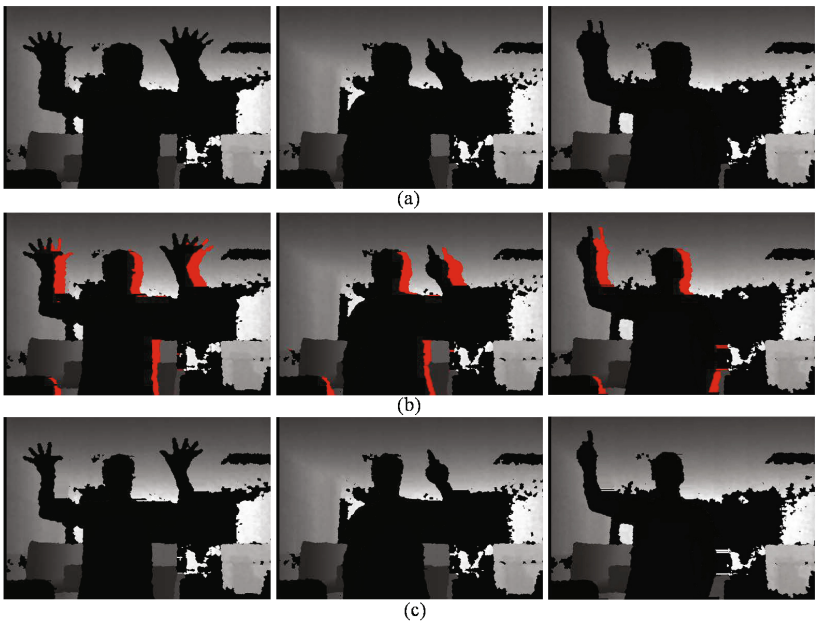


Fig. 7. Results of video 3. (a) the raw depth maps. (b) shadow detection results. (c) the final results.

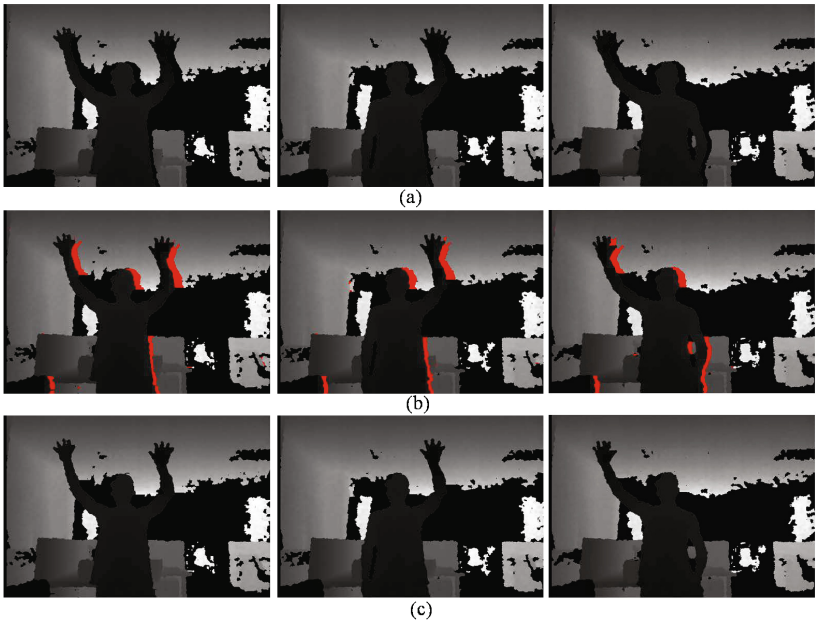


Fig. 8. Results of video 2. (a) the raw depth maps. (b) shadow detection results. (c) the final results.

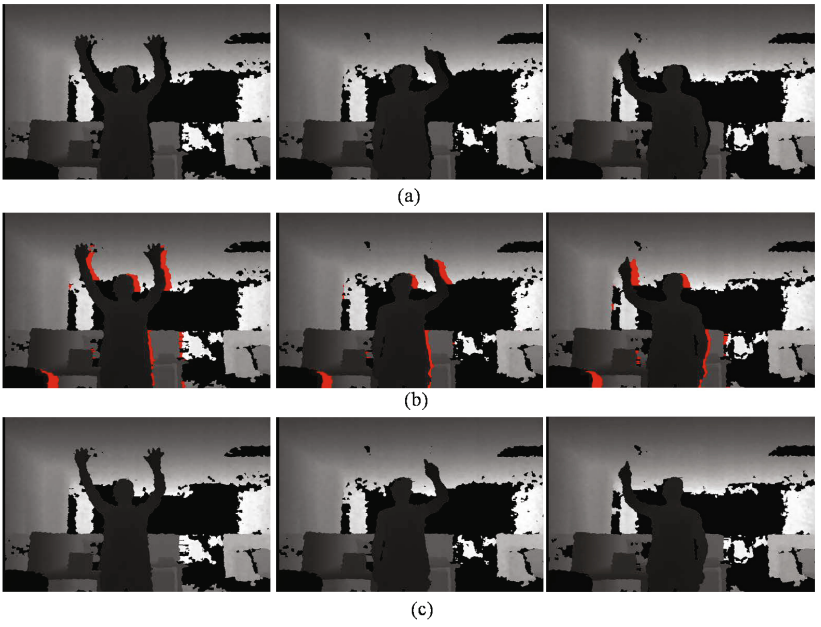


Fig. 9. Results of video 1. (a) the raw depth maps. (b) shadow detection results. (c) the final results.

4.2 Experiment on Time Cost

As is mentioned above, our method of shadow repair is a preprocessing for depth map. As a result, the method should be fast enough to leave some time for following procedures. At least, the time cost per frame should be less than 33ms, the time interval between two frames.

Table 3 shows the Test Result on Time Cost. The average processing time is 4.3298ms, which is fast enough as a preprocessing. In addition, the average time, maximum time and minimum time are close to each other, so the proposed method is fairly stable on time cost.

Table 3. Test Result on Time Cost

Average Time (ms)	Maximum Time (ms)	Minimum Time (ms)
4.3298	5.8433	4.0543

5 Conclusion and Future Works

In this paper, a shadow repair approach is proposed to make the depth data more complete. We firstly define shadow and explain its cause using the principle of Kinect depth measurement. Then a discriminant is designed to distinguish shadow from other NMD regions. Finally, the repair approach is proposed. The approach consists of two procedures. A horizontal scanning method and an exhaustion method are applied to detect shadow regions row by row, then a simple filling strategy is used to remove the regions detected. Results show our method is time saving and reliable.

In the future, two works could be done to improve the proposed method. Firstly, the present filling strategy is fairly rough. Therefore, more information like color from RGB video could be used to estimate the no-measured depth. Secondly, the feature of other noises would also be studied to make the proposed method more comprehensive.

Acknowledgement. We would like to thank for the support from NSF of China (Grand No.90920008).

References

1. Camplani, M., Salgado, L.: Adaptive spatio-temporal filter for low-cost camera depth maps. In: IEEE International Conference on Emerging Signal Processing Applications, pp. 33–36 (2012)
2. Gangwal, O.P., Djapic, B.: Real-time implementation of depth map post-processing for 3D-TV in dedicated hardware. In: International Conference on Consumer Electronics, pp. 173–174 (2010)
3. Mueller, M., Zilly, F., Kauff, P.: Adaptive cross-trilateral depth map filtering. In: 3DTV Conference, pp. 1–4 (2010)

4. Yang, Q., Yang, R., Davis, J., Nister, D.: patial-depth super resolution for range images. In: International Conference on Computer Vision and Pattern Rcognition, pp. 1–8 (2007)
5. Zhu, J., Wang, L., Yang, R., Davis, J.: Fusion of time-of-flight depth and stereo for high accuracy depth maps. In: International Conference on Computer Vision and Pattern Rcognition, pp. 1–8 (2007)
6. Camplani, M., Salgado, L.: Efficient Spatio-Temporal Hole Filling Strategy for Kinect Depth Maps. Proceedings of SPIE 8290 (2012)
7. Matyunin, S., Vatolin, D., Berdnikov, Y., Smirnov, M.: Temporal filtering for depth maps generated by Kinect depth camera. In: 3DTV Conference, pp. 1–4 (2011)
8. Khoshelham, K., Elberink, S.O.: Accuracy and resolution of kinect depth data for indoor mapping applications. Sensors 12, 1437–1454 (2012)
9. OpenNI.: PrimeSense Sensor Module (2011)

A Unified Framework for Line Extraction in Dioptric and Catadioptric Cameras

Jesus Bermudez-Cameo, Gonzalo Lopez-Nicolas, and Jose J. Guerrero

Instituto de Investigación en Ingeniería de Aragón
Universidad de Zaragoza, Spain
{bermudez,gonlopez,jguerrer}@unizar.es

Abstract. Many of the omnidirectional visual systems have revolution symmetry and, consequently, they can be described by the radially symmetric distortion model. Following this projection model, straight lines are projected on curves called line-images. In this paper we present a novel unified framework to deal with these line-images directly on the image which is valid for any central system. In order to validate this framework we have developed a method to extract line-images with a 2-points RANSAC, which makes use of the camera calibration. The proposed method also gives the adjacent regions of line-images which can be used for matching purposes. The line-images extractor has been implemented and tested with simulated and real images.

1 Introduction

Line-images have been extensively used in computer vision. When a projection system is central, the 3D line and the viewpoint (optical center) lies on the same plane $\mathbf{\Pi}$ and the projection is described by a vector normal $\mathbf{n}$ to this plane $\mathbf{\Pi}$. In general, any point $\mathbf{X}$ contained in plane $\mathbf{\Pi}$ is projected on the line-image and satisfies a nice constraint like $\mathbf{n}^\mathsf{T}\mathbf{X} = 0$. In perspective cameras this constraint is transformed to the image plane resulting a 2D line. When the projection system is not perspective the relationship is not linear and the projected line-image is a curve.

Many approaches, e.g [1,2], solve the constraint for collinear points in the unitary sphere. The intersection of the plane $\mathbf{\Pi}$ with the unitary sphere is a great circle which is related with the image using the projection model. Instead of working on the sphere the problem can be tackled directly on the image. This approach has been extensively used in catadioptric images. The catadioptric line projection is modelled by the proposal of Geyer et al. [3]. In this particular case line-images are conics [4,5]. For the case of fisheyes, line-images have not been extensively used.

Most of conventional and non-conventional cameras have revolution symmetry. Even when this constraint is not perfectly satisfied differences with the model can be encapsulated in an additional linear transformation. Main advantage of radially symmetric distortion is that it can be used to model many different devices including perspective cameras, fisheyes and catadioptric systems.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 627–639, 2013.

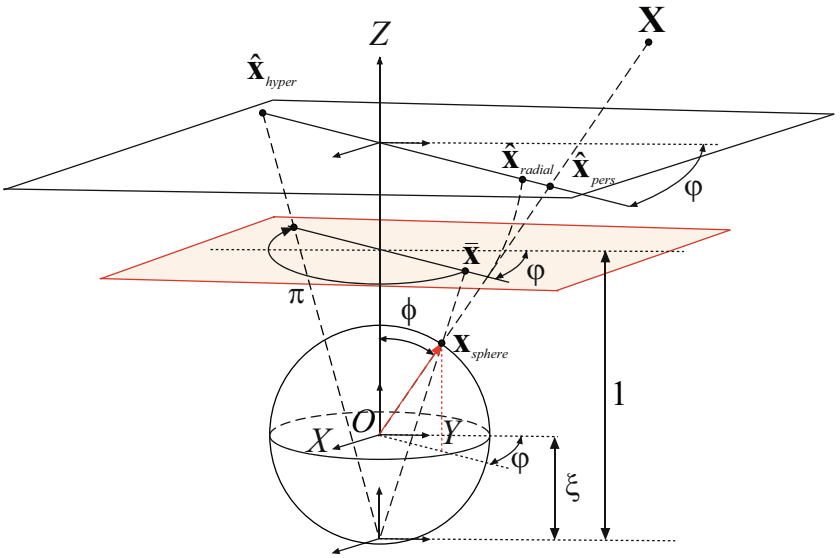


Fig. 1. Catadioptric sphere camera model: The 3D point X is projected onto the sphere. Then this point is backprojected to a normalized plane through a virtual optic center located a distance ξ from the effective viewpoint. This point $\bar{\mathbf{x}}$ is transformed to the image plane using the collineation H_c. Fisheye camera models: The radius of the point on the image is distorted by a function $\hat{r} = h(\phi)$.

As the line projection on the raw image is defined by more than two points it contains information about the calibration and distortion model and can be used to correct the image distortion [6,7,8]. These works are not about the line-image as geometric form, however the line-image is implicitly contained in them. In spite of not being expressed explicitly, the workspace used in Tardif et al. [9] is very related with the space in which a line-image is represented in our proposal.

Once the line-projection model is performed a direct application is line-image extraction. When the direct line projection model is known a Hough transform approach could be used for line-detection. This is the approach used for line-extraction with catadioptric systems in [10,11,12]. In [2] a scheme of split and merge is proposed to extract line-images in catadioptric systems. This approach use the inverse point projection model to back-project the points to the unitary sphere where the robust fitting is done. In [13,14] a line-extraction method for hypercatadioptric systems solving the equation of the conic on the normalized-plane is proposed.

In this paper, we present a framework for line-image extraction in central systems following radially symmetric distortion models. This unified framework is a generalization of the method presented by Bermudez-Cameo et al. in [13]. This generalization expands the results obtained for hypercatadioptric systems to other catadioptric systems and dioptric systems with revolution symmetry.

Explicit analytic expressions have been obtained for paracatadioptric, equiangular-fisheye, stereographic-fisheye and orthogonal-fisheye models. We show an expression for the homogeneous line-image equation which is coherent with radially symmetric distortion models. The image-space in which the line-image is represented is similar to the space used in [9] for self-calibration. Main difference with this work is that we focused on the line-image and that in our proposal the distortion function is analytically solved for each projection model instead of having an empirical solution. In general our proposal is analytically solved when the inverse point projection model exists. The line-image homogeneous equation defines an algebraic distance measured in pixels which approximates the distance from any point of the image to the projected curve. We define a new robust method to extract line-images using this expression valid for catadioptric systems and fisheyes. We show the behaviour of the line-images and compare this model with the extension of the catadioptric sphere model for fisheyes presented in [15]. The extraction method is used to obtain the adjacent regions of image segments.

The rest of the paper is organized as follows. In Section 2 we describe the catadioptric sphere model and the fisheye projection models. In Section 3 we present a unified description to represent line-images in revolution symmetry systems. In Section 4 we show the line-extraction method. In Section 5 we test the line extraction method for simulated and real images. Finally we present the conclusions.

2 Projection Models for Central Systems with Revolution Symmetry

When a projection system conserves symmetry around an axis it could be described using cylindrical coordinates. If the system is central the projected rays lie on a common point called fixed viewpoint $\mathbf{O}$. In this case, both constrains are well represented by the spherical coordinate system. Let $\mathbf{X}$ be a 3D point in homogeneous coordinates $\mathbf{X} = (X\ Y\ Z\ 1)^{\mathsf{T}}$. This point is transformed to the reference system of the camera in which the origin is the fixed viewpoint $\mathbf{O}$ of the system and the Z-axis is aligned with the axis of revolution. This transformation consists of a rotation R and a translation $\mathbf{t}$, therefore the projection matrix is $\mathsf{P} = (\mathsf{R}|\mathbf{t})$. The point is projected onto a unitary sphere around the viewpoint $\mathbf{O}$ of the system. It is defined with two angular coordinates ϕ and φ as,

$$\mathbf{x} = (\sin\phi\cos\varphi, \sin\phi\sin\varphi, \cos\phi)^{\mathsf{T}} \ . \tag{1}$$

Depending on the projection model this point is mapped on the image using different expressions. Notice that any point lying on the revolution axis is projected on an image point called principal point. If the camera is correctly aligned with the axis of revolution we can observe that the coordinate θ of a polar system in the image centred in the principal point, is related with the spherical coordinate φ via the pixel aspect ratio k_{par}, as $\tan\theta = \pm k_{par}\tan\varphi$[1].

[1] The sign in this expression is used to model reflections in catadioptric systems.

Catadioptric and dioptric systems are projection systems which conserve the revolution symmetry. Many projection models are used to model this devices. In the following descriptions we assume that image points are expressed in a reference centred in the principal point. We also assume that pixel aspect ratio is equal to one which is valid in digital imagery. A point in this reference system is denominated with the notation $\hat{\mathbf{x}}$. The transformation from this reference to the final image coordinate system is the following,

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & s & u_0 \\ 0 & k_{par} & v_0 \\ 0 & 0 & 1 \end{pmatrix} \hat{\mathbf{x}} \ . \tag{2}$$

2.1 Projection Models for Catadioptric Systems

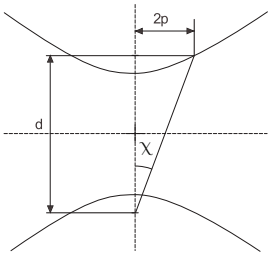
Under the sphere camera model [3] all central catadioptric systems can be modelled by a projection to the unitary sphere followed by a perspective projection via a virtual viewpoint located a distance ξ from the effective viewpoint (see Fig. 1). Let $\hat{\mathbf{x}} = (\hat{x}, \hat{y}, 1)^T$ be a point on an image referenced to the principal point and given the spherical coordinates ϕ and φ of the corresponding point on the unitary sphere then,

$$\hat{x} = \frac{f\eta \sin\phi \cos\varphi}{\cos\phi + \xi} \quad \text{and} \quad \hat{y} = -\frac{f\eta \sin\phi \sin\varphi}{\cos\phi + \xi} \ . \tag{3}$$

In polar coordinates the point is described by $\hat{\theta} = -\varphi$ and

$$\hat{r} = \frac{f\eta \sin\phi}{\cos\phi + \xi} = \frac{f\eta \tan\phi}{1 + \xi\sqrt{\tan^2\phi + 1}} \ . \tag{4}$$

The geometry of the projection system is described by parameters ξ and η which have a different definition depending on the system. In particular, when using hypercatadioptric systems the mirror parameters ξ and η are related via a single parameter χ which is related with the semi-latus rectum of the generational hyperbola and the distance between foci (see Fig. 2).



	ξ	η
Parabolic:	1	$2p$
Hyperbolic:	$cos\chi = \frac{d}{\sqrt{d^2+4p^2}}$	$sin\chi = \frac{2p}{\sqrt{d^2+4p^2}}$
Planar:	0	1

d: distance between focus
$4p$: latus rectum

Fig. 2. Parameters of the unified sphere model for catadioptric systems

2.2 Fisheye Models

Several models are used to describe point projection in dioptric systems depending on the manufacturing procedure of the lens [16,17,18]. Assuming square pixel, these models are expressed in polar coordinates $(\hat{r}, \hat{\theta})$. For all these models $\hat{\theta} = \varphi$ and the radius changes depending on the camera type (see Table 1).

Table 1. Fish-eye projection models

Equiangular-Fisheye	Stereographic-Fisheye	Orthogonal-Fisheye
$\hat{r} = f\phi$	$\hat{r} = 2f \tan\left(\frac{\phi}{2}\right)$	$\hat{r} = f \sin(\phi)$

Some authors have used the catadioptric sphere model to calibrate fisheye models [15]. In the case of the stereographic projection both models are equivalent when $\xi = 1$ and $\eta = 2$. For other cases it is assumed that $\xi > 1$. As we will show in the following sections, the catadioptric sphere model and the rest of the fisheye models are not equivalent and it is only a good approximation when the field of view (FOV) is less than 180 degrees.

3 Unified Description for Line Projection in Central Systems with Revolution Symmetry

Let $\mathbf{\Pi} = (n_x, n_y, n_z, 0)^\mathsf{T}$ be a plane defined by a 3D line and the viewpoint of the system $\mathbf{O}$. The projected line associated to the 3D line can be represented by $\mathbf{n} = (n_x, n_y, n_z)^\mathsf{T}$. Then, the points $\mathbf{X}$ lying in the 3D line are projected to points $\mathbf{x}$. These points satisfy $\mathbf{n}^\mathsf{T}\mathbf{x} = 0$. Using the spherical representation (1) and assuming that $\hat{\theta} = \pm\varphi$ (square pixel) this equality could be expressed as

$$\sin\phi\,(n_x\hat{x} \pm n_y\hat{y}) + n_z\hat{r}\cos\phi = 0\ . \tag{5}$$

With the change of variable $\hat{\alpha} = \dfrac{n_x\hat{x} \pm n_y\hat{y}}{n_z}$ we can isolate the model parameters from the normal describing the line, obtaining the expression,

$$\hat{\alpha} = -\hat{r}\cot\phi\ . \tag{6}$$

Notice that $\hat{\alpha} = \hat{\alpha}(\hat{r})$, as a result of $\phi = h^{-1}(\hat{r})$ when we have symmetry of revolution and square pixel. Therefore, the constraint for points on the line projection in image coordinates for systems with symmetry of revolution is

$$n_x\hat{x} \pm n_y\hat{y} - n_z\hat{\alpha}(\hat{r}) = 0\ , \tag{7}$$

where $\hat{\alpha}$ is a different expression for each camera model depending on the radius and the model parameters (see Table 2).

Table 2. $\hat{\alpha}$ depending on the projection model

Perspective	Para Catadioptric	Hyper Catadioptric	Equiangular Fisheye	Stereographic Fisheye	Orthogonal Fisheye
$-f$	$\frac{\hat{r}^2}{4fp} - fp$	$\frac{-f+\cos\chi\sqrt{\hat{r}^2+f^2}}{\sin\chi}$	$-\hat{r}\cot\frac{\hat{r}}{f}$	$\frac{\hat{r}^2}{4f} - f$	$-\sqrt{f^2-\hat{r}^2}$

3.1 Line-Image Curve Representation

Equation (7) is the homogeneous representation of the line projection on the image. There exist two particular cases common to all the projection models showed above. First we have the case in which 3D lines are coplanar to the revolution axis. In this case $n_z = 0$ and the resulting line-image is a radial straight line passing through the principal point.

$$n_x\hat{x} \pm n_y\hat{y} = 0 \ . \tag{8}$$

The second particular case happens when $\mathbf{n} = (0, 0, 1)^\mathsf{T}$. In this case the line-image is the projection of the vanishing line. This projection is a circle centred at principal point and with radius $\hat{r}_{VL}$. This radius depends on the calibration and differs with the projection model (see Table 3). The line-image equation in this case has the form,

$$\hat{\alpha}\left(\hat{r}\right) = 0 \ . \tag{9}$$

Table 3. $\hat{r}_{VL}$ for different projection models

Perspective	Para Catadioptric	Hyper Catadioptric	Equiangular Fisheye	Stereographic Fisheye	Orthogonal Fisheye
∞	$2fp$	$f\tan\chi$	$f\frac{\pi}{2}$	$2f$	f

The general form for a line-image is a curve. The catadioptric case has been deeply studied in [5], and it has been proven that the line-image is a conic. The stereographic case could be expressed directly in terms of a catadioptric projection. The orthographic line-image is also a conic but not in terms of a catadioptric projection. For the other cases in general the curve is not a conic.

In Fig. 3, we show a parametric representation of line-images depending on the elevation angle of the normal $\mathbf{n}$ describing the projection plane of a 3D line. Each image has been simulated for a different device but with the same $\hat{r}_{VL}$. In all the cases line-images are well approximated by conics when the points of the segment are inside the limits of the vanishing line projection (FOV lower than 180 degrees)[19]. However in the case of equiangular and orthogonal fisheye (b)(d) the line-images are not well fitted by conics when we are in regions of the

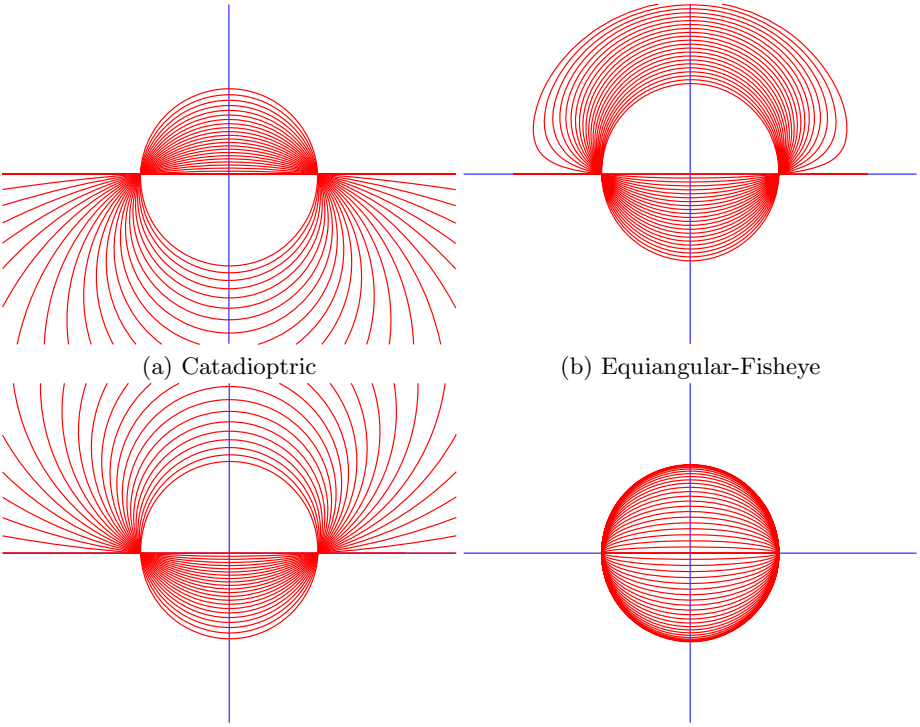


Fig. 3. Representation of line-images on the image plane depending on the projection model and with different values of the elevation of the normal **n**

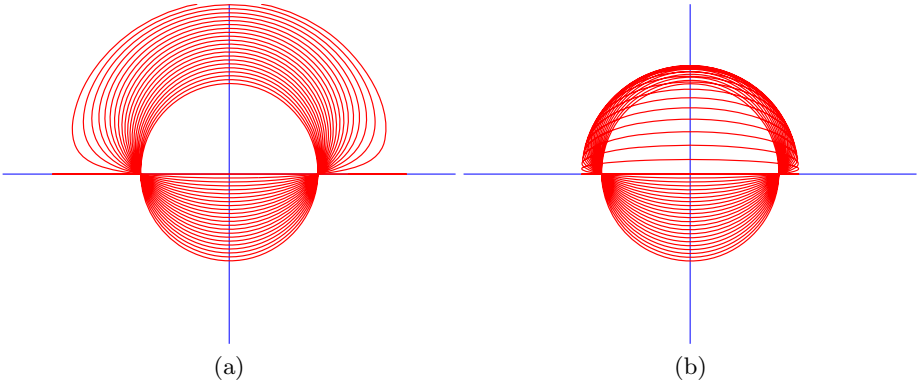


Fig. 4. Comparison of line-images on the image plane using the equiangular-fisheye projection model and the catadioptric sphere model with $\xi > 1$. Each line-image corresponds to a different value of the elevation of the normal **n**.

image corresponding to FOV greater than 180 degrees. In Fig. 4 we show a comparison of the line-images in an equiangular fisheye and an approximation using the catadioptric sphere model [15]. Notice that the line-image is well fitted inside the vanishing line projection but not outside. We also show how each pair of conics intersects in four points instead of two giving a sense of non-geometric coherence.

3.2 The Line-Image Homogeneous Equation as a Measure of Distance

The homogeneous expression of the line-image (7) defines a family of curves located to an algebraic distance from the original curve.

$$d(\hat{x}, \hat{y}) = n_x \hat{x} \pm n_y \hat{y} - n_z \hat{\alpha} \ . \tag{10}$$

This algebraic distance is an approximation of the metric distance from a point to the line-image and is defined in pixels. When using an algebraic distance based on conics (e.g for hypercatadioptric systems $d = \sqrt{\mathbf{x}^\mathsf{T} \Omega_{hyper} \mathbf{x}}$) is known that given a fixed threshold the region around the conic have a different thickness depending on the elevation angle of the vector $\mathbf{n}$. With our proposal the distance is a good approximation in regions close to the line-image.

In Fig. 5 (a) we show a comparison between the minimum distance of a point to the line-image (blue dotted) and the proposed algebraic distance (red) for hypercatadioptric images. The algebraic distance approximates the real distance in regions which are close to the line-image, therefore can be used to discriminate if a point lies on a line-image or not. In Fig. 5 (b) we show the same comparison but using the algebraic distance defined by the expression of a conic on the image ($d = \sqrt{\mathbf{x}^\mathsf{T} \Omega_{hyper} \mathbf{x}}$). We can see how this distance does not approximate well the metric distance in regions close to the curve. We also show that this distance is lower than the metric distance in vertical lines but higher when the lines are horizontal. In practice that means that the thickness of a region defined by a threshold varies considerably if elevation of $\mathbf{n}$ changes.

Therefore we conclude that the proposed algebraic distance (10) is useful to discriminate if a point belongs to a line-image in catadioptric systems. However, we have observed that in orthogonal systems the defined region is not constant. In this case it is necessary to use additional criteria to determine if a point lies on a conic or not (this will be detailed in Section 4.2).

4 Two Points RANSAC for Image Fitting

In this section we present a generalization of the method presented by Bermudez-Cameo et al. in [13] to fit line-images in central projection systems with revolution symmetry. First, we show how to define a line-image using two points and the calibration of the system. Then we describe the computation of the gradient used in RANSAC and how to robustly fit the line-image.

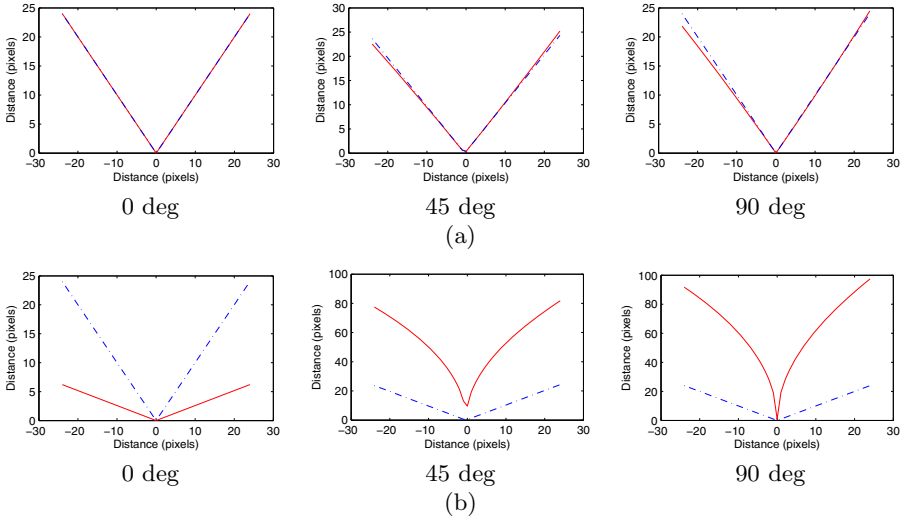


Fig. 5. Comparison between metric distance (blue dotted) and algebraic distances (red solid): (a) Our proposal (10). (b) Conic based algebraic distance.

4.1 Line-Image Definition with Two Points

Having a collection of at least two points lying on a line-image we can build an homogeneous linear system using (7). The solution of this linear system is the normal $\mathbf{n}$ describing the projection plane of a 3D line:[2]

$$M \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} = \begin{pmatrix} \hat{x}_1 & \pm\hat{y}_1 & -\hat{\alpha}_1 \\ \hat{x}_2 & \pm\hat{y}_2 & -\hat{\alpha}_2 \\ \vdots & \vdots & \vdots \\ \hat{x}_n & \pm\hat{y}_n & -\hat{\alpha}_n \end{pmatrix} \begin{pmatrix} n_x \\ n_y \\ n_z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} . \tag{11}$$

Depending on the device type, the way to compute the variable $\hat{\alpha}$ differs (see Table 2). The system is solved using a Singular Values Decomposition (SVD). In particular with two points and solving for n_x, n_y and n_z we have

$$n_x = \hat{y}_1\hat{\alpha}_2 - \hat{y}_2\hat{\alpha}_1 \ , \quad n_y = \pm\left(\hat{x}_2\hat{\alpha}_1 - \hat{x}_1\hat{\alpha}_2\right) \quad \text{and} \quad n_z = \hat{x}_2\hat{y}_1 - \hat{x}_1\hat{y}_2 \ . \tag{12}$$

In contrast with [13], here points are defined in the image plane instead of the normalized-plane[3]. Therefore, in our proposal the residual vector $\delta = \mathsf{M}\mathbf{n}$ is measured in pixel units.

[2] The sign when '$\pm$' in the following equations is positive for dioptric systems and negative for catadioptrics.

[3] Points are referenced to the principal point. The normalized-plane is an intermediate projection plane described in the sphere-model (see Fig. 1).

4.2 Gradient of the Line-Image Curve

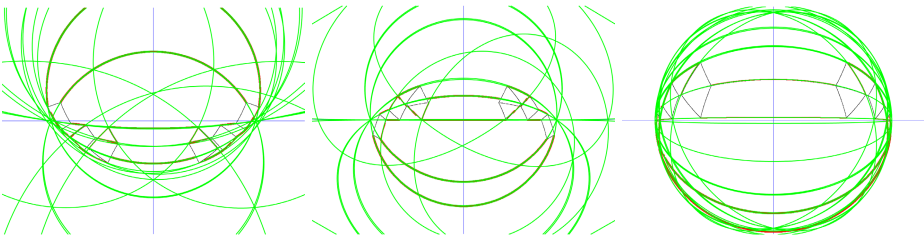
The gradient of the algebraic distance (10) is a vector perpendicular to the line-image in each point of the curve.

$$\frac{\partial d}{\partial \hat{x}} = n_x - n_z \frac{\partial \hat{\alpha}}{\partial \hat{r}} \frac{\hat{x}}{\hat{r}} \qquad \frac{\partial d}{\partial \hat{y}} = \pm n_y - n_z \frac{\partial \hat{\alpha}}{\partial \hat{r}} \frac{\hat{y}}{\hat{r}} \ . \tag{13}$$

Table 4. $\frac{\partial \hat{\alpha}}{\partial \hat{r}} \frac{1}{\hat{r}}$ used in Gradient Computing

Perspective	Para Catadioptric	Hyper Catadioptric	Equiangular Fisheye	Stereographic Fisheye	Orthogonal Fisheye
0	$\frac{1}{2fp}$	$\frac{\cot\chi}{\sqrt{\hat{r}^2+f^2}}$	$\frac{1}{f}\left(1-\frac{f}{\hat{r}}\cot\frac{\hat{r}}{f}+\cot^2\frac{\hat{r}}{f}\right)$	$\frac{1}{2f}$	$\frac{1}{\sqrt{\hat{r}^2-f^2}}$

The gradient of the line-image is used for several purposes. One of them is to define a more accurate threshold for the algebraic distance criterion. Having the two defining points of the curve and the gradient in each point we compute the coordinates of a point located to a metric distance from the curve. Then we compute the algebraic distance d of this point using the expression (10). As d is monotone the obtained distance could be used as threshold for the algebraic distance. The analytical gradient of the line-image is also used as additional criterion in the voting process. Having the orientation of the gradient in each point of the curve we can compute the angular distance between this and the gradient obtained in the Canny edge detection. Finally, the gradient is used to extract adjacent regions around the fitted segment of the curve.



(a) (b) (c)

Fig. 6. Line-image extraction example on simulated images: (a) Hypercatadioptric System. (b) Fisheye Equiangular. (c) Fisheye Orthogonal.

4.3 Robust Extraction on the Image

Line-image extraction can be explained as follows. First we detect the edges using the Canny algorithm and stored them in connected components. From the Canny algorithm we also obtain the gradient of each pixel of the image.

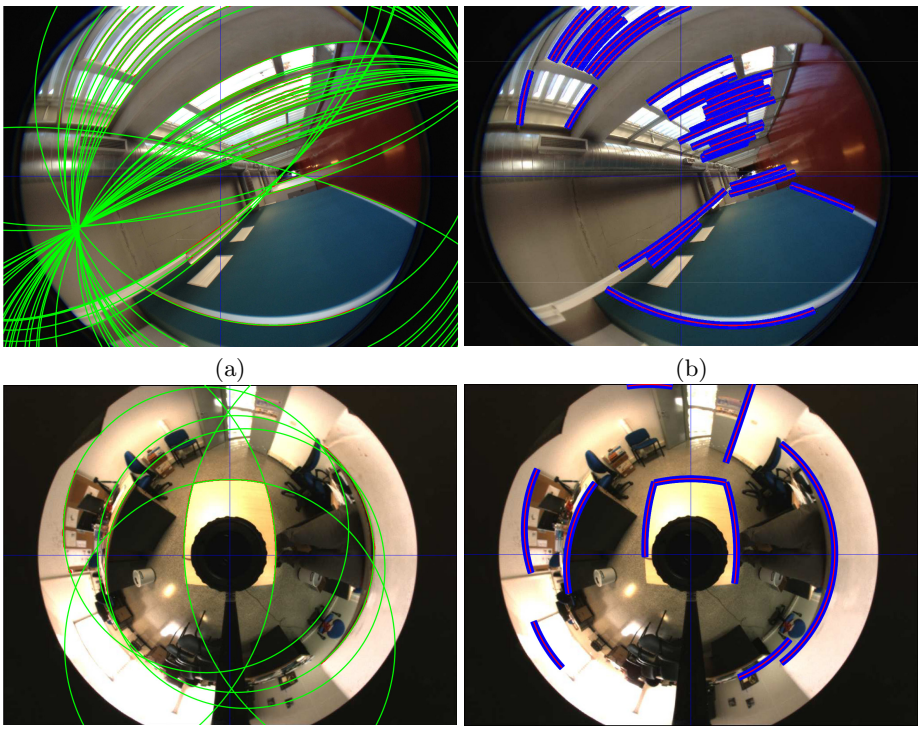


Fig. 7. (a) Line-image extraction example on real dioptric image. (b) Adjacent region extraction example on dioptric image. (c) Line-image extraction example on real hypercatadioptric image. (d) Adjacent region extraction on hypercatadioptric image.

For each component we launch a RANSAC algorithm to robustly extract line-images. Two points of the connected component are selected randomly. With these two points a line-image is computed using the two-points line-image definition presented above (Section 4.1). Two distances to the curve are computed from the rest of points of the connected component. First distance is algebraic distance shown in Section 3.2. The second distance is an angular distance between the gradient in each point computed from the line-image (Section 4.2) and the gradient computed by image processing in the Canny edge detection. Points with both distances smaller than a threshold vote for this line-image. The candidate which collects more votes is selected as the best fit. Notice that

this proposal assumes that a component contains at least a line-image. When a component is the projection of another shape (e.g a circle on a planar surface) the algorithm does not fit the whole boundary. Instead of that, the algorithm extracts the line-image which better fits the given component. Once the line is fitted we extract the adjacent region surrounding the curve. Given a region thickness, the analytical gradient of each point of the segment is used to obtain the coordinates of the region. This image regions can be used for computing local-descriptors in order to perform a line-matching approach.

5 Experiments

We have tested the line extraction method using synthetic and real images. The synthetic images have been generated via Matlab simulation. From each normal vector $\mathbf{n}$ we compute the points of the intersection between the plane $\mathbf{\Pi}$ and the sphere. Then points are projected using the corresponding projection model. We have generated images for hypercatadioptric, equiangular an orthographic fisheye systems with a resolution of 1024x768 pixels. In Fig. 6(a-c) we show the line-images extracted for three simulated images in hypercatadioptric, equiangular and orthogonal systems. We show how the shape of the extracted curves are quite different depending on the projection model. Two different omnidirectional systems have been used to acquire the real images. The real dioptric images have been taken with an iPhone 4S camera with a commercial equiangular fisheye[4] with a resolution of 3264x2498 pixels. The real hypercatadioptric images have been acquired with a firewire camera with an hyperbolic mirror and a resolution of 1024x768 pixels. In Fig. 7(a) we show the behaviour of the line-image extractor with a real equiangular image. In Fig. 7(b) we show the same image in which segments and its adjacent regions have been extracted. This test has been repeated for a hypercatadioptric image in Fig. 7(c-d).

6 Conclusions

We have presented a framework to deal with line-projection in any radially symmetric central projection system. This framework allows to perform line algorithms valid for different classes of omnidirectional systems. Working on the image allows us to extract the adjoining regions surrounding each line-image. This image-regions can be used to compute region-based local descriptors in omnidirectional images. Notice that the extracted regions conserve invariance to orientation. Experimental results have been showed for catadioptric, equiangular-fisheye and orthogonal-fisheye models and the framework can be easily extended to other projection systems if they can expressed in the form $\phi = h^{-1}(\hat{r})$.

Acknowledgement. This work was supported by the Spanish projects VISPA DPI2009-14664-C02-01, VINEA DPI2012-31781, DGA-FSE(T04) and FEDER funds. First author was supported by the FPU program AP2010-3849.

[4] `http://www.pixeet.com/fisheye-lens.`

References

1. Ying, X., Hu, Z., Zha, H.: Fisheye Lenses Calibration Using Straight-Line Spherical Perspective Projection Constraint. In: Narayanan, P.J., Nayar, S.K., Shum, H.-Y. (eds.) ACCV 2006 Part II. LNCS, vol. 3852, pp. 61–70. Springer, Heidelberg (2006)
2. Bazin, J.C., Demonceaux, C., Vasseur, P.: Fast Central Catadioptric Line Extraction. In: Martí, J., Benedí, J.M., Mendonça, A.M., Serrat, J. (eds.) IbPRIA 2007 Part II. LNCS, vol. 4478, pp. 25–32. Springer, Heidelberg (2007)
3. Geyer, C., Daniilidis, K.: A Unifying Theory for Central Panoramic Systems and Practical Implications. In: Vernon, D. (ed.) ECCV 2000 Part II. LNCS, vol. 1843, pp. 445–461. Springer, Heidelberg (2000)
4. Geyer, C., Daniilidis, K.: Catadioptric projective geometry. International Journal of Computer Vision 45, 223–243 (2001)
5. Barreto, J.P., Araujo, H.: Geometric properties of central catadioptric line images and their application in calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence 27, 1327–1333 (2005)
6. Devernay, F., Faugeras, O.: Straight lines have to be straight. Machine Vision and Applications 13, 14–24 (2001)
7. Alvarez, L., Gómez, L., Sendra, J.: An algebraic approach to lens distortion by line rectification. Journal of Mathematical Imaging and Vision 35, 36–50 (2009)
8. Brown, D.: Close-range camera calibration. Photogrammetric engineering 37, 855–866 (1971)
9. Tardif, J.-P., Sturm, P., Roy, S.: Self-calibration of a General Radially Symmetric Distortion Model. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006 Part V. LNCS, vol. 3954, pp. 186–199. Springer, Heidelberg (2006)
10. Vasseur, P., Mouaddib, E.M.: Central catadioptric line detection. In: British Machine Vision Conference (2004)
11. Ying, X., Hu, Z.: Catadioptric line features detection using hough transform. In: 17th International Conference on Pattern Recognition, ICPR, vol. 4, pp. 839–842 (2004)
12. Mei, C., Malis, E.: Fast central catadioptric line extraction, estimation, tracking and structure from motion. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pp. 4774–4779 (2006)
13. Bermudez-Cameo, J., Puig, L., Guerrero, J.J.: Hypercatadioptric line images for 3D orientation and image rectification. Robotics and Autonomous Systems 60(6), 755–768 (2012)
14. Puig, L., Bermudez, J., Guerrero, J.J.: Self-orientation of a hand-held catadioptric system in man-made environments. In: IEEE International Conference on Robotics and Automation, ICRA, pp. 2549–2555 (2010)
15. Courbon, J., Mezouar, Y., Eck, L., Martinet, P.: A generic fisheye camera model for robotic applications. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, pp. 1683–1688 (2007)
16. Kingslake, R.: A history of the photographic lens. Academic Press (1989)
17. Stevenson, D., Fleck, M.: Nonparametric correction of distortion. In: 3rd IEEE Workshop on Applications of Computer Vision, pp. 214–219 (1996)
18. Ray, S.: Applied photographic optics: Lenses and optical systems for photography, film, video, electronic and digital imaging. Focal Press (2002)
19. Sturm, P., Ramalingam, S., Tardif, J., Gasparini, S., Barreto, J.: Camera models and fundamental concepts used in geometric computer vision. Foundations and Trends in Computer Graphics and Vision 6(1-2), 1–183 (2011)

Fusion of Time-of-Flight and Stereo for Disambiguation of Depth Measurements

Ouk Choi and Seungkyu Lee

Samsung Advanced Institute of Technology, Republic of Korea

Abstract. The complementary nature of time-of-flight and stereo has led to their fusion systems, providing high quality depth maps robustly against depth bias and random noise of the time-of-flight camera as well as the lack of scene texture. This paper shows that the fusion system is also effective for disambiguating time-of-flight depth measurements caused by phase wrapping, which records depth values that are much less than their actual values if the scene points are farther than a certain maximum range. To recover the unwrapped depth map, we build a Markov random field based on a constraint that an accurately unwrapped depth value should minimize the dissimilarity between its projections on the stereo images. The unwrapped depth map is then adapted to stereo matching, reducing the matching ambiguity and enhancing the depth quality in textureless regions. Through experiments we show that the proposed method extends the range use of the time-of-flight camera, delivering unambiguous depth maps of real scenes.

1 Introduction

Time-of-Flight (ToF) cameras provide real-time depth measurements robustly against the lack of scene texture. For this reason, ToF cameras have been applied to various applications such as 3D object tracking and modeling.

In order to extend the applicability of ToF cameras towards high-quality 3D modeling, there are several drawbacks to be improved. Commercial ToF cameras [1,2] do not usually provide any color information, which is indispensable for visually pleasing 3D models. In addition, to our best knowledge, the highest resolution of the cameras is as low as about 320×240 pixels [2]. The cameras also suffer from depth errors originating from various sources such as depth bias [3], random noise [4], and the aliasing effect due to phase wrapping [5,6,7,8,9,10] that records depth values that are much less than their actual values if the scene points are farther than a certain maximum range.

Fusion systems [11,12,13,14,15,16,17,18,19,20,21,22] combining ToF and color cameras naturally improve many of the aforementioned drawbacks since color cameras provide color information, and their resolution is usually higher than that of ToF cameras. Thus even the minimal set-up combining a single ToF and a single color camera [13,22] provides high resolution depth maps with reduced noise.

K.M. Lee et al. (Eds.): ACCV 2012, Part IV, LNCS 7727, pp. 640–653, 2013.

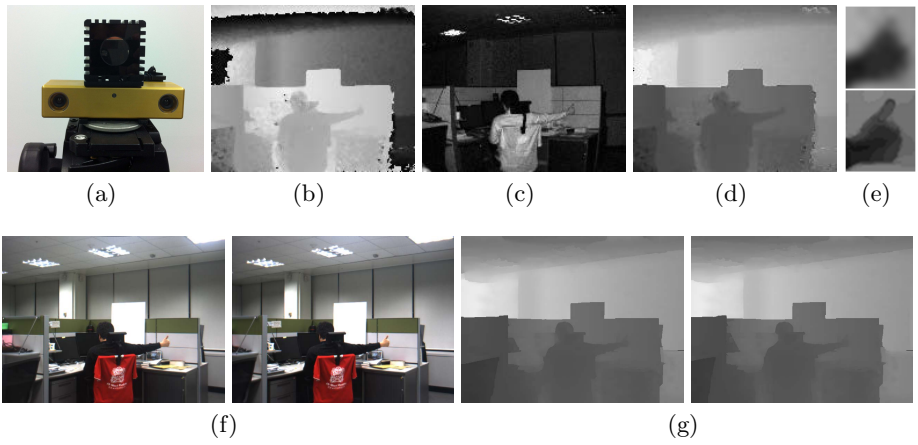


Fig. 1. Input and output of the proposed method. (a) Fusion system. (b) ToF depth map and (c) its amplitude image. The depth map suffers from phase wrapping, and shows wrapping boundaries with high contrast. (d) Unwrapped depth map. (e) Regions (hands) extracted from (d) and (g): upper from (d) and lower from (g). The intensity has been linearly adjusted to show the difference. (f) Stereo images. (g) Stereo depth maps. The intensity of the depth maps are proportional to the Z value. The maximum intensity (255 in gray level) corresponds to 5m in (b) and 10m in (d) and (g).

Among the fusion systems, the combination of a single ToF and a stereo camera [11,12,14,15,16,17,19,20,21] is of special interest due to their complementary nature: Stereo vision hardly delivers reliable depth information on textureless regions such as a white wall, on which ToF cameras give reliable depth measurements. On the other hand, ToF cameras hardly provide accurate depth measurements on objects with low infrared reflectivity, on which stereo vision give accurate depth information as long as distinctive scene texture is present.

For statistically optimal fusion of ToF depth and stereo data, Beder et al. [12] estimate planar surface patches using a weighted least square technique based on the variances of image noise of stereo pixels and the distance uncertainty of ToF pixels. More recent approaches [14,15,19,20] model the fusion problem using Markov random fields, and use global optimization techniques. The core principle in [14,15] is that a ToF camera provides reliable initial depth estimates, which are refined by using the stereo data, whereas the reliability of each data is modeled in [19,20], enhancing the robustness to unreliable ToF depth data.

To our best knowledge, none of the previous fusion approaches [11,12,13,14,15,16,17,18,19,20,21] handles with phase wrapping [5,6,7,8,9,10]. In scaled environments, the confidence of ToF depth data is not accurately measured by the amplitude of the received infrared light signal due to phase wrapping: The accuracy of depth values rapidly changes across the wrapping boundaries while the amplitude values do not, as shown in Fig. 1(b) and (c). Thus the approaches [17,20] that determine the ToF depth confidence using the amplitude value are prone to errors in scaled environments.

On the other hand, the fusion system has an innate ability to resolve phase wrapping: An accurate ToF depth value should minimize the dissimilarity between its projections on the stereo images as long as the scene point is visible from all the cameras. Based on this fact, we build a Markov random field to find the unwrapped depth values robustly against noise and occlusion. The unwrapped depth map is then up-sampled to provide an additional evidence for stereo matching, reducing the matching ambiguity in textureless regions. Finally, the stereo depth map is refined by using the up-sampled ToF depth map, and structures on textureless surfaces are recovered.

The recently developed Kinect sensor [23] provides a color image and its corresponding high-resolution depth map with 640×480 pixels at a video rate. Although the depth map suffers from time-varying errors and contains many holes, the sensor has its potential to provide high quality depth maps [24]. The working range of the Kinect sensor is, however, limited from 0.8m to 4m [23].

The paper is organized as follows. The remainder of this section is devoted to a brief introduction on phase wrapping of ToF cameras. Section 2 presents the proposed method for fusion of ToF and stereo for disambiguating depth measurements. Section 3 demonstrates the effectiveness of the proposed method through experiments on real scenes. Finally, Section 4 concludes the paper.

1.1 Phase Wrapping

A ToF camera measures the distance to scene surfaces by calculating the traveling time of a modulated infrared light signal, which is continuously emitted from the camera and reflected from the surfaces. The reflected light signal is detected at each pixel and converted to electric charges that are mixed with internal signals with known phases [5]. The time-of-flight is proportional to the phase ϕ of the detected signal, which is calculated using the mixed signals that are periodic functions of ϕ with a period of 2π. The distance r is then given by $\frac{c\phi}{4\pi f}$, where c is the speed of light and f is the modulation frequency.

Since $\phi + 2n\pi$ gives exactly the same mixed signals with ϕ, they are ambiguous with each other for a nonnegative integer n. For this reason, each modulation frequency f has its maximum range $r_{\max} = \frac{c}{2f}$ encoded without ambiguity. For any scene points farther than $r_{\max}$, the measured distance r has a modular error, and the unknown number of wrappings n needs to be estimated to recover the actual distance $r + nr_{\max}$.

There exist phase unwrapping approaches for estimating the unknown number of wrappings [6,7,8,9,10] of ToF depth maps. For the approaches to produce good results, two or more ToF depth maps should be successively acquired at different modulation frequencies [8], resulting in temporal differences between the depth maps, which makes the approaches hardly deal with moving cameras and objects. Recent development of a hardware system [10] enables simultaneous acquisition of a pair of ToF depth maps at two different modulation frequencies within a single shot. The acquisition time should be, however, longer than that of a single ToF depth map of the same quality, and such a hardware system has not yet been equipped in commercial ToF cameras. Since the proposed method uses a single

ToF depth map and a pair of color images that are simultaneously acquired, it has its full potential to deal with moving cameras and objects.

2 Proposed Method

Our fusion system consists of commercial ToF and stereo cameras, as shown in Fig. 1(a). The ToF camera, SR4000 [1] delivers the 3D coordinates $\mathbf{X}^{ToF}$ of a scene point at each pixel, which is calculated from the distance $r = \|\mathbf{X}^{ToF}\|$ and the intrinsic camera parameters. The camera also simultaneously provides an amplitude image of the detected infrared light signal. The stereo camera, Bumblebee2 [25] acquires a pair of left and right color images, and provides their rectified image pair and the intrinsic and extrinsic camera parameters. Fig. 1 shows sample images acquired by the fusion system, where the ToF depth and stereo image resolutions are 176×144 and 640×480 pixels, respectively. In contrast, the fields-of-view of the cameras are almost identical.

Given a single ToF depth map and a pair of rectified stereo images, the proposed method returns an unwrapped ToF depth map and a pair of stereo depth maps as shown in Fig. 1(d) and (g). The proposed method consists of four steps: off-line calibration and on-line phase unwrapping, stereo matching, and refinement, which are described in the following subsections.

2.1 Calibration

Since our cameras are separately calibrated ones, we find the extrinsic parameters $(\mathbf{S}, \mathbf{R}, \mathbf{t})$ for mapping a ToF 3D point $\mathbf{X}^{ToF}$ to its projections $\mathbf{x}^L$ and $\mathbf{x}^R$ on the left and right stereo images:

$$\begin{aligned} \mathbf{X}^L &= \mathbf{S}(\mathbf{R}\mathbf{X}^{ToF} + \mathbf{t}), \\ \mathbf{X}^R &= \mathbf{X}^L - \mathbf{b}, \\ \mathbf{x}^L &= \mathbf{K}\mathbf{X}^L,\ \mathbf{x}^R = \mathbf{K}\mathbf{X}^R, \end{aligned} \tag{1}$$

where $\mathbf{R}$ and $\mathbf{t}$ are a 3×3 rotation matrix and a 3×1 translation vector, respectively. $\mathbf{S}$ is a 3×3 diagonal matrix for scaling each element of $\mathbf{R}\mathbf{X}^{ToF} + \mathbf{t}$. $\mathbf{K}$ and $\mathbf{b} = (b, 0, 0)^T$ are the stereo camera parameters, which are given.

In addition to conventional extrinsic parameters represented by $(\mathbf{R}, \mathbf{t})$, we use the scaling matrix $\mathbf{S}$ for compensating depth bias of the ToF camera as well as the calibration error that may not have been fully reduced in the manufacturing process. $\mathbf{S}$ can be considered as an extension of the scaling parameter in [20].

The extrinsic parameters are estimated as follows. First, images of checkerboard patterns are acquired by the fusion system, and their corresponding pixels are manually labeled. The checkerboard patterns are placed at various positions with different orientations, ranging from 1m to 4m from the fusion system, covering most of the calibrated range of the ToF camera [1]. To help the manual labeling and to simulate a variety of infrared reflectivity, we use checkerboard patterns of three different sizes with different gray level intensities. From the manually labeled pixels, a set of K triples of corresponding points are collected:

$\{\mathbf{X}_k^{ToF}, \mathbf{x}_k^L, \mathbf{x}_k^R : k = 1, ..., K\}$. By successively applying the direct linear transformation algorithm [26] and the Levenberg-Marquardt algorithm [27], we can find the extrinsic parameters $(\mathbf{S}, \mathbf{R}, \mathbf{t})$ that minimize the sum of squared projection errors under the orthogonality constraint of the rotation matrix $\mathbf{R}$.

After the estimation, we can calculate several statistical values such as the root mean square projection error $\sigma_{\mathbf{x}}$ and the root mean square disparity error σ_d, which are used in later steps for compensating time-varying ToF depth errors:

$$\begin{aligned} \sigma_{\mathbf{x}} &= \sqrt{\frac{1}{2K}\sum_k \|\mathbf{x}_k^L - \mathbf{x}_k^{ToF\to L}\|^2 + \|\mathbf{x}_k^R - \mathbf{x}_k^{ToF\to R}\|^2}, \\ \sigma_d &= \sqrt{\frac{1}{K}\sum_k ((x_k^L - x_k^R) - (x_k^{ToF\to L} - x_k^{ToF\to R}))^2}, \end{aligned} \tag{2}$$

where $\mathbf{x}_k^{ToF\to L}$ and $\mathbf{x}_k^{ToF\to R}$ denote the projections of $\mathbf{X}_k^{ToF}$ onto the left and right images, respectively, and $x_k^{ToF\to L}$ and $x_k^{ToF\to R}$ denote their x coordinates. We also calculate $\bar{I}$, the mean amplitude of the manually labeled ToF pixels.

2.2 Phase Unwrapping

We estimate the number of wrappings n_i at each ToF pixel i by minimizing an energy function E^{ToF} based on Markov random field (MRF) modeling:

$$E^{ToF} = \sum_{i\in ToF} C_i(n_i) + \lambda \sum_{(i,j)\in\mathcal{N}} U(n_i, n_j), \tag{3}$$

where $C_i(n_i)$ denotes the cost of assigning n_i to pixel i, and is referred to as the data cost. $U(n_i, n_j)$ denotes the cost of assigning n_i and n_j to adjacent pixels i and j, and is referred to as the discontinuity cost. $\mathcal{N}$ is the eight-neighborhood system on pixels, and λ is a balancing coefficient. In this subsection, we define $C_i(n_i)$ and $U(n_i, n_j)$ so that the number of wrappings n_i will be robustly estimated against noise and occlusion.

If n_i is given, we can recover the unwrapped ToF 3D point $\mathbf{X}_i^{ToF}(n_i)$ by

$$\mathbf{X}_i^{ToF}(n_i) = \frac{\|\mathbf{X}_i^{ToF}\| + n_i r_{\max}}{\|\mathbf{X}_i^{ToF}\|}\mathbf{X}_i^{ToF}, \tag{4}$$

where $\mathbf{X}_i^{ToF}$ is the measured 3D point. $\mathbf{X}_i^{ToF}(n_i)$ is projected onto $\mathbf{x}_i^{ToF\to L}(n_i)$ and $\mathbf{x}_i^{ToF\to R}(n_i)$ in the left and right images, respectively, and we denote their texture and color dissimilarity by

$$Dissim(\mathbf{x}_i^{ToF\to L}(n_i), d_i(n_i)), \tag{5}$$

where the disparity $d_i(n_i)$ equals to $x_i^{ToF\to L}(n_i) - x_i^{ToF\to R}(n_i)$. If n_i is correct, we can expect the dissimilarity to be small. $\mathbf{X}_i^{ToF}$, however, suffers from time-varying depth errors that are not reduced by the calibration procedure, so the dissimilarity may be calculated from inaccurate projections that do not correspond to each other. To incorporate the accurate projections into consideration,

we search for the minimum dissimilarity value in the vicinity of $\mathbf{x}_i^{ToF\to L}(n_i)$ and $d_i(n_i)$, and define the data cost $C_i(n_i)$ as the minimum value:

$$C_i(n_i) = \min_{\|\Delta\mathbf{x}\|<T_\mathbf{x}, \Delta d<T_d} Dissim(\mathbf{x}_i^{ToF\to L}(n_i) + \Delta\mathbf{x}, d_i(n_i) + \Delta d), \quad (6)$$

where $T_\mathbf{x}$ and T_d, which determine the search range, are set to $3\sigma_\mathbf{x}$ and $3\sigma_d$.

Our dissimilarity measure $Dissim$ is the weighted sum of the Birchfield and Tomasi's measure (BT) [28] applied to the a and b channels in the CIE Lab color space and the adaptive normalized cross correlation ($ANCC$) [29], and both measures are aggregated by using the adaptive support window [30]:

$$Dissim = \frac{1}{2T_{BT}} BT + \frac{1}{2} ANCC, \quad (7)$$

where T_{BT} is a threshold for truncating color difference. By combining the two different dissimilarity measures, we obtain the balance between the distinctiveness of color information and the robustness to illumination.

Since the viewpoints of the cameras are all different, a ToF 3D point can be occluded from the viewpoints of the stereo camera, resulting in large dissimilarity between accurate projections. In addition, our dissimilarity measure is ambiguous if all the unwrapped points of $\mathbf{X}_i^{ToF}$ with different n_i are projected onto the same textureless region. To handle with the occlusion and ambiguity, we define $U(n_i, n_j)$ in a manner of penalizing pixels i and j if the proximity between their ToF 3D points is broken by assigning different numbers of wrappings. In this manner, a pixel with ambiguous data costs can be identically labeled with its proximate neighbors that may have distinctive data costs:

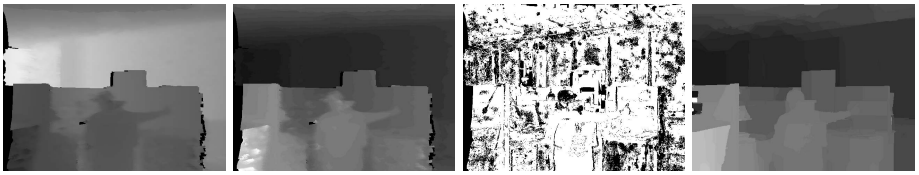
$$U(n_i, n_j) = \begin{cases} \exp(\frac{\Delta I_{ij}^2}{2\sigma_I^2}) \exp(-\frac{\Delta \mathbf{X}_{ij}^2}{2\sigma_U^2})/\Delta_{ij}, & \text{if } \Delta\mathbf{X}_{ij} < T_U, n_i \neq n_j, \\ 0, & \text{otherwise}, \end{cases} \quad (8)$$

where ΔI_{ij} and Δ_{ij} are the amplitude difference and pixel coordinate distance between i and j, respectively, and $\Delta\mathbf{X}_{ij} = \|\mathbf{X}_i^{ToF} - \mathbf{X}_j^{ToF}\|$. σ_I^2 and σ_U^2 are the mean values of ΔI_{ij}^2 and $\Delta\mathbf{X}_{ij}^2$, respectively. The threshold T_U is set to the smaller value between $3\sigma_U$ and $0.5r_{\max}$.

We minimize E^{ToF} in Eq. (3) using the α-expansion algorithm [31], and obtain an unwrapped depth map. As a post processing step, we apply a median filter and a bilateral filter [32] sequentially to the unwrapped depth map, to refine incorrectly unwrapped depth values and to reduce noise. Fig. 1(d) shows an unwrapped depth map obtained by the proposed method.

2.3 Stereo Matching

After the phase unwrapping step, a mesh model can be constructed by connecting adjacent ToF 3D points. We project the mesh model onto the left and right images to obtain up-sampled depth maps. Fig. 2(a) shows one of the up-sampled depth maps. Because of the different viewpoints of the cameras, the pixels in



(a) (b) (c) (d)

Fig. 2. Stereo matching process and result. (a) Left up-sampled depth map LU and (b) its converted disparity map UD. The pixels without valid values are colored in black. The intensity in (b) is linear with the disparity values. (c) Data cost confidence map. If $\alpha_p > \beta_p$, pixel p is colored in black, and otherwise in white. Refer to the text for α_p and β_p. (d) Stereo matching result. From (a) to (d), only the left images are shown.

the vicinity of the depth discontinuities do not have valid values in the up-sampled depth maps. In addition, ToF depth values suffer from time-varying errors causing inaccurate projection.

To improve the quality of the up-sampled depth maps, we use a stereo matching and refinement approach that utilizes both of the up-sampled depth maps and stereo images. Inspired by the success of the MRF-based optimization methods for stereo matching [33], we find the disparity values that minimize MRF energy functions. The energy functions are defined for both left and right images; however, we present the energy for the left image, to avoid redundancy:

$$E^{Stereo} = \sum_{p \in L} D_p(d_p) + \gamma \sum_{(p,q) \in \mathcal{N}} V(d_p, d_q), \tag{9}$$

where $D_p(d_p)$ is the cost of assigning disparity d_p to pixel p, and $V(d_p, d_q)$ is the cost of assigning d_p and d_q to adjacent pixels p and q. γ is a balancing coefficient.

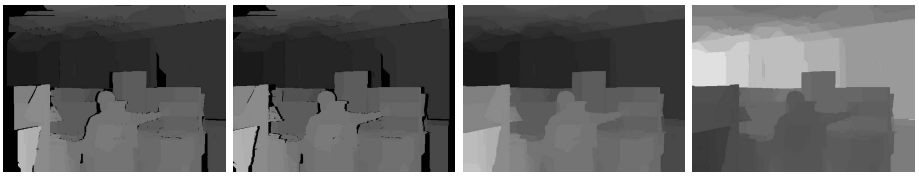
$D_p(d_p)$ is defined as the weighted sum of the stereo image-based cost $Dissim$ in Eq. (5) and a depth-based cost D^{UD}:

$$D_p(d_p) = \alpha_p Dissim(\mathbf{x}_p, d_p) + \beta_p D^{UD}(\mathbf{x}_p, d_p). \tag{10}$$

The left up-sampled depth map LU can be converted to a disparity map UD using the stereo camera parameters, and we use UD to define $D^{UD}(\mathbf{x}_p, d_p)$. Using the thresholds $T_{\mathbf{x}}$ and T_d, we can determine a set $PD(\mathbf{x}_p, T_{\mathbf{x}}, T_d)$ of possible disparity values of pixel p, consisting of all the disparity values $d = d_k + \Delta d$ of all the pixels k located at $\mathbf{x}_p + \Delta\mathbf{x}$ in UD, where $\Delta d < T_d$ and $\|\Delta\mathbf{x}\| < T_{\mathbf{x}}$. We define D^{UD} in a manner of assigning low penalty if $d_p \in PD(\mathbf{x}_p, T_{\mathbf{x}}, T_d)$ and otherwise high penalty:

$$D^{UD}(\mathbf{x}_p, d_p) = \begin{cases} \min\limits_{d}(Dissim(\mathbf{x}_p, d)), & \text{if } d_p \in PD(\mathbf{x}_p, T_{\mathbf{x}}, T_d), \\ \max\limits_{d}(Dissim(\mathbf{x}_p, d)), & \text{otherwise,} \end{cases} \tag{11}$$

where the low and high penalties are adaptively determined using the values of $Dissim$, to obtain the balance between $Dissim$ and D^{UD}.



(a) (b) (c) (d)

Fig. 3. Stereo depth map refinement process. (a) Left and (b) right disparity maps after the consistency check. The inconsistent pixels are colored in black. (c) Left refined disparity map obtained by using D_{occl}, and (d) its converted depth map DM.

α_p is the same confidence measure with those in [19,20] based on the distinctiveness of the dissimilarity values:

$$\alpha_p = 1 - Dissim_{best}(\mathbf{x}_p)/Dissim_{second}(\mathbf{x}_p). \tag{12}$$

β_p is set to the mean of the weighted distinctiveness of the ToF 3D points that are projected onto the neighborhood of pixel p:

$$\beta_p = E[w_i \times (1 - C_{i,best}/C_{i,second})], \text{ for } i \in ToF \text{ such that } \|\mathbf{x}_i^{ToF \to L} - \mathbf{x}_p\| < T_{\mathbf{x}}, \tag{13}$$

where $C_{i,best}$ and $C_{i,second}$ are the minimum and the second minimum value of $C_i(n_i)$ in Eq. (6). w_i is the confidence of ToF pixel i, which is 1 if its amplitude I_i is greater than $\bar{I}$, and otherwise is $\sqrt{I_i/\bar{I}}$ based on the theoretical derivation that the time-varying depth error is approximately proportional to $1/\sqrt{I_i}$ [34]. α_p and β_p are normalized to satisfy $\alpha_p + \beta_p = 1$. Fig. 2(c) shows a confidence map showing that $\alpha_p > \beta_p$ in highly textured or low infrared reflective regions. In contrast, we can observe that $\alpha_p < \beta_p$ in textureless regions.

We define $V(d_p, d_q)$ in a manner of encouraging pixels with similar color values to have similar disparity values:

$$V(d_p, d_q) = \exp(-\frac{\|\mathbf{c}_p - \mathbf{c}_q\|^2}{2\sigma_{\mathbf{c}}^2}) \min((d_p - d_q)^2, T_V)/\Delta_{pq}, \tag{14}$$

where $\mathbf{c}$ denotes the color values of a pixel, and $\sigma_{\mathbf{c}}^2$ is the mean of $\|\mathbf{c}_p - \mathbf{c}_q\|^2$. T_V is a threshold for truncating $V(d_p, d_q)$. Δ_{pq} is the pixel coordinate distance between p and q. We also optimize E^{Stereo} using the α-expansion algorithm [31], and Fig. 2 shows a disparity map obtained by our stereo matching method.

2.4 Refinement

After the stereo matching step, we obtain a pair of left and right disparity maps, for which the left-right consistency is checked to find the pixels with erroneous disparity values and the pixels in occluded regions. Fig. 3(a) and (b) show consistency-checked left and right disparity maps.

Since $Dissim$ does not provide reliable evidence in the occluded regions, we fill the inconsistent regions by minimizing a MRF energy with different data costs. For the inconsistent pixels with valid disparity values in UD, we use D^{UD} as the data cost D^{Occl}, while we use the following data cost for the remaining inconsistent pixels:

$$D^{Occl}(\mathbf{x}_p, d_p) = \begin{cases} \min\limits_{d}(Dissim(\mathbf{x}_p, d)), \text{ if } d_p = d_{left}, \\ \max\limits_{d}(Dissim(\mathbf{x}_p, d)), \text{ otherwise,} \end{cases} \tag{15}$$

where d_{left} is the disparity of the closest consistent pixel in the left direction. We note that d_{left} is replaced with d_{right} for the right image. Fig. 3(c) shows a refined disparity map obtained by minimizing the new MRF energy.

A refined disparity map, for example, the left refined disparity map can be inversely converted to a depth map DM, in which the depth discontinuities are enhanced but the structures in textureless regions are lost. To recover the lost structures, we apply a filter iteratively to each pixel p of DM, whose neighboring pixel q is searched in both LU and DM:

$$Z_p^{DM} \leftarrow \sum_{q \in \mathcal{W}_p} w_q^{DM} Z_q^{DM} + w_q^{LU} Z_q^{LU}, \tag{16}$$

where Z denotes the Z values of the pixels in each depth map, and w denotes the filtering weights that satisfy $\sum_{q \in \mathcal{W}_p} w_q^{DM} + w_q^{LU} = 1$. $\mathcal{W}_p$ denotes the filtering window. If the difference between d_p and d_q are larger than T_d, w_q^{LU} and w_q^{DM} are all set to 0 to preserve the depth discontinuity, while they are otherwise set to a trilateral weight [22] determined by the differences in Z, $\mathbf{c}$, and $\mathbf{x}$ between p and q. Fig. 1(g) shows refined stereo depth maps obtained by the filtering.

3 Experiments

This section provides experimental results demonstrating the effectiveness of the proposed method. First, we quantitatively compare our phase unwrapping results with those obtained by the state-of-the-art multi-frequency phase unwrapping method (MFPU) [8]. Second, we qualitatively compare three kinds of depth maps: depth maps obtained by using the fused data cost D, the stereo image-based data cost $Dissim$, and the depth-based data cost D^{UD}.

For the experiments, we acquired 57 sets of a single 31MHz ToF depth map and stereo images from four different places: office, corridor, auditorium, and laboratory. In addition, we acquired ground-truth 10MHz depth maps from the same viewpoints of their corresponding input images one after another. The maximum ranges of the two frequencies are 4.84m and 15m, respectively. Although our method is ready to deal with dynamic scenes, we acquired static scenes so as to make precise comparison between the estimated and the ground-truth depth maps. The integration time was set to 4×8.2ms (corresponding to 20.5fps due to the read-out time) for the input depth data to simulate real-time

Table 1. Fraction of accurately estimated pixels. $N_{\max} = 2$ and $N_{\max} = 1$ mean that the actual largest distances are smaller than the three times and twice of $r_{\max}$ (14.51m and 9.68m), respectively. In the entire 57 sets, $N_{\max} = 2$ for 46 sets, and $N_{\max} = 1$ for 31 sets. These sets were used to calculate the fractions.

$N_{\max}$	Proposed	MFPU [8]
2	96.65%	88.25%
1	96.32%	96.88%

depth acquisition, and 4×16.4ms for the ground-truth depth data to suppress random noise, which decreases with the integration time. In addition we adopt the internal median filter our ToF camera [1] as preprocessing.

We used fixed parameters in the experiments: $T_{\mathbf{x}} = 5.48$, $T_{\mathbf{d}} = 1.12$, and $\bar{I} = 6279.74$ that were obtained as the byproducts of the calibration; and $T_{BT} = 20$, $\lambda = 2.5$, $\gamma = 0.15$, and $T_V = 5$ that were manually chosen. The maximum number of wrappings and disparity were set to 5 and 100, which determines the theoretical maximum range of the ToF camera as 29.03m and the minimum depth of the stereo camera as 0.97m. We follow the parameter settings in [30] for the data cost aggregation except the window size, which was set to 19 to save the memory use. We also note that the parameters c_f and c_d of MFPU [8] were both set to 0.9 based on a grid search.

In the quantitative evaluation, we consider the number of wrappings n_i to be correct if $|r_i + n_i r_{\max} - r_i^\star| < 0.5 r_{\max}$, where $r_i^\star$ is the distance value at pixel i in the ground-truth depth map. If no $n_i \in \{0, \ldots, 5\}$ satisfies the above inequality because of noise larger than $0.5 r_{\max}$, we discard pixel i from the evaluation. We note that the fraction of the discarded pixels are within 1% of the entire pixels.

Table 1 shows the fraction of pixels with accurately estimated distance values. Since MFPU is able to extend the maximum range up to twice [8], the success rate of the proposed method is higher than that of MFPU when $N_{\max}$ is 2 (refer to Table 1 for the definition of $N_{\max}$). In contrast, the success rate of MFPU is slightly higher than that of the proposed method when $N_{\max} = 1$. We could not evaluate the success rates on the sets with $N_{\max} > 2$ since 10MHz is the lowest frequency supported by our ToF camera. We also note that the corridor scene data was not used for the quantitative evaluation not only because $N_{\max} > 2$ for most of the images but also because the ground-truth depth maps suffer from erroneous depth measurements due to high reflection from the window and floor as shown in Fig. 4(a) and (b).

Fig. 4 shows sample phase unwrapping results. Fig. 4(a) shows results on the corridor scene, which demonstrates the potential of the proposed method to extend the maximum range of the ToF camera up to four times. In Fig. 4(b) and (c), it is also clearly seen that the maximum range is extended up to three times with high success rates. On the other hand, Fig. 4(d) and (e) show the drawbacks of the proposed method: If a large textureless object is present in the scene as shown in the left part of 4(d,ii), the scene points from the object can be projected onto a single homogeneous region regardless of the value of n_i, resulting in ambiguous data costs. If a foreground object, for example, the chair

Fig. 4. Phase unwrapping results. (i) Input depth map. (ii) Left image. Unwrapped depth maps obtained by (iii) the proposed method and (iv) MFPU. (v) Ground-truth depth map. (vi) Estimated number of wrappings corresponding to (iii). The maximum intensity (255 in gray level) in (i) corresponds to 5m except the last row (15m). It corresponds to $N \times 5$m in the other depth maps in (iii–v), where $N = 3$ for (a),(b), $N = 2$ for (c),(d),(e), and $N = 4$ for (f). The gray level intensity in (vi) is linear with the number of wrappings ranging from 0 to 3. The red pixels are with erroneously estimated numbers of mods, while the green pixels are those discarded from the evaluation.

heads in Fig. 4(e,i), has similar original depth values with its background, the discontinuity cost of assigning different numbers of mods to the foreground and background pixels becomes large, resulting in erroneous phase unwrapping.

On the other hand, it is interesting to notice that the actual number of wrappings of the '+'-marked pixel i in Fig. 4(b,vi) is 3, while it is estimated as 2 by the proposed method. The 3D points $\mathbf{X}_i^{ToF}(n_i)$ for $n_i = 2$ and 3 are projected onto pixel positions $\mathbf{x}_2 = (527.62, 230.38)^T$ and $\mathbf{x}_3 = (526.45, 231.52)^T$ in the left image, respectively. Their corresponding disparity values are $d_2 = 8.39$ and $d_3 = 5.99$. Thus, the search ranges $\{\mathbf{x}_2 + \Delta\mathbf{x}, d_2 + \Delta d\}$ and $\{\mathbf{x}_3 + \Delta\mathbf{x}, d_3 + \Delta d\}$ for the minimum $Dissim$ overlap each other under the current thresholds $T_{\mathbf{x}}$ and T_d, resulting in ambiguous data costs with a constant value for $n_i \geq 2$. This kind of ambiguity is observed from other pixels of the corridor scene data, setting the practical maximum range of the current fusion system to three times of that of the ToF camera.

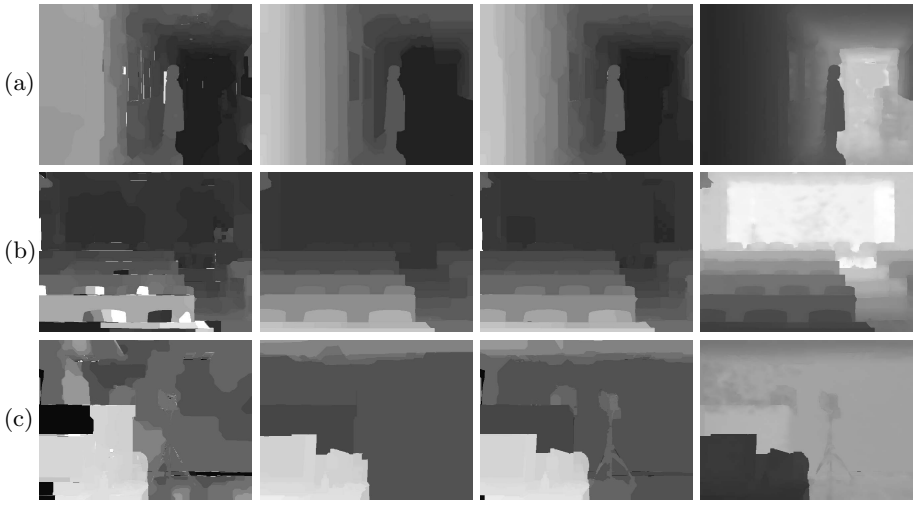


Fig. 5. Stereo matching and refinement results. (a–c) Left disparity maps and refined depth maps corresponding to the images in Fig. 4(b–d). (i–iii) Disparity maps obtained by using *Dissim*, D^{UD}, and D as the stereo matching data cost. (iv) Refined depth map obtained from (iii). In (c), the intensity in (i–iii) has been non-linearly adjusted to show the difference.

The maximum range can be extended more in many ways: We can reduce $T_{\mathbf{x}}$ and T_d by increasing the integration time, losing the system's potential to be applied in real-time tasks. We can also decrease the modulation frequency, for example, to 10MHz as shown in Fig. 4(f), at the expense of increased time-varying depth error. We can also increase the baseline of the stereo camera to separate the search ranges, although the current system does not allow it.

Fig. 5 shows stereo matching and refinement results on the input images shown in Fig. 4. In Fig. 5(a,i), the smooth structure on the left wall is lost by using *Dissim* for lack of texture, while it can be observed in Fig. 5(a,ii) and (a,iii). Similarly, in Fig. 5(b,i), the disparity values of the front chairs are not accurately estimated for lack of texture and due to aliasing, while they are accurately estimated in Fig. 5(b,ii) and (b,iii). In contrast, in the middle of Fig. 5(c,i) and (c,iii), the background object is identifiable while it is impossible in Fig. 5(c,ii), showing the complementary nature of stereo and time-of-flight.

On the other hand, the disparity values of the floor should be similar with those of the ceiling in Fig. 5(a) since the system is looking forward; however, neither by stereo nor by time-of-flight could they be accurately estimated due to the reflection of the far-range objects on the floor. In Fig. 5(c,i), the disparity of the right part of the mis-unwrapped foreground object could be accurately estimated by using *Dissim*; however, it could not be applied in Fig. 5(c,iii) because the mis-unwrapped object exhibits a high amplitude value that increases the confidence of D^{UD}, while the distinctiveness of *Dissim* is low, showing that

the distinctiveness and amplitude-based confidence measure is inadequate when the depth values are mis-unwrapped.

4 Conclusion

In this paper, we showed that fusion of time-of-flight and stereo is effective for correcting the modular ToF depth error caused by phase wrapping. Based on the constraint that an accurately unwrapped ToF 3D point should be projected onto its corresponding stereo pixels with similar color and texture, we built a Markov random field for estimating the number of wrappings. Through the experiments, we showed that the proposed method successfully extends the range use of the ToF camera up to three times, which is expected to be increased by an elaborated design of the fusion system. In addition, we showed that an accurately unwrapped depth map greatly improves the stereo matching results by reducing the matching ambiguity and providing the structure on textureless surfaces.

To overcome the drawbacks of the fusion system, we will consider using prior knowledge on the structure of indoor scenes, so that the depth values of highly reflective surfaces can be robustly estimated.

References

1. http://www.mesa-imaging.ch/prodview4k.php
2. http://en.wikipedia.org/wiki/Time-of-flight_camera
3. Lindner, M., Schiller, I., Kolb, A., Koch, R.: Time-of-flight sensor calibration for accurate range sensing. CVIU 114, 1318–1328 (2010)
4. Kim, Y.S., Kang, B., Lim, H., Choi, O., Lee, K., Kim, J.D.K., Kim, C.Y.: Parametric model-based noise reduction for ToF depth sensors. In: IS&T/SPIE EI (2012)
5. Göktürk, S.B., Yalcin, H., Bamji, C.: A time-of-flight depth sensor—system description, issues and solutions. In: CVPR Workshops (2004)
6. Choi, O., Lim, H., Kang, B., Kim, Y.S., Lee, K., Kim, J.D.K., Kim, C.Y.: Range unfolding for time-of-flight depth cameras. In: ICIP (2010)
7. Droeschel, D., Holz, D., Behnke, S.: Probabilistic phase unwrapping for time-of-flight cameras. In: Joint 41st International Symposium on Robotics and 6th German Conference on Robotics (2010)
8. Droeschel, D., Holz, D., Behnke, S.: Multifrequency phase unwrapping for time-of-flight cameras. In: IROS (2010)
9. McClure, S.H., Cree, M.J., Dorrington, A.A., Payne, A.D.: Resolving depth-measurement ambiguity with commercially available range imaging cameras. In: Image Processing: Machine Vision Applications III (2010)
10. Jongenelen, A.P.P., Carnegie, D.A., Payne, A.D., Dorrington, A.A.: Maximizing precision over extended unambiguous range for TOF range imaging systems. In: IEEE Instrumentation & Measurement Technology Conference (2010)
11. Kuhnert, K.D., Stommel, M.: Fusion of stereo-camera and PMD-camera data for real-time suited precise 3D environment reconstruction. In: IROS (2006)
12. Beder, C., Bartczak, B., Koch, R.: A Combined Approach for Estimating Patchlets from PMD Depth Images and Stereo Intensity Images. In: Hamprecht, F.A., Schnörr, C., Jähne, B. (eds.) DAGM 2007. LNCS, vol. 4713, pp. 11–20. Springer, Heidelberg (2007)

13. Lindner, M., Lambers, M., Kolb, A.: Data fusion and edge-enhanced distance refinement for 2D RGB and 3D range images. International Journal of Intelligent Systems Technologies and Applications 5, 344–354 (2008)
14. Gudmundsson, S.Á., Aanæs, H., Larsen, R.: Fusion of stereo vision and time-of-flight imaging for improved 3D estimation. International Journal of Intelligent Systems Technologies and Applications 5, 425–433 (2008)
15. Hahne, U., Alexa, M.: Combining time-of-flight depth and stereo images without accurate extrinsic calibration. International Journal of Intelligent Systems Technologies and Applications 5, 325–333 (2008)
16. Netramai, C., Melnychuk, O., Joochim, C., Roth, H.: Combining PMD and stereo camera for motion estimation of a mobile robot. In: IFAC World Congress (2008)
17. Hahne, U., Alexa, M.: Depth Imaging by Combining Time-of-Flight and On-Demand Stereo. In: Kolb, A., Koch, R. (eds.) Dyn3D 2009. LNCS, vol. 5742, pp. 70–83. Springer, Heidelberg (2009)
18. Kim, Y.M., Theobalt, C., Diebel, J., Kosecka, J., Miscusik, B., Thrun, S.: Multi-view image and ToF sensor fusion for dense 3D reconstruction. In: IEEE Workshop on 3-D Digital Imaging and Modeling (2009)
19. Zhu, J., Wang, L., Gao, J., Yang, R.: Spatial-temporal fusion for high accuracy depth maps using dynamic MRFs. IEEE T. PAMI 32, 899–909 (2010)
20. Zhu, J., Wang, L., Yang, R., Davis, J.E., Pan, Z.: Reliability fusion of time-of-flight depth and stereo geometry for high quality depth maps. IEEE T. PAMI 33, 1400–1414 (2011)
21. Hansard, M., Horaud, R., Amat, M., Lee, S.: Projective alignment of range and parallax data. In: CVPR (2011)
22. Choi, O., Lim, H., Kang, B., Kim, Y.S., Lee, K., Kim, J.D.K., Kim, C.Y.: Discrete and continuous optimizations for depth image super-resolution. In: IS&T/SPIE EI (2012)
23. http://msdn.microsoft.com/en-us/library/hh438998.aspx
24. Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohli, P., Shotton, J., Hodges, S., Fitzgibbon, A.: Kinectfusion: Real-time dense surface mapping and tracking. In: ISMAR (2011)
25. http://www.ptgrey.com/products/bumblebee2/bumblebee2_stereo_camera.asp
26. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge University Press (2003)
27. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes in C++. Cambridge University Press (2002)
28. Birchfield, S., Tomasi, C.: A pixel dissimilarity measure that is insensitive to image sampling. IEEE T. PAMI 20, 401–406 (1998)
29. Heo, Y.S., Lee, K.M., Lee, S.U.: Illumination and camera invariant stereo matching. In: CVPR (2008)
30. Yoon, K.J., Kweon, I.S.: Adaptive support-weight approach for correspondence search. IEEE T. PAMI 28, 650–656 (2006)
31. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE T. PAMI 23, 1222–1239 (2001)
32. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: ICCV (1998)
33. http://vision.middlebury.edu/stereo/eval/
34. Buttgen, B., Lustenberger, F., Seitz, P.: Demodulation pixel based on static drift fields. IEEE Transactions on Electron Devices 53, 2741–2747 (2006)

Author Index