Anders Heyden Fredrik Kahl
Carl Olsson Magnus Oskarsson
Xue-Cheng Tai (Eds.)

Energy Minimization Methods in Computer Vision and Pattern Recognition

LNCS 8081

9th International Conference, EMMCVPR 2013
Lund, Sweden, August 2013
Proceedings

Lecture Notes in Computer Science 8081

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

Editorial Board

David Hutchison
Lancaster University, UK

Takeo Kanade
Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler
University of Surrey, Guildford, UK

Jon M. Kleinberg
Cornell University, Ithaca, NY, USA

Alfred Kobsa
University of California, Irvine, CA, USA

Friedemann Mattern
ETH Zurich, Switzerland

John C. Mitchell
Stanford University, CA, USA

Moni Naor
Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz
University of Bern, Switzerland

C. Pandu Rangan
Indian Institute of Technology, Madras, India

Bernhard Steffen
TU Dortmund University, Germany

Madhu Sudan
Microsoft Research, Cambridge, MA, USA

Demetri Terzopoulos
University of California, Los Angeles, CA, USA

Doug Tygar
University of California, Berkeley, CA, USA

Gerhard Weikum
Max Planck Institute for Informatics, Saarbruecken, Germany

Anders Heyden Fredrik Kahl
Carl Olsson Magnus Oskarsson
Xue-Cheng Tai (Eds.)

Energy Minimization Methods in Computer Vision and Pattern Recognition

9th International Conference, EMMCVPR 2013
Lund, Sweden, August 19-21, 2013
Proceedings

Springer

Volume Editors

Anders Heyden
Fredrik Kahl
Carl Olsson
Magnus Oskarsson

Lund University
Centre for Mathematical Sciences
Lund, Sweden
E-mail: {heyden; fredrik; calle; magnuso}@maths.lth.se

Xue-Cheng Tai
University of Bergen
Deptartment of Mathematics
Bergen, Norway
E-mail: tai@mi.uib.no

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-40394-1 e-ISBN 978-3-642-40395-8
DOI 10.1007/978-3-642-40395-8
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013945420

CR Subject Classification (1998): F.2, E.1, G.2, I.3.5, I.4-5, G.1, C.2

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

© Springer-Verlag Berlin Heidelberg 2013
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

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

Energy minimization methods have become an established paradigm for resolving a variety of challenging problems in the fields of computer vision and pattern recognition. In spite of the lack of a "unifying theory of vision," notable progress has been achieved over the last few decades and we are now able to solve application problems that were not possible before. Much of this progress can be contributed to improved optimization strategies. In fact, most of the state-of-the-art methods are nowadays based on the concept of computing solutions to a given problem by minimizing the respective energies using modern optimization methods.

This volume contains the papers presented at the 9th International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR 2013), held in Lund, Sweden, during August 19–21, 2013. These papers demonstrate the broad scope of the research field, ranging from applications in medical imaging to semantic scene analysis and image editing. The energy minimization methods include everything from discrete graph theoretical approaches and Markov Random Fields to variational methods and partial differential equations. We received 40 submissions and based on the reviewers' recommendations, after a double-blind review process, 26 high-quality papers were selected for publication, all of them as oral presentations.

Furthermore, we were delighted that three leading experts from the fields of computer vision and optimization, namely, Stephen Boyd (Stanford University), Olga Veksler (University of Western Ontario) and Thomas Pock (Graz University of Technology), agreed to further enrich the conference with inspiring keynote lectures.

We would like to express our gratitude to those who made this event possible and contributed to its success. In particular, the international Program Committee of top international experts in the field provided excellent reviews. A major donation from the Swedish Research Council is gratefully acknowledged. We are also grateful for the local support from all the people in the Mathematical Imaging Group at Lund University, as well as for the financial back-up from the Swedish Society for Automated Image Analysis. It is our belief that this conference helped to advance the field of energy minimization methods and to further establish the mathematical foundations of computer vision.

May 2013

Anders Heyden
Fredrik Kahl
Carl Olsson
Magnus Oskarsson
Xue-Cheng Tai

Organization

EMMCVPR 2013 was organized by the Centre for Mathematical Sciences, Lund Univeristy, Sweden, on behalf of the Swedish Society for Automated Image Analysis.

Executive Committee

Conference Chair

Fredrik Kahl	Lund University, Sweden

Program Chairs

Anders Heyden	Lund University, Sweden
Fredrik Kahl	Lund University, Sweden
Carl Olsson	Lund University, Sweden
Magnus Oskarsson	Lund University, Sweden
Xue-Cheng Tai	University of Bergen, Norway

Invited Speakers

Stephen Boyd	Stanford University, USA
Olga Veksler	University of Western Ontario, Canada
Thomas Pock	Graz University, Austria

Program Committee

H. Aanaes	J. Clark	B. Flach
L. Agapito	J. Corso	D. Geiger
S. Agarwal	J. Coughlan	L. Grady
K. Alahari	D. Cremers	E. Hancock
E. Bae	J. Darbon	R. Hartley
D. Batra	A. Delong	H. Ishikawa
J.-C. Bazin	G. Doretto	D. Jacobs
I. Ben Ayed	C.-H. Ek	I. Jermyn
M. Blaschko	O. Enqvist	C. Kervrann
Y. Boykov	A. Eriksson	R. Kimmel
A. Bruhn	P. Favaro	T. Kohlberger
M. Björkman	M. Felsberg	I. Kokkinos
A. Chambolle	M. Figueiredo	V. Kolmogorov
X. Chen	A. Fix	N. Komodakis

A. Konushin
L. Latecki
J. Lellman
H. Li
S. Li
X. Liu
S. Masnou
S. Maybank
M. Ng
C. Nieuwenhuis
M. Pelillo
T. Pock
A. Rangarajan
B. Rosenhahn
S. Roth
J. Shi
D. Schlesinger
F. Schmidt
C. Schnoerr
A. Schwing
Y. Seo
A. Shekhovtsov
C. Sminchisescu
J. Sturm
H. Talbot
M. Tappen
C. Taylor
O. Veksler
R. Vidal
V. Vineet
J. Warrell
J. Weickert
T. Werner
O. Woodford
X. Wu
J. Yuan
C. Zach
J. Zerubia
S.-C. Zhu

Sponsoring Institutions

Centre for Mathematical Sciences, Lund University
Swedish Society for Automated Image Analysis
The Swedish Research Council

Table of Contents

Image Restoration

Scene Understanding

Segmentation I

Superpixels

Statistical Methods and Learning

Segmentation II

Rapid Mode Estimation for 3D Brain MRI Tumor Segmentation

Haithem Boussaid, Iasonas Kokkinos, and Nikos Paragios

Center for Visual Computing, Ecole Centrale de Paris, France
Galen, INRIA Saclay, France
{haithem.boussaid,iasonas.kokkinos,nikos.paragios}@ecp.fr

Abstract. The efficient definition of the tumor area is crucial in brain tumor resection planning. But current methods embedded in computer aided diagnosis systems can be time consuming, while the initialization of the segmentation mask may be possible. In this work, we develop a method for rapid automated segmentation of brain tumors.

The main contribution of our work is an efficient method to initialize the segmentation by casting it as nonparametric density mode estimation, and developing a Branch and Bound-based method to efficiently find the mode (maximum) of the density function. Our technique is exact, has guaranteed convergence to the global optimum, and scales logarithmically in the volume dimensions by virtue of recursively subdividing the search space through Branch-and-Bound and Dual-Tress data structures.

This estimated mode provides our system with an initial tumor hypothesis which is then refined by graph-cuts to provide a sharper outline of the tumor area.

We demonstrate a 12-fold acceleration with respect to a standard mean-shift implementation, allowing us to accelerate tumor detection to a level that would facilitate a high-degree brain tumor resection planning.

1 Introduction

The most common treatment of brain tumors is surgical resection, while a determining factor for the quality of the intervention is efficient identification in the planning of the surgical margins. Conventional segmentation techniques rely on prior knowledge and smoothness constraint to overcome the enormous variability of tumors both in terms of location as well as in terms of geometric characteristics. Recent studies indicate statistical preferential locations for tumors in the brain [1]. [2] proved that using this information improves substantially the results. In this work, we develop a rapid method for automated segmentation initialization that relies on nonparametric density mode estimation. The mode of nonparametric density estimate corresponds to the center of the tumor, and localizing it in a three-dimensional volume is used to initialize a 3D segmentation using graph-cuts.

Clustering, segmentation and nonparametric density mode estimation are related problems whose combination has been particularly studied in 2D images in

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 1–11, 2013.
© Springer-Verlag Berlin Heidelberg 2013

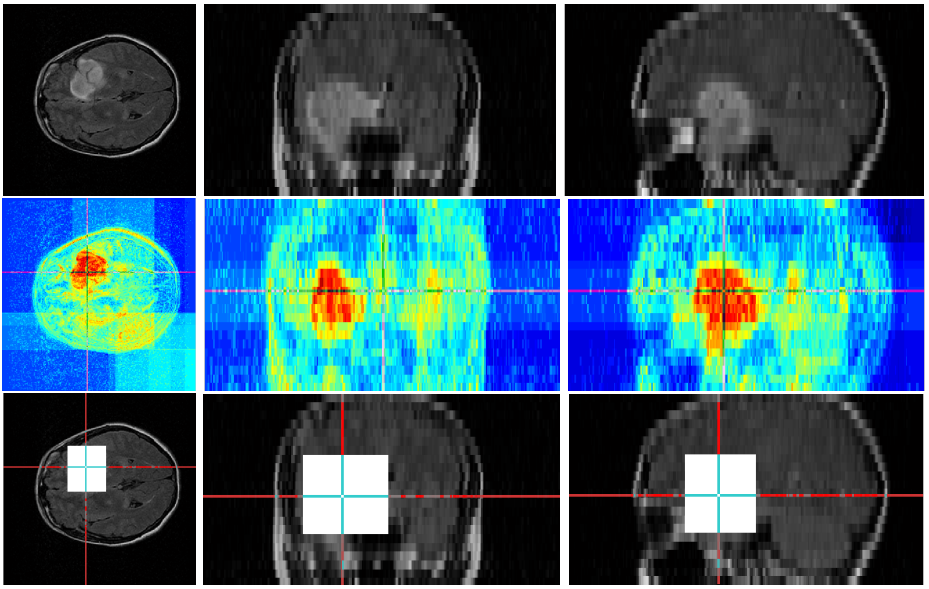


Fig. 1. First row: patient MRI, Second row: adaboost scores, Third row: detection bounding boxes. From left to right: x-y plane, x-z plane, y-z plane where x,y,z are the coordinates of the tumor center.

the thread of works developed around the Mean-Shift method [3]. This method is also used as a component in a number of diagnosis tools such as vessel tracking [4], Multiple Sclerosis brain segmentation [5] and MRI brain clustering [6], and is a general tool that applies transversally to a host of problems in medical imaging.

Our goal is to address the computational complexity of the mode estimation problem. The original Mean Shift method [3] is iterative, scales linearly in the number of points used in the Kernel Density Estimation (KDE) (as it follows the trajectory of every one of them) and can require careful checking of convergence. Faster variants of Mean Shift exist including Medoid Shift [7], Quick Shift [8], Fast Gauss transforms [9] as well as the Dual Tree variant of Mean Shift [10]. However, those of them that are exact [9,10] are 'dense' i.e. evaluate the KDE over a dense set of locations; as such they may be inappropriate for application to 3D medical image volumes. Alternatively, while those that focus on the modes [7,8] are only approximate, and have complexity proportional to $O(KN)$ where N is the number of pixels and K is the typical neighborhood size.

The main contribution of our work is a rapid mode estimation technique for 3D MRI image segmentation. Dealing with three dimensional data challenges several algorithms which are reasonably efficient for 2D medical image analysis. In this paper, we leverage recent developments using Branch-and-Bound (BB) in object detection [11], which demonstrated that detection is possible in time sub-linear in the image size.

The main thrust of our work is the adaptation of this idea to the mode finding problem in KDE, typically addressed through Mean Shift. We propose an algorithm that can find the mode of the density with best-case complexity being logarithmic in the size of the search space.

We apply our algorithm to the setting of 3D brain tumor segmentation. Our algorithm takes the scores of a discriminatively trained classifier for tumor voxels and uses them to construct weights for a KDE-based estimate of the tumor location. Using standard mean shift would require tracking the trajectory of each voxel, and identifying the largest basin of attraction. Instead our algorithm narrows down the location of the maximum through an iterative branch-and bound algorithm. In specific, we construct bounds on the value of the KDE over intervals of the search space, and use these bounds to devise a prioritized search strategy for the density's mode. We demonstrate substantial speedups when comparing to the standard mean-shift algorithm.

Furthermore, we couple the mode estimation results with a post-processing step using graph-cuts, which allows us to boost the segmentation performance of our algorithm. Systematic evaluations are performed on clinical datasets demonstrating a 12-fold acceleration in speed over classical Mean-Shift while at the same time achieving robust tumor detection and segmentation.

2 3D Structure Detection

Our goal is to devise an algorithm that can quickly detect the largest region corresponding to a class (tumor in our case) within a 3D volume. This problem is particularly challenging for standard segmentation algorithms as it is hard to define exact boundaries between tumor and normal tissue [12]. Moreover, relying only on a classifier to separate the tumor class from the remaining structures in the MRI volume is tricky, due to the similarity between tumor and normal tissue and the high diversity in appearance of tumor tissue among different patients [12].

We start by phrasing our problem as mode seeking for a Kernel Density Estimate and then proceed to describing our Branch-and-Bound based optimization algorithm. We note that even though we focus on tumor segmentation, the same approach could potentially be useful to other maximization problems in 3D space.

2.1 Problem Formulation

We consider that we are provided with a scoring function that estimates the probability w_i with which a voxel x_i in $\Re^3$ can belong to the considered class (i.e. a tumor vs non-tumor classifier). Namely, we have a mapping:

$$f : \Re^3 \rightarrow [0, 1], x_i \mapsto w_i, \tag{1}$$

where f encapsulates the feature extraction around x_i and the subsequent formation of the class posterior. In specific, this score can be the output of a soft classifier or a likelihood function on the density distribution of the tumor class.

In order to pool information from multiple voxels and obtain a regularized labeling of the 3D volume, we phrase our problem in terms of a Kernel-based Density Estimator of the form:

$$KDE(x) = \sum_{i=1}^{n} w_i K_h (x - x_i) \tag{2}$$

We consider that K_h is a separable decreasing kernel, with the parameter h determining the used amount of smoothing. In the context of our application, we work with the finite-support Epanechnikov kernel [13]:

$$K_h(x - x_i) = \begin{cases} 0 \text{ if } \|x - x_i\| > h \\ \frac{3}{4}\left(1 - \left(\frac{\|x-x_i\|}{h}\right)^2\right) \text{ else,} \end{cases} \tag{3}$$

even though any other separable decreasing kernel could be used, e.g. an uniform or a Gaussian kernel [2]. We also note that in principle we should normalize the w_i elements to have unit sum, but the subsequent tasks are unaffected by this normalization. We address the problem of region detection in terms of mode estimation for the KDE above, namely we set out to find:

$$x^* = \underset{\Re^3}{\text{argmax}}\ S(x) = \underset{\Re^3}{\text{argmax}}\ \sum_{i=1}^{n} w_i K_h (x - x_i) \tag{4}$$

Instead of the iterative procedure employed by Mean-Shift, we now describe how Brand-and-Bound can be used to directly recover the solution of Eq.4.

2.2 Branch and Bound Algorithm

Branch-and-Bound is an optimization method that searches for global maximum of a function $S(x)$. To this end, the algorithm employs a recursive subdivision of an interval of solutions X in its prioritized search for the maximum. The priority of an interval is determined by the function's upper bound $\overline{S}$ within it. So, if we consider the maximum value of function S within the interval X, say $S(X) = \max_{x \in X} S(x)$, we bound it with $\overline{S}(X) \geq S(X)$. Moreover, we require $\overline{S}(\{x\}) = S(x)$

At each iteration a candidate domain X is popped from the priority queue, and split into subintervals. A new upper bound for each subinterval is computed and inserted in the priority queue. The bounding function drives BB to the most promising intervals until the first singleton interval, say x, is reached. Since the bound is tight for singletons, we know that the solutions contained in the remaining intervals of the priority queue will score below x, since the upper bound of their scores is below the returned singleton's score. This guarantees that once a singleton is popped from the priority queue, it will be the global maximum of S.

2.3 Bounding the KDE Score

Having phrased the general setting of Branch-and-Bound, we now turn to how we can apply it to mode finding for Kernel Density Estimation; the main mathematical construct that we need is a bound on the score of a KDE within an interval of solutions. Namely, we need a function $\overline{S}(X)$ which gives us for an interval X an upper bound to the score of the KDE score within X:

$$\overline{S}(X) \geq \max_{x_j \in X} \sum_{x_i \in X'}^{n} w_i K_h \left(x_j - x_i\right) = S(X). \tag{5}$$

We call points contained in X' the source locations and points in X the domain locations, with the intuition that the points in X' contribute to a score in X [14].

We now decompose the computation of the upper bound in Eq. 5 into smaller parts by using the partitions $X = \cup_{d \in D} X_d$ and $X' = \cup_{s \in S} X$. Our decomposition is based on the fact that $\max_x f(x) + g(x) \leq \max_x f(x) + \max_x g(x)$. For Eq. 5 this means that if we separately maximize some of the summands and add them back, this gives us something that will be larger than $S(X)$ (and as such, a valid candidate for $\overline{S}(X)$).

Based on this observation we introduce the quantity:

$$\mu_d^s = \max_{x_j \in X_d} \sum_{x_i \in X_s} w_i K_h \left(x_j - x_i\right) \tag{6}$$

and upper bound the right-hand side of Eq. 5 as:

$$S(X) \leq \max_d \sum_s \mu_d^s \tag{7}$$

where we have brought the summation over s outside the maximization over x_j. This means that if we can construct separately upper bounds to μ_d^s, we can add them up and obtain a valid upper bound for $S(X)$. This will then be used by Branch-and-Bound to prioritize the search over intervals that contain the density's mode.

In particular, we can upper bound $\mu_X^{X'}$ with $\overline{\mu}_X^{X'}$ as follows:

$$\overline{\mu}_X^{X'} \doteq \left(\sum_{i \in X'_s} w_i\right) \max_{i \in X} \max_{j \in X'} K(x_i, x_j) \tag{8}$$

Thus, the upper bound $\overline{S}(X)$ for $S(X)$ can be written as:

$$\overline{S}(X) \doteq \max_X \sum_{X'} \overline{\mu}_X^{X'} \geq S(X) \tag{9}$$

The first term in Eq. 8 can be computed rapidly over large intervals using fine-to-coarse summation. The second term can also be efficiently computed by exploiting the fact that X and X' are hyperrecangles, as also illustrated in the right of Fig. 2, and detailed in [14,11].

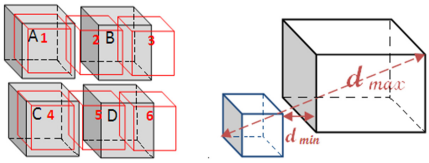


Fig. 2. Left: an example of Dual Trees interaction: points belonging to Source node 6 have insignificant contribution to the objective sum computed in Domain points in node A. Right: Distance bounds between nodes in dual trees.

3 3D Brain Tumor Segmentation

Once a region of interest is efficiently selected, we proceed to segmentation in order to delineate the tumor from the normal tissue. Many segmentation methods have been proposed in the literature for tumor segmentation. MRF based-segmentation [15] has proved its performance and robustness in many applications. Therefore, we formulate the task of tumor segmentation from the 3D volume of interest as a discrete energy minimization problem. The 3D volume V is viewed as a lattice $\{\vartheta, \varepsilon\}$ where each voxel v_p forms a node in the graph. The MRF energy is written as:

$$E(V) = \sum_{p \in \vartheta} \Theta_p(v_p) + \sum_{(p,q) \in \varepsilon} \Theta_{pq}(v_p, v_q) \tag{10}$$

The function f serves as the unary potential energy. In order to improve the classification results, we use a regularization expressed by the binary potential energy. The conventional 4-neighborhood system is extended in 3D so that each voxel has 8 neighbors. We consider, in this work, the Potts model modulated by the contrast of normalized intensities as our regularization function.

This global criterion measures both the total dissimilarity among the two groups and the total similarity inside them. Global minimum of the considered energy is efficiently computed with the graph cut/max flow minimization algorithm[15,16].

4 Experimental Evaluation

To evaluate our method on a real dataset, we use adaboost to provide us with the scores of individual voxels. It is based on the idea that a combination of weak classifiers such as decision trees can create a strong classifier. We use 40 randomly selected images to train the classifier with the following features: normalized intensities, locations (x,y,z), intensities of smoothed image at 3 half octave scales, gradient magnitude, Laplacian of Gaussian features at 3 half octave scales, absolute of Laplacian of Gaussian features at 3 different scales. Our classifier was trained with 50 rounds of boosting and we employed Decision Tress of Depth 3.

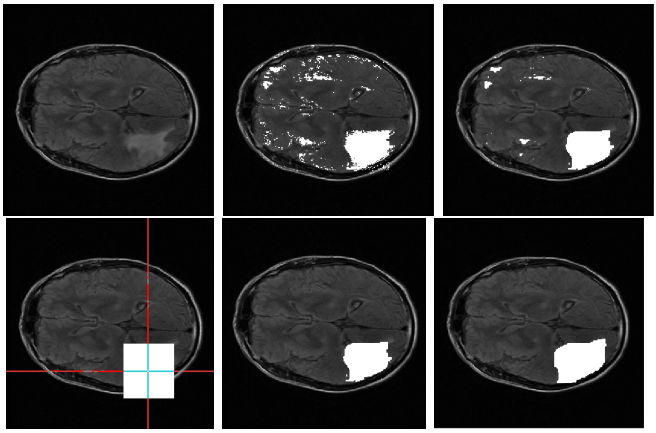


Fig. 3. First row: a) patient MRI, b) adaboost segmentation result, c) graph cut result considering the whole volume. Second row: a) our method: bounding box b) our method segmentation results, c) ground truth segmentation.

Table 1. Average computational time comparison

image size	profile	our method	Mean-Shift	exhaustive search
256x256x24	detection	2.5s	31s	60s
	overall time	17s	46.5s	75.5s
512x512x33	detection	8s	223s	319s
	overall time	93s	293.5s	389s

4.1 The Dataset

We did our experiments on a dataset composed of 113 patients with low grade gliomas. The patient age ranged from 21 to 65 years, and tumor size between 3.5 and 250 cm^3. The MRI volume size varied from 256x256x24 to 512x512x33. The voxel resolution ranged from 0.4x0.4 to 0.9x0.9 mm^2 in the (x,y) plane and 5.3 to 6.4 mm in the z plane. The 3D images were rigidly aligned using *medInria* [17]. The dataset comes with a manual segmentation of the gliomas tumor done by experts, which is considered as our ground truth data.

4.2 Validation Methodology

In order to assess the quality of the segmentation results, we compute the Dice similarity coefficient, which reflects the overlapping rate between the segmented volume and the volume defined by experts. We evaluate the efficiency of our algorithm by comparing the computational time of the detection part with the Mean-Shift procedure and convolving the 3D volume with the kernel. Since the tumor size can achieve 250 cm^3 we convolve with an Epanechnikov kernel whose

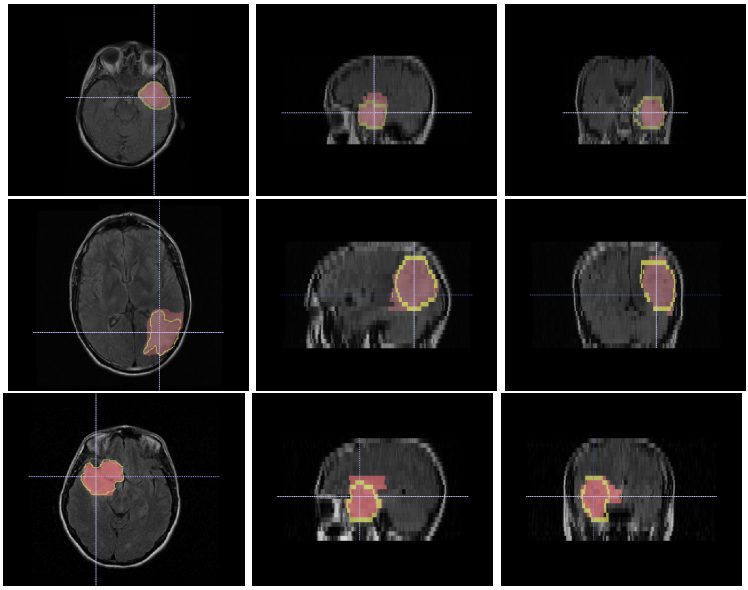


Fig. 4. Segmentation results. The manual segmentation (yellow) is superimposed to the automatic segmentation (red).

scale equals nearly the quarter of the volume size. This value matches the maximal size of the ground truth segmented gliomas. We use the most efficient available CPU version of convolution. The used package detects automatically if the kernel is separable and optimizes the convolution computation.

4.3 Results

The average Dice computed on our database is 0.73 (cf. Fig. 5) which is comparable to the results produced by the state of the art methods [18,2]. we report from [18] that the computational cost is between 20 and 120 seconds and the average DICE coefficient is 0.77. Our average computational time is 19 seconds. Mode estimation is a principal ingredient of the proposed method, as the results become less accurate if we only use either adaboost classifier or graph-cuts (cf. Fig. 5, Fig. 4). We compare the computational time between our work, a standard implementation of Mean Shift and an exhaustive search over volume locations after evaluating KDE in all locations cf. Table 1. We run the algorithms on a 4-core Intel Xeon computer which frequence is 2.67GHz. We use, though, a single core in the computation.

5 Discussion

While our method was inspired from previous work [14,11], we recall that[14] does not use Branch-and-Bound and provides a technique for the efficient

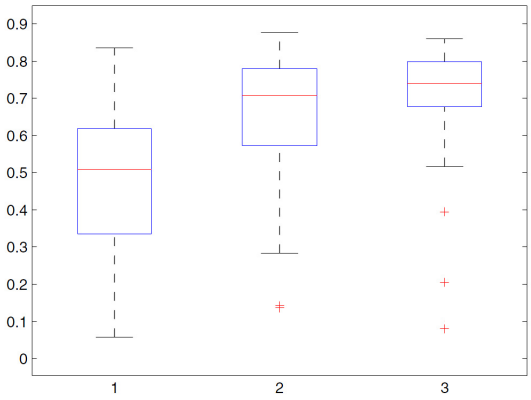


Fig. 5. Boxplots of the Dice values. From left to right: segmentation results with boosting only, boosting and pairwise regularization, boosting, rapid mode estimation and pairwise regularization.

computation of a KDE score everywhere, on all 'domain points'. Similarly, the multipole method [19] evaluates a KDE on all candidate locations, and is thus linear in the number of points. The aforementioned methods are excellent for KDE evaluation- but for mode estimation they perform an 'overkill', since they exactly evaluate the score everywhere, while we only want the location of the maximum. Instead our technique searches directly for the maximum location, and effectively 'ignores' less promising locations. In particular we discard chunks of points quickly by using cheaply computable upper bounds to their score. This allows us to work in time sublinear (practically logarithmic) in the number of possible locations. This is crucial for 3D medical data, where increasing the resolution by a factor of 2 will result in an 8-fold slowdown for Multipole/Dual Trees, but will require only 3 more iterations for our method (cf. Table. 1).

To the best of our knowledge, branch and bound has not been used before for mode estimation of KDEs. It was used in [11] for Object Detection, but with a different score function. We thus expect that our work will be of interest to other researchers working on mode estimation.

6 Conclusion

In this paper, we have presented a Branch-and-Bound based method for efficient mode estimation in KDE. We used our method for brain tumor detection and segmentation of 3D MR images. We demonstrate that our method results in a 12-fold speedup over standard Mean-Shift. Our approach is more robust than applying graph cut on the whole volume. The largest part of the computational time is taken by feature computation which can easily accelerated with graphic

processing units. Future directions include evaluating and adapting the proposed approach to the 3D liver tumor tracking in radiation therapy where real time is crucial.

Acknowledgements. This work has been funded by grant ANR-10-JCJC-0205; and by the EU Project MOBOT FP7-ICT-2011-600796. We would like to thank Hugues Duffau for providing us with the 3D MRI brain tumor dataset.

References

1. Duffau, H., Capelle, L.: Preferential brain locations of low-grade gliomas. Cancer 100, 2622–2626 (2004)
2. Parisot, S., Duffau, H., Chemouny, S., Paragios, N.: Graph-based detection, segmentation & characterization of brain tumors. In: CVPR, pp. 988–995 (2012)
3. Comaniciu, D., Meer, P.: Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. 24(5), 603–619 (2002)
4. van Walsum, T., Schaap, M., Metz, C.T., van der Giessen, A.G., Niessen, W.J.: Averaging centerlines mean shift on paths. In: Metaxas, D., Axel, L., Fichtinger, G., Székely, G. (eds.) MICCAI 2008, Part I. LNCS, vol. 5241, pp. 900–907. Springer, Heidelberg (2008)
5. Prima, D.G.L.S., Collins, D.L., Arnold, D.L., Morrissey, S.P., Barillot, C.: Combining robust expectation maximization and mean shift algorithms for multiple sclerosis brain segmentation. In: MICCAI Workshop on Medical Image Analysis on Multiple Sclerosis (2008)
6. Mayer, A., Greenspan, H.: An adaptive mean-shift framework for mri brain segmentation. IEEE Transactions on Medical Imaging (2009)
7. Sheikh, Y.A., Khan, E.: Kanade, T.: Mode-seeking by medoidshifts. In: ICCV, Number 141 (2007)
8. Vedaldi, A., Soatto, S.: Quick shift and kernel methods for mode seeking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 705–718. Springer, Heidelberg (2008)
9. Yang, C., Duraiswami, R., Gumerov, N.A., Davis, L.S.: Improved fast gauss transform and efficient kernel density estimation. In: ICCV, pp. 464–471 (2003)
10. Wang, P., Lee, D., Gray, A.G., Rehg, J.M.: Fast mean shift with accurate and stable convergence. Journal of Machine Learning Research - Proceedings Track 2, 604–611 (2007)
11. Kokkinos, I.: Rapid deformable object detection using dual-tree branch-and-bound. In: Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K. (eds.) NIPS, pp. 2681–2689 (2011)
12. Birkbeck, N., Cobzas, D., Jägersand, M., Murtha, A., Kesztyues, T.: An interactive graph cut method for brain tumor segmentation. In: WACV, pp. 1–7 (2009)
13. Scott, D.: Multivariate density estimation: theory, practice, and visualization. Wiley series in probability and mathematical statistics: Applied probability and statistics. Wiley (1992)
14. Gray, A.G.: Nonparametric density estimation: toward computational tractability. In: SIAM International Conference on Data Mining (2003)
15. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans. Pattern Anal. Mach. Intell. 26, 1124–1137 (2004)

16. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts. PAMI 26, 65–81 (2004)
17. Toussaint, N., Souplet, J.C., Fillard, P.: Medinria: Medical image navigation and research tool by inria. In: Proc. of MICCAI 2007 Workshop on Interaction in Medical Image Analysis and Visualization (2007)
18. Bauer, S., Nolte, L.-P., Reyes, M.: Fully automatic segmentation of brain tumor images using support vector machine classification in combination with hierarchical conditional random field regularization. In: Fichtinger, G., Martel, A., Peters, T. (eds.) MICCAI 2011, Part III. LNCS, vol. 6893, pp. 354–361. Springer, Heidelberg (2011)
19. Engheta, N., Murphy, W.D., Rokhlin, V., Vassiliou, M.: The fast multipole method for electromagnetic scattering computation. IEEE Transactions on Antennas and Propagation 40, 634–641 (1985)

Jointly Segmenting Prostate Zones in 3D MRIs by Globally Optimized Coupled Level-Sets

Jing Yuan[1], Eranga Ukwatta[1], Wu Qiu[1], Martin Rajchl[1], Yue Sun[1], Xue-Cheng Tai[2], and Aaron Fenster[1]

[1] Medical Imaging Labs, Robarts Research Institute,
University of Western Ontario,
London, Ontario, Canada N6A 5K8
{jyuan,eukwatta,wqiu,mrajchl,ysun,afenster}@robarts.ca
[2] Mathematics Department, University of Bergen,
Bergen, Norway
tai@math.uib.no

Abstract. It is of great interest in image-guided prostate interventions and diagnosis of prostate cancer to accurately and efficiently delineate the boundaries of prostate, especially its two clinically meaningful sub-regions/zones of the central gland (CZ) and the peripheral zone (PZ), in the given magnetic resonance (MR) images. We propose a novel coupled level-sets/contours evolution approach to simultaneously locating the prostate region and its two sub-regions, which properly introduces the recently developed convex relaxation technique to jointly evolve two coupled level-sets in a global optimization manner. Especially, in contrast to the classical level-set methods, we demonstrate that the two coupled level-sets can be simultaneously moved to their globally optimal positions at each discrete time-frame while preserving the spatial inter-surface consistency; we study the resulting complicated combinatorial optimization problem at each discrete time evolution by means of convex relaxation and show its global and exact optimality, for which we introduce the novel coupled continuous max-flow model and demonstrate its duality to the investigated convex relaxed optimization problem with the region constraint. The proposed coupled continuous max-flow model naturally leads to a new and efficient algorithm, which enjoys great advantages in numerics and can be readily implemented on GPUs. Experiments over 10 T2-weighted 3D prostate MRIs, by inter- and intra-operator variability, demonstrate the promising performance of the proposed approach.

Keywords: Convex Optimization, 3D Prostate Zonal Segmentation.

1 Introduction

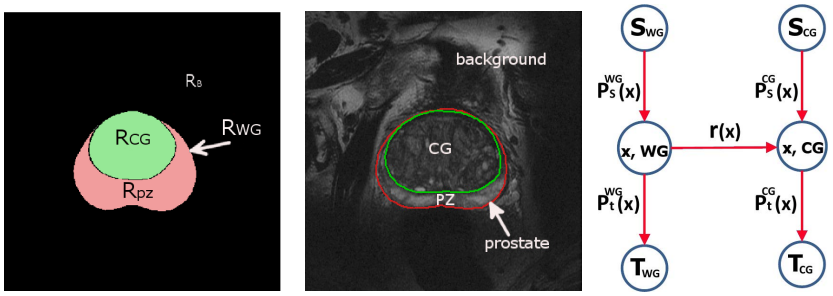
Prostate cancer is one of major health problems in the western world, with one in six men affected during their lifetime [1]. In diagnosing prostate cancer, transrectal ultrasound (TRUS) guided biopsies have become the gold standard. However, the accuracy of the TRUS guided biopsy relies on and is limited by

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 12–25, 2013.
© Springer-Verlag Berlin Heidelberg 2013

the fidelity. Magnetic resonance (MR) imaging is an attractive option for guiding and monitoring such interventions due to its superior visualization of not only the prostate, but also its substructure and surrounding tissues [2, 3]. The fusion of 3D TRUS and MRI provides an effective way to target biopsy needles in the 3D TRUS image toward the prostate region containing MR identified suspicious lesions, which is regarded as an alternative to the more expensive and inefficient MRI-based prostate biopsy [4] and the less accurate conventional 2D TRUS-guieded prostate biopsy. On the other hand, during guidance of the biopsy, the prostate region is usually recognized by two visually meaningful subregions in a prostate MRI: the *central gland* (CG) and the *peripheral zone* (PZ) [5], and up to 80% of prostate cancers are located within the PZ region [6]. The ability to superimpose the 3D TRUS image used to guide the biopsy onto these pre-segmented prostate zones(subregions) of interest in MRIs is highly desired in a fused 3D TRUS/MRI guided biopsy system. In addition, computer aided diagnosis (CAD) techniques for prostate cancer can also benefit from the correct interpretation of the prostate zonal anatomy since the occurrence and appearance of the cancer depends on its zonal location [7, 8]; and the ratio of CG volume to whole prostate gland (WG) can be used to monitor prostate hyperplasia [9]. To this end, efficient and accurate extraction of the prostate region, in particular its sub-regions of CG and PZ, from 3D prostate MRIs is of great interest in both image-guided prostate interventions and diagnosis of prostate cancer.

Many studies focused their efforts on the segmentation of the whole prostate in 3D MR images (especially in T2w 3D MRIs), see [10] for a review; where the obvious intensity inhomogeneity of prostate makes the segmentation task challenging. However, only few studies focused on the segmentation of the prostate sub-regions/zones in 3D MRIs. Allen *et al.* [11] proposed a method for the automatic delineation of the prostate boundaries and CG, which was limited to the middle region of the prostate (where T2w contrast permits accurate segmentation), and ignored the apex and base of the gland. Yin *et al.* [12] proposed an automated CG segmentation algorithm based on Layered Optimal Graph Image Segmentation of Multiple Objects and Surfaces (LOGISMOS). The first paper about segmenting the prostate into two regions of PZ and CG was proposed by Makni *et al.* [13]. The authors proposed a modified version of the evidential C-means algorithm to cluster voxels into their respective zones incorporating the spatial relation between voxels in 3D multispectral MRIs including a T2w image, a diffusion weighted image (DWI), and a contrast enhanced MRI (CEMRI). More recently, Litjens *et al.* [14] proposed a pattern recognition method to classify the voxels using anatomical, intensity and texture features in multispectral MRIs. However, in [13] and [14], the segmentation of prostate peripheral zone relies on the manual segmentation of the whole prostate gland.

Contributions: Based on recent developments of the new global optimization technique to the single level-set/contour propagation [15–17], we propose a new global optimization-based coupled level-set evolution approach to delineating the whole prostate gland (WG) and its subregions of CG and PZ jointly from a



(a) (b) (c)

Fig. 1. (a) shows the proposed layout of anatomically consistent regions: the whole prostate(WG) $\mathcal{R}_{WG}$ and its two zones: central gland(CG) $\mathcal{R}_{CG}$ and peripheral zone(PZ) $\mathcal{R}_{PZ}$, which are mutually distinct from the background region R_B. (b) illustrates the segmented contours overlaid on a T2w prostate MRI slice.

single input 3D T2 weighted prostate MR image. The proposed method matches the intensity distribution models of the two prostate sub-regions CG and PZ to guide the simultaneous propagation of two coupled level-sets. We efficiently and globally solve the resulted challenging combinatorial optimization problem, the so-called *coupled min-cut model*, during each discrete time evolution by means of convex relaxation. We propose a novel spatially continuous flow-maximization model, i.e. the *coupled continuous max-flow model*, and demonstrate its duality to the studied convex relaxed optimization problem with the region consistency constraint. The *coupled continuous max-flow model* directly leads to a new and efficient *continuous max-flow based* algorithm, which enjoys great advantages in numerics and can be readily implemented on GPUs. Experiments over 10 T2-weighted 3D prostate MRIs, by inter- and intra-operator variability, demonstrate the promising performance of the proposed approach. The proposed method can be easily applied to other image segmentation tasks.

The classical level-set methods [18] are based on locally computing the associated convention PDE in a time-explicit manner and converge slowly; especially, an extremely complex scheme is required for correctly propagating multi-class level-sets. In contrast, the global optimization based contour evolution technique introduces a new implicit-time contour convenction scheme which allows the large time step-size to accelerate convergence, and the inter-level-set contraints can be easily adapted into the propagation process in a global optimization way (as the proposed approach in this work).

2 Global Optimization to Coupled Contour Evolution

Now we target to segment a given 3D T2w prostate MR image $I(x)$ into the prostate region $\mathcal{R}_{WG}$ together with its two mutually distinct sub-regions: the

central gland $\mathcal{R}_{CG}$ and the peripheral zone $\mathcal{R}_{PZ}$, where $\mathcal{R}_B$ denotes the background (see Figure 1(a)), i.e.

$$\Omega \,=\, \mathcal{R}_{WG} \cup \mathcal{R}_B\,, \quad \mathcal{R}_{WG} \cap \mathcal{R}_B \,=\, \emptyset\,, \tag{1}$$

where the two spatially coherent sub-regions: the R_{CG} and R_{PZ} constitute the whole prostate region $\mathcal{R}_{WG}$ such that

$$\mathcal{R}_{WG} \,=\, \mathcal{R}_{CG} \cup \mathcal{R}_{PZ}\,; \quad \mathcal{R}_{CG} \cap \mathcal{R}_{PZ} \,=\, \emptyset\,. \tag{2}$$

In this context, we propose a novel global optimization based approach to jointly evolving two coupled contours $\mathcal{C}^{WG}$ and $\mathcal{C}^{CG}$ to the correct boundaries of the prostate and the central gland, while keeping the inter-contour relationship

$$\mathcal{R}_{CG} \,\subset\, \mathcal{R}_{WG}\,; \tag{3}$$

i.e. the inclusion region $\mathcal{R}_{CG}$ of $\mathcal{C}^{CG}$ is covered by the inclusion region $\mathcal{R}_{WG}$ of $\mathcal{C}^{WG}$. Clearly, once the two contours $\mathcal{C}^{CG}$ and $\mathcal{C}^{WG}$ are computed, the peripheral zone $\mathcal{R}_{PZ}$ is determined by the complementary region $\mathcal{R}_{WG} \backslash \mathcal{R}_{CG}$. We show that the resulting combinatorial optimization at each discrete-time contour propagation can be solved globally and exactly by convex relaxation, which means that the two contours can be moved to their 'best' positions during each discrete time evolution. With this respect, we propose and investigate a unified framework in terms of *coupled continuous max-flow model*. In addition, the new optimization theory can be used to drive an efficient *coupled continuous max-flow* based algorithms, which have great numerical advantages and can be readily implemented on graphics processing units (GPU) to achieve a high computation performance. The proposed optimization theory and algorithm can be directly extended to the general case of evolving $n > 2$ contours $\mathcal{C}^i$, $i = 1 \ldots n$, while preserving the order $\mathcal{C}_t^n \subset \mathcal{C}_t^{n-1} \ldots \subset \mathcal{C}_t^1$; and also applied to other image segmentation applications.

2.1 Matching Multiple Intensity Distribution Models

One major challenge to segment a typical T2w prostate MR image is the strong intensity inhomogeneity of prostate (see Figure 1(a)), where the zones R_{CG} and R_{PZ} of $\mathcal{R}_{WG}$ have their distinct intensity appearances, hence constitute the complex appearance model of the prostate region $\mathcal{R}_{WG}$. In this work, we propose to model the intensity appearance of the prostate region $\mathcal{R}_{WG}$ by the two independent appearance models of its two sub-regions $\mathcal{R}_{CG}$ and $\mathcal{R}_{PZ}$, which are distinct to each other. This sets up a proper composite appearance description of the entire prostate region $\mathcal{R}_{WG}$ in practice. Such a composite intensity appearance model is shown to be more accurate than the often-used mixture appearance model in practice [19]. Indeed, the two separated appearance models can be obtained much easier and more accurately, with less influence by sampling statistics, than the direct mixture appearence model of $\mathcal{R}_{WG}$. We propose to match the two distinct appearance models of the prostate sub-regions R_{CG} and R_{PZ} in stead of the mixture model of the prostate $\mathcal{R}_{WG}$.

Let $\pi_i(z)$, $i \in \{CG, PZ\}$, be the intensity probability density function (PDF) of the respective prostate sub-region $\mathcal{R}_i$ and $z \in \mathcal{Z}$ gives the photometric value of intensities. Also, let $\pi_B(I(x))$ be the PDF of the background region $\mathcal{R}_B$. In practice, such PDFs of intensities of the interesting object regions provide a reliable and global description of the segmented objects [20], which can be learned from either sampled pixels or given training datasets.

Given the indicator functions $u_i(x) \in \{0, 1\}$, $i \in \{CG, WG\}$, of the inclusion region of the contour $\mathcal{C}^i$:

$$u_i(x) := \begin{cases} 1\,, \text{where } x \text{ is inside } \mathcal{C}^i \\ 0\,, \text{otherwise} \end{cases}, \quad i \in \{CG, WG\}\,, \tag{4}$$

the Bhattacharyya distance [20] is used for matching the PDFs of the three distinct regions: $\mathcal{R}_{CG}$, $\mathcal{R}_{PZ}$ and $\mathcal{R}_B$; which results in the following model-matching term:

$$E_m(u) = -\sum_{z \in \mathcal{Z}} \left\{ \sqrt{\pi_{CG}(z)\,\phi_{CG}(z)} + \sqrt{\pi_{PZ}(z)\,\phi_{PZ}(z)} + \sqrt{\pi_B(z)\,\phi_B(z)} \right\} \tag{5}$$

where $\phi_{CG,PZ,B}(u, z)$ are the respective PDFs for the estimated regions of $\mathcal{R}_{CG}$, $\mathcal{R}_{PZ}$ and $\mathcal{R}_B$, and computed by the Parzen method:

$$\phi_{CG}(z) = \frac{\int_\Omega K(z - I(x))\, u_{CG}\, dx}{\int u_{CG}\, dx}\,, \quad \phi_{PZ}(z) = \frac{\int_\Omega K(z - I(x))\, (u_{WG} - u_{CG})\, dx}{\int (u_{WG} - u_{CG})\, dx}$$

and

$$\phi_B(z) \;=\; \frac{\int_\Omega K(z - I(x))\, (1 - u_{WG})\, dx}{\int (1 - u_{WG})\, dx}$$

where $K(\cdot)$ is the Gaussian kernel function $K(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-x^2/2\sigma^2)$.

Optimization Model: In view of the histogram matching energy function (5) and region constraint (3), we propose to compute the region indictor functions $u_{CG}(x), u_{WG}(x) \in \{0, 1\}$ by minimizing the following energy function

$$\min_{u_{CG}, u_{WG}(x) \in \{0,1\}} E_m(u) \;+\; \int_\Omega g(x)\, |\nabla u_{CG}(x)|\; dx \;+\; \int_\Omega g(x)\, |\nabla u_{WG}(x)|\; dx \tag{6}$$

subject to the inter-region constraint

$$u_{CG}(x) \;\leq\; u_{WG}(x)\,, \quad \forall x \;\in\; \Omega\,; \tag{7}$$

where the total-variation functions properly approximates the weighted areas of $\mathcal{R}_{CG}$ and $\mathcal{R}_{WG}$ and (7) corresponds to (3).

2.2 Global Optimization and Coupled Contour Evolution

Now we study the optimization problem (6) and introduce a novel global optimization based approach to evolving the two contours $\mathcal{C}^{CG}$ and $\mathcal{C}^{WG}$, w.r.t. $\mathcal{R}_{CG}$ and $\mathcal{R}_{WG}$, simultaneously while preserving the constraint (3).

Single Contour Evolution and Min-Cut: In contrast to the classical level-set evolution theory, the recent developments [16, 17] of the global optimization theory to the evolution of a single contour $\mathcal{C}$ proves that the propagation of the contour $\mathcal{C}_t$ at time t to its new position $\mathcal{C}_{t+h}$ at time $t+h$ can be modeled and globally optimized in terms of computing the *min-cut* problem:

$$\mathcal{C}_{t+h} \;:=\; \min_{\mathcal{C}} \int_{\mathcal{C}^+} c^+(x)\,dx \;+\; \int_{\mathcal{C}^-} c^-(x)\,dx \;+\; \int_{\partial\mathcal{C}} g(s)\,ds\,, \tag{8}$$

where

1. $\mathcal{C}^+$ indicates the region expansion w.r.t. $\mathcal{C}_t$: for $\forall x \in \mathcal{C}^+$, it is initially outside $\mathcal{C}_t$ at time t, and 'jumps' to be inside $\mathcal{C}_{t+h}$ at $t+h$; for such a 'jump', it pays the cost:
$$c^+(x) = \big(\mathrm{dist}(x, \partial\mathcal{C}_t) + f(x)\big)/h\,; \tag{9}$$
2. $\mathcal{C}^-$ indicates the region shrinkage w.r.t. $\mathcal{C}_t$: for $\forall x \in \mathcal{C}^-$, it is initially inside $\mathcal{C}_t$ at t, and 'jumps' to be outside $\mathcal{C}_{t+h}$ at $t+h$; for such a 'jump', it pays the cost $c^-(x)$:
$$c^-(x) = \big(\mathrm{dist}(x, \partial\mathcal{C}_t) - f(x)\big)/h\,. \tag{10}$$

The function $\mathrm{dist}(x, \partial\mathcal{C}_t)$ gives the distance of any $x \in \Omega$ to the current contour $\mathcal{C}_t$, where Ω is the image domain; the outer force function $f(x)$ is data-associated and is chosen based on the specified application: for example, $f(x)$ can be defined using the first-order variation of the distribution matching function, e.g. the Bhattacharyya distance. Obviously, the time step-size h is implicitly adapted in the cost functions (9)-(10), which allows a large value in numerical practice to speed-up the evolution of contours towards convergence.

To be more clear, we define the cost functions $D_s(x)$ and $D_t(x)$ as follows:

$$D_s(x) \;:=\; \begin{cases} c^-(x)\,, & \text{where } x \in \mathcal{C}_t \\ 0\,, & \text{otherwise} \end{cases}\,, \quad D_t(x) \;:=\; \begin{cases} c^+(x)\,, & \text{where } x \notin \mathcal{C}_t \\ 0\,, & \text{otherwise} \end{cases}\;; \tag{11}$$

which can be interpreted as the cost of assigning each pixel x to be foreground or background, respectively. Let $u(x) \in \{0, 1\}$ be the labeling function of the new contour $\mathcal{C}$ in (8). Therefore, the proposed optimization model (8) to contour evolution can be equally reformulated as the *min-cut model*:

$$\min_{u(x)\in\{0,1\}} \; \langle 1-u, D_s\rangle + \langle u, D_t\rangle + \int_{\Omega} g(x)\,|\nabla u|\;dx\,, \tag{12}$$

which can be solved globally and exactly with various efficient algorithms of graph-cut and convex optimization!

Coupled Contour Evolution and Coupled Min-Cuts: Following the same ideas of (8), for the evolution of the contour $\mathcal{C}^{CG}$ during the discrete time-frame from t to $t+h$, we, correspondingly, define the cost functions $c^+_{CG}(x)$ and $c^-_{CG}(x)$ w.r.t. region expansion $\mathcal{C}^+_{CG}$ and shrinkage $\mathcal{C}^-_{CG}$; for the evolution of $\mathcal{C}^{WG}$, we

define $c^+_{WG}(x)$ and $c^-_{WG}(x)$ as the respective costs to region changes. Therefore, we optimize the problem (6) by evolving the two contours $\mathcal{C}^{CG}$ and $\mathcal{C}^{WG}$, which propagates the contours $\mathcal{C}^{CG}_t$ and $\mathcal{C}^{WG}_t$ at time t to their new positions $\mathcal{C}^{CG}_{t+h}$ and $\mathcal{C}^{WG}_{t+h}$ while preserving the constraint (3): $\mathcal{R}_{CG} \subset \mathcal{R}_{WG}$ by the following minimization problem:

$$\min_{\mathcal{C}^{CG},\mathcal{C}^{WG}} \sum_{i\in\{CG,WG\}} \Big\{ \int_{\mathcal{C}^+_i} c^+_i(x) \, + \int_{\mathcal{C}^-_i} c^-_i(x) \, + \int_{\partial\mathcal{C}^i} g(s) \Big\} \tag{13}$$

subject to the region constraint $\mathcal{R}_{CG} \subset \mathcal{R}_{WG}$.

Similar as (11), we define the label assignment cost functions $D^i_s(x)$ and $D^i_t(x)$, $i \in \{CG, WG\}$, such that:

$$D^i_s(x) := \begin{cases} c^-_i(x)\,, \text{ where } x \in \mathcal{C}^i_t \\ 0\,, \qquad \text{otherwise} \end{cases}, \quad D^i_t(x) := \begin{cases} c^+_i(x)\,, \text{ where } x \notin \mathcal{C}^i_t \\ 0\,, \qquad \text{otherwise} \end{cases}. \tag{14}$$

In consequence, the optimization problem (13) can be equally represented by

$$\min_{u_{CG,WG}(x)\in\{0,1\}} \sum_{i\in\{CG,WG\}} \Big\{ \langle 1-u_i, D^i_s\rangle + \langle u_i, D^i_t\rangle + \int_\Omega g\,|\nabla u_i|\; dx \Big\} \tag{15}$$

subject to the linear inequality region constraint (7), i.e. $u_{CG}(x) \leq u_{WG}(x)$.

Clearly, without the constraint (7), the formulation (15) gives rise to two independent min-cut problems; the region constraint (7) conjoins these two independent min-cut problems with each other. Hence, in this paper, we call (15) the model of *coupled min-cuts* in the spatially continuous setting, i.e. the *coupled continuous min-cut model*.

Convex Relaxation and Coupled Continuous Max-Flow Model: In this work, we investigate the proposed challenging combinatorial optimization problems (15) by convex relaxation, where the binary constrained labeling functions $u_{CG,WG}(x) \in \{0, 1\}$ are relaxed to be the convex constraint $u_{CG,WG}(x) \in [0, 1]$. Therefore, we have the corresponding convex relaxation problem:

$$\min_{u_{CG,WG}\in[0,1]} \sum_{i\in\{CG,WG\}} \Big\{ \langle 1-u_i, D^i_s\rangle + \langle u_i, D^i_t\rangle + \int_\Omega g\,|\nabla u_i|\; dx \Big\}, \tag{16}$$

subject to the region constraint (7).

In the following sections, we propose the novel dual model to the convex optimization problem (16), which corresponds to maximize the streaming flows upon a novel coupled flow-maximization setting, i.e. the *coupled continuous max-flow model*. With help of the new *coupled continuous max-flow model*, we show the convex relaxed optimization problem (16) solves the original combinatorial optimization (16) exactly and globally! This means the two coupled contours $\mathcal{C}^{CG}$ and $\mathcal{C}^{WG}$ can be moved to their globally optimized positions, i.e. best positions, during each discrete time frame. In addition, we can directly derive the new *coupled continuous max-flow algorithm* which avoids tackling the non-smooth function terms and linear constraint in (16) and enjoys a fast convergence.

Coupled Continuous Max-Flow Model: To motivate the *coupled continuous max-flow model*, we introduce a novel flow configuration (shown in Fig. 1(c)), which is the combination of two independent standard flow-maximization settings (see [21, 22] etc.), linked by an additional directed flow $r(x)$ in-between:

- We set up two copies Ω_{CG} and Ω_{WG} of Ω w.r.t. the essential two continuous min-cuts; for Ω_i, $i \in \{CG, WG\}$, two extra nodes s_i and t_i are added as the source and sink terminals; we link s_i to each pixel $x \in \Omega_i$ and link x to t_i, see Fig. 1(c); moreover, within each of Ω_i, $i \in \{CG, WG\}$, we define the source flow $p_s^i(x)$ which is directed from s_i to each x and the sink flow $p_t^i(x)$ which is directed from each x to t_i; also, within each of Ω_i, there is the spatial flow $q^i(x)$ around each pixel x.
- At each pixel x, there exists an extra flow $r(x)$ directed from Ω_2 to Ω_1.

For the two source flow fields $p_s^{CG}(x)$ and $p_s^{WG}(x)$, we define the flow capacity constraints:

$$p_s^{CG}(x) \,\leq\, D_s^{CG}(x)\,, \quad p_s^{WG}(x) \,\leq\, D_s^{WG}(x)\,; \quad \forall x \,\in\, \Omega\,. \tag{17}$$

Likewise, for the two sink flow fields: $p_t^{CG}(x)$ and $p_t^{WG}(x)$, and the spatial flows: $q^{CG}(x)$ and $q^{WG}(x)$, we define the respective flow capacity constraints:

$$p_t^{CG}(x) \,\leq\, D_t^{CG}(x)\,, \quad p_t^{WG}(x) \,\leq\, D_t^{WG}(x)\,; \quad \forall x \,\in\, \Omega\,; \tag{18}$$

and

$$\left|q^{CG}(x)\right| \,\leq\, g(x)\,, \quad \left|q^{WG}(x)\right| \,\leq\, g(x)\,; \quad \forall x \,\in\, \Omega\,. \tag{19}$$

Moreover, the extra directed flow field $r(x)$ for each x at Ω_{WG} to the same position at Ω_{CG} is constrained by

$$r(x) \,\geq\, 0\,; \quad \forall x \,\in\, \Omega\,. \tag{20}$$

In addition to the above flow capacity constraints, at each pixel $x \in \Omega_i$, $i \in \{CG, WG\}$, all the flow fields $p_s^i(x)$, $p_t^i(x)$, $q_i(x)$ and $r(x)$ are balanced such that

$$R_{CG}(x) \,:=\, \mathrm{div}\, q^{CG}(x) \,+\, p_t^{CG}(x) \,-\, p_s^{CG}(x) \,-\, r(x) \,=\, 0\,; \tag{21}$$

and

$$R_{WG}(x) \,:=\, \mathrm{div}\, q^{WG}(x) \,+\, p_t^{WG}(x) \,-\, p_s^{WG}(x) \,+\, r(x) \,=\, 0\,. \tag{22}$$

Therefore, we propose the novel *coupled continuous max-flow model* which achieves the maximum total flows directed from s_1 and s_2, i.e.

$$\max_{p_s, p_t, q, r} \int_\Omega p_s^{CG}\, dx \,+\, \int_\Omega p_s^{WG}\, dx \tag{23}$$

subject to the flow capacity conditions (17) - (20) and the flow conservation conditions (21) and (22).

Duality and Global Optimum to (15)**:** Note that (23) provides two independent continuous max-flow problems, which are linked by the extra directed flow field $r(x)$. We can prove the duality between the *coupled continuous max-flow model* (23) and the convex relaxed optimization problem (16), (see [22, 23])

Proposition 1. *The proposed* coupled continuous max-flow model (23) *is equivalent or dual to the convex relaxed coupled continuous min-cut formulation* (16)*:*

$$(23) \iff (16)\,.$$

Clearly, for the given convex relaxation problem (16), the global optimum exists. In addition, with helps of the proposed continuous max-flow model (23), we can prove thresholding the global optimum of (16) also solve the original combinatorial optimization problem (15). This means that the two contours $\mathcal{C}^{CG}$ and $\mathcal{C}^{WG}$ can be moved to their 'best' position(s), i.e. the global optimum, during each discrete time frame!

Proposition 2. *Let* $u^*_{CG}(x), u^*_{WG}(x) \in [0,1]$ *be any global optimum of the convex relaxed coupled continuous min-cut formulation* (16)*, their thresholds* $u^\ell_{CG}(x) \in \{0,1\}$ *and* $u^\ell_{WG}(x) \in \{0,1\}$*:*

$$u^\ell_i(x) \;=\; \begin{cases} 1\,, & \textit{when } u^*_i(x) > \ell \\ 0\,, & \textit{when } u^*_i(x) \le \ell \end{cases}\,, \quad i \in \{CG, WG\} \tag{24}$$

for any $\ell \in [0,1)$*, solves the original binary-constrained coupled continuous min-cut problem* (15) *globally and exactly.*

Actually, the functions $u^\ell_{CG}(x), u^\ell_{WG}(x) \in \{0,1\}$ *indicate the new positions of the two thresholded level-sets* $\mathcal{C}^{CG}$ *and* $\mathcal{C}^{WG}$ *respectively, which are the globally optimized contours to* (13)*.*

The proofs of Prop. 1 and Prop. 2 are omitted here due to the limit space.

Coupled Continuous Max-Flow Algorithm: On the other hand, the proposed *coupled continuous max-flow model* (23) naturally leads to an efficient *coupled continuous max-flow based algorithm* in a multiplier-augmented way [24] (similar as [22, 23]). Based on the augmented Lagrangian algorithmic scheme, we introduce the multiplier functions $u_{CG}(x)$ and $u_{WG}(x)$ to (21) and (22) respectively and define the Lagrangian function:

$$L(p_s, p_t, q, r, u) := \int_\Omega p^{CG}_s(x)\,dx + \langle u_{CG}, R_{CG}\rangle + \int_\Omega p^{WG}_s(x)\,dx + \langle u_{WG}, R_{WG}\rangle\,.$$

We also define the following augmented Lagrangian function

$$L_c(p_s, p_t, q, r, u) := L(p_s, p_t, q, r, u) - \frac{c}{2}\,\|R_{CG}\|^2 - \frac{c}{2}\,\|R_{WG}\|^2$$

where $c > 0$ is constant.

We proposed the *coupled continuous max-flow algorithm* which explores the following iteration till convergence: each k-th iteration consists of the flow maximization steps over the flow functions p_s, p_t, q and r and corresponding flow constraints, and the label updating steps:

1. Maximize $L_c(p_s, p_t, q, r, u)$ over the spatial flows $\left|q^i(x)\right| \leq g(x)$, $i \in \{CG, WG\}$, by fixing the other variables, which gives

$$(q^i)^{k+1} := \underset{|q^i(x)|\leq g(x)}{\arg\max} -\frac{c}{2}\left\|\operatorname{div} q^i - F_i^k\right\|^2,$$

where

$$F_i^k(x) = \left((p_s^i)^k - (p_t^i)^k - r^k + (u_i)^k\right)(x)/c\,.$$

It can be implemented by the one-step of gradient-projection procedure [25].

2. Maximize $L_c(p_s, p_t, q, r, u)$ over the source flows $p_s^i(x) \leq D_s^i(x)$, $i \in \{CG, WG\}$, by fixing the other variables, which gives

$$(p_s^i)^{k+1} := \underset{p_s^i(x)\leq D_s^i(x)}{\arg\max} \int_\Omega p_s^i\,dx - \frac{c}{2}\left\|p_s^i - G_i^k\right\|^2,$$

where

$$G_i^k(x) = \left(\operatorname{div} q_i^{k+1} + (p_t^i)^k + r^k - (u_i)^k/c\right)(x)\,.$$

It can be solved exactly by:

$$(p_s^i)^{k+1}(x) \;=\; \min\left(G_i^k(x) + 1/c\,,\; D_s^i(x)\right). \tag{25}$$

3. Maximize $L_c(p_s, p_t, q, r, u)$ over the sink flows $p_t^i(x) \leq D_t^i(x)$, $i \in \{CG, WG\}$, by fixing the other variables, which gives

$$(p_t^i)^{k+1} := \underset{p_t^i(x)\leq D_t^i(x)}{\arg\max} -\frac{c}{2}\left\|p_t^i + H_i^k\right\|^2,$$

where

$$H_i^k(x) = \left(\operatorname{div} q_i^{k+1} - (p_s^i)^{k+1} + r^k - (u_i)^k/c\right)(x)\,.$$

It can be solved exactly by:

$$(p_t^i)^{k+1}(x) \;=\; \min\left(-H_i^k(x)\,,\; D_t^i(x)\right). \tag{26}$$

4. Maximize $L_c(p_s, p_t, q, r, u)$ over the coupled flow field $r(x) \geq 0$ by fixing the other variables, which gives

$$r^{k+1} := \underset{r(x)\geq 0}{\arg\max} -\frac{c}{2}\left\|r - J_{CG}^k\right\|^2 - \frac{c}{2}\left\|r + J_{WG}^k\right\|^2,$$

where $J_i(x)$, $i \in \{CG, WG\}$, are fixed. It can be compted exactly by

$$r^{k+1}(x) \;=\; \max\left(0, (J_1 - J_2)/2\right).$$

5. Update the multiplier functions $u_i^{k+1}(x)$, $i \in \{CG, WG\}$, by

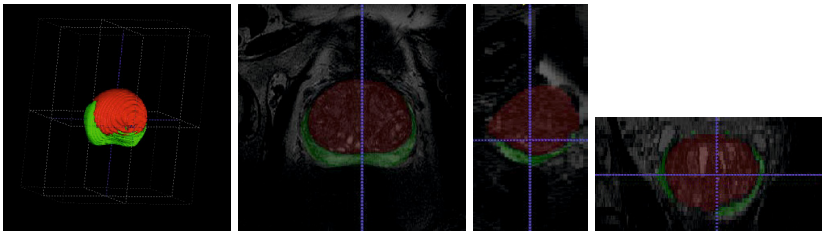
$$u_i^{k+1}(x) \;=\; u_i^k(x) - c\,R_i^{k+1}(x)\,. \tag{27}$$

The *coupled continuous max-flow* algorithm successfully avoids directly handling the non-smooth functions and linear constraint in the corresponding convex relaxation model (16). The experiments show that the proposed algorithm also obtains a much faster convergence rate in practice. In addition, the *coupled continuous max-flow algorithm* can be readily implemented on GPUs to significantly speed-up computation.

3 Experiments and Results

Experiment Implementation: We applied the proposed continuous max-flow algorithm on 10 T2w MR images acquired using a body coil. Subjects were scanned at 3 Tesla with a GE Excite HD MRI system (Milwaukee, WI, USA). All images were acquired at $512\times512\times36$ voxels with spacing of $0.27\times0.27\times2.2mm^3$. Two closed surfaces were constructed via a thin-plate spline fitting with ten to twelve user selected initial points on the WG and CG surface, respectively, which were used as the initial CG and WG surfaces for surface evolution. The original input 3D image was also cropped by enlarging the bounding box of the initial WG surface by 30 voxels in order to speed up computations. The initial PDFs for the regions of R_{CZ} and R_{PZ} were calculated based on the intensities in the user-initialized CG and PZ regions, respectively.

Evaluation Metrics: The proposed segmentation method was evaluated by comparing the results to manual segmentations in terms of DSC, the mean absolute surface distance (MAD), and the maximum absolute distance (MAXD) [26, 27]. All validation metrics were calculated for the entire prostate gland, central gland and peripheral zone. In addition, the coefficient-of-variation (CV) of DSC [27] was used to evaluate the intra-observer variability of our method introduced by manual initialization.



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

Fig. 2. Segmentation result of one prostate. (a) rendered resulting surface, (b) axial view, (c) sagittal view, and (d) coronal view, Green: the segmented PZ, red: the segmented CG.

Accuracy: Visual inspections in Fig. 2 show that the PZ and CG regions segmented by the proposed approach agree well with the objects. Quantitative experiment result for 10 patient images using the proposed method is shown in Table 1. The mean DSC was $89.2\pm4.5\%$ for the whole prostate gland, $84.7\pm5.2\%$ for the central gland, and $68.5\pm6.9\%$ for the peripheral zone. In addition, the evaluation results of MAD and MAXD are provided in Table 1, which give similar information to DSC.

Reliablility: Ten images were also segmented three times by the same observer to assess the intra-observer variability introduced by the user initialization. The proposed method initialized by three repetitions yielded a CV of 7.5%, 5.6%, and 5.0% for PZ, CG, and WG, respectively. It can be seen that the proposed

Table 1. Mean segmentation results in terms of DSC, MAD and MAXD for 10 patient images

	DSC (%)	MAD (mm)	MAXD (mm)
PZ	68.5 ± 6.9	4.8 ± 2.1	20.1 ± 11.5
CG	84.7 ± 5.2	3.2 ± 1.2	12.3 ± 3.8
WG	89.2 ± 4.5	2.9 ± 0.9	12.2 ± 4.8

Table 2. Intra-observer variability results in terms of DSC (%) using three repetitions of the same observer for ten patient images

	PZ	CG	WG
experiment 1	68.5 ± 6.9	84.7 ± 5.2	89.2 ± 4.5
experiment 2	67.9 ± 5.8	85.3 ± 4.5	88.7 ± 4.0
experiment 3	69.2 ± 6.5	84.5 ± 4.5	89.0 ± 3.8
CV (%)	7.5	5.6	5.0

method demonstrated low intra-observer segmentation variability for the CG and WG, suggesting a good reproducibility.

Computational Time: The proposed approach was implemented in Matlab (Natick, MA) using CUDA (NVIDIA Corp., Santa Clara, CA). The experiments were performed on a Windows desktop with an Intel i7-2600 CPU (3.4 GHz) and a GPU of NVIDIA Geforce 580X. The mean run time of three repeated segmentations for each 3D MR image was used to estimate the segmentation time in this study. The mean segmentation time was 8 ± 0.5s (converged with 3 - 5 surface evolutions) in addition to 40 ± 5s for initialization, resulting in a total segmentation time of less than $50s$ for each 3D image ($512 \times 512 \times 36$ voxels).

4 Discussions and Conclusions

In this work, we propose and evaluate a new global optimization-based coupled contour evolution approach to simultaneously extracting the boundaries of prostate and its component zones from the input 3D prostate T2w MRI, which address the challenge of segmenting multiple prostate regions in a numerically stable and efficient way. In contrary to the classical level-set methods, the proposed approach demonstrates great advantages in terms of numerical efficiency and moving the coupled contours to their 'best' positions simultaneously while preserving the inter-contour relationship. The introduced algorithm shows reliable performance results with minimal user interactions using ten patient images, suggesting itself for potential clinical use in 3D TRUS/MR image guided prostate interventions and computer aided diagnosis of prostate cancer.

The experimental results using ten 3D MR patient prostate images showed that the proposed *continuous max-flow algorithm* is capable of providing a robust and efficient segmentation for different prostate zones at the same time, such as PZ, CG and WG, with promising accuracy and reliability. In terms of accuracy,

DSC of $89.2 \pm 4.5\%$ for the whole prostate region(WG), based on the introduced composite intensity appearance model, is better than the result of $86.2 \pm 3.0\%$ obtained by the state-of-art mixture intensity model; DSCs of $68.5 \pm 6.9\%$ and $84.7 \pm 5.2\%$ for PZ and CG yielded by our methods are lower than $75.0 \pm 7.0\%$ and $89.0 \pm 3.0\%$ reported in [14] or $76.0 \pm 6.0\%$ and $87.0 \pm 4.0\%$ reported in [13]. However, these two methods made use of multi-spectral MR information and required manual WG segmentations as initialization. In addition, comparing to these methods, the proposed method also needs less user interactions and computation time. In order to improve the segmentation accuracy, our future studies might put emphasis on incorporating additional prior information, such as texture and shape, or relying on information from multi-spectral MR imaging.

Acknowledgments. The authors are grateful for funding from the Canadian Institutes of Health Research (CIHR) and the Ontario Institute of Cancer Research (OICR). E. Ukwatta acknowledges the support from Natural Sciences and Engineering Research Council (NSERC) Canada Research Scholarship (CGS). A. Fenster holds a Canada Research Chair in Biomedical Engineering, and acknowledges the support of the Canada Research Chair Program.

References

1. Siegel, R., Naishadham, D., Jemal, A.: Cancer statistics, 2012. CA: A Cancer Journal for Clinicians 62(1), 10–29 (2012)
2. Leslie, S., Goh, A., Lewandowski, P.M., Huang, E.Y.H., de Castro Abreu, A.L., Berger, A.K., Ahmadi, H., Jayaratna, I., Shoji, S., Gill, I.S., Ukimura, O.: 2050 contemporary image-guided targeted prostate biopsy better characterizes cancer volume, gleason grade and its 3d location compared to systematic biopsy. The Journal of Urology 187(4, suppl.), e827 (2012)
3. Doyle, S., Feldman, M.D., Tomaszewski, J., Madabhushi, A.: A boosted bayesian multiresolution classifier for prostate cancer detection from digitized needle biopsies. IEEE Trans. Biomed. Engineering 59(5), 1205–1218 (2012)
4. Beyersdorff, D., Winkel, A., Hamm, B., Lenk, S., Loening, S.A., Taupitz, M.: MR imaging-guided prostate biopsy with a closed MR unit at 1.5 T: initial results. Radiology 234(2), 576–581 (2005)
5. Villeirs, G., De Meerleer, G.: Magnetic resonance imaging (mri) anatomy of the prostate and application of mri in radiotherapy planning. European Journal of Radiology 63(3), 361–368 (2007)
6. Haffner, J., Potiron, E., Bouyé, S., Puech, P., Leroy, X., Lemaitre, L., Villers, A.: Peripheral zone prostate cancers: location and intraprostatic patterns of spread at histopathology. The Prostate 69(3), 276–282 (2009)
7. Reinsberg, S., Payne, G., Riches, S., Ashley, S., Brewster, J., Morgan, V., et al.: Combined use of diffusion-weighted mri and 1h mr spectroscopy to increase accuracy in prostate cancer detection. American Journal of Roentgenology 188(1), 91–98 (2007)
8. Kitajima, K., Kaji, Y., Fukabori, Y., Yoshida, K., Suganuma, N., Sugimura, K.: Prostate cancer detection with 3 t mri: Comparison of diffusion-weighted imaging and dynamic contrast-enhanced mri in combination with t2-weighted imaging. Journal of Magnetic Resonance Imaging 31(3), 625–631 (2010)
9. Kirby, R., Gilling, P.: Fast facts: benign prostatic hyperplasia. Health Press Limited (2011)

10. Ghose, S., Oliver, A., Martí, R., Lladó, X., Vilanova, J., Freixenet, J., Mitra, J., Sidibé, D., Meriaudeau, F.: A survey of prostate segmentation methodologies in ultrasound, magnetic resonance and computed tomography images. Computer Methods and Programs in Biomedicine 108(1), 262–287 (2012)
11. Allen, P., Graham, J., Williamson, D., Hutchinson, C.: Differential segmentation of the prostate in mr images using combined 3d shape modelling and voxel classification. In: 3rd IEEE International Symposium on Biomedical Imaging: Nano to Macro, pp. 410–413. IEEE (2006)
12. Yin, Y., Fotin, S., Periaswamy, S., Kunz, J., Haldankar, H., Muradyan, N., Turkbey, B., Choyke, P.: Fully automated 3d prostate central gland segmentation in mr images: a logismos based approach. In: SPIE Medical Imaging, International Society for Optics and Photonics, pp. 83143B–83143B (2012)
13. Makni, N., Iancu, A., Colot, O., Puech, P., Mordon, S., Betrouni, N., et al.: Zonal segmentation of prostate using multispectral magnetic resonance images. Medical Physics 38(11), 6093 (2011)
14. Litjens, G., Debats, O., van de Ven, W., Karssemeijer, N., Huisman, H.: A pattern recognition approach to zonal segmentation of the prostate on MRI. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) MICCAI 2012, Part II. LNCS, vol. 7511, pp. 413–420. Springer, Heidelberg (2012)
15. Luckhaus, S., Sturzenhecker, T.: Implicit time discretization for the mean curvature flow equation. Calc. Var. Partial Differential Equations 3(2), 253–271 (1995)
16. Boykov, Y., Kolmogorov, V., Cremers, D., Delong, A.: An integral solution to surface evolution PDEs via geo-cuts. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3953, pp. 409–422. Springer, Heidelberg (2006)
17. Yuan, J., Ukwatta, E., Tai, X.C., Fenster, A., Schnoerr, C.: A fast global optimization-based approach to evolving contours with generic shape prior. Technical report CAM-12-38, UCLA (2012)
18. Osher, S., Fedkiw, R.: Level set methods and dynamic implicit surfaces. Applied Mathematical Sciences, vol. 153. Springer, New York (2003)
19. Delong, A., Gorelick, L., Schmidt, F.R., Veksler, O., Boykov, Y.: Interactive segmentation with super-labels. In: Boykov, Y., Kahl, F., Lempitsky, V., Schmidt, F.R. (eds.) EMMCVPR 2011. LNCS, vol. 6819, pp. 147–162. Springer, Heidelberg (2011)
20. Michailovich, O., Rathi, Y., Tannenbaum, A.: Image segmentation using active contours driven by the bhattacharyya gradient flow. IEEE Transactions on Image Processing 16(11), 2787–2801 (2007)
21. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, 2001 (2001)
22. Yuan, J., Bae, E., Tai, X.: A study on continuous max-flow and min-cut approaches. In: CVPR 2010 (2010)
23. Yuan, J., Bae, E., Tai, X.-C., Boykov, Y.: A continuous max-flow approach to potts model. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part VI. LNCS, vol. 6316, pp. 379–392. Springer, Heidelberg (2010)
24. Bertsekas, D.P.: Nonlinear Programming. Athena Scientific (September 1999)
25. Chambolle, A.: An algorithm for total variation minimization and applications. Journal of Mathematical Imaging and Vision 20(1), 89–97 (2004)
26. Garnier, C., Bellanger, J.J., Wu, K., Shu, H., Costet, N., Mathieu, R., de Crevoisier, R., Coatrieux, J.L.: Prostate segmentation in HIFU therapy. IEEE Trans. Med. Imag. 30(3), 792–803 (2011)
27. Qiu, W., Yuan, J., Ukwatta, E., Tessier, D., Fenster, A.: Rotational-slice-based prostate segmentation using level set with shape constraint for 3D end-firing TRUS guided biopsy. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) MICCAI 2012, Part I. LNCS, vol. 7510, pp. 537–544. Springer, Heidelberg (2012)

Linear Osmosis Models for Visual Computing

Joachim Weickert[1], Kai Hagenburg[1], Michael Breuß[2], and Oliver Vogel[1]

[1] Mathematical Image Analysis Group
Dept. of Mathematics and Computer Science, Campus E1.1
Saarland University, 66041 Saarbrücken, Germany
{weickert,hagenburg,vogel}@mia.uni-saarland.de
[2] Institute for Applied Mathematics and Scientific Computing
Platz der Deutschen Einheit 1, Brandenburg University of Technology
03046 Cottbus, Germany
breuss@tu-cottbus.de

Abstract. Osmosis is a transport phenomenon that is omnipresent in nature. It differs from diffusion by the fact that it allows nonconstant steady states. In our paper we lay the foundations of osmosis filtering for visual computing applications. We model filters with osmotic properties by means of linear drift-diffusion processes. They preserve the average grey value and the nonnegativity of the initial image. Most interestingly, however, we show how the nonconstant steady state of an osmosis evolution can be steered by its drift vector field. We interpret this behaviour as a data integration mechanism. In the integrable case, we characterise the steady state as a minimiser of a suitable energy functional. In the nonintegrable case, we can exploit osmosis as a framework to fuse incompatible data in a visually convincing way. Osmotic data fusion differs from gradient domain methods by its intrinsic invariance under multiplicative grey scale changes. The osmosis framework constitutes a novel class of methods that can be taylored to solve various problems in image processing, computer vision, and computer graphics. We demonstrate its versatility by proposing osmosis models for compact image respresentation, shadow removal, and seamless image cloning.

Keywords: osmosis, drift–diffusion, Fokker–Planck equation, diffusion filters, gradient domain methods, shadow removal, image editing.

1 Introduction

While diffusion processes are frequently used in image processing, computer vision and computer graphics, there is a closely related transport phenomenon in nature that is basically unexplored in visual computing applications: It is called osmosis [1]. Osmosis describes transport through a semipermeable membrane in such a way that in its steady state, the liquid concentrations on both sides of the membrane can differ. Osmosis is the primary mechanism for transporting water in and out of cells, and it has many applications in medicine and engineering. It can be seen as the nonsymmetric counterpart of diffusion. Since diffusion can only model symmetric transport processes, it leads to flat steady states [2].

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 26–39, 2013.
© Springer-Verlag Berlin Heidelberg 2013

Our Contributions. The goal of our paper is to lay the foundations of osmosis filtering for visual computing applications. In contrast to osmosis in natural systems we do not need two different phases (water and salt) and a membrane that is only permeable for one of them: We can obtain nonconstant steady states within a single phase that represents the grey value. All we have to do is to supplement diffusion with a drift term. The resulting drift-diffusion process is also in divergence form and thus preserves the average grey value of the initial image, but it allows to have full control over its nonflat steady state: We show that we can design osmosis filters that converge to any specified image. Most importantly, we shall see that osmosis has the ability to integrate conflicting gradient data in a visually convincing way. This enables many applications beyond classical data regularisation tasks. In particular, we show the potential of osmosis for three prototypical problems: compact data representation, shadow removal, and image editing. Interestingly, these applications do not require any nonlinearities: The richness of the drift term permits to reach these goals already within a linear setting.

Paper Structure. In Section 2 we describe our drift-diffusion framework for continuous osmosis filters, analyse its essential properties, and interprete osmosis processes as models for data integration. Afterwards we sketch a simple numerical scheme in Section 3. Applications of osmosis models to visual computing problems are described in Section 4, and our paper is concluded with a summary in Section 5.

Related Work. While diffusion filters are often combined with data fidelity terms, there are not many combinations with a drift term in divergence form so far. Hagenburg et al. [3] have proposed a lattice Boltzmann model for dithering that approximates a nonlinear drift-diffusion equation in the continuous limit. However, they did not investigate this continuous model any further. In [4] it was shown that a combination of a discrete osmosis model with a stabilised backward diffusion filter is useful for designing numerical schemes for hyperbolic conservation laws that benefit from low numerical diffusion. Moreover, this discrete osmosis model has been interpreted as a nonsymmetric Markov chain, while discrete diffusion filters lead to symmetric Markov chains. Illner and Neunzert [5] have investigated so-called directed diffusion processes that converge to a specified background image, but did not apply them to image processing problems.

With respect to their ability to integrate incompatible gradient data, osmosis methods can be related to gradient domain methods. In computer vision, gradient domain methods are used for shape from shading [6], for shadow removal [7], and as models for retinex [8]. In computer graphics they are useful for a number of image editing and fusion problems; see e.g. [9, 10]. Relations between osmosis and gradient domain methods are discussed in Section 2.3, and Section 4.3 gives an experimental comparison. With respect to their invariance under multiplicative brightness changes, osmosis methods also resemble Georgiev's covariant derivative framework [11], but appear to be easier to understand.

Since our drift-diffusion formulation of osmosis filtering can be interpreted in a stochastic way as Fokker-Planck equation [12], it has some structural similarities to work by Sochen [13] that deals with a stochastic justification of the Beltrami flow. He mentions the potential benefit of drift terms but did not carry out any experiments. The Fokker-Planck equation has also been used by Wang and Hancock [14] for performing probabilistic relaxation labelling on graphs.

While the present paper focuses on the continuous theory and introduces specific models for different visual computing applications, we have also authored a companion paper that establishes a fully discrete theory for osmosis and studies efficient numerical methods [15].

2 Continuous Linear Osmosis Filtering

2.1 Drift–Diffusion Model

We consider a rectangular image domain $\Omega \subset \mathbb{R}^2$ with boundary $\partial\Omega$. A reasonable osmosis theory for greyscale images requires a positive initial image $f : \Omega \to \mathbb{R}_+$. Moreover, we assume that we can choose some *drift vector field* $\boldsymbol{d} : \Omega \to \mathbb{R}^2$. As we will see below, it allows us to steer the osmosis process to a desired nonflat steady state. Then a *(linear) osmosis filter* computes a family $\{u(\boldsymbol{x}, t) \mid t \geq 0\}$ of processed versions of $f(\boldsymbol{x})$ by solving the drift-diffusion equation

$$\partial_t u \;=\; \Delta u \;-\; \operatorname{div}(\boldsymbol{d}u) \qquad \text{on } \Omega \times (0, T], \tag{1}$$

with f as initial condition,

$$u(\boldsymbol{x}, 0) \;=\; f(\boldsymbol{x}) \qquad \text{on } \Omega, \tag{2}$$

and homogeneous Neumann boundary conditions:

$$\langle \boldsymbol{\nabla} u - \boldsymbol{d}u,\, \boldsymbol{n} \rangle \;=\; 0 \qquad \text{on } \partial\Omega \times (0, T]. \tag{3}$$

Here $\langle ., . \rangle$ denotes the Euclidean inner product, and $\boldsymbol{n}$ is the outer normal vector to the image boundary $\partial\Omega$. Thus, the boundary conditions specify a vanishing flux across the image boundaries.

Extending linear osmosis to colour images does not create specific problems: One proceeds separately in each RGB channel using individual drift vector fields in each channel.

2.2 Theoretical Properties

While the main focus of our paper is on modelling aspects, successful modelling is impossible without some insights into essential theoretical properties. They are summarised in the following proposition.

Proposition 1. [Theory for Continuous Linear Osmosis]
A classical solution of the linear osmosis process (1)–(3) with positive initial image $f : \Omega \to \mathbb{R}_+$ and drift vector field $\boldsymbol{d} : \Omega \to \mathbb{R}^2$ satisfies the following properties:

(a) The average grey value is preserved:

$$\frac{1}{|\Omega|}\int_\Omega u(\boldsymbol{x},t)\,d\boldsymbol{x} \;=\; \frac{1}{|\Omega|}\int_\Omega f(\boldsymbol{x})\,d\boldsymbol{x} \qquad \forall\, t>0\,. \tag{4}$$

(b) The evolution preserves nonnegativity:

$$u(\boldsymbol{x},t)\geq 0 \qquad \forall \boldsymbol{x}\in\Omega, \quad \forall t>0. \tag{5}$$

(c) If $\boldsymbol{d}$ satisfies

$$\boldsymbol{d} \;=\; \boldsymbol{\nabla}(\ln v) \;=\; \frac{\boldsymbol{\nabla} v}{v} \tag{6}$$

with some positive image v, then the follwoing holds:
The steady state equation

$$\Delta u \;-\; \operatorname{div}(\boldsymbol{d}u) \;=\; 0 \tag{7}$$

is equivalent to the Euler-Lagrange equation of the energy functional

$$E(u) \;=\; \int_\Omega v\left|\boldsymbol{\nabla}\left(\frac{u}{v}\right)\right|^2 d\boldsymbol{x}. \tag{8}$$

Moreover, the steady state solution of the osmosis process is given by $w(\boldsymbol{x}) = \frac{\mu_f}{\mu_v} v(\boldsymbol{x})$, where μ_f and μ_v denote the average grey values of f and v.

Proof

(a) Let $\mu(t) := \frac{1}{|\Omega|}\int_\Omega u(\boldsymbol{x},t)\,d\boldsymbol{x}$ denote the average grey value at time $t \geq 0$. Using the divergence theorem and the homogeneous Neumann boundary conditions we obtain

$$\begin{aligned}\frac{d\mu}{dt} \;&=\; \frac{1}{|\Omega|}\int_\Omega \partial_t u\,d\boldsymbol{x} \;=\; \frac{1}{|\Omega|}\int_\Omega \operatorname{div}(\boldsymbol{\nabla}u - \boldsymbol{d}u)\,d\boldsymbol{x}\\ &=\; \int_{\partial\Omega}\langle \boldsymbol{\nabla}u - \boldsymbol{d}u, \boldsymbol{n}\rangle\,dS \;=\; 0.\end{aligned} \tag{9}$$

Thus, the average grey value remains constant over time.

(b) Assume that $T>0$ is the smallest time where $\min_{\boldsymbol{x},t} u(\boldsymbol{x},t) = 0$, and that this minimum is attained in some inner point $\boldsymbol{\xi}$. Thus, $\boldsymbol{\nabla}u(\boldsymbol{\xi},T) = \boldsymbol{0}$, and we have

$$\partial_t u(\boldsymbol{\xi},T) \;=\; \Delta u(\boldsymbol{\xi},T) \;-\; \underbrace{u(\boldsymbol{\xi},T)}_{=0}\operatorname{div}\boldsymbol{d} \;-\; \boldsymbol{d}^\top \underbrace{\boldsymbol{\nabla}u(\boldsymbol{\xi},T)}_{=\boldsymbol{0}}. \tag{10}$$

This shows that in $(\boldsymbol{\xi},T)^\top$ the osmosis evolution behaves like the diffusion equation $\partial_t u = \Delta u$. It is well known that for diffusion with homogeneous Neumann boundary conditions the minimum cannot decrease in time; see e.g. [2]. Thus, the solution of the osmosis process remains nonnegative.

(c) The energy functional (8) can be rewritten as

$$E(u) = \int_\Omega F(u, \nabla u)\, d\boldsymbol{x} \tag{11}$$

with

$$F(u, \nabla u) \;=\; \frac{|v\nabla u - u\nabla v|^2}{v^3}\,. \tag{12}$$

From the calculus of variations we know that any minimiser of $E(u)$ satisfies the Euler-Lagrange equation

$$0 \;=\; F_u - \partial_x F_{u_x} - \partial_y F_{u_y} \tag{13}$$

with homogeneous Neumann boundary conditions, where $\boldsymbol{x} = (x, y)^\top$ and subscripts denote partial derivatives. With F from (12) this becomes after some simplifications

$$0 \;=\; -2v\,\mathrm{div}\left(\frac{v\nabla u - u\nabla v}{v^3}\right) \;-\; \frac{4\,\nabla^\top v\,(v\nabla u - u\nabla v)}{v^3}\,. \tag{14}$$

Using

$$\mathrm{div}\left(v\,\nabla\left(\frac{u}{v}\right)\right) \;=\; v^2\,\mathrm{div}\left(\frac{v\nabla u - u\nabla v}{v^3}\right) \;+\; \frac{2\,\nabla^\top v\,(v\nabla u - u\nabla v)}{v^2} \tag{15}$$

the Euler-Lagrange equation can be written as

$$0 \;=\; -\frac{2}{v}\,\mathrm{div}\left(v\,\nabla\left(\frac{u}{v}\right)\right). \tag{16}$$

It is easy to check that this is equivalent to (7) if $\boldsymbol{d} = \frac{\nabla v}{v}$ with $v > 0$. Straightforward computations also show that one obtains (3) as boundary condition on $\partial\Omega$.

It is clear that an image v with $\boldsymbol{d} = \frac{\nabla v}{v}$ also fulfils the steady state equation (7) of the osmosis evolution with homogeneous Neumann boundary conditions. However, note that with v also cv with any constant c is a solution of this problem. Since the osmosis evolution preserves the average grey value and the nonnegativity of the initial image, it can only converge to a rescaled version w of v that is nonnegative and has the same average grey value as the original image f. Thus, $w(\boldsymbol{x}) = \frac{\mu_f}{\mu_v} v(\boldsymbol{x})$. □

Preservation of the average grey value does not distinguish osmosis from diffusion filtering [2]. However, while diffusion filtering satisfies a maximum–minimum principle [2], osmosis only fulfils a weaker form of stability, namely preservation of nonnegativity. We conjecture that it is possible to establish preservation of strict positivity, since this also holds in the fully discrete case [15].

Proposition 1 implies that osmosis permits nontrivial steady states. This is a fundamental difference to diffusion that allows only flat steady states [2].

Fig. 1. Convergence of osmosis to a specified image. **From left to right: (a)** Original image, 512×512 pixels. Each channel has the same mean value as the *mandrill* test image. **(b)** Osmosis result at evolution time $t = 50$. **(c)** $t = 1000$. **(d)** $t = 250000$ gives a steady state that is identical to the mandrill image.

Of course, these steady state results should be accompanied by a formal convergence analysis. This is mathematically more involved and will be presented in a journal version of our paper.

Figure 1 illustrates such a convergence behaviour. Starting from a flat initial image, we can choose the drift vector field such that osmosis converges to the *mandrill* test image: If $\boldsymbol{v} = (v_1, v_2, v_3)^\top$ is the RGB image of the mandrill, the drift vector in channel i is chosen as $\boldsymbol{\nabla}(\ln v_i)$.

2.3 Osmosis as a Process for Data Integration

We have seen that for $\boldsymbol{d} := \boldsymbol{\nabla}(\ln v)$, osmosis converges to a multiplicatively rescaled version of v. This motivates us to call $\boldsymbol{d}[v] := \boldsymbol{\nabla}(\ln v)$ the *canonical drift vector field* of the image v. Since $\boldsymbol{d}[v]$ contains derivative information of the steady state, we may regard osmosis as a process for data integration.

Obviously it is not very exciting to design an osmosis process that converges to an image which we know already. However, much more interesting situations arise when we modify the drift vector field, e.g. by setting certain components to zero, or by fusing the canonical drift vector fields of different images. Such applications will be considered in Section 4. Although in general the new drift vector field will be nonintegrable, osmosis will still create a steady state that aims at finding a good compromise between all conflicting constraints. In that sense osmosis resembles gradient domain methods that are popular both in computer vision [6–8] and in computer graphics [10, 9]. Let us analyse these connections in more detail.

Gradient domain methods integrate a (possibly nonintegrable) gradient field approximation $\boldsymbol{p} = (p_1, p_2)^\top$ by minimising the energy

$$E(u) = \int_\Omega |\boldsymbol{\nabla} u - \boldsymbol{p}|^2 \, d\boldsymbol{x}. \tag{17}$$

The corresponding Euler–Lagrange equation that a minimiser w has to fulfil is given by the Poisson equation

$$\Delta w \;=\; \operatorname{div} \boldsymbol{p}. \tag{18}$$

In the integrable case with $\boldsymbol{p} = \boldsymbol{\nabla} v$, it is clear that an additive shift of v gives the same $\boldsymbol{p}$. Thus, gradient domain methods can recover v up to an additive constant.

On the other hand, an osmotic steady state w satisfies

$$\Delta w \;=\; \operatorname{div}(\boldsymbol{d} w). \tag{19}$$

An integrable osmosis setting with $\boldsymbol{d} = \boldsymbol{\nabla}(\ln v)$ is invariant under multiplicative rescalings of v. In this sense it resembles Georgiev's covariant derivative framework [11], but appear to be easier to comprehend. For computer vision applications where illumination changes are often modelled as multiplicative changes of the grey values, this multiplicative invariance of osmosis is preferable over the additive invariance of gradient domain methods.

In the nonintegrable case, gradient domain methods and osmosis can give different results that cannot be transformed into each other by simple additive or multiplicative grey value changes.

3 A Simple Numerical Scheme

To keep our paper self-contained, let us now sketch a simple explicit finite difference scheme for our osmosis model. For more numerical details and more efficient schemes we refer to [15], where a general fully discrete theory for osmosis filtering is established.

We consider a grid size h in x- and y-direction, and a time step size $\tau > 0$. Moreover, we denote by $u_{i,j}^k$ an approximation to u in the grid point $((i-\frac{1}{2})h, (j-\frac{1}{2})h))^\top$ at time $k\tau$. Setting $\boldsymbol{d} = (d_1, d_2)^\top$, a straightforward finite difference discretisation of (1) is given by

$$\begin{aligned}
\frac{u_{i,j}^{k+1} - u_{i,j}^k}{\tau} \;&=\; \frac{u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - 4u_{i,j}^k}{h^2} \\
&-\; \frac{1}{h}\left(d_{1,i+\frac{1}{2},j} \frac{u_{i+1,j}^k + u_{i,j}^k}{2} - d_{1,i-\frac{1}{2},j} \frac{u_{i,j}^k + u_{i-1,j}^k}{2} \right) \\
&-\; \frac{1}{h}\left(d_{2,i,j+\frac{1}{2}} \frac{u_{i,j+1}^k + u_{i,j}^k}{2} - d_{2,i,j-\frac{1}{2}} \frac{u_{i,j}^k + u_{i,j-1}^k}{2} \right).
\end{aligned} \tag{20}$$

It allows to compute the results at time level $k+1$ from the data at level k. This scheme also holds for boundary points, if we mirror the image at its boundaries and assume a zero drift vector across boundaries.

For some positive image v, we obtain a discrete approximation of its canonical drift vector field $(d_1[v], d_2[v])^\top = \frac{\boldsymbol{\nabla} v}{v}$ at intermediate grid points via

$$d_{1,i+\frac{1}{2},j} = \frac{2\,(v_{i+1,j} - v_{i,j})}{h\,(v_{i+1,j} + v_{i,j})}, \qquad d_{2,i,j+\frac{1}{2}} = \frac{2\,(v_{i,j+1} - v_{i,j})}{h\,(v_{i,j+1} + v_{i,j})}. \tag{21}$$

In [15] we show that the scheme (20)–(21) preserves positivity and converges to its unique steady state if the time step size satisfies

$$\tau < \frac{h^2}{8}. \tag{22}$$

4 Application to Visual Computing Problems

In order to illustrate the potential of osmosis models to solve visual computing problems, we study three fairly different applications: compact image representation, shadow removal, and seamless image cloning. All results below display steady states that have been computed with the numerical scheme from Section 3 with $h := 1$ and $\tau := 0.1$. One can achieve positivity of a bytewise coded initial image by adding an offset value of $\epsilon > 0$ such that each channel lies in the range $[\epsilon, 255+\epsilon]$. Offset values should not be too large to avoid that they have a visible impact on the result. We choose $\epsilon := 1$.

4.1 Compact Data Representation

Let us now investigate if osmosis processes can be useful for compact image representations. There has been a long tradition of reconstructing images from their information near edges; see e.g. [16, 17]. One may for instance store the grey values on both sides of the edges as Dirichlet data, and interpolate the remaining data by solving the Laplace equation $\Delta u = 0$ in between. While this requires to store two grey values per edge point, it appears tempting to use osmosis and keep only the magnitude of the induced drift vector at each edge pixel, since we know that its direction is orthogonal to the edge contour. All drift vectors that are not adjacent to edges are set to zero, such that homogeneous diffusion interpolation is performed. This is illustrated in Figure 2. We observe that this compact image representation works well at step edges, while the contrasts that are reproduced at smooth edges appear somewhat too low. This proof-of-concept application indicates that osmosis can become a valuable tool for encoding step edges within a more comprehensive image compression approach based in partial differential equations (PDEs). More details will be reported in forthcoming publications.

4.2 Shadow Removal

Many computer vision tasks such as segmentation, tracking, and object recognition benefit from the removal of shadows. To this end, one wants to identify the shadow region and adapt its brightness to the brightness of the rest of the image. Several methods have been proposed to find suitable shadow edges; see [18] and the references therein. Here we assume that the shadow edges are given, and we concentrate on the brightness adaptation problem. So far, this brightness adaptation has been achieved for example with gradient domain methods [7] or with pyramid-based approaches [18].

Fig. 2. Osmosis for compact image representation. **(a) Left:** Original image. **(b) Middle:** Canny edges, amounting to 9.6% of all pixels. **(c) Right:** Reconstruction using the average grey value of the orginal image and the canonical drift vectors in the edges.

Interestingly the invariance of osmosis under global multiplicative greyscale changes offers a particularly elegant solution for this task: If one models shadows as a local multiplicative illumination change within the image, then this only affects the canonical drift vectors at the transition between the shadow and the rest of the image. Thus, shadow removal can be accomplished by simply modifying the canonical drift vectors at the shadow boundaries. Setting them to zero at these locations turns osmosis locally into homogoneneous diffusion and guarantees a continuous transition. An osmosis evolution that starts with the original image and uses these modified drift vectors converges to a steady state where the shadow has been removed. Osmosis shadow removal has no problems recovering texture details in the shadow part which in an important criterion in state-of-the-art methods [19].

Figure 3 illustrates these ideas. If one uses the original image as initialisation, the results will be somewhat too dark due to the shadow region and the fact that osmosis preserves the average grey value (or colour value in each channel). As a remedy, one may want to rescale the results such that the values in the non-shadow region approximate the ones in the initial image. This is done in Figure 3(d).

4.3 Seamless Image Cloning

The property of osmosis to fuse incompatible information can also be used for seamless image cloning. Figure 4 illustrates the problem: Two images f_1 and f_2 are to be merged such that f_2 replaces image information of f_1. The rectangular image domain of the original image f_1 is denoted by Ω, and the image domain that is to be inserted is Γ. Its boundary is given by $\partial\Gamma$.

The classical gradient domain method for seamless image cloning is *Poisson image editing* [10]. It creates a fused image by solving the Poisson equation (18) with gradient data $\boldsymbol{p} = \boldsymbol{\nabla} f_2$ in Γ and Dirichlet boundary conditions $u = f_1$ on $\partial\Gamma$. By construction, this localises the influence of the patch to the domain Γ.

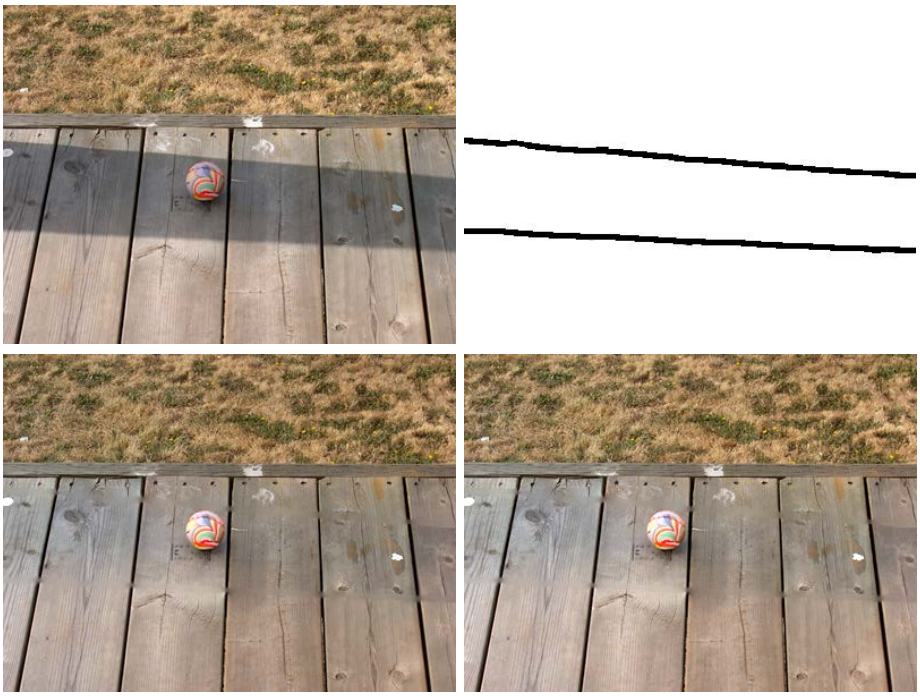


Fig. 3. Shadow removal by osmosis. **(a) Top left:** Original image (400×299 pixels). **(b) Top right:** User-selected shadow boundaries. In these boundaries all drift vectors are set to zero (homogeneous diffusion). In the other areas, the canonical drift vectors are used. **(c) Bottom left:** Osmosis reconstruction with (a) as initialisation. **(d) Bottom right:** Multiplicative rescaling of (c) such that the colours in the non-shadow regions approximate the ones in (a).

To provide an osmosis-based alternative to Poisson image editing, we proceed as follows: We use the canonical drift vectors of f_1 in $\Omega \setminus \Gamma$, and the ones of f_2 in Γ. At the interface $\partial\Gamma$, we use the arithmetic mean of both drift vectors. The process is initialised with f_1 on the entire rectangular image domain Ω, and its steady state gives the cloned image.

Figure 5 juxtaposes the results of Poisson image editing and osmosis image editing for an application where we want to clone the face of Lagrange on the body of Euler. While both methods give seamless results, Poisson image editing is unable to adapt the higher contrast of the face of Lagrange to the lower contrast of the Euler image. Section 2.3 gives an explanation for this phenomenon: Gradient domain methods allow only additive grey value shifts, and additive shifts cannot influence the contrast. Osmosis image editing, on the other hand, enables multiplicative changes that can also adapt the contrast. This is one reason why the osmosis result in Fig. 5(f) comes visually closer to the original Euler image in Fig. 5(a). A second reason results from the averaging of the canonical

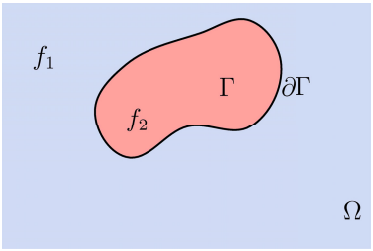


Fig. 4. Illustration of the image editing problem

Fig. 5. Seamless image cloning (cf. [4]). **(a) Top left:** Painting of Euler. **(b) Top middle:** Drawing of Lagrange (with selected face region). **(c) Top right:** Direct cloning of Lagrange on top of Euler's face. **(d) Bottom left:** Poisson image editing with Dirichlet data of Euler at the interface. **(e) Bottom right:** Osmosis image editing with averaged drift vector fields at the interface. Source of original images: Wikimedia Commons.

Fig. 6. Comparison between Poisson image editing and osmosis image editing when the contrast of the inserted patch is not optimal. **(a) Top left:** Original image (400×300 pixels). **(b) Top right:** Direct cloning of a patch where the RGB values have been multiplied by a factor 0.25. **(c) Bottom left:** Poisson image editing suffers from the poor contrast whithin the cloned patch. **(d) Bottom right:** Osmosis image editing gives a much better contrast reconstruction.

drift vectors at the interface $\partial\Gamma$. This makes the influence of the osmosis editing global, which also contributes to a more harmonic impression.

Fig. 6 presents a synthetic experiment that is taylored to visualise the differences between gradient domain editing and osmosis editing. We have reduced the contrast in the mouth region of the teddy bear by multiplying the RGB values by 0.25. While gradient domain editing cannot increase the contrast in the cloned patch, this is no problem for osmosis image editing.

These discussions show that there are two reasons for the superiority of osmosis image editing over Poisson image editing: boundary conditions that avoid a strict localisation, and the ability of osmosis to perform multiplicative brightness adaptations instead of additive ones.

5 Summary and Conclusions

We have advocated osmosis as a novel concept for visual computing. It is surprising that after decades of intensive research on PDE methods in image analysis,

this important process in nature has been widely ignored by our research community so far. While osmosis filters differ from homogeneous diffusion filtering "only" by their drift term, this term has a fundamental consequence: It creates nonconstant steady states that can be controlled in a transparent way by the drift vector field. This offers interesting application areas that go far beyond the classical image regularisation and enhancement applications of diffusion methods. We have illustrated this potential by using osmosis for compact image representation, shadow removal, and seamless image cloning. While many diffusion filters rely on nonlinear concepts and may even require singular diffusivities or anisotropic diffusion tensors, our osmosis models show their high versatility already within a purely linear setting. Moreover, unlike gradient domain methods, osmosis is intrinsically invariant under global multiplicative changes of the grey values. In view of their promising potential, it is our hope that osmosis modelling will become a widely applied framework for visual computing.

Obviously our paper can only serve as a starting point, and there are many ways to continue research on osmosis processes for image analysis and synthesis. While our applications have exploited the nontrivial steady states of osmosis processes, it would also be interesting to see if the evolution itself has useful applications. On a theoretical side, we are working on a complete well-posedness and convergence theory for continuous linear osmosis processes, and we are also establishing a linear osmosis theory for semidiscrete and fully discrete processes; see [15] for first results. Last but not least, we will also consider nonlinear generalisations of our linear osmosis framework.

Acknowledgments. This work has been performed while all authors were members of the Mathematical Image Analysis Group. Financial support by the *Deutsche Forschungsgemeinschaft (DFG)* is gratefully acknowledged.

References

1. Borg, F.: What is osmosis? Explanation and understanding of a physical phenomenon. Technical report, Chydenius Institute, Jyväskylä University, Karleby, Finland (2003) arXiv:physics/0305011v1
2. Weickert, J.: Anisotropic Diffusion in Image Processing. Teubner, Stuttgart (1998)
3. Hagenburg, K., Breuß, M., Vogel, O., Weickert, J., Welk, M.: A lattice Boltzmann model for rotationally invariant dithering. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Kuno, Y., Wang, J., Pajarola, R., Lindstrom, P., Hinkenjann, A., Encarnação, M.L., Silva, C.T., Coming, D. (eds.) ISVC 2009, Part II. LNCS, vol. 5876, pp. 949–959. Springer, Heidelberg (2009)
4. Hagenburg, K., Breuß, M., Weickert, J., Vogel, O.: Novel schemes for hyperbolic PDEs using osmosis filters from visual computing. In: Bruckstein, A.M., ter Haar Romeny, B.M., Bronstein, A.M., Bronstein, M.M. (eds.) SSVM 2011. LNCS, vol. 6667, pp. 532–543. Springer, Heidelberg (2012)
5. Illner, R., Neunzert, H.: Relative entropy maximization and directed diffusion equations. Mathematical Methods in the Applied Sciences 16, 545–554 (1993)
6. Frankot, R., Chellappa, R.: A method for enforcing integrability in shape from shading algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence 10(4), 439–451 (1988)

7. Finlayson, G.D., Hordley, S.D., Drew, M.S.: Removing shadows from images. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002, Part IV. LNCS, vol. 2353, pp. 823–836. Springer, Heidelberg (2002)
8. Morel, J.M., Petro, A.B., Sbert, C.: A PDE formalisation of retinex theory. IEEE Transactions on Image Processing 19(11), 2825–2837 (2010)
9. Fattal, R., Lischinski, D., Werman, M.: Gradient domain high dynamic range compression. In: Proc. SIGGRAPH 2002, San Antonio, TX, pp. 249–256 (July 2002)
10. Pérez, P., Gagnet, M., Blake, A.: Poisson image editing. ACM Transactions on Graphics 22(3), 313–318 (2003)
11. Georgiev, T.: Covariant derivatives and vision. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 56–69. Springer, Heidelberg (2006)
12. Risken, H.: The Fokker–Planck Equation. Springer, New York (1984)
13. Sochen, N.A.: Stochastic processes in vision: From Langevin to Beltrami. In: Proc. Eighth International Conference on Computer Vision, Vancouver, Canada, vol. 1, pp. 288–293. IEEE Computer Society Press (July 2001)
14. Wang, H., Hancock, E.R.: Probabilistic relaxation labelling using the Fokker–Planck equation. Pattern Recognition 41(11), 3393–3411 (2008)
15. Vogel, O., Hagenburg, K., Weickert, J., Setzer, S.: A fully discrete theory for linear osmosis filtering. In: Kuijper, A., Bredies, K., Pock, T., Bischof, H. (eds.) SSVM 2013. LNCS, vol. 7893, pp. 368–379. Springer, Heidelberg (2013)
16. Carlsson, S.: Sketch based coding of grey level images. Signal Processing 15, 57–83 (1988)
17. Mainberger, M., Bruhn, A., Weickert, J., Forchhammer, S.: Edge-based compression of cartoon-like images with homogeneous diffusion. Pattern Recognition 44(9), 1859–1873 (2011)
18. Shor, Y., Lischinski, D.: The shadow meets the mask: Pyramid-based shadow removal. Computer Graphics Forum 22(2), 577–586 (2008)
19. Salamati, N., Germain, A., Süsstrunk, S.: Removing shadows from images using color and near-infrared. In: Proc. 2011 IEEE International Conference on Image Processing, Brussels, Belgium, pp. 1713–1716 (September 2011)

Analysis of Bayesian Blind Deconvolution

David Wipf[1] and Haichao Zhang[2]

[1] Visual Computing Group, Microsoft Research, Beijing, P.R. China
[2] School of Computer Science, Northwestern Polytechnical University, Xi'an, P.R. China
{davidwipf,hczhang1}@gmail.com

Abstract. Blind deconvolution involves the estimation of a sharp signal or image given only a blurry observation. Because this problem is fundamentally ill-posed, strong priors on both the sharp image and blur kernel are required to regularize the solution space. While this naturally leads to a standard MAP estimation framework, performance is compromised by unknown trade-off parameter settings, optimization heuristics, and convergence issues stemming from non-convexity and/or poor prior selections. To mitigate these problems, several authors have recently proposed substituting a variational Bayesian (VB) strategy that marginalizes over the high-dimensional image space leading to better estimates of the blur kernel. However, the underlying cost function now involves both integrals with no closed-form solution and complex, function-valued arguments, thus losing the transparency of MAP. To elucidate these issues, we demonstrate that the VB methodology can be recast as an unconventional MAP problem with a very particular penalty/prior that couples the image, blur kernel, and noise level in a principled way. This unique penalty has a number of useful characteristics pertaining to relative concavity, local minima avoidance, and scale-invariance that allow us to rigorously explain the success of VB including its existing implementational heuristics and approximations. It also provides strict criteria for choosing the optimal image prior that, perhaps counter-intuitively, need not reflect the statistics of natural scenes. In so doing we challenge the prevailing notion of why VB is successful for blind deconvolution while providing a transparent platform for introducing enhancements and extensions.

1 Introduction

Image blur is an undesirable phenomenon that often accompanies the image formation process and may arise, for example, because of camera-shake during acquisition. Blind image deconvolution or deblurring strategies aim to recover a sharp image from only a blurry, compromised observation, a long-standing problem [8] that remains an active research topic [5,16,11,4,6]. Assuming a convolutional blur model with additive noise [5,16], the low quality image observation process is commonly modeled as

$$\mathbf{y} = \mathbf{k} * \mathbf{x} + \mathbf{n}, \tag{1}$$

where $\mathbf{k}$ is the point spread function (PSF) or blur kernel, $*$ denotes the 2D convolution operator, and $\mathbf{n}$ is a noise term commonly assumed to be zero-mean Gaussian with covariance $\lambda\mathbf{I}$ (although as we shall see, these assumptions about the noise distribution can easily be relaxed via the framework described herein). The task of blind deconvolution

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 40–53, 2013.
© Springer-Verlag Berlin Heidelberg 2013

is to estimate both the sharp image $\mathbf{x}$ and blur kernel $\mathbf{k}$ given only the blurry observation $\mathbf{y}$, where we will mostly be assuming that $\mathbf{x}$ and $\mathbf{y}$ represent filtered (e.g., gradient domain) versions of the original pixel-domain images. Because $\mathbf{k}$ is non-invertible, some (typically) high frequency information is lost during the observation process, and thus even if $\mathbf{k}$ were known, the non-blind estimation of $\mathbf{x}$ is ill-posed. However, in the blind case where $\mathbf{k}$ is also unknown, the difficulty is exacerbated considerably, with many possible image/kernel pairs explaining the observed data equally well.

To alleviate this problem, prior assumptions must be adopted to constrain the space of candidate solutions, which naturally suggests a Bayesian framework. In Section 2, we briefly review the two most common classes of Bayesian algorithms for blind deconvolution used in the literature, (i) Maximum a Posteriori (MAP) estimation and (ii) Variational Bayes (VB), and then later detail their fundamental limitations, which include heuristic implementational requirements and complex cost functions that are difficult to disentangle. Section 3 uses ideas from convex analysis to reformulate these Bayesian methods promoting greater understanding and suggesting useful enhancements, such as rigorous criteria for choosing appropriate image priors. Experiments are carried out in Section 4 to provide corroborating empirical evidence for some of our theoretical claims. Finally, concluding remarks are contained in Section 5.

2 MAP versus VB

To compensate for the ill-posedness of the blind deconvolution problem, a strong prior is required for both the sharp image and kernel to regularize the solution space. Recently, natural image statistics have been invoked to design prior (regularization) models, e.g., [9,7], and MAP estimation using these priors has been proposed for blind deconvolution, e.g., [5,16,12]. While some specifications may differ, the basic idea is to find the mode of the posterior distribution $p(\mathbf{x}, \mathbf{k}|\mathbf{y})$, which is equivalent to solving

$$\min_{\mathbf{x},\mathbf{k}} -2\log p(\mathbf{y}|\mathbf{x},\mathbf{k})p(\mathbf{x})p(\mathbf{k}) \equiv \min_{\mathbf{x},\mathbf{k}} \frac{1}{\lambda}\|\mathbf{k} * \mathbf{x} - \mathbf{y}\|_2^2 + g_{\mathbf{x}}(\mathbf{x}) + g_{\mathbf{k}}(\mathbf{k}), \qquad (2)$$

where $g_{\mathbf{x}}(\mathbf{x})$ is a penalty term over the desired image while $g_{\mathbf{k}}(\mathbf{k})$ regularizes the blur kernel, both of which generally have embedded parameters that must be balanced along with λ. It is also typical to assume that $\sum_i k_i = 1$, with $k_i \geq 0$ and we will adopt this assumption throughout (however, Section 3.5 will discuss a type of scale invariance such that this assumption becomes irrelevant anyway).

Although straightforward, there are many problems with existing MAP approaches including ineffective global minima, e.g., poor priors may lead to degenerate solutions like the delta kernel (frequently called the no-blur solution), or too many local minima and subsequent convergence issues. Therefore, the generation of useful solutions (or to guide the algorithm carefully to a proper local minima) requires a delicate balancing of various factors such as dynamic noise levels, trade-off parameter values, and other heuristic regularizers such as salient structure selection [16,4,6] (we will discuss such issues more in Sections 3.3 and 3.4). To mitigate some of these shortcomings of MAP, the influential work by Levin *et al.* and others proposes to instead solve [11]

$$\max_{\mathbf{k}} p(\mathbf{k}|\mathbf{y}) \equiv \min_{\mathbf{k}} -2\log p(\mathbf{y}|\mathbf{k})p(\mathbf{k}), \qquad (3)$$

where $p(\mathbf{y}|\mathbf{k}) = \int p(\mathbf{x}, \mathbf{y}|\mathbf{k})d\mathbf{x}$. This technique is sometimes referred to as Type II estimation in the statistics literature.[1] Once $\mathbf{k}$ is estimated in this way, $\mathbf{x}$ can then be obtained via conventional non-blind deconvolution techniques. One motivation for the Type II strategy is based on the inherent asymmetry in the dimensionality of the image relative to the kernel [11]. By integrating out (or averaging over) the high-dimensional image, the estimation process can then more accurately estimate the few remaining low-dimensional parameters in $\mathbf{k}$.

The challenge of course with (3) is that the evaluation of $p(\mathbf{y}|\mathbf{k})$ requires a marginalization over $\mathbf{x}$, which is a computationally intractable integral given realistic image priors. Consequently a variational Bayesian (VB) strategy is used to approximate the troublesome marginalization [12]. A similar idea has also been proposed by a number of other authors [13,5,1]. In brief, VB provides a convenient way of computing a rigorous upper bound on $-\log p(\mathbf{y}|\mathbf{k})$, which can then be substituted into (3) for optimization purposes leading to an approximate Type II estimator.

The VB methodology can be easily applied whenever the image prior $p(\mathbf{x})$ is expressible as a Gaussian scale mixture (GSM) [15], meaning

$$p(\mathbf{x}) = \exp\left[-\frac{1}{2}g_{\mathbf{x}}(\mathbf{x})\right] = \prod_i \exp\left[-\frac{1}{2}g_x(x_i)\right] = \prod_i \int \mathcal{N}(x_i; 0, \gamma_i)p(\gamma_i)d\gamma_i, \quad (4)$$

where each $\mathcal{N}(x_i; 0, \gamma_i)$ represents a zero mean Gaussian with variance γ_i and prior distribution $p(\gamma_i)$. The role of this decomposition will become apparent below. Also, with some abuse of notation, $p(\gamma_i)$ may characterize a discrete distribution, in which case the integral in (4) can be reduced to a summation. Note that all prior distributions expressible via (4) will be supergaussian [15], and therefore will to varying degrees favor a sparse $\mathbf{x}$. Given this $p(\mathbf{x})$, the negative log of $p(\mathbf{y}|\mathbf{k})$ can be upper bounded via

$$-\log p(\mathbf{y}|\mathbf{k}) \leq \underbrace{-\iint q(\mathbf{x}, \boldsymbol{\gamma}) \log \frac{p(\mathbf{x}, \boldsymbol{\gamma}, \mathbf{y}|\mathbf{k})}{q(\mathbf{x}, \boldsymbol{\gamma})} d\mathbf{x}d\boldsymbol{\gamma}}_{F[q(\mathbf{x},\boldsymbol{\gamma}),\mathbf{k}]},$$

where $F[q(\mathbf{x}, \boldsymbol{\gamma}), \mathbf{k}]$ is called the *free energy*, $q(\mathbf{x}, \boldsymbol{\gamma})$ is an arbitrary distribution over $\mathbf{x}$, and $\boldsymbol{\gamma} = [\gamma_1, \gamma_2, \ldots]^T$, the vector of all the variances from (4). Equality is obtained when $q(\mathbf{x}, \boldsymbol{\gamma}) = p(\mathbf{x}, \boldsymbol{\gamma}|\mathbf{y}, \mathbf{k})$. In fact, if we were able to iteratively minimize this F over $q(\mathbf{x}, \boldsymbol{\gamma})$ and $\mathbf{k}$ (i.e., a form of coordinate descent), this would be exactly equivalent to the standard expectation-maximization (EM) algorithm for minimizing $-\log p(\mathbf{y}|\mathbf{k})$ with respect to $\mathbf{k}$, treating $\boldsymbol{\gamma}$ and $\mathbf{x}$ as hidden data and assuming $p(\mathbf{k})$ is flat within the specified constraint set mentioned previously (see [2, Ch.9.4] for a detailed examination of this fact). However, optimizing over $q(\mathbf{x}, \boldsymbol{\gamma})$ is intractable since $p(\mathbf{x}, \boldsymbol{\gamma}|\mathbf{y}, \mathbf{k})$ is generally not available in closed-form. Likewise, there is no closed-form update for $\mathbf{k}$, and hence no exact EM solution is possible.

The VB theory shows that if we restrict the form of $q(\mathbf{x}, \boldsymbol{\gamma})$ via structural assumptions, the updates can now actually be computed, albeit approximately. For this purpose

[1] To be more specific, Type II estimation refers to the case where we optimize over one set of unknown variables after marginalizing out another set, in our case the image $\mathbf{x}$. In this context, standard MAP over both $\mathbf{x}$ and $\mathbf{k}$ can be viewed as Type I.

Algorithm 1. VB Blind Deblurring [12,15,1]

1: **Input:** a blurry image $\mathbf{y}$, noise level reduction factor β ($\beta > 1$), minimum noise level λ_0, an image prior $p(\mathbf{x}) = \exp[-\frac{1}{2}g_{\mathbf{x}}(\mathbf{x})] = \prod_i \exp[-\frac{1}{2}g_x(x_i)]$
2: **Initialize:** blur kernel $\mathbf{k}$, noise level λ
3: **While** stopping criteria is not satisfied, do

- **Update sufficient statistics for** $q(\boldsymbol{\gamma}) = \prod_i q(\gamma_i)$
  $\omega_i \triangleq \mathrm{E}_{q(\gamma_i)}[\gamma_i^{-1}] \leftarrow \frac{{g_x}'(\sigma_i)}{2\sigma_i}$, with $\sigma_i^2 \triangleq \mathrm{E}_{q(x_i)}[x_i^2] = \mu_i^2 + C_{ii}$.
- **Update sufficient statistics for** $q(\mathbf{x}) = \prod_i q(x_i)$
  $\boldsymbol{\mu} \triangleq \mathrm{E}_{q(\mathbf{x})}[\mathbf{x}] \leftarrow \mathbf{A}^{-1}\mathbf{b}$, $C_{ii} \triangleq \mathrm{Var}_{q(x_i)}[x_i] \leftarrow A_{ii}^{-1}$, where $\mathbf{A} = \frac{\mathbf{H}^T\mathbf{H}}{\lambda} + \mathrm{diag}[\boldsymbol{\omega}]$, $\mathbf{b} = \frac{\mathbf{H}^T\mathbf{y}}{\lambda}$, $\mathbf{H}$ is the convolution matrix of $\mathbf{k}$.
- **Update** $\mathbf{k}$
  $\mathbf{k} \leftarrow \arg\min_{\mathbf{k}\geq 0} \|\mathbf{y} - \mathbf{W}\mathbf{k}\|_2^2 + \sum_j c_j k_j^2$, where $c_j = \sum_i C_{i+j,i+j}$ and $\mathbf{W}$ is the convolution matrix of $\boldsymbol{\mu}$.
- **Noise level reduction** If $\lambda > \lambda_0$, then $\lambda \leftarrow \lambda/\beta$.

4: **End**

the most common constraint is that $q(\mathbf{x}, \boldsymbol{\gamma})$ must be factorized, sometimes called a mean-field approximation [2, Ch.10.1]:

$$\min_{q(\mathbf{x},\boldsymbol{\gamma}),\mathbf{k}} F\left[q(\mathbf{x}, \boldsymbol{\gamma}), \mathbf{k}\right], \quad \text{s.t. } q(\mathbf{x}, \boldsymbol{\gamma}) = \prod_i q(x_i)q(\gamma_i). \tag{5}$$

The requisite update rules are shown in Algorithm 1. Numerous methods fall within this category with some implementational differences. Note also that the full distributions for each $q(x_i)$ and $q(\gamma_i)$ are generally not needed; only a few sufficient statistics are required (certain means and variances, see Algorithm 1), analogous to standard EM. These can be efficiently computed using techniques from [15] for any $p(\mathbf{x})$ produced by (4). In the VB algorithm from [12], the sufficient statistic for $\boldsymbol{\gamma}$ is computed using an alternative methodology which applies only to finite Gaussian scale mixtures. However, the resulting updates are nonetheless equivalent to Algorithm 1 as shown in [19].

While possibly well-motivated in principle, the Type II approach relies on rather severe factorial assumptions which may compromise the original high-level justifications. In fact, at any minimizing solution denoted $q^*(x_i), q^*(\gamma_i), \forall i, \mathbf{k}^*$, it is easily shown that the gap between F and $-\log p(\mathbf{y}|\mathbf{k}^*)$ is given explicitly by the KL divergence between the distributions $\prod_i q^*(x_i)q^*(\gamma_i)$ and $p(\mathbf{x}, \boldsymbol{\gamma}|\mathbf{y}, \mathbf{k}^*)$. Because the posterior $p(\mathbf{x}, \boldsymbol{\gamma}, |\mathbf{y}, \mathbf{k})$ is generally highly coupled (non-factorial), this divergence will typically be quite high, indicating that the associated approximation could be of low quality. We therefore have no reason to believe that this $\mathbf{k}^*$ is anywhere near the maximizer of $p(\mathbf{y}|\mathbf{k})$, which was the ultimate goal and motivation of Type II to begin with.

Other problems persist as well. For example, the free energy cost function, which involves both integration and function-valued arguments, is not nearly as transparent as the standard MAP estimation from (2). Moreover for practical use, like most deconvolution algorithms including MAP, VB requires an appropriate schedule for reducing the

noise variance λ during each iteration (see Algorithm 1), otherwise performance can be quite poor.

It therefore becomes difficult to rigorously explain exactly why VB has often been empirically more successful than MAP in practice (see [1,12] for empirical comparisons), nor how to decide which image priors operate best in the VB framework. While Levin *et al.* have suggested that at a high level, marginalization over the latent image using natural-image-statistic-based priors is a good idea to overcome some of the problems faced by MAP estimation [11], this argument only directly motivates substituting (3) for (2) rather than providing explicit rationalization for (5). Thus, we intend to more meticulously investigate the mechanism by which VB operates, explicitly accounting for all of the approximations and assumptions involved by drawing on sparse estimation concepts from [15,18]. This endeavor will also naturally motivate extensions to the VB framework and a simple prescription for choosing an appropriate image prior $p(\mathbf{x})$. Overall, we hope that we can further demystify VB providing an entry point for broader improvements such as robust non-uniform deblurring.

Several surprising, possibly counterintuitive conclusions emerge from our investigation in Section 3 which challenge some of the prevailing wisdom regarding why and how Bayesian algorithms can be advantageous for blind deconvolution. These include:

- The optimal image prior for blind deconvolution purposes using VB or MAP is likely not the one which most closely reflects natural images statistics. Rather, we argue that it is a highly-sparse distribution that most significantly discriminates between blurry and sharp images.
- The advantage of VB over MAP is not directly related to the dimensionality differences between $\mathbf{k}$ and $\mathbf{x}$ and the conventional benefits of marginalization over the latter. In fact, we prove in Section 3.1 that the underlying cost functions are formally equivalent in ideal noiseless environments given the factorial assumptions required by practical VB algorithms. Instead, there is an intrinsic mechanism built into VB that allows bad locally minimizing solutions to be largely avoided even when using the highly non-convex, discriminative priors needed to distinguish between blurry and sharp images. This represents a new perspective on the relative advantages of VB.
- The VB algorithm can be reformulated in such a way that non-Gaussian noise models, non-uniform blur operators, and other extensions are easily incorporated, circumventing one important perceived advantage of MAP.

3 Analysis of Variational Bayes

Following [5] and [12], we work in the derivative domain of images for ease of modeling and better performance, meaning that $\mathbf{x}$ and $\mathbf{y}$ denote the lexicographically ordered image derivatives of sharp and blurry images respectively obtained via a particular derivative filter. Given that convolution is commutative, the blur kernel is unaltered.

For latent sharp image derivatives of size $M \times N$ and blur kernel of size $P \times Q$, we denote the lexicographically ordered vector of sharp image derivatives, blurry image derivatives, and blur kernel as $\mathbf{x} \in \mathbb{R}^m$, $\mathbf{y} \in \mathbb{R}^n$ and $\mathbf{k} \in \mathbb{R}^l$ respectively, with $m \triangleq MN$, $n \triangleq (M-P+1)(N-Q+1)$, and $l \triangleq PQ$. This assumes a single

derivative filter. The extension to multiple filters, typically one for each image dimension, follows naturally. For simplicity of notation however, we omit explicit referencing of multiple filters throughout this paper, although all related analysis follow through in a straightforward manner.

The likelihood model (1) can be rewritten as $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} = \mathbf{W}\mathbf{k} + \mathbf{n}$, where $\mathbf{H} \in \mathbb{R}^{n\times m}$ and $\mathbf{W} \in \mathbb{R}^{n\times l}$ are the convolution matrices constructed from the blur kernel and sharp image respectively. We introduce a matrix $\bar{\mathbf{I}} \in \mathbb{R}^{l\times m}$, where the j-th row of $\bar{\mathbf{I}}$ is a binary vector with 1 indicating that the j-th element of $\mathbf{k}$ (i.e. k_j) appears in the corresponding column of $\mathbf{H}$ and 0 otherwise. We define $\|\bar{\mathbf{k}}\|_2 \triangleq \sqrt{\sum_j k_j^2 \bar{I}_{ji}}$, which is equivalent to the norm of the i-th column of $\mathbf{H}$. It can also be viewed as the effective norm of $\mathbf{k}$ accounting for the boundary effects. [2] The element-wise magnitude of $\mathbf{x}$ is given by $|\mathbf{x}| \triangleq [|x_1|, |x_2|, \cdots]^T$.

Finally we introduce the definition of *relative concavity* [14] which will serve subsequent analyses:

Definition 1. *Let u be a strictly increasing function on $[a, b]$. The function ν is **concave relative** to u on the interval $[a, b]$ if and only if $\nu(y) \leq \nu(x) + \frac{\nu'(x)}{u'(x)}[u(y) - u(x)]$ holds $\forall x, y \in (a, b)$.*

We will use $\nu \prec u$ to denote that ν is concave relative to u on $[0, \infty)$. This can be understood as a natural generalization of the traditional notion of a concavity, in that a concave function is equivalently *concave relative to a linear function* per Definition 1. The notion of relative concavity induces an ordering for many of the common sparsity promoting penalty functions. Intuitively, a non-decreasing function ν of $|x_i|$ is more aggressive in promoting sparsity than some u if it is concave relative to u.

3.1 Connecting VB with MAP

As mentioned previously, the VB algorithm of [12] can be efficiently implemented using any image prior expressible in the form of (4). However, for our purposes we require an alternative representation with roots in convex analysis. Based on [15], it can be shown that any prior given by (4) can also be represented as a maximization over scaled Gaussians with different variances leading to the alternative representation

$$p(x_i) = \exp\left[-\frac{1}{2}g_x(x_i)\right] = \max_{\gamma_i \geq 0} \mathcal{N}(x_i; 0, \gamma_i) \exp\left[-\frac{1}{2}f(\gamma_i)\right], \tag{6}$$

where f is some non-negative energy function; the associated exponentiated factor is sometimes treated as a hyperprior, although it will not generally integrate to one. This f, which determines the form of g_x, will ultimately play a central role in how VB penalizes $\mathbf{x}$ as will be explored via the results of this section. We now re-express the

[2] Technically $\|\bar{\mathbf{k}}\|_2$ depends on i, the index of image pixels, but it only makes a difference near the image boundaries. We prefer to avoid an explicit notational dependency on i to keep the presentation concise. The subsequent analysis will also omit this dependency although all of the results carry through in the general case. The same is true for other quantities that depend on $\|\bar{\mathbf{k}}\|_2$, e.g., the ρ parameter defined later in (7).

VB cost function in an alternative form amenable to direct comparisons with MAP from (2):

Theorem 1. *Consider the objective function*

$$\mathcal{L}(\mathbf{x},\mathbf{k}) \triangleq \frac{1}{\lambda}\|\mathbf{y}-\mathbf{k}*\mathbf{x}\|_2^2 + \sum_i \left[g_{\text{VB}}(x_i,\rho) + \log\|\bar{\mathbf{k}}\|_2^2\right], \tag{7}$$

where

$$g_{\text{VB}}(x_i,\rho) \triangleq \min_{\gamma_i\geq 0}\left[\frac{x_i^2}{\gamma_i} + \log(\rho+\gamma_i) + f(\gamma_i)\right], \quad \text{and } \rho \triangleq \frac{\lambda}{\|\bar{\mathbf{k}}\|_2^2}. \tag{8}$$

Algorithm 1 minimizing (5) is equivalent to coordinate descent minimization of (7) over $\mathbf{x}$, $\mathbf{k}$, *and the latent variables* $\boldsymbol{\gamma} = [\gamma_1,\gamma_2,\ldots]^T$.

Proofs will be deferred to [19]. The penalty on $\mathbf{k}$ in (7) is not unlike those incorporated into standard MAP schemes from (2). However, quite unlike MAP, the image penalty g_{VB} is dependent on both the noise level λ and the kernel $\mathbf{k}$ through the parameter ρ, the ratio of the noise level to the squared kernel norm. Moreover, with a general $\lambda \neq 0$, it is easily shown that g_{VB} is non-separable in $\mathbf{k}$ and $\mathbf{x}$, meaning $g_{\text{VB}}(x_i,\rho) \neq h_1(x_i) + h_2(\mathbf{k})$ for any possible functions h_1 and h_2. The remainder of Section 3 will explore the consequences of this crucial, yet previously unexamined distinction from typical MAP formulations.

In contrast, with $\lambda = 0$, both MAP and VB possess a formally equivalent penalty on each x_i via the following corollary:

Corollary 1. *If* $\lambda = 0$, *then* $g_{\text{VB}}(x_i,0) = g_x(x_i) \equiv -2\log p(x_i)$.

Therefore the underlying VB cost function *is effectively no different than regular MAP from (2) in the noiseless setting*, a conclusion that seems to counter some of the prevailing understanding of VB deconvolution algorithms.

3.2 Evaluating the VB Image Penalty g_{VB}

We will now attempt to explore in more depth exactly how the image penalty g_{VB} from (7) contributes to the success of VB. While in a few special cases g_{VB} can be computed in closed-form for general $\rho \neq 0$ leading to greater transparency, as we shall see below the VB algorithm and certain attendant analyses can nevertheless be carried through even when closed-form solutions for g_{VB} are not possible. Importantly, we can assess properties that may potentially affect the sparsity and quality of resulting solutions as λ and $\|\bar{\mathbf{k}}\|_2^2$ are varied.

A highly sparse prior, and therefore penalty function, is generally more effective in differentiating sharp images with fine structures from blurry ones (more on this later). Recall that concavity with respect to coefficient magnitudes is a signature property of such sparse penalties [18]. A potential advantage of MAP is that it is very straightforward to characterize the associated image penalty; namely, if g_x from (2) is a highly concave, nondecreasing function of each $|x_i|$, then we may expect that sparse image gradients will be heavily favored. And for two candidate image penalties ${g_x}^{(1)}$ and ${g_x}^{(2)}$, if ${g_x}^{(1)} \prec {g_x}^{(2)}$, then we may expect the former to promote an even sparser solution than the latter (provided we are not trapped at a bad local solution).

In contrast, with VB it is completely unclear to what degree g_{VB} favors sparse solutions. We now explicitly describe sufficient and necessary conditions for g_{VB} to be a concave, nondecreasing function of $|x_i|$, which turn out to be much stricter than the conditions required for MAP.

Theorem 2. *The VB penalty g_{VB} will be a concave, non-decreasing function of $|x_i|$ for any ρ if and only if f from (6) is a concave, non-decreasing function on $[0, \infty)$. Moreover, at least $m-n$ elements of $\mathbf{x}$ will equal zero at any locally minimizing solution to (7) (however typically many more will equal zero in practice).*

Theorem 2 explicitly quantifies what class of image priors leads to a strong, sparsity-promoting $\mathbf{x}$ penalty when fully propagated through the VB framework. Yet while this attribute may anchor VB as a legitimate sparse estimator in the image (filter) domain given an appropriate f, it does not explain precisely why VB often produces superior results to MAP. In fact, the associated MAP penalty g_x (when generated from the same f) will actually promote sparse solutions under much weaker conditions as follows:

Corollary 2. *The MAP penalty g_x will be a concave, non-decreasing function of $|x_i|$ if and only if $\vartheta(z) \triangleq \log(z) + f(z)$ is a concave, non-decreasing function on $[0, \infty)$.*

The extra $\log$ factor implies that f itself need not be concave to ensure that g_x is concave. For example, the selection $f(z) = z - \log(z)$ it not concave and yet the associated g_x still will be since now $\vartheta(z) = z$, which is concave and non-decreasing as required by Corollary 2. Moving forward then, to really understand VB we must look deeper and examine the role of ρ in modulating the effective penalty on $\mathbf{x}$.

For this purpose, consider the simplest choice for f satisfying the conditions of Theorem 2; namely, $f(\gamma) = b$ for some constant b. This in turn implies that $p(x_i)$ is the improper Jeffreys non-informative prior on the coefficient magnitudes $|x_i|$, i.e., $p(x) \propto 1/|x|$, after solving the maximization from (6), and is attractive in part because there are no embedded hyperparameters (the constant b is irrelevant).

Theorem 3. *In the special case where $f(\gamma_i) = b$, then*

$$g_{\text{VB}}(x_i, \rho) \equiv \frac{2|x_i|}{|x_i| + \sqrt{x_i^2 + 4\rho}} + \log\left(2\rho + x_i^2 + |x_i|\sqrt{x_i^2 + 4\rho}\right). \tag{9}$$

Figure 1 (a) displays a 2D plot of this penalty function. In the limit as $\rho \to 0$, the first term in (9) converges to the indicator function $I[x_i \neq 0]$, and thus when we sum over i we obtain the ℓ_0 norm of $\mathbf{x}$.[3] The second term in (9), when we again sum over i, converges to $\sum_i \log|x_i|$, ignoring a constant factor. Sometimes referred to as Gaussian entropy, this term can also be connected to the ℓ_0 norm via the relations $\|\mathbf{x}\|_0 \equiv \lim_{p\to 0} \sum_i |x_i|^p$ and $\lim_{p\to 0} \frac{1}{p}\sum_i (|x_i|^p - 1) = \sum_i \log|x_i|$ [18]. Thus the cumulative effect when ρ becomes small is an image prior that closely mimics the highly non-convex ℓ_0 norm. In contrast, when ρ becomes large, it can be shown that both terms in (9), when combined for all i, approach scaled versions of the convex ℓ_1 norm. Additionally, if we assume a fixed kernel and ignore boundary effects, this scaling turns

[3] Although with $\rho = 0$, this term reduces to a constant, and therefore has no impact.

out to be optimal in a particular Bayesian sense (this technical point will be addressed further in a future publication).

For intermediate values of ρ between these two extremes, we obtain a g_{VB} that becomes *less* concave with respect to each $|x_i|$ as ρ increases in the specific sense of relative concavity discussed previously. To help formalize this notion, we define the function $g_{\text{VB}}^{\rho_\alpha} : \mathbb{R}^+ \rightarrow \mathbb{R}$ as $g_{\text{VB}}^{\rho_\alpha}(z) = g_{\text{VB}}(z, \rho = \rho_\alpha)$, with $z \geq 0$. Note that because g_{VB} is a symmetric function with respect to the origin, we may conveniently examine its concavity properties considering only the positive half of the real line. We then have the following:

Corollary 3. *If* $f(\gamma_i) = b$, *then* $g_{\text{VB}}^{\rho_1} \prec g_{\text{VB}}^{\rho_2}$ *for* $\rho_1 < \rho_2$.

Thus, as the noise level λ is increased, ρ increases and we have a penalty that behaves more like a convex (less sparse) function, and so becomes less prone to local minima. In contrast, as $\|\bar{\mathbf{k}}\|_2^2$ is increased, meaning that ρ is now reduced, the penalty actually becomes *more* concave with respect to $|x_i|$. This phenomena is in some ways similar to certain homotopy sparse estimation schemes (e.g., [3]), where heuristic hyperparameters are introduced to gradually introduce greater non-convexity into canonical compressive sensing problems, but without any dependence on the noise or other factors. The key difference here with VB is that penalty shape modulation is explicitly dictated by both the noise level λ and the kernel $\mathbf{k}$ in an entirely integrated fashion.

3.3 Noise Dependency Analysis

The success of practical VB blind deconvolution algorithms is heavily dependent on some form of stagewise coarse-to-fine approach, whereby the kernel is repeatedly reestimated at successively higher resolutions. One way to implement this approach is to initially use large values of λ such that only dominant, primarily low-frequency image structures dictate the optimization [11]. During subsequent iterations as the blur kernel begins to reflect the correct coarse shape, λ can be gradually reduced to allow the recovery of more detailed, fine structures.

A highly sparse (concave) prior can ultimately be more effective in differentiating sharp images and fine structures from blurry ones (see [19] and references within). However, if such a prior is applied at the initial stages of estimation, the iterations are likely to become trapped at suboptimal local minima, of which there will always be a combinatorial number. Moreover, in the early stages, the effective noise level is actually high due to errors contained in the estimated blur kernel, and exceedingly sparse image penalties are likely to produce unstable solutions. Given the reformulation outlined above, we can now argue that VB implicitly avoids these problems by beginning with a large λ (and therefore a large ρ), such that the penalty function is initially nearly convex in $|x_i|$. As the iterations proceed and fine structures need to be resolved, the penalty function becomes less convex as λ is reduced, but the risk of local minima and instability is ameliorated by the fact that we are likely to be already in the neighborhood of a desirable basin of attraction. Additionally, the implicit noise level (or modeling error) is now substantially less. This kind of automatic 'resolution' adaptive penalty shaping is arguably superior to conventional MAP approaches based on (2), where the concavity/shape of the induced separable penalty function is kept fixed regardless of the variation in the noise level or scale.

3.4 Blur Dependency Analysis

The shape parameter ρ is also affected by the kernel norm, with larger values of $\|\bar{\mathbf{k}}\|_2^2$ leading to less convexity of the penalty function g_{VB} while small values increase the convexity. With the standard assumptions $\sum_j k_j = 1$ and $k_j \geq 0$, $\|\bar{\mathbf{k}}\|_2^2$ is bounded between $1/l$ and 1, where l is the number of pixels in the kernel.[4] An increase of $\|\bar{\mathbf{k}}\|_2^2$ indicates that the kernel is more sparse, with the extreme case of $\mathbf{k} = \delta$ leading to $\|\bar{\mathbf{k}}\|_2^2 = 1$. In this situation, g_{VB} is the most concave in $|x_i|$ (per the analysis of Section 3.2), which is reasonable, as this is the easiest kernel type to handle so the sparsest penalty function can be used without much concern over local minima. In contrast, $\|\bar{\mathbf{k}}\|_2^2$ is the smallest when all elements are equal, which is the more challenging case corresponding with a broad diffuse image blur, with many local minima. In this situation, the penalty function is more convex and conservative. In general, a highly concave prior is not needed to disambiguate a highly blurred image from a relatively sharp one.

Additionally, at the beginning of the learning process when λ is large and before any detailed structures have been resolved, the $\log \|\bar{\mathbf{k}}\|_2^2$ penalty on $\mathbf{k}$ from (7) will naturally favor a blurry, diffuse kernel in the absence of additional information. This will help ensure that g_{VB} is relatively convex and less aggressive during the initial VB iterations. However, as the algorithm proceeds, λ is reduced, and some elements of $\mathbf{x}$ are pushed towards zero, the penalty g_{VB}, with its embedded $\mathbf{k}$ dependency, will gradually become less convex and can increasingly dominate the overall cost function (since for small λ and large $\|\bar{\mathbf{k}}\|_2^2$ the lower bound on g_{VB} can drop arbitrarily per the above described concavity modulation). Because g_{VB} is minimized as $\mathbf{k}$ becomes relatively sparse, a more refined $\mathbf{k}$ can be explored at this stage to the extent that $\mathbf{x}$ can be pushed towards greater sparsity as well (if $\mathbf{x}$ is not sparse, then there is no real benefit to refining $\mathbf{k}$). Again, this desirable effect occurs with relatively limited risk of local minima because of the gradual, intrinsically-calibrated introduction of increased concavity. In contrast, successful MAP algorithms must add various structure selection heuristics or penalty terms and associated trade-off parameters to (2) in order to avoid suboptimal minima and improve performance [16,4,17].

3.5 Other Choices for f

Because essentially any sparse prior on $\mathbf{x}$ can be expressed using the alternative variational form from (6), choosing such a prior is tantamount to choosing f which then determines g_{VB}. Theorem 2 suggests that a concave, non-decreasing f is useful for favoring sparsity (assumed to be in the gradient domain). Moreover, Theorem 3 and subsequent analyses suggest that the simplifying choice where $f(\gamma) = b$ possesses several attractive properties regarding the relative concavity of the resulting g_{VB}. But what about other selections for f and therefore g_{VB}?

While directly working with g_{VB} can sometimes be limiting (except in certain special cases like $f(\gamma) = b$ from before), the variational form of (8) allows us to closely examine

[4] Actually, because of natural invariances embedded into the VB cost function, the assumption $\sum_j k_j = 1$ is not needed for the analysis that follows. See Section 3.5 for more details.

the relative concavity of a useful proxy. Let $\psi(\gamma_i, \rho) \triangleq \log(\rho + \gamma_i) + f(\gamma_i)$. Then for fixed λ and $\mathbf{k}$ the VB estimation problem can equivalently be viewed as solving

$$\min_{\mathbf{x},\boldsymbol{\gamma}\geq 0} \frac{1}{\lambda}\|\mathbf{y} - \mathbf{k} * \mathbf{x}\|_2^2 + \sum_i \left[\frac{x_i^2}{\gamma_i} + \psi(\gamma_i, \rho)\right]. \tag{10}$$

It now becomes clear that the sparsity of $\mathbf{x}$ and $\boldsymbol{\gamma}$ are intimated related. More concretely, assuming f is concave and non-decreasing, then there is actually a one-to-one correspondence in that whenever $x_i = 0$, the optimal $\gamma_i = 0$, and vice versa.[5] Therefore we may instead examine the relative concavity of ψ for different ρ values, which will directly determine the sparsity of $\boldsymbol{\gamma}$ and in turn, the sparsity of $\mathbf{x}$. This then motivates the following result:

Theorem 4. *Let $\rho_1 < \rho_2$ and assume that f is a concave, non-decreasing function. Then $\psi^{\rho_1} \prec \psi^{\rho_2}$ if and only if $f(\gamma) = a\gamma + b$, with $a \geq 0$.*

Thus, although we have not been able to formally establish a relative concavity result for all general g_{VB} directly, Theorem 4 provides a nearly identical analog allowing us to draw similar conclusions to those detailed in Sections 3.3 and 3.4 whenever a general affine f is adopted. Perhaps more importantly, it also suggests that as f deviates from an affine function, we may begin to lose some of the desirable effects regarding the described penalty shape modulation.

While previously we closely scrutinized the special affine case where $f(\gamma) = b$, it still remains to examine the more general affine form $f(\gamma) = a\gamma + b$, $a > 0$. In fact, it is not difficult to show that as a is increased, the resulting penalty on $\mathbf{x}$ increasingly resembles an ℓ_1 norm with lesser dependency on ρ, thus severely muting the effect of the shape modulation that appears to be so effective (see arguments above and empirical results section below). So there currently does not seem to be any advantage to choosing some $a > 0$ and we are left, out of the multitude of potential image priors, with the conveniently simple choice of $f(\gamma) = b$, where the value of b is inconsequential. Experimental results support this conclusion: namely, as a is increased from zero performance gradually degrades (results not shown for space considerations).

As a final justification for simply choosing $f(\gamma) = b$, there is a desirable form of invariance that uniquely accompanies this selection.

Theorem 5. *If $\mathbf{x}^*$ and $\mathbf{k}^*$ represent the optimal solution to (7) under the constraint $\sum_i k_i = 1$, then $\alpha^{-1}\mathbf{x}^*$ and $\alpha\mathbf{k}^*$ will always represent the optimal solution under the modified constraint $\sum_i k_i = \alpha$ if and only if $f(\gamma) = b$.*

This is unlike the myriad of MAP algorithms or VB with other choices of f, where the exact calibration of the constraint can fundamentally alter the form of the optimal solution beyond a mere rescaling. Moreover, if such a constraint on $\mathbf{k}$ is omitted altogether, these other methods must then carefully tune associated trade-off parameters, so in one way or another this lack of invariance may require additional tuning.

[5] To see this first consider $x_i = 0$. The x_i^2/γ_i term can be ignored and so the optimal γ_i need only minimize $\log(\rho + \gamma_i) + f(\gamma_i)$, which is concave and non-decreasing whenever f is. Therefore the optimal γ_i is trivially zero. Conversely if $\gamma_i = 0$, then there is effectively an infinite penalty on x_i, and so the optimal x_i must also be zero.

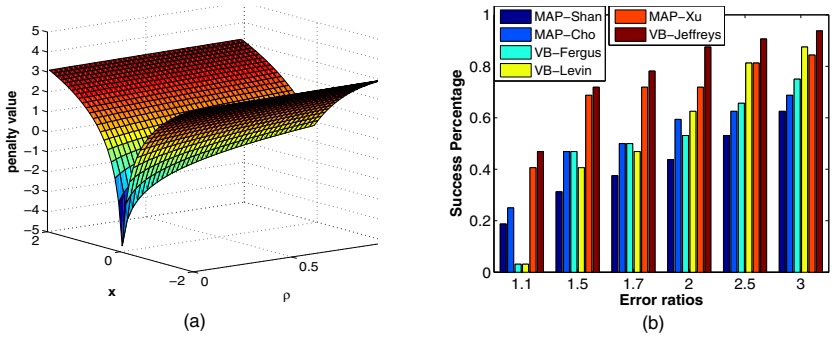


Fig. 1. (a) A 2D example surface plot of the coupled penalty function $g_{\text{VB}}(x, \rho)$; f is a constant. (b) Evaluation of the restoration results: Cumulative histogram of the deconvolution error ratio across 32 test examples. The height of the bar indicates the percentage of images having error ratio below that level. High bars indicate better performance.

4 Experimental Results

We emphasize that the primary purpose of this paper is the formal analysis of state-of-the-art VB methodology. Empirical support for recent VB algorithms, complementary to our theoretical presentation, can be found in [1,12]. Nonetheless, motivated by our results herein, we now briefly evaluate two simple enhancements of Algorithm 1. In particular, we (i) learn the λ parameter from Algorithm 1 based on updates from [19] derived using the VB formulation given in Section 3.1, and (ii) incorporate an image prior obtained using a flat f in (6) instead of a prior based on natural image statistics. We will refer to this algorithm as VB-Jeffreys since the underlying image prior is based on the improper Jeffreys distribution as described previously. Interestingly, Babacan *et al.* [1] experiment with a variety of VB algorithms using different underlying image priors, and empirically find that f as a constant works best; however, no explanation is given for why this should be the case. Thus, our results in Section 3 provide a powerful theoretical confirmation of this selection, along with a number of useful attendant intuitions.[6]

We reproduce the experiments from [12] using the benchmark test data from [11].[7] This consists of 4 base images of size 255×255 and 8 different blurring effects, leading to a total of 32 blurry images. We compare the VB-Jeffreys method with the algorithms of Shan *et al.* [16], Xu *et al.* [17], Cho *et al.* [4], Fergus *et al.* [5] and Levin *et al.* [12]. The methods of Fergus *et al.* and Levin *et al.* are two related VB approaches, labeled VB-Fergus and VB-Levin respectively. The other three methods [16,17,4] follow the MAP framework, and are labeled as MAP-Shan, MAP-Xu and MAP-Cho respectively.

[6] Based on a strong simplifying assumption that the covariance $\mathbf{C}$ from Algorithm 1 is a constant, Babacan *et al.* [1] provide some preliminary discussion regarding possibly why VB may be advantageous over MAP. However, this material mostly exists in the sparse estimation literature (e.g., see [15,18] and related references) and therefore the behavior of VB blind deconvolution remains an open question.

[7] This data is available online at `http://www.wisdom.weizmann.ac.il/~levina/papers/LevinEtalCVPR09Data.rar`

The SSD (Sum of Squared Difference) metric defined in [11] is used for measuring the error between estimated and the ground-truth images. To normalize for the fact that harder kernels give a larger image reconstruction error even when the true kernel is known, the SSD ratio between the image deconvolved with the estimated kernel and the image deconvolved with the ground-truth kernel is used as the final evaluation measure.

The results of MAP-Cho, VB-Fergus, and VB-Levin are from the dataset accompanying [12] directly. The results of MAP-Shan and MAP-Xu are produced using the software provided by the authors, for which we adjust the parameters carefully. For all algorithms we run every test image with the same parameters and non-blind deconvolution step, similar to [11]. The cumulative histogram of the error ratios is shown in Figure 1(b). The height of the bar indicates the percentage of images having error ratio below that level. High bars indicate better performance. As mentioned by Levin *et al.*, the results with error ratios above 2 may already have some visually implausible regions [11]. The VB-Jeffreys algorithm can achieve around 90% success with error ratio below 2, well above the other algorithms.

In general, all of the VB algorithms exhibit good performance, especially given that they do not benefit from any additional prior information or regularization heuristics that facilitate blur-adaptive structure selection, meaning additional refinements based on domain knowledge added to (2) that boost all of the MAP algorithms. Notably, even without this additional regularization factors, VB-Jeffreys significantly outperforms the MAP algorithms. However, one curious phenomenon is that both VB-Fergus and VB-Levin experience a relatively large drop-off in performance when the error ratio reduces from 1.5 to 1.1. While it is difficult to be absolutely certain, one plausible explanation for this decline relates to the prior selection employed by these algorithms. In both cases, the prior is based on a finite mixture of zero mean Gaussians with different variances tuned to natural image statistics. While such a prior does heavily favor approximately sparse signals, it will never produce any exactly sparse estimates at any resolution of the course-to-fine hierarchy, and hence, especially at high resolutions the penalty shape modulation effect of VB will be highly muted. Thus these algorithms may not be optimal for resolving extremely fine details, which is required for reliably producing image estimates with the lowest error ratios. In contrast, to achieve high error ratios only lower resolution features need be resolved, and in this regime VB-Levin, which is the closest algorithmically to VB-Jeffreys, performs nearly the best. Again, this reinforces the notion that natural image statistics may not be the optimal basis for image priors within the VB framework.

5 Conclusion

This paper carefully examines the underlying VB objective function in a transparent form, leading to better understanding and a principled criteria for choosing the optimal image prior. In this context, we have motivated a unique selection, out of the infinite set of possible sparse image priors, that simultaneously allows for maximal discrimination between blurry and sharp images, displays a desirable form of scale invariance, and leads to an intrinsic coupling between the blur kernel, noise level, and image penalty such that bad local minima can largely be avoided. From this perspective, it is no longer

difficult to enhance performance and generality by inheriting additional penalty functions or noise models (e.g., Laplacian, Poisson, etc.) commonly reserved for MAP. To the best of our knowledge, this represents a new viewpoint for understanding VB algorithms. While technically these conclusions only apply to the uniform blur model described by (1), we envision that many of the underlying principles can nonetheless be applied to more general models in the future. Initial results using a non-uniform blur model are promising [20].

References

1. Babacan, S.D., Molina, R., Do, M.N., Katsaggelos, A.K.: Bayesian blind deconvolution with general sparse image priors. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VI. LNCS, vol. 7577, pp. 341–355. Springer, Heidelberg (2012)
2. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Heidelberg (2006)
3. Chartrand, R., Yin, W.: Iteratively reweighted algorithms for compressive sensing. In: ICASSP (2008)
4. Cho, S., Lee, S.: Fast motion deblurring. In: ACM SIGGRAPH ASIA (2009)
5. Fergus, R., Singh, B., Hertzmann, A., Roweis, S.T., Freeman, W.T.: Removing camera shake from a single photograph. In: ACM SIGGRAPH (2006)
6. Krishnan, D., Tay, T., Fergus, R.: Blind deconvolution using a normalized sparsity measure. In: CVPR (2011)
7. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-Laplacian priors. In: NIPS (2009)
8. Kundur, D., Hatzinakos, D.: Blind image deconvolution. IEEE Signal Process. Mag. 13(3), 43–64 (1996)
9. Levin, A., Fergus, R., Durand, F., Freeman, W.T.: Deconvolution using natural image priors. Technical report, MIT (2007)
10. Levin, A., Weiss, Y., Durand, F., Freeman, W.: Understanding and evaluating blind deconvolution algorithms. In: CVPR (2009)
11. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Understanding blind deconvolution algorithms. IEEE Trans. Pattern Anal. Mach. Intell. 33(12), 2354–2367 (2011)
12. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Efficient marginal likelihood optimization in blind deconvolution. In: CVPR (2011)
13. Miskin, J.W., MacKay, D.J.C.: Ensemble learning for blind image separation and deconvolution. In: Advances in Independent Component Analysis (2000)
14. Palmer, J.A.: Relatve convexity. Technical report, UCSD (2003)
15. Palmer, J.A., Wipf, D.P., Kreutz-Delgado, K., Rao, B.D.: Variational EM algorithms for non-Gaussian latent variable models. In: NIPS (2006)
16. Shan, Q., Jia, J., Agarwala, A.: High-quality motion deblurring from a single image. In: ACM SIGGRAPH (2008)
17. Xu, L., Jia, J.: Two-phase kernel estimation for robust motion deblurring. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 157–170. Springer, Heidelberg (2010)
18. Wipf, D.P., Rao, B.D., Nagarajan, S.S.: Latent variable Bayesian models for promoting sparsity. IEEE Tran. Info. Theory 57(9), 6236–6255 (2011)
19. Wipf, D., Zhang, H.: Revisiting Bayesian blind deconvolution. MSRA Tech Report (March 2013)
20. Zhang, H., Wipf, D.: Non-uniform blind deblurring with a spatially adaptive prior. MSRA Tech Report (April 2013)

A Variational Method for Expanding the Bit-Depth of Low Contrast Image

Motong Qiao, Wei Wang, and Michael K. Ng

Department of Mathematics, Hong Kong Baptist University

Abstract. Traditionally, bit-depth expansion is an image processing technique to display a low bit-depth image on a high bit-depth monitor. In this paper, we study a variational method for expanding the bit-depth of low contrast images. Our idea is to develop a variational approach containing an energy functional to determine a local mapping function $f(r, x)$ for bit-depth expansion via a smoothing technique, such that each pixel can be adjusted locally to a high bit-depth value. In order to enhance low contrast images, we make use of the histogram equalization technique for such local mapping function. Both bit-depth expansion and equalization terms can be combined together into the resulting objective function. In order to minimize the differences among the local mapping function at the nearby pixel locations, the spatial regularization of the mapping is incorporated in the objective function. Experimental results are reported to show that the performance of the proposed method is competitive with the other compared methods for several testing low contrast images.

Keywords: bit-depth expansion, variational methods, low contrast, spatial regularization.

1 Introduction

The bit-depth of an image refer to the number of bits used to represent a pixel value. A high bit-depth implies greater ability to store the information. The most widely used images are 8-bit in gray-level or 24-bit in color, which are suitable for displaying on traditional Cathode Ray Tube (CRT) monitors. There are some legacy images stored in a low bit-depth. Also high bit-depth images are emerging along with the development of capturing and displaying instruments in imaging sciences. A low bit-depth image has its advantages on saving storage space and transmission. However, content and contrast are distorted in low bit-depth images. In particular, it would be useful and interesting to display a high bit-depth image from a low bit-depth and low contrast image and This paper aims at investigating an variational method to expand the bit-depth of low contrast images.

1.1 Bit-Depth Expansion Methods

The conventional methods for bit-depth expansion include zero-padding (ZP), multiplication-by-an-ideal-gain (MIG) [1], bit-replication (BR) [1], Gamma-expansion (GE) [2], and high dynamic range imaging (HDRI). Whereas HDRI

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 54–65, 2013.
© Springer-Verlag Berlin Heidelberg 2013

usually requires a collection of photographs taken with different exposures, for instance the methods in [3–6], we focus on the problem of expanding the bit-depth from one given image only.

The bit-depth expansion problem came up from the inverse image dithering problem. The problem can be depicted as follows. Given a $p-$bit image X with a range $[0, 2^p - 1]$ and a $q-$bit $(q > p)$ image Y with a range $[0, 2^q - 1]$, the goal is to convert X to Y. Obviously the most simple idea is through zero padding (ZP): $[Y]_{i,j} = [X]_{i,j} \times 2^{(q-p)}$. Another basic method is to multiply an ideal gain (MIG) and then round to the nearest integer:

$$[Y]_{i,j} = \text{Round}\left([X]_{i,j} \times \frac{2^q - 1}{2^p - 1}\right)$$

The other method combining the gamma selection is used in Akyuz's psychophysical experiments [2], namely gamma expansion (GE) method:

$$Y = k(\frac{X - X_{\min}}{X_{\max} - X_{\min}})^{\gamma},$$

where k represents the highest possible value of Y and γ determines the nonlinearity of the scaling. In [1], a bit replication (BR) method is proposed:

$$[X]_{i,j} = x_{p-1}x_{p-2}\cdots x_1x_0; \quad [Y]_{i,j} = x_{p-1}x_{p-2}\cdots x_1x_0x_{p-1}x_{p-2}\cdots,$$

where x_k is the k^th bit of the (i,j)th pixel value $[X]_{i,j}$ of X. These methods achieve an one-to-one mapping, which means a gray level in a low bit-depth image can only be mapped to a certain gray level in a high bit-depth image. It is clear that these methods do not consider spatial pixel locations in adjusting pixel values. Thus the resulting image may suffer from contouring artifacts. Recently, a more sophisticated method is proposed in [7]. The idea is to use MIG as initialization of the high bit-depth image, and then remove the contouring artifacts by a spatial variant filter based on the segmentation of the contour suffered region. The results of this two-step method depend on whether the contour region can be correctly divided from smooth regions. In [8], other low-pass filters with adaptive windows size are developed to obtain a better performance in edge-preservation and contour removal.

1.2 The Contribution

In this paper, we propose and develop a variational approach containing an energy functional to determine a local mapping function $f(r, x)$ for bit-depth expansion, such that each pixel can be adjusted locally to a high bit-depth value. Here r refers to the variable for the number of bits used and x refers to the variable for pixel locations. A smoothing technique can be employed in the expansion process by considering the regularization based on the first-order derivative of f with respect to r: $\int_x \int_r f_r(r, x)^2$. Since low bit-depth images may be degraded due to exposure reasons [9], the visual appearance (contrast) of the

displayed image should be enhanced in the variational method. We can make use of the histogram equalization technique to find a local mapping function to enhance low contrast images: $\int_x \int_r f_r(r,x)^2/h(r,x)$ where $h(r,x)$ is the local histogram of a low contrast image. As both bit-expansion and histogram equalization processes involves the first-order derivative of f with respect to r, we combine them together into the resulting objective function. In order to minimize the differences among the local mapping function at the nearby pixel locations, the spatial regularization of the mapping is also incorporated in the functional: $\int_x \int_r \|\nabla f(r,x)\|_2^2$. To adjust a local mapping function, we also incorporate another penalty term to require the mean brightness of the displayed image can be close to that of the input image. Experimental results are reported to show that the performance of the proposed method is competitive with the other compared methods for several testing low contrast images.

The outline of this paper is organized as follows. In Section 2, we will describe the proposed model. In Section 3, we will present the algorithm to solve the proposed model numerically. In Section 4, we will present numerical examples to show the effectiveness of the proposed model. Finally, the concluding remarks will be given in Section 5.

2 The Proposed Model

Our main idea is to make use of histogram equalization and smoothing techniques to determine an local mapping function to display low bit-depth and low contrast images. Let r and s represent the normalized variables for a low contrast low bit-depth image and its displayed bit-expanded image respectively, i.e., $0 \leq r, s \leq 1$. Assume that $h(r)$ represents the normalized histograms for the input low bit-depth image. A function $s = f(r)$, is used to map each input level r to a new level s to achieve the required enhancement objective. It has been shown in [10] that the histogram equalization problem can be considered a variational minimization problem as follows:

$$\min_f J(f) = \int_0^1 \frac{1}{h(r)} f_r^2 \mathrm{dr}, \tag{1}$$

where f_r is the first derivative of f with respect to r.

Suppose Ω denotes the image domain. In order to preserve more local image details and to make the enhancement according to such details of an input low bit-depth image, we employ a local transformation $f(r,x)$, where $(r,x) \in \Lambda = (0,1) \times \Omega$. Here, at each pixel location x, we design a transformation $f(r,x)$. Then a term of an objective function should contain

$$\int_\Lambda \frac{1}{h(r,x)} f_r(r,x)^2 \mathrm{dxdr}. \tag{2}$$

On the other hand, a smoothing technique is used for bit-depth expansion. We can include a term

$$\int_\Lambda f_r(r,x)^2 \mathrm{dxdr}$$

Here $f_r(r,x)$ is the first-order derivative of f with respect to r. As both bit-expansion and histogram equalization processes involves $f_r(r,x)$, we combine them together by considering (2) only into the resulting objective function.

In order to minimize the differences among the local transformations at the nearby pixel locations, the spatial regularization of the transformation is also incorporated in the functional for the equalization process. In particular, we consider the H_1-norm regularization $|\nabla f|^2$ of f in the model, where ∇ denotes the gradient operator of f with respect to the horizontal and vertical directions of an image. Moreover, we can incorporate another penalty term that the mean brightness of the displayed high bit-depth image can be close to that of the input low bit-depth image. The proposed variational model is given as follows:

$$\Phi(f) = \int_\Lambda \frac{1}{h(r,x)} f_r(r,x)^2 \mathrm{dxdr} + \gamma_1 \int_\Lambda |\nabla f|^2 \mathrm{dxdr} + \gamma_2 \int_\Omega \left(\int_0^1 f h(r,x) \mathrm{dr} - \mu \right)^2 \mathrm{dx}, \tag{3}$$

where γ_1 and γ_2 are two positive regularization parameters, and μ is the mean brightness of the input low bit-depth image.

According to [7, 8], the disadvantage of ZP, MIG, BR and GE methods is that they only provide one-to-one mapping from a low to high bit-depth image. Indeed, most of possible values in high bit-depth image are not used. In the proposed, we make use of the variational method to determine a local transformation which gives a one-to-many mapping from a low to high bit-depth image. According to local transformation based on local image details, the same pixel value at two different locations may be mapped to two different pixel values, and the two pixel values at the nearby pixel locations may be mapped to close pixel values. We expect the noise and contour artifacts of the displayed high bit-depth image can be resolved, and its local contrast can also be enhanced.

In a discrete setting, the functional in (3) can be written as follows:

$$\Phi_d(\mathbf{f}) = \sum_{i=1}^{R} \sum_{j=1}^{N} \frac{(\mathbf{D}_{i,j}^{(r)} \mathbf{f})^2}{h_{i,j}} + \gamma_1 \sum_{i=1}^{R} \sum_{j=1}^{N} \left\| \mathbf{D}_{i,j}^{(x)} \mathbf{f} \right\|_2^2 + \gamma_2 \sum_{j=1}^{N} \left(\sum_{i=1}^{R} \mathbf{f}_{i,j} h_{i,j} - \mu \right)^2, \tag{4}$$

where R and N refers to the number of high bit-depth levels and the number of pixel values respectively, $\mathbf{f} = [\mathbf{f}_{i,j}]$ is the n-vector containing the transformation function values at the i-th high bit-depth level and j-th pixel location (the lexicographic ordering of $\mathbf{f}$ is used), $n = R \times N$ refers to the total number of unknowns. For a displayed high bit-depth image of q bits, R is equal to 2^q. Moreover, $\mathbf{D}_{i,j}^{(r)} \mathbf{f}$ is the discrete derivative value of $\mathbf{f}$ with respect to the i-th high bit-depth level at j-th pixel location, and $\mathbf{D}_{i,j}^{(x)} \mathbf{f}$ is the discrete gradient vector of $\mathbf{f}$ with respect to the j-th pixel location at the i-th high bit-depth level (the Euclidean norm $\|\mathbf{D}_{i,j}^{(x)} \mathbf{f}\|_2^2$ is used). It is clear that $\mathbf{D}_{i,j}^{(r)}$ is an 1-by-n matrix and $\mathbf{D}_{i,j}^{(x)}$ is an 2-by-n matrix for $1 \le i \le R$ and $1 \le j \le N$.

As a summary, the following minimization problem is employed for displaying a high bit-depth image:

$$\min_{\mathbf{f}} \Phi_d(\mathbf{f}) \quad \text{subject to } 0 \le \mathbf{f} \le 1 \tag{5}$$

3 The Algorithm

In practice, we employ the alternating direction method of multipliers (ADMM) to solve the constrained optimization problem in (5). For simplicity, we form n-by-n matrix $\mathbf{D}^{(r)} = [(\mathbf{D}_{i,j}^{(r)})^T]$ and $2n$-by-n matrix $\mathbf{D}^{(x)} = [(\mathbf{D}_{i,j}^{(x)})^T]$. By using the Lagrangian multipliers λ_1, λ_2, λ_3, λ_4 to the linear constraints:

$$\mathbf{u} = \mathbf{D}^{(r)}\mathbf{f}, \quad \text{and} \quad \mathbf{v} = \mathbf{w} = \mathbf{z} = \mathbf{f},$$

with $\mathbf{u} = [u_{i,j}]$, the augmented Lagrangian function is given by

$$\begin{aligned}
&\mathcal{L}(\mathbf{f}, \mathbf{u}, \mathbf{v}, \mathbf{w}, \mathbf{z}, \lambda_1, \lambda_2, \lambda_3, \lambda_4) \\
&= \iota(\mathbf{z}) + \sum_{i=1}^{R}\sum_{j=1}^{N} \frac{\mathbf{u}_{i,j}^2}{h_{i,j}} + \gamma_1 \sum_{i=1}^{R}\sum_{j=1}^{N} \left\| \mathbf{D}_{i,j}^{(x)}\mathbf{v} \right\|_2^2 + \gamma_2 \sum_{j=1}^{N} \left(\sum_{i=1}^{R} \mathbf{w}_{i,j} h_{i,j} - \mu \right)^2 + \\
&< \lambda_1, \mathbf{u} - \mathbf{D}^{(r)}\mathbf{f} > + < \lambda_2, \mathbf{v} - \mathbf{f} > + < \lambda_3, \mathbf{w} - \mathbf{f} > + < \lambda_4, \mathbf{z} - \mathbf{f} > + \\
&\beta(||\mathbf{u} - \mathbf{D}^{(r)}\mathbf{f}||_2^2 + ||\mathbf{v} - \mathbf{f}||_2^2 + ||\mathbf{w} - \mathbf{f}||_2^2 + ||\mathbf{z} - \mathbf{f}||_2^2),
\end{aligned}$$

where

$$\iota(\mathbf{z}) := \begin{cases} 0, & 0 \le \mathbf{z} \le 1, \\ +\infty, & \text{otherwise}, \end{cases}$$

and $< \cdot, \cdot >$ is the inner product of Euclidean space.

The Algorithm:

(i) Set $\mathbf{f}^0 = \tilde{\mathbf{f}}$, $\lambda_1^0 = \tilde{\lambda}_1$, $\lambda_2^0 = \tilde{\lambda}_2$, $\lambda_3^0 = \tilde{\lambda}_3$, $\lambda_4^0 = \tilde{\lambda}_4$ be the initial input data;

(ii) At the kth iteration:

– Given $\mathbf{f}^k$, λ_1^k, λ_2^k, λ_3^k, λ_4^k, and compute $\mathbf{u}^{k+1}$, $\mathbf{v}^{k+1}$, $\mathbf{w}^{k+1}$, $\mathbf{z}^{k+1}$ by solving:

$$\min_{\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{z}} \mathcal{L}(\mathbf{f}^k, \mathbf{u}, \mathbf{v}, \mathbf{w}, \mathbf{z}, \lambda_1^k, \lambda_2^k, \lambda_3^k, \lambda_4^k); \tag{6}$$

– Given $\mathbf{u}^{k+1}, \mathbf{v}^{k+1}, \mathbf{w}^{k+1}, \mathbf{z}^{k+1}$, and compute $\mathbf{f}^{k+1}$ by solving:

$$\mathcal{L}(\mathbf{f}, \mathbf{u}^{k+1}, \mathbf{v}^{k+1}, \mathbf{w}^{k+1}, \mathbf{z}^{k+1}, \lambda_1^k, \lambda_2^k, \lambda_3^k, \lambda_4^k); \tag{7}$$

– Updating $\lambda_1^{k+1}, \lambda_2^{k+1}, \lambda_3^{k+1}, \lambda_4^{k+1}$ by using:

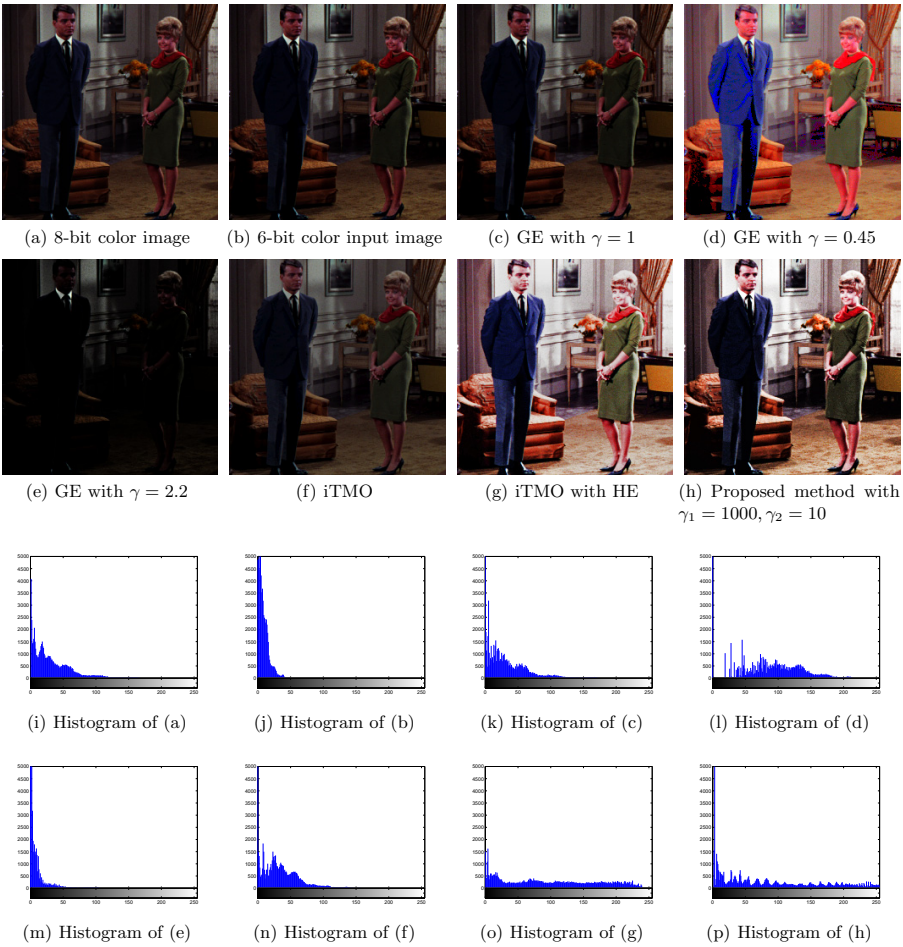
$$\begin{aligned}
\lambda_1^{k+1} &= \lambda_1^k + 2 * \beta(\mathbf{u}^{k+1} - \mathbf{D}^{(r)}\mathbf{f}^{k+1}), \\
\lambda_2^{k+1} &= \lambda_2^k + 2 * \beta(\mathbf{v}^{k+1} - \mathbf{f}^{k+1}), \\
\lambda_3^{k+1} &= \lambda_3^k + 2 * \beta(\mathbf{w}^{k+1} - \mathbf{f}^{k+1}), \\
\lambda_4^{k+1} &= \lambda_4^k + 2 * \beta(\mathbf{z}^{k+1} - \mathbf{f}^{k+1}).
\end{aligned}$$

(iii) Go back to Step (ii) until $\frac{||\mathbf{f}^{k+1} - \mathbf{f}^k||}{||\mathbf{f}^{k+1}||} \leq \epsilon$.

For the subproblem in (6), all the unknowns can be solved separately. In the next section, we test the proposed algorithm to expand low bit-depth and low contrast images.

4 Numerical Results

For ease and obviousness of displaying the results, we test our variational method on 8-bit (in each channel) RGB images of size 256×256 as shown in Figures 1-5(a). The input images in Figures 1-5(b) are obtained from 8-bit original images in Figures 1-5(a) by averaging, and some of them are manually lowered the contrast



(a) 8-bit color image (b) 6-bit color input image (c) GE with $\gamma = 1$ (d) GE with $\gamma = 0.45$

(e) GE with $\gamma = 2.2$ (f) iTMO (g) iTMO with HE (h) Proposed method with $\gamma_1 = 1000, \gamma_2 = 10$

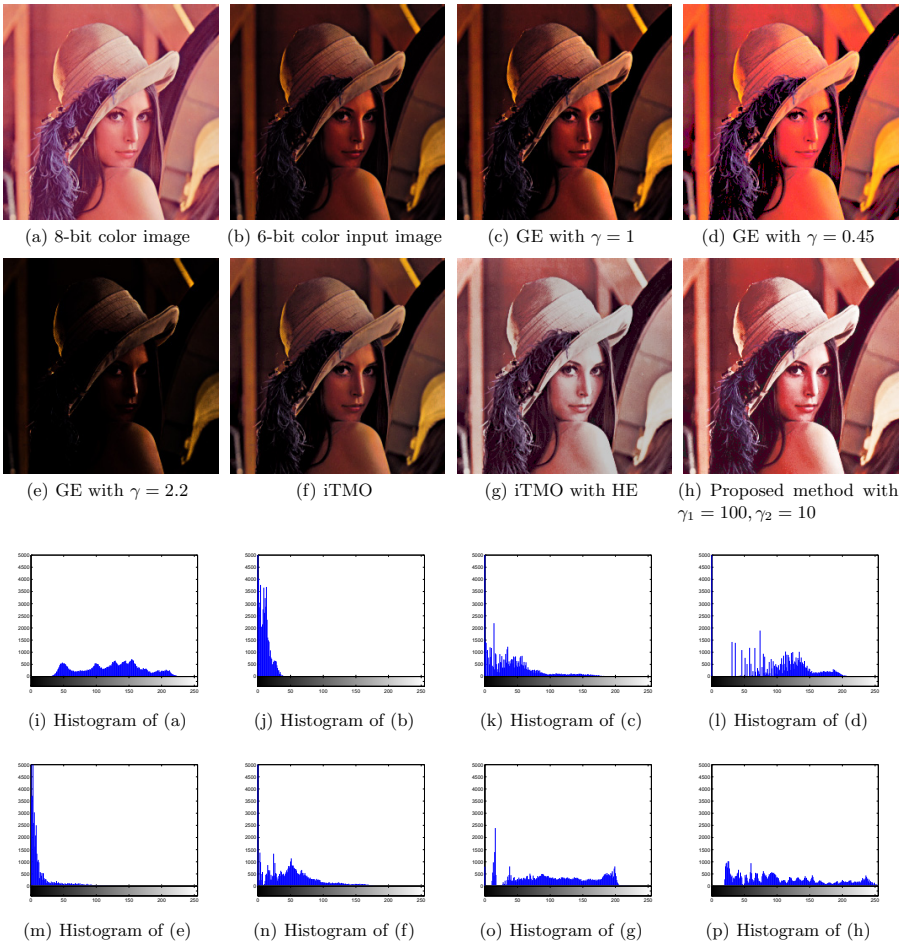
(i) Histogram of (a) (j) Histogram of (b) (k) Histogram of (c) (l) Histogram of (d)

(m) Histogram of (e) (n) Histogram of (f) (o) Histogram of (g) (p) Histogram of (h)

Fig. 1. The Comparison of different methods for the "Couple" image

Table 1. PSNR value comparison

Input	PSNR (dB)					
Images	Proposed	GE with $\gamma = 1$	GE with $\gamma = 0.45$	GE with $\gamma = 2.2$	iTMO	iTMO with HE
Lena	**19.49**	19.09	9.66	6.65	17.09	17.46
Boats	20.22	20.17	13.13	7.53	17.56	**20.42**
Monolake	**24.33**	9.24	12.14	10.23	12.03	20.23
Peppers	**29.35**	7.13	10.09	17.17	9.84	19.22



(a) 8-bit color image (b) 6-bit color input image (c) GE with $\gamma = 1$ (d) GE with $\gamma = 0.45$

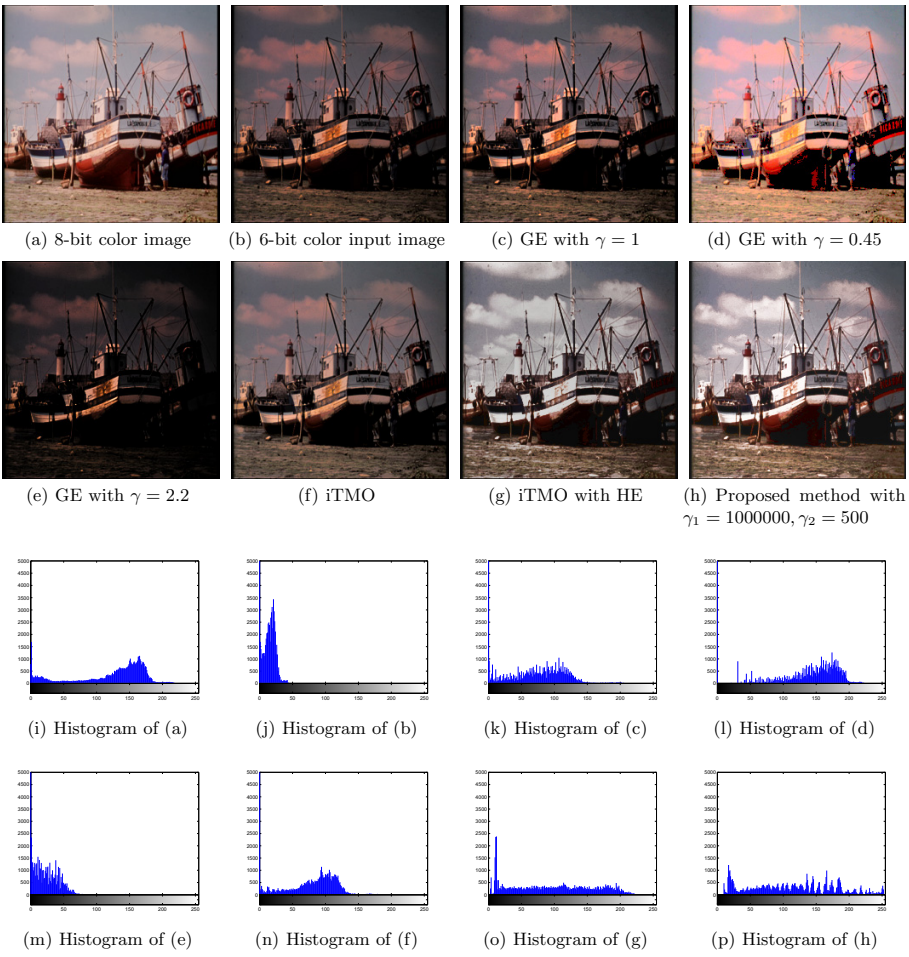
(e) GE with $\gamma = 2.2$ (f) iTMO (g) iTMO with HE (h) Proposed method with $\gamma_1 = 100, \gamma_2 = 10$

(i) Histogram of (a) (j) Histogram of (b) (k) Histogram of (c) (l) Histogram of (d)

(m) Histogram of (e) (n) Histogram of (f) (o) Histogram of (g) (p) Histogram of (h)

Fig. 2. The Comparison of different methods for the "Lena" image

or biased the exposure to simulate different degradation cases. We see from Figures 1(h)-(i), the histogram are biased toward the small values, and this input image in is underexposed. Here the histogram combines all the pixel values in the red, green and blue channels. We see from Figures 2-5(h) that the histograms of Figures 2-



(a) 8-bit color image (b) 6-bit color input image (c) GE with $\gamma = 1$ (d) GE with $\gamma = 0.45$

(e) GE with $\gamma = 2.2$ (f) iTMO (g) iTMO with HE (h) Proposed method with $\gamma_1 = 1000000, \gamma_2 = 500$

(i) Histogram of (a) (j) Histogram of (b) (k) Histogram of (c) (l) Histogram of (d)

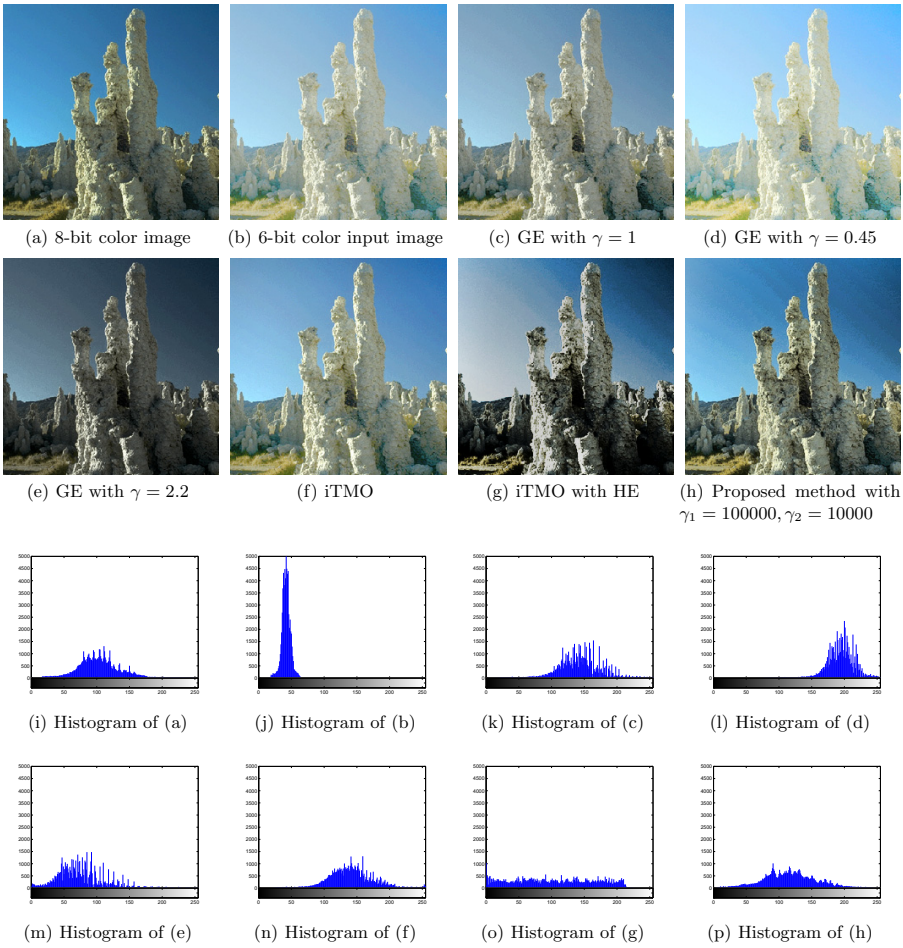
(m) Histogram of (e) (n) Histogram of (f) (o) Histogram of (g) (p) Histogram of (h)

Fig. 3. The Comparison of different methods for the "Boats" image

5(a) are not biased toward the small or large values. Therefore, we manually adjust their pixel values to under-exposed images (see Figures 2-3(b) and 2-3(i)) and the over-exposed images (Figures 4-5(b) and 4-5(i)).

We compare our results with Gamma expansion (GE) method [2] and Banterle's inverse tone mapping operators (iTMO) method [11]. GE method has the effect of adjusting the contrast and iTMO method simulates the camera response function. To keep comparison fair, we also consider doing the histogram equalization to the input images first and then applying iTMO method. The PSNR values between the resulting image and the original 8-bit image are shown in Table 1 except for Figure 1 where it is a low contrast image already.

The parameters in GE method is set as the same in Akyuz's psychophysical investigations in [2] with three gamma alternatives $\gamma = 1, 0.45, 2.2$. The iTMO method inverts the Reinhard's tone mapping operators in [12]. The initial local



(a) 8-bit color image (b) 6-bit color input image (c) GE with $\gamma = 1$ (d) GE with $\gamma = 0.45$

(e) GE with $\gamma = 2.2$ (f) iTMO (g) iTMO with HE (h) Proposed method with $\gamma_1 = 100000, \gamma_2 = 10000$

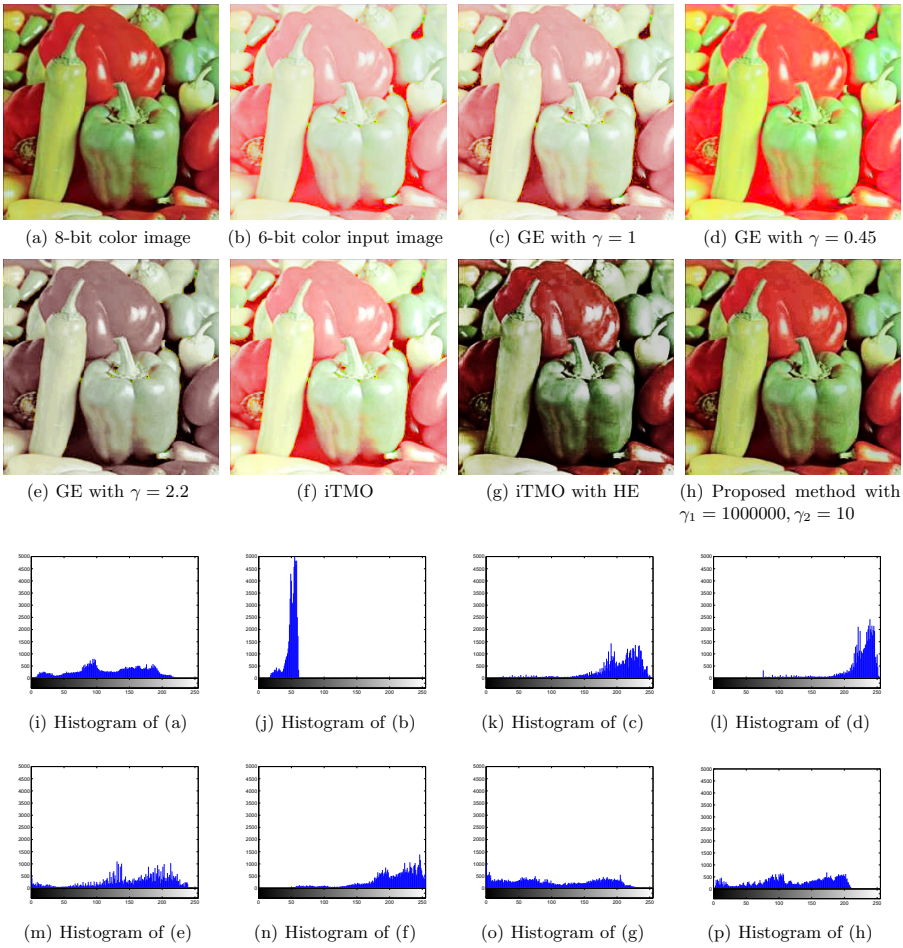
(i) Histogram of (a) (j) Histogram of (b) (k) Histogram of (c) (l) Histogram of (d)

(m) Histogram of (e) (n) Histogram of (f) (o) Histogram of (g) (p) Histogram of (h)

Fig. 4. The Comparison of different methods for the "Monolake" image

histogram $h_{i,j}$ in equation (4) was obtained by calculating the local histograms of the input low contrast image and then linearly project them onto the desired dynamic range with zeros padding. The values of parameters β, γ_1 and γ_2 are tuned according to the PSNR values between the displayed image and the original 8-bit image. In the tests, we set the fixed value of the penalty parameter β to be 100 in the ADMM method. The stopping criterion of the ADMM method is that the relative difference between the successive iterates ϵ is less than 1×10^{-4}.

According to Figures 1-5, it can be shown from the results that the proposed method achieves better visual appearance and obtain higher PSNR values than the other testing methods in most cases. The local contrast is enhanced to increase detailed visibility. In the meanwhile, the over-exposed or under-exposed images can be adjusted through the histogram redistribution process to achieve a observer-friendly display in high bit-depth setting. These results are also re-



(a) 8-bit color image (b) 6-bit color input image (c) GE with $\gamma = 1$ (d) GE with $\gamma = 0.45$

(e) GE with $\gamma = 2.2$ (f) iTMO (g) iTMO with HE (h) Proposed method with $\gamma_1 = 1000000, \gamma_2 = 10$

(i) Histogram of (a) (j) Histogram of (b) (k) Histogram of (c) (l) Histogram of (d)

(m) Histogram of (e) (n) Histogram of (f) (o) Histogram of (g) (p) Histogram of (h)

Fig. 5. The Comparison of different methods for the "Peppers" image

flected from the more equalized histograms generated by the proposed method, see Figures 1-5(n). However, the histograms generated by the other methods are still biased toward either small or large pixel values, see Figures 1-5(j), 1-5(k), 1-5(l), and 1-5(m). Indeed, the results by iTMO show that the contrast can also be enhanced but only on the limited middle range. Since it only reverts by using one nonlinear *s*-shaped camera response function. it cannot deal with the regions with small or large pixel values, especially when the input image is under-exposed or over-exposed. The GE method can also control the contrast uniformly, but it cannot improve local contrast of input images.

Finally, we report the ADMM method converges very fast. In our examples, it takes around 5-10 iterations to obtain the resulting high bit-depth image, and

the computational time is about 15 seconds in average. Since we are doing local histogram calculation, the algorithm is easy to be modified to parallel computing for acceleration.

5 Concluding Remarks

In this paper, we have presented a variational method to generate a high bit-depth image from a single low bit-depth image so that it can be appropriately displayed on a high bit-depth monitor or projector. The pixel values from a low bit-depth image are mapped to that on a high bit-depth image by spatial variant mapping functions, derived from the constrained variational histogram equalization method. The energy minimization problem can be solved efficiently by ADMM algorithm. From the experimental results, we can see that the detailed visibility can be enhanced as well as avoiding the over-enhancement of the noise and other artifacts. The proposed method can also deal with the under-exposed or over-exposed input images to enhance their local contrast in the resulting high bit-depth image.

References

1. Ulichney, R., Cheung, S.: Pixel Bit-Depth Increase by Bit Replication. Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts III, Proceeding of SPIE, pp. 232–241 (1998)
2. Akyüz, A.O., Fleming, R., Riecke, B.E., Reinhard, E., Bülthoff, H.H.: Do HDR displays support LDR content?: a psychophysical evaluation. ACM Transactions on Graphics (TOG) 26(3), 38 (2007)
3. Robertson, M.A., Borman, S., Stevenson, R.L.: Dynamic Range Improvement through Multiple Exposures. In: Proceedings of International Conference on Image Processing, vol. 3, pp. 159–163 (1999)
4. Debevec, P.E., Malik, J.: Recovering High Dynamic Range Radiance Maps from Photographs. ACM SIGGRAPH, classes. 31 (2008)
5. Kao, W.-C.: High Dynamic Range Imaging by Fusing Multiple Raw Images and Tone Reproduction. IEEE Transactions on Consumer Electronics 54(1), 10–15 (2008)
6. Grossberg, M.D., Nayar, S.K.: High Dynamic Range from Multiple Images: Which Exposures to Combine? In: Proceedings of ICCV Workshop on Color and Photometric Methods in Computer Vision (2003)
7. Liu, C.-H., Au, O.C., Wong, P.H.W., Kung, M.C., Chao, S.-C.: Bit-Depth Expansion by Adaptive Filter. In: IEEE Inernational Symposium on Circuits and Systems, pp. 496–499 (2008)
8. Taguchi, A., Nishiyama, J.: Bit-Length Expansion by Inverse Quantization Process. In: Proceedings of the 20th European Signal Processing Conference (EUSIPCO), pp. 1543–1547 (2012)
9. Jongseong, C., Min Kyu, P., Moon Gi, K.: High Dynamic Range Image Reconstruction with Spatial Resolution Enhancement. The Computer Journal 52(1), 114–125 (2009)

10. Altas, I., Louis, J., Belward, J.: A Variational Approach to the Radiometric Enhancement of Digital Imagery. IEEE Transactions on Image Processing 4(6), 845–849 (1995)
11. Banterle, F., Ledda, P., Debattista, K., Chalmers, A.: Inverse Tone Mapping. In: Proceedings of 4th International Conference on Computer Graphics and Interactive Techniques in Australisia and Southeast Asia, pp. 349–356 (2006)
12. Reinhard, E., Stark, M., Shirley, P., Ferwerda, J.: Photographic Tone Reproduction for Digital Images. ACM Transactions on Graphics 21(3), 267–276 (2002)

Variational Shape from Light Field

Stefan Heber, Rene Ranftl, and Thomas Pock*

Institute for Computer Graphics and Vision,
Graz University of Technology, Austria
{stefan.heber,ranftl,pock}@icg.tugraz.at
http://www.icg.tu-graz.ac.at/

Abstract. In this paper we propose an efficient method to calculate a high-quality depth map from a single raw image captured by a light field or plenoptic camera. The proposed model combines the main idea of Active Wavefront Sampling (AWS) with the light field technique, *i.e.* we extract so-called sub-aperture images out of the raw image of a plenoptic camera, in such a way that the virtual view points are arranged on circles around a fixed center view. By tracking an imaged scene point over a sequence of sub-aperture images corresponding to a common circle, one can observe a virtual rotation of the scene point on the image plane. Our model is able to measure a dense field of these rotations, which are inversely related to the scene depth.

Keywords: Light field, depth, continuous optimization.

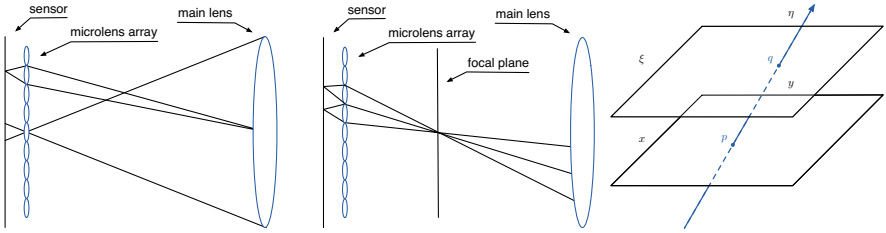
1 Introduction

In geometrical optics, rays are used to model the propagation of light. The amount of light propagated by a ray is denoted as radiance, and the radiance along all rays in a 3D space is called the plenoptic function [1]. The plenoptic function is a 5D function, due to the fact that a ray in 3D space can be parametrized via a position (x, y, z) and a direction (ξ, η). However, the 5D plenoptic function contains redundant information, because the radiance along a ray remains constant till it hits an object. Thus, the redundant information is one dimensional which reduces the 5D plenoptic function to the 4D light field [14] or Lumigraph [12].

There are different devices to capture light fields. The simplest ones are single moving cameras (gantry constructions), which allow a large baseline between captured viewpoints, but are limited to static scenes. Another way of capturing light fields is via multiple cameras. This approach allows to capture dynamic scenes, but is very hardware intensive. In order to capture a full light field, multiple cameras have to be arranged in a 2D array [22]. A further device which re-attracts attention in recent years is the light field [14] or plenoptic camera [2]. As long ago as in the year 1908, Lippmann [15] introduced the basic idea of such a camera. The technique has then been developed and improved by various authors [8–10], but it needed nearly a century till the first commercial plenoptic camera has become available [19, 18].

* This work was funded by the Austrian Science Fund (FWF).

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 66–79, 2013.
© Springer-Verlag Berlin Heidelberg 2013



(a) plenoptic 1.0 (b) plenoptic 2.0 (c) parametrization

Fig. 1. (a) and (b) illustrates different types of light field or plenoptic cameras defined by Lumsdaine and Georgiev [16]. (a) Sketch of a traditional light field camera, also called *plenoptic 1.0 camera*. (b) Sketch of a focused light field camera, also denoted as *plenoptic 2.0 camera*. (c) Sketch of the used light field parametrization. The 4D light field $L(x, y, \xi, \eta)$ is rewritten as $L(p, q)$, where p is a point in the image plane and q is a point in the lens plane. Thus p represents the spatial component and q represents the directional component.

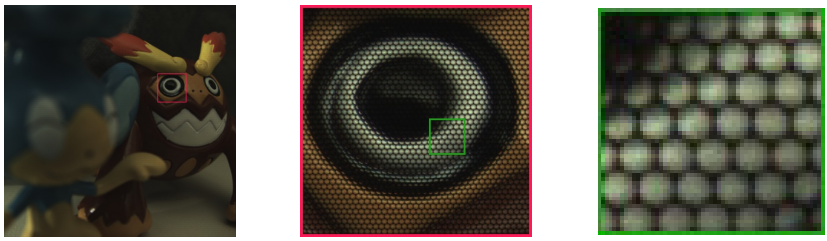
Compared to a conventional camera (2D photograph), which only captures the total amount of light striking each point on the image sensor, a light field camera records the amount of light of individual light rays, which contribute to the final image. Recording the additional directional information is achieved by inserting a micro-lens-array into the optical train of a conventional camera. This has the effect, that the micro-lenses separate the incoming light into several rays of different directions. The individual light rays are then captured at different locations on the sensor.

Depending on the focusing position of the main lens and the micro-lenses, Lumsdaine and Georgiev [16] distinguished between two types of plenoptic cameras, *plenoptic 1.0* and *plenoptic 2.0* (cf Figure 1(a) and 1(b)). In a traditional plenoptic camera (*plenoptic 1.0*) [2, 19, 18] the main lens is focused at the micro-lens plane and the micro-lenses are focused at the main lens (optical infinity). Thus, the position of each micro-lens captures spatial information, and the part of the sensor under each micro-lens captures angular information. Note, that each micro-lens spans the same angular range. Thus, the size of the individual micro-lenses sets the spatial sampling resolution, which leads to a fundamental trade-off: For a fixed sensor resolution, the increase of directional resolution, simultaneously decreases the spatial resolution of the final image. In a focused plenoptic camera (*plenoptic 2.0*) the micro-lenses are focused on the focal plane of the main lens, which has on the one hand the effect, that angular information is spread across different micro-lenses, but on the other hand the camera now records dense spatial information, rather than dense directional information, which results in a higher spatial resolution of the final image.

Using a two-plane parametrization (cf Figure 1(c)) the general structure of a light field can be considered as a 4D function

$$L : \Omega \times \Pi \to \mathbb{R}, \qquad (\boldsymbol{p}, \boldsymbol{q}) \mapsto L(\boldsymbol{p}, \boldsymbol{q}) \tag{1}$$

where $\boldsymbol{p} := (x, y)^{\mathrm{T}}$ and $\boldsymbol{q} := (\xi, \eta)^{\mathrm{T}}$ represent coordinate pairs in the image plane $\Omega \subset \mathbb{R}^2$ and in the lens plane $\Pi \subset \mathbb{R}^2$, respectively.



(a) (b) (c)

Fig. 2. Raw image data captured with a *plenoptic 1.0* camera. (a) shows the complete raw image and (b) and (c) are closeup views, which show the effect of the micro-lens array. Each micro-lens splits the incoming light into rays of different directions, where each ray hits the image sensor behind the micro-lens at a different location. Thus the use of such a micro-lens array makes it possible to capture the 4D light field.

There are several different visualizations of the light field data. The most obvious one, in the case of a plenoptic camera, is the raw image recorded by the sensor itself. This raw image is a composition of small discs, where each disc represents the image of a specific micro-lens. A typical example of a raw image obtained by a plenoptic camera is shown in Figure 2. Another representation can be obtained by extracting all values out of the raw image, which correspond to light-rays with the same direction. In the case of a *plenoptic 1.0* this means one has to consider all image values which are located at the same position in the disc-like image of each micro-lens. As in [18], images obtained in such a way will be referred to as sub-aperture images. In terms of the 4D light field $L(\boldsymbol{p}, \boldsymbol{q})$, a sub-aperture image is an image obtained by holding a direction $\boldsymbol{q}$ fixed and varying over all image positions $\boldsymbol{p}$. This sub-aperture representation provides an interesting interpretation of the light field data as a series of images with slightly different viewpoints, which are parallel to a common image plane. This shows that the light field provides information about the scene geometry. We also want to mention a more abstract visualization of the light field, which goes under the name epipolar image. An epipolar image is a slice of the light field, where one coordinate of the position $\boldsymbol{p}$ and one coordinate of the direction $\boldsymbol{q}$ is held constant.

It has been shown, that the light field can be used for different image processing tasks, like *e.g.* digital refocusing [13, 18], extending the depth of field [18], digital correction of lens aberrations [18], super-resolution [4, 24], and depth estimation [3, 23]. In this paper we will focus mainly on the latter task of depth calculation.

There has been done a lot of research in developing algorithm for traditional stereo estimation, but there is a lack of algorithms, which are capable of exploiting the structure within a light field. To the best of our knowledge, there are two recent works, which consider the task of depth estimation using a plenoptic camera. First, Bishop and Favaro [3] proposed an iterative multi-view method for depth estimation which considers all possible combinations of sub-aperture images on a rectangular grid aligned with the micro-lens center. Second, Wanner and Goldluecke [23] proposed a method for depth labeling of light fields, where they make use of the epipolar representation of

the light field. They also propose to enforce additional global visibility constraints, but even without this additional step, their method is computational expensive.

2 Methodology

The proposed method is motivated by the idea of Active Wavefront Sampling (AWS)[11], which is a 3D surface imaging technique, that uses a conventional camera and a so called AWS module. The simples type of an AWS module is an off-axis aperture, which is rotated around the optical axis. This circular movement of the aperture results in a rotation of a scene point's image on the image plane. This has the effect, that the scene point's depth is encoded by the radius of the according rotation on the image plane. A scene point located on the in-focus plane of the main lens will have a zero radius and thus its image will remain constant throughout all aperture positions, whereas scene points located at increasing distances from the in-focus plane will rotate on circles with increasing radii. Note, that a scene point which is located behind the in-focus plane will have an image rotation that is shifted by π on the rotation circle, compared to a scene point in-front of the in-focus plane. After calculating the circle rotation, the true depth can be obtained by simple geometric considerations.

Due to the fact that an image recorded with a traditional camera and an additional AWS module is similar to a specific sub-aperture image extracted from the light field, it is possible to apply the main idea of AWS to the light field setting. More precisely, in the case of a *plenoptic 1.0 camera*, *i.e.* that we have to extract sub-aperture images, where the directional positions lie on a circle centered at the origin of the lens plane Π.

Contrary to AWS, the light field provides much more information than just sub-aperture images corresponding to a single rotating off-axis aperture. The light field data allows to extract sub-aperture images corresponding to arbitrary circles and, what is even more important, it also allows to extract the center view. Moreover, it should also be mentioned, that in the light field setting the different sub-aperture images correspond to the same point in time. In the AWS setting images are captured at different times, where the time-difference depends on the time needed to mechanically move the rotating off-axis aperture from one position to the next.

We also want to note, that, in the AWS setting as well as in the light field setting, a circle movement is not the only path, which can be used. In general an arbitrary path can be used, and depth can be recovered as long as the path is known. However, in this work we will restrict ourselves to circular paths. On the one hand circular paths are used due to the fact that such patterns match the raw image data better than rectangular ones, and thus reduce unwanted effects like vignetting. On the other hand the use of circular patterns also simplifies the model.

3 Shape from Light Field Model

We now continue with formulating the proposed stereo model, which is based on variational principles. As an example we will briefly describe the general variational problem for stereo estimation, which is usually written as follows:

$$\min_d \lambda \Psi(d; I_L, I_R) + \Phi(d)\,, \tag{2}$$

where Ψ measures the data fidelity between two image I_L and I_R for a given disparity map d, Φ is a regularization function, and $\lambda \geqslant 0$ weights the influence of the data term. Common fidelity terms are *e.g.* the Euclidean norm, or the ℓ^1 norm.

The problem of stereo matching is in general ill-posed, therefore additional assumptions about the disparity map d are needed. This prior knowledge is added via the regularization function Φ. A popular regularization term is the Total Variation $\mathrm{TV}(x) = \int \mathrm{d}|\nabla x|$, which favors piecewise constant solutions.

In what follows we will present the proposed model for estimating a depth map for a given light field $L(\boldsymbol{p}, \boldsymbol{q})$. The model can be seen as a specialized multi-view stereo matching approach, where the rotation radius of a scene point's image is measured with respect to a given center position. We will first describe the data-fidelity term and discuss afterwards suitable regularization functions.

3.1 Data Fidelity Term

In our model we assume a *plenoptic 1.0 camera*, and we denote with $u : \Omega \to \mathbb{R}$ a function which defines for each scene point, captured at the position $\boldsymbol{p} \in \Omega$ in the center view $L(\cdot, \mathbf{0})$, the corresponding scene point's largest image rotation radius. This allows us to state the following energy in the continuous setting:

$$E_{data}(u) = \int_{\Omega} \int_{0}^{R} \int_{0}^{2\pi} \psi_{s,r}(\boldsymbol{p}, u(\boldsymbol{p}))\ \mathrm{d}(s,\, r,\, \boldsymbol{p})\,, \tag{3}$$

with

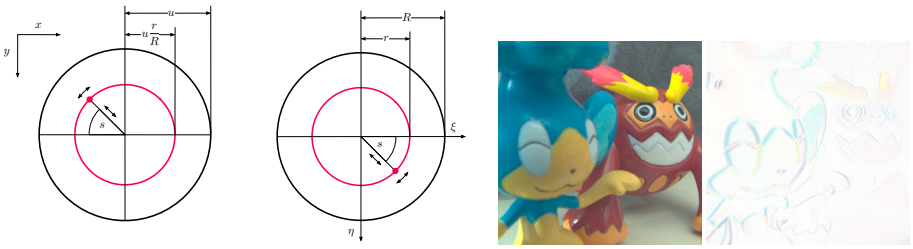
$$\psi_{s,r}(\boldsymbol{p}, u(\boldsymbol{p})) = \theta\left(L\left(\boldsymbol{p}, \mathbf{0}\right) - L\left(\boldsymbol{p} - u(\boldsymbol{p})\frac{\varphi_{s,r}}{R}\,, \varphi_{s,r}\right)\right)\,, \tag{4}$$

where $\varphi_{s,r} = r\left(\cos(s), \sin(s)\right)^{\mathrm{T}}$ is a circle parametrization with radius r and center at the origin, $\Omega \subset \mathbb{R}^2$ is the image domain, $\theta(\cdot)$ denotes an error estimator, and $R > 0$ is the predefined largest allowed circle radius in the lens plane, which corresponds to the largest possible aperture radius of the main lens. Thus, $\psi_{s,r}(\boldsymbol{p}, u(\boldsymbol{p}))$ measures the brightness difference between the center view $L(\cdot, \mathbf{0})$ at position $\boldsymbol{p}$ and the sub-aperture image $L(\cdot, \varphi_{s,r})$ at position $\boldsymbol{p} - u(\boldsymbol{p})\frac{\varphi_{s,r}}{R}$. Here, the latter position describes for varying s a circle in the image plane centered at $\boldsymbol{p}$ and with radius $u(\boldsymbol{p})\frac{r}{R}$ (cf Figure 3).

As already mentioned above, common fidelity terms are the Euclidean norm or the ℓ^1 norm. The corresponding error functions are the quadratic differences and the absolute differences, respectively. The Euclidean norm provides the advantage of being differentiable, which allows to apply standard optimization techniques. But it comes with the disadvantage of not being robust to outliers, which occur in areas of occlusions. The ℓ^1 norm on the other hand is non-smooth, but it is more robust to outliers, and hence we will make us of it, *i.e.* we choose $\theta(x) = |x|$.

In order to obtain a convex approximation of the data term (3) we use first-order Taylor approximations for the sub-aperture images, *i.e.*

$$\begin{aligned} &L\Big(\boldsymbol{p} - u(\boldsymbol{p})\frac{\varphi_{s,r}}{R}, \varphi_{s,r}\Big) \approx \\ &\quad L\left(\boldsymbol{p} - u_0(\boldsymbol{p})\frac{\varphi_{s,r}}{R}, \varphi_{s,r}\right) + \left(u(\boldsymbol{p}) - u_0(\boldsymbol{p})\right)\ \frac{r}{R}\,\nabla_{-\frac{\varphi_{s,r}}{r}} L\left(\boldsymbol{p} - u_0(\boldsymbol{p})\frac{\varphi_{s,r}}{R}, \varphi_{s,r}\right), \end{aligned} \tag{5}$$



(a) rotation in Ω (b) rotation in Π (c) center view (d) difference image

Fig. 3. (a) and (b) illustrate the parametrization used in (3). (a) sketches a scene point's image position (purple dot) and the corresponding rotation circle, and (b) shows the according directional sampling position in the lens plane for extracting the sub-aperture image. (c) and (d) provide a visualization of the scene point's image rotation. (c) is the center view and (d) shows the color inverted difference image between two sub-aperture images . The two fixed directional components of the light field, which are used to extract the sup-aperture images, are chosen to lie opposite to each other on a circle centered at zero.

where $\nabla_{-\frac{\varphi_{s,r}}{r}}$ denotes the directional derivative, with direction $[-\frac{\varphi_{s,r}}{r}, \mathbf{0}]^{\mathrm{T}}$.

Finally we want to note, that although the ℓ^1 norm is robust to outliers it is not robust to varying illumination. Thus, *e.g.* vignetting effects of the main lens and the microlenses, might lead to problems. In order to be more robust against illumination changes, we apply a structure-texture decomposition [25] on the sub-aperture images, *i.e.* we pre-process each image by removing its low frequency component.

3.2 Regularization Term

In this section we will briefly discuss different regularization terms, which can be added to the data-fidelity term proposed in Section 3.1. A regularization term is needed due to the fact, that the problem of minimizing (3) with respect to u is ill-posed, and therefore the fidelity term alone does not provide sufficient information to calculate a reliable solution. Thus, we additionally assume that u varies smoothly almost everywhere in the image domain. In order to add this assumption to our model, we will use an extension of Total Generalized Variation (TGV) [5]. As indicated by the name, TGV is a generalization of the famous Total Variation (TV). Whereas TV favors piecewise constant solutions, TGV^k favors piecewise polynomial solutions of order $k-1$, *e.g.* TGV^2 will favor piecewise linear solutions. Following the work by Ranftl *et al.* [21], we extend TGV^2 by using an anisotropic diffusion tensor $D^{\frac{1}{2}}$. This diffusion tensor connects the prior with the image content, which leads to solutions with a lower degree of smoothness around depth edges. This image-driven TGV regularization term can be written as

$$E_{reg}(u) = \min_{\boldsymbol{w}} \alpha_1 \|D^{\frac{1}{2}}(\nabla u - \boldsymbol{w})\|_{\mathcal{M}} + \alpha_0 \|\boldsymbol{\nabla} \boldsymbol{w}\|_{\mathcal{M}}\,, \tag{6}$$

where $\|\cdot\|_{\mathcal{M}}$ denotes a Radon norm for vector-valued and matrix-valued Radon measures, and $\alpha_0, \alpha_1 > 0$. Furthermore, as in [21]

$$D^{\frac{1}{2}} = \exp(-\gamma\,|\nabla I|^{\beta})\,\boldsymbol{n}\boldsymbol{n}^{\mathrm{T}} + \boldsymbol{n}^{\perp}\boldsymbol{n}^{\perp^{\mathrm{T}}}, \tag{7}$$

where I denotes the center view $L(\cdot, \mathbf{0})$, $\boldsymbol{n}$ is the normalized gradient, $\boldsymbol{n}^{\perp}$ is a vector perpendicular to $\boldsymbol{n}$, and γ and β are predefined scalars. Combining the data term in (3) and the above regularization term (6) leads to a robust model, which is capable of reconstructing subpixel accurate depth maps.

3.3 Discretization

By discretizing the spatial domain Ω, and the involved circles in (3) we obtain

$$\hat{E}_{data}(u) = \sum_{\boldsymbol{p}\in\hat{\Omega}} \sum_{i=1}^{M} \sum_{j=1}^{N} \psi_{s_j,r_i}(\boldsymbol{p}, u(\boldsymbol{p})), \quad \text{with} \quad s_j = \frac{2\,\pi(j-1)}{N} \text{ and } r_i = \frac{R\,i}{M}, \tag{8}$$

where $\hat{\Omega} := \{(x, y)^{\mathrm{T}} \in \mathbb{N}_0^2 \,|\, x < n,\, y < m\}$ denotes the discrete image domain. Furthermore, M and N are the number of different sampling circles, and the number of uniform sampling positions on each circle, respectively.

3.4 Optimization

In order to optimize the complete discretized problem

$$\min_{u\in\mathbb{R}^{mn}} \lambda\,\hat{E}_{data}(u) + \hat{E}_{reg}(u) \tag{9}$$

we use a primal-dual algorithm, proposed by Chambolle *et al.* [7]. Therefore, we first have to rewrite (9) as a saddle point problem. Note, that $u \in \mathbb{R}^{mn}$ is now represented as a column vector, and $\hat{E}_{reg}(u)$ denotes the discrete version of (6). To simplify notation we first define $\tilde{A}_{ij}$ and $B_{ij} \in \mathbb{R}^{mn}$ as

$$\tilde{A}_{ij} := \left(\frac{r_i}{R}\nabla - \frac{\varphi_{s_j,r_i}}{r_i} L\left(\boldsymbol{p} - u_0(\boldsymbol{p})\frac{\varphi_{s_j,r_i}}{R}, \varphi_{s_j,r_i}\right)\right)_{\boldsymbol{p}\in\hat{\Omega}}, \tag{10}$$

$$B_{ij} := \left(L\,(\boldsymbol{p}, \mathbf{0}) - L\left(\boldsymbol{p} - u_0(\boldsymbol{p})\frac{\varphi_{s_j,r_i}}{R}, \varphi_{s_j,r_i}\right)\right)_{\boldsymbol{p}\in\hat{\Omega}}, \tag{11}$$

and by setting $A_{ij} := \mathrm{diag}(\tilde{A}_{ij})$, it is possible to formulate (9) equivalently as the following saddle point problem

$$\min_{u,\boldsymbol{w}} \max_{\substack{\|\boldsymbol{p_u}\|_\infty \le 1 \\ \|\boldsymbol{p_w}\|_\infty \le 1 \\ \|\boldsymbol{p_{ij}}\|_\infty \le 1}} \Big\{ \lambda \sum_{i=1}^{M}\sum_{j=1}^{N} \langle B_{ij} - A_{ij}(u - u_0), \boldsymbol{p_{ij}}\rangle + \alpha_1 \Big\langle D^{\frac{1}{2}}(\nabla u - \boldsymbol{w}), \boldsymbol{p_u}\Big\rangle + \alpha_0\,\langle \boldsymbol{\nabla w}, \boldsymbol{p_w}\rangle \Big\}, \tag{12}$$

Algorithm 1. Primal-Dual Algorithm for Shape from Light Field

Require: Choose $\sigma > 0$ and $\tau > 0$, s.t. $\tau\sigma = 1$. Set Σ_{p_u}, Σ_{p_w}, $\Sigma_{p_{ij}}$, T_u, and T_w as in (13), $n = 0$, and the rest arbitrary.

while $n < iter$ **do**

// Dual step

$$\boldsymbol{p_u^{n+1}} \leftarrow \mathcal{P}_{\{\|\boldsymbol{p_u}\|_\infty \leqslant 1\}} \left(\boldsymbol{p_u^n} + \sigma \Sigma_{p_u} \alpha_1 \left(D^{\frac{1}{2}} \left(\nabla \bar{u}^n - \bar{\boldsymbol{w}}^{\boldsymbol{n}}\right)\right)\right)$$

$$\boldsymbol{p_w^{n+1}} \leftarrow \mathcal{P}_{\{\|\boldsymbol{p_w}\|_\infty \leqslant 1\}} \left(\boldsymbol{p_w^n} + \sigma \Sigma_{p_w} \alpha_0 \left(\boldsymbol{\nabla} \bar{\boldsymbol{w}}^{\boldsymbol{n}}\right)\right)$$

$$\boldsymbol{p_{ij}^{n+1}} \leftarrow \mathcal{P}_{\{\|\boldsymbol{p_{ij}}\|_\infty \leqslant 1\}} \left(\boldsymbol{p_{ij}^n} + \sigma\lambda \left(B_{ij} - A_{ij}(\bar{u}^n - u_0)\right)\right)$$

// Primal step

$$u^{n+1} \leftarrow u^n - \tau \,\mathrm{T}_u \left(\alpha_1 \nabla^{\mathrm{T}} \left(D^{\frac{1}{2}} \boldsymbol{p_u^{n+1}}\right) + \lambda \textstyle\sum_{i,j} A_{ij} \boldsymbol{p_{ij}^{n+1}}\right)$$

$$\boldsymbol{w^{n+1}} \leftarrow \boldsymbol{w^n} - \tau \,\mathrm{T}_w \left(\alpha_0 \boldsymbol{\nabla}^{\mathrm{T}} \boldsymbol{p_w^{n+1}} - \alpha_1 D^{\frac{1}{2}} \boldsymbol{p_u^{n+1}}\right)$$

$$\bar{u}^{n+1} \leftarrow 2\, u^{n+1} - u^n$$

$$\bar{\boldsymbol{w}}^{\boldsymbol{n+1}} \leftarrow 2\, \boldsymbol{w^{n+1}} - \boldsymbol{w^n}$$

// Iterate

$$n \leftarrow n + 1$$

end while

where the dual variables $\boldsymbol{p_{ij}}$ have the same dimension as B_{ij}. Now we can directly apply the algorithm proposed in [7] to solve (12).

An improvement with respect to convergence speed can be obtained by using adequate symmetric and positive definite preconditioning matrices [20], leading to the update scheme shown in Algorithm 1, which is iterated for a fixed number of iterations or till a suitable convergence criterion is fulfilled. Here $\mathcal{P}$ denotes a reprojection operator, and Σ_*, and T_* are the preconditioning matrices, given as follows

$$\begin{aligned} \Sigma_{p_u} = \frac{1}{3} I, \;\; \Sigma_{p_w} = \frac{1}{2} I, \;\; \Sigma_{p_{ij}} = I, \;\; \mathrm{T}_w = \frac{1}{\alpha_1^2 + 4\alpha_0^2} I \\ \mathrm{T}_u = \left(\alpha_1 \,\mathrm{diag}\left(D_x^{\mathrm{T}} D_x + D_y^{\mathrm{T}} D_y\right)\right)^{-1} + \sum_{i,j} \left(\lambda\, A_{ij}\right)^{-2}, \end{aligned} \tag{13}$$

where I denotes the identity matrix, $\left(\begin{smallmatrix} D_x \\ D_y \end{smallmatrix}\right) := D^{\frac{1}{2}} \nabla$, and $\mathrm{diag}(X)$ takes a matrix X and sets all elements, which are not on the main diagonal to zero.

Due to the fact, that the linear approximation (5) is only accurate in a small area around u_0 we embed the complete algorithm into a coarse-to-fine warping scheme [6].

Fig. 4. Examples of generated synthetic data. The figure shows pairs of images, where one is a closeup view of the synthetic raw image data and the other represents a corresponding sub-aperture image. From left to right the pairs show clean data, data with added vignetting effects, and data with added vignetting effect and additive Gaussian noise.

4 Experimental Results

In this section we first evaluate our method using the Light Field Benchmark Dataset (LFBD)[1], which is a dataset of synthetic light fields created with Blender[2]. All light field scenes within the dataset have a directional resolution of 9×9 pixels per micro-lens, and varying spatial resolutions, which are listed in Table 1. After the synthetic evaluation we will also present some qualitative results for real world data, where we use raw images captured with a Lytro[3] camera as input for the proposed algorithm.

4.1 Synthetic Experiments

We first evaluate our algorithm on synthetic images similar to the ones taken with a plenoptic camera (cf Figure 4), where the images are created using the LFBD. For the synthetic experiments we set $M = 1$ and $N = 8$, which means that we use a single circle and 8 sampling positions per circle. Other parameters are tuned for the different scenes. Figure 5 shows qualitative depth map results for the proposed approach as well as the ground truth data provided in the LFBD. It can be observed, that the proposed approach is capable of creating piecewise smooth depth maps with clear depth discontinuities. Further, we also visualize the relative depth errors in green (red), which are smaller (larger) than 0.2% and 1.0% in column four and five, respectively. It can be seen that the remaining errors are concentrated at occlusion boundaries or at positions of specular highlights.

Next we simulate vignetting effects of the micro-lenses and the main lens as well as additive image noise. For the vignetting effect of the main lens and micro-lenses we reduce the brightness of pixel based on their distance from the image center and the micro-lens center, respectively. As image noise we use additive Gaussian noise with zero mean and a variance with a σ equal to 2% of the image dynamic range. Example images are shown in Figure 4 and the quantitative results can be found in Table 1. To justify the high quality of our depth map results we also provide the results for the variational

[1] http://lightfield-analysis.net
[2] http://www.blender.org/
[3] https://www.lytro.com/

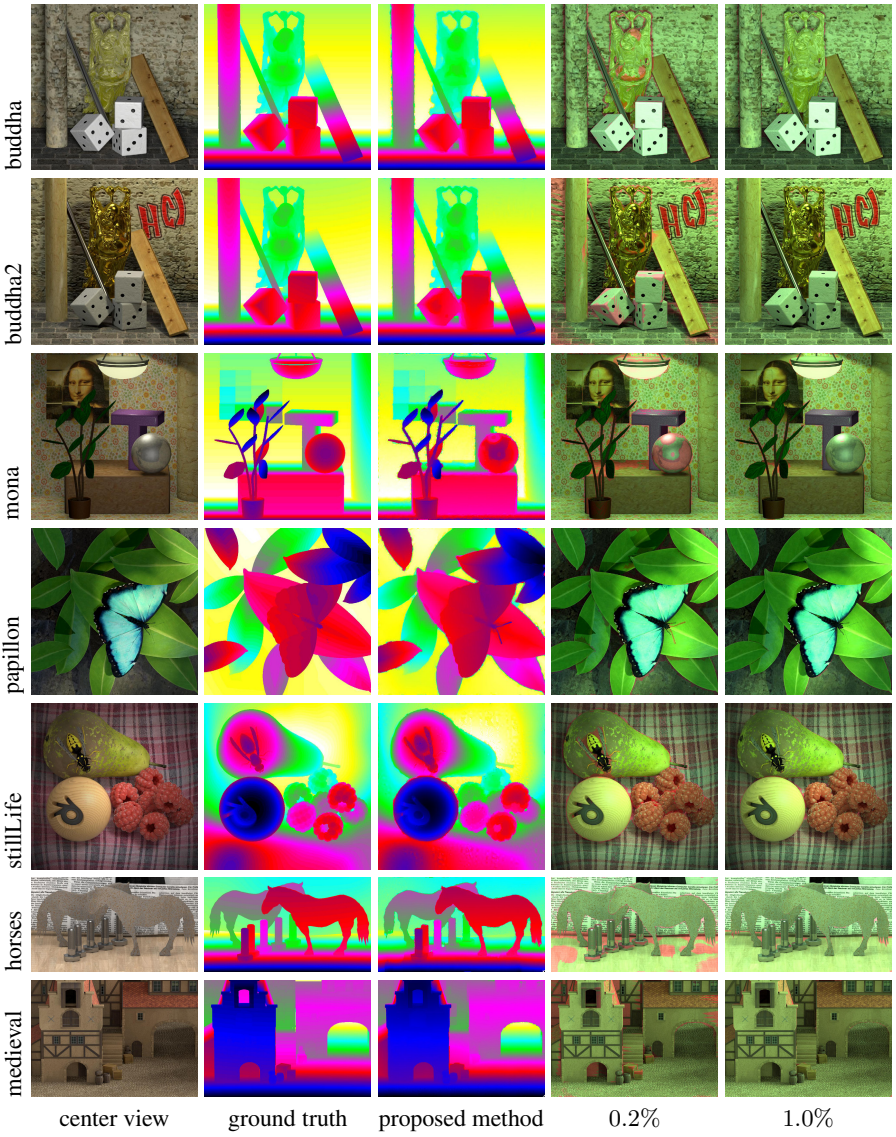


Fig. 5. Qualitative results for synthetic scenes. All scenes have a directional resolution of 9×9 pixels per micro-lens, and varying spatial resolutions, which are listed in Table 1. The figure shows from left to right, the center view, the color coded ground truth depth map, the color coded depth map results for the proposed method, and maps which indicate in green (red) the pixels with a relative depth error of less (more) than 0.2% and 1.0%, respectively.

Table 1. Quantitative results for the scenes shown in Figure 5. The table shows the percentage of pixels with a relative depth error of more than 0.2%, 0.5%, and 1.0%, for different synthetic scenes.

	clean			**vignetting**			**vignetting & noise**			spatial
	1.0%	0.5%	0.2%	1.0%	0.5%	0.2%	1.0%	0.5%	0.2%	resolution
buddha	2.12	3.91	8.37	2.23	4.41	10.38	2.42	4.97	15.28	768×768
buddha2	1.15	2.44	15.05	1.69	4.40	21.81	1.53	3.86	23.33	768×768
mona	2.08	4.66	12.90	2.44	6.47	19.54	2.82	9.46	22.11	768×768
papillon	2.42	3.82	8.79	2.37	4.17	12.97	3.08	6.26	19.12	768×768
stillLife	0.96	2.32	6.33	1.00	2.17	6.89	0.94	2.17	6.02	768×768
horses	1.98	4.47	16.83	2.56	6.98	21.37	3.71	11.05	27.86	1024×576
medieval	1.53	2.93	11.09	1.70	3.50	18.78	1.69	4.54	19.74	1024×720
average	1.75	3.51	11.34	2.00	4.59	15.96	2.31	6.04	19.07	

Table 2. Quantitative results of methods proposed by Wanner and Goldluecke [23] evaluated on an older version of the LFBD. The results are taken from [23], and provided here as a reference point to the results shown in Table 1. The scenes are quite similar to the ones shown in Figure 5 (especially the buddha* scene is nearly identical to the buddha2 scene of the current LFBD). The table shows the percentage of pixels with a relative depth error of more than 0.2% and 1.0%, for the different methods proposed in [23].

	Local [23]		**Global [23]**		**Consistent [23]**		directional	spatial
	1.0%	0.2%	1.0%	0.2%	1.0%	0.2%	resolution	resolution
conehead*	22.9	78.5	1.3	51.0	1.1	48.9	21×21	500×500
mona*	57.0	91.9	25.7	87.7	19.9	84.5	41×41	512×512
buddha*	20.4	73.6	4.1	61.7	2.9	60.4	21×21	768×768
average	33.4	81.3	10.4	66.8	8.0	64.6		

depth labeling approach presented in [23] (cf Table 2). These results were generated using an older version of the LFBD, which is unfortunately no longer available. However, certain scenes like *e.g.* the buddha* scene is nearly identical to the buddha2 scene in the current LFBD, which allows to draw a comparison. By doing so, we see that the proposed method outperforms the method in [23] in terms of accuracy by a large margin. Moreover, depending on the spatial resolution of the input images, the proposed approach takes about 5-10 seconds to compute, whereas the global approach and the consistent approach proposed in [23] take 2-10 minutes and several hours, respectively.

4.2 Real World Experiments

In this section we present some qualitative real world results obtained by using the proposed shape from light field method. For light field capturing we use a Lytro camera. Such a camera provides a spatial resolution of around 380×330 micro-lenses and a

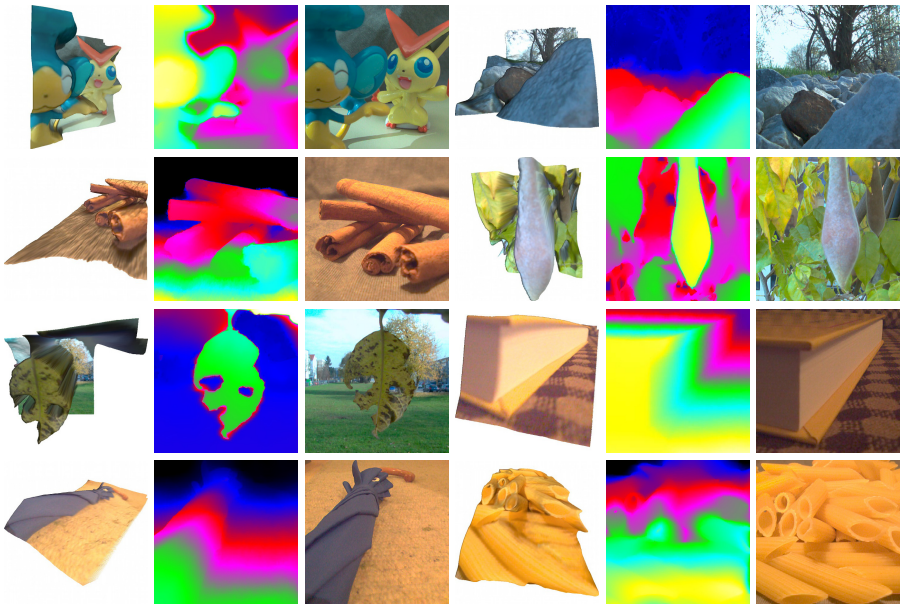


Fig. 6. Qualitative results, which demonstrate the effectiveness of the proposed model. The figure shows depth map results in terms of 3D views and color-coded disparity maps as well as all-in-focus results for various scenes.

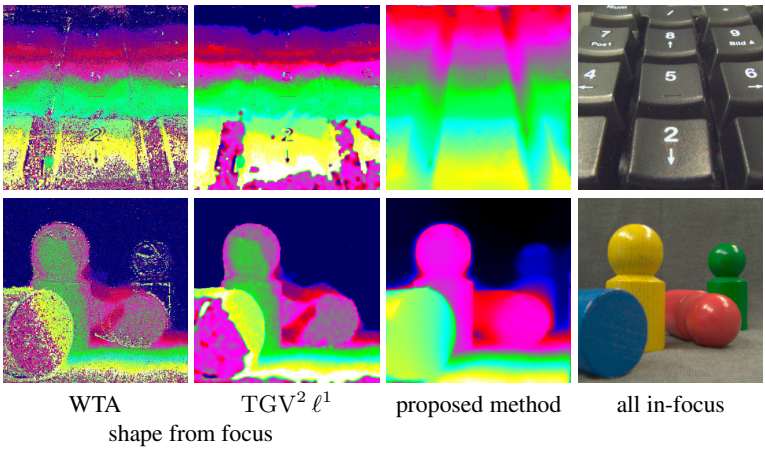


Fig. 7. Comparison to shape-from-focus. The figure shows shape from focus results in terms of the *winner-takes-all* (WTA) solution and a TGV^2 regularized version, where 12 digital refocused images (provided by the commercial Lytro software) were used as input. Furthermore, the figure shows the calculated depth maps of the proposed method, and corresponding all in-focus images.

directional resolution of about 10×10 pixels per micro-lens. For the real world experiments we set $M = 1$ and $N = 16$, and we again tune the other parameters. Figure 6 provides results of the proposed method for different scenes. Among others, Figure 6 shows 3D views created using the calculated depth maps, as well as color-coded disparity maps. Although the spatial resolution provided by the Lytro camera is quite low and the extracted sub-aperture images are quite noisy, the proposed method is again able to calculate piecewise smooth depth maps, with clearly visible depth discontinuities.

It should also be mentioned, that by using the calculated image rotation u, one can easily generate an all-in-focus image by summing up corresponding image locations in the sub-aperture images. All-in-focus results are shown in the third and sixth column of Figure 6.

In a final experiment we compare our results to the commercial Lytro software. Here we compare the proposed approach with a shape-from-focus approach [17]. For the shape-from-focus approach we use 12 digital refocused images, provided by the Lytro software, where we apply a high-pass filter on the images to measure the in-focus area in each image. The qualitative results in terms of the *winner-takes-all* (WTA) solution and a TGV^2 regularized versions are shown in Figure 7. It can be seen, that the results generated with the proposed approach are clearly superior.

5 Conclusion

In this work we proposed a method for calculating a high quality depth map out of a given light field, which was captured with a plenoptic camera. We first showed that it is possible to extract sub-aperture images out of the raw light field data, which are similar to those images captured with a rotating off-axis aperture in the AWS setting. Based on this observation we formulated a variational model which measures a dense field of scene point's image rotation radii over certain sub-aperture images, where these rotation radii encode depth information of the scene points in the image.

In the experiment section we showed on synthetic and real world examples that our model is capable of generating high-quality depth maps. We simulated vignetting effects and image noise on synthetic data. Moreover, we also showed that our model is robust enough to generate piecewise smooth depth maps with sharp depth discontinuities, out of the noisy and highly aliased sub-aperture images extracted form the raw light field data, which was captured with a Lytro camera.

References

1. Adelson, E.H., Bergen, J.R.: The plenoptic function and the elements of early vision. In: Computational Models of Visual Processing, pp. 3–20. MIT Press (1991)
2. Adelson, E.H., Wang, J.Y.A.: Single lens stereo with a plenoptic camera. IEEE Transactions on Pattern Analysis and Machine Intelligence 14(2), 99–106 (1992)
3. Bishop, T., Favaro, P.: Plenoptic depth estimation from multiple aliased views. In: 12th International Conference on Computer Vision Workshops (ICCV Workshops), pp. 1622–1629. IEEE (2009)

4. Bishop, T.E., Favaro, P.: The light field camera: Extended depth of field, aliasing, and superresolution. IEEE Transactions on Pattern Analysis and Machine Intelligence 34(5), 972–986 (2012)
5. Bredies, K., Kunisch, K., Pock, T.: Total generalized variation. SIAM Journal on Imaging Sciences 3(3), 492–526 (2010)
6. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3024, pp. 25–36. Springer, Heidelberg (2004)
7. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision 40, 120–145 (2011)
8. Coffey, D.F.W.: Apparatus for making a composite stereograph (December 1936)
9. Dudnikov, Y.A.: Autostereoscopy and integral photography. Optical Technology 37(3), 422–426 (1970)
10. Fife, K., Gamal, A.E., Philip Wong, H.S.: A 3mpixel multi-aperture image sensor with 0.7μm pixels in 0.11μm cmos (February 2008)
11. Frigerio, F.: 3-dimensional Surface Imaging Using Active Wavefront Sampling. PhD thesis, Massachusetts Institute of Technology (2006)
12. Gortler, S.J., Grzeszczuk, R., Szeliski, R., Cohen, M.F.: The lumigraph. In: SIGGRAPH, pp. 43–54 (1996)
13. Isaksen, A., McMillan, L., Gortler, S.J.: Dynamically reparameterized light fields. In: SIGGRAPH, pp. 297–306 (2000)
14. Levoy, M., Hanrahan, P.: Light field rendering. In: SIGGRAPH, pp. 31–42 (1996)
15. Lippmann, R.: La photographie intégrale. Comptes-Rendus, Académie des Sciences 146, 446–551 (1908)
16. Lumsdaine, A., Georgiev, T.: The focused plenoptic camera. In: Proc. IEEE ICCP, pp. 1–8 (2009)
17. Nayar, S., Nakagawa, Y.: Shape from Focus. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(8), 824–831 (1994)
18. Ng, R.: Digital Light Field Photography. Phd thesis, Stanford University (2006)
19. Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., Hanrahan, P.: Light field photography with a hand-held plenoptic camera. Technical report, Stanford University (2005)
20. Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In: International Conference on Computer Vision (ICCV), pp. 1762–1769. IEEE (2011)
21. Ranftl, R., Gehrig, S., Pock, T., Bischof, H.: Pushing the limits of stereo using variational stereo estimation. In: Intelligent Vehicles Symposium, pp. 401–407. IEEE (2012)
22. Vaish, V., Wilburn, B., Joshi, N., Levoy, M.: Using plane + parallax for calibrating dense camera arrays. In: Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2–9 (2004)
23. Wanner, S., Goldluecke, B.: Globally consistent depth labeling of 4D lightfields. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2012)
24. Wanner, S., Goldluecke, B.: Spatial and angular variational super-resolution of 4D light fields. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 608–621. Springer, Heidelberg (2012)
25. Wedel, A., Pock, T., Zach, C., Bischof, H., Cremers, D.: An Improved Algorithm for TV-L^1 Optical Flow. In: Cremers, D., Rosenhahn, B., Yuille, A.L., Schmidt, F.R. (eds.) Visual Motion Analysis. LNCS, vol. 5604, pp. 23–45. Springer, Heidelberg (2009)

Simultaneous Fusion Moves for 3D-Label Stereo*

Johannes Ulén and Carl Olsson

Centre for Mathematical Sciences
Lund University, Sweden
{ulen,calle}@maths.lth.se

Abstract. Second derivative regularization methods for dense stereo matching is a topic of intense research. Some of the most successful recent methods employ so called binary fusion moves where the combination of two proposal solutions is computed. In many cases the fusion move can be solved optimally, but the approach is limited to fusing pairs of proposals in each move. For multiple proposals iterative binary fusion may potentially lead to local minima.

In this paper we demonstrate how to simultaneously fuse more than two proposals at the same time for a 2nd order stereo regularizer. The optimization is made possible by effectively computing a generalized distance transform. This allows for computation of messages in linear time in the number of proposals. In addition the approach provides a lower bound on the globally optimal solution of the multi-fusion problem. We verify experimentally that the lower bound is very close to the computed solution, thus providing a near optimal solution.

1 Introduction

Dense stereo matching is a classical Computer Vision problem. The problem is in most cases ill posed due to ambiguous texture and therefore regularization terms have to be incorporated. The most popular ones are the first order regularizers [2,3,6]. These typically penalize assignments where neighboring pixels have different disparity. Their popularity is in large part due to the fact that they often result in submodular formulations when applying move-making algorithms such as α-expansion [2] or fusion moves [8]. On the downside this type of regularization favors fronto-parallel planes since surfaces with a non-zero disparity derivative are penalized.

To address this problem [1] used 3D-labels corresponding to arbitrary 3D-planes. The interaction between planes is a Potts model and therefore this approach is suitable for piecewise planar scenes. Li and Zucker [9] use pairwise interactions between 3D labels (encoding disparity and disparity derivatives) to penalize non-smooth surfaces. In contrast Woodford et.al. [17] use regular disparity labels to encode second derivative priors. The optimization problem is however made substantially more difficult due to the introduction of non-submodular triple-interactions.

In [10] it is shown how to encode second derivative priors using labels similar to those used by Li and Zucker [9]. In addition it is shown both theoretically and experimentally that resulting optimization problems are often nearly submodular making

* This work has been funded by the Swedish Research Council (grant 2012-4213), the Swedish Foundation for Strategic Research (SSF) through the program *Future Research Leaders*, the *ELLIIT* program and by the European Research Council (GlobalVision grant no. 209480).

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 80–93, 2013.
© Springer-Verlag Berlin Heidelberg 2013

them easier to solve than the corresponding triple clique versions. On the downside the label space is much bigger than that of [17].

In this paper we show that it is possible to simultaneously fuse more than two proposals solutions for the formulation of [10] using message-passing-based methods such as TRW-S [5]. We will refer to this formulation as *3D-label stereo*. We show that it is possible to compute generalized distance transforms in linear time using methods similar to those of [3] despite the fact that our labels are 3-dimensional. This opens up the possibility of extending the fusion approach of [8] by fusing more than two models at a time. We show that this approach results in solutions with reduced energy compared to the standard fusion approach.

For standard first order stereo regularization, where each label represent a disparity, fusing multiple models at the same time have already been explored in [15]. However their method requires the regularizer to be convex in the label index, which is not true for 3D-label stereo.

1.1 Background: Fusion Moves for Stereo

Consider an arbitrary second order pseudo-boolean function of n variables, usually expressed as,

$$E(\boldsymbol{x}) = \sum_p U_p(x_p) + \sum_p \sum_{q \in \mathcal{N}(p)} V_{pq}(x_p, x_q), \tag{1}$$

here $\boldsymbol{x} \in \boldsymbol{L}^n$, where $\boldsymbol{L}$ is some discrete set and $\mathcal{N}$ is some connectivity. In vision E is usually called the energy function and our goal is to find a minimizer of it. Lempitsky et al. [8] proposed to a minimization method that iteratively reduces the energy by fusing candidate solutions. Two assignments $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ are *fused* together into *one* solution with lower energy by solving

$$\min_{\boldsymbol{z} \in \{0,1\}^n} E\big(\boldsymbol{z} \cdot \boldsymbol{x}_0 + (\mathbf{1} - \boldsymbol{z}) \cdot \boldsymbol{x}_1\big), \tag{2}$$

where $\cdot$ is element-wise multiplication. We refer to solving (2) as performing a binary fusion move, and a possible solution to (1) as a *proposal*. It is possible to optimize a given energy function by generating a lot of proposals and then iteratively perform binary fusion.

For the problem of stereo matching it was shown in [10] how to formulate a second derivative surface prior in the form of (1). Let $\mathcal{I}$ be the image and $\mathcal{D} : \mathcal{I} \to \mathbb{R}$ be the disparity function (disparities assigned to every pixel of $\mathcal{I}$). To each pixel p we assign 3 dimensional labels $\boldsymbol{x}_p$ consisting of the disparity $\mathcal{D}(p)$ and the gradient of the disparity $\nabla\mathcal{D}(p)$. Let the function $\mathcal{T}_p\mathcal{D} : \mathcal{I} \to \mathbb{R}$ be the tangent at the point p seen as a function of the whole image, that is

$$\mathcal{T}_p\mathcal{D}(x) = \mathcal{D}(p) + \nabla\mathcal{D}(p)^T (x - p). \tag{3}$$

If τ is a threshold and we use

$$V_{pq} = \min(|\mathcal{T}_p\mathcal{D}(q) - \mathcal{D}(q)|, \tau) = \min(|\mathcal{D}(p) + \nabla\mathcal{D}(p)^T (q - p) - \mathcal{D}(q)|, \tau), \tag{4}$$

then according to the Taylor expansion we get

$$V_{pq} \approx \min(|\frac{1}{2}(q-p)^T \nabla^2 \mathcal{D}(p)(q-p)|, \tau). \tag{5}$$

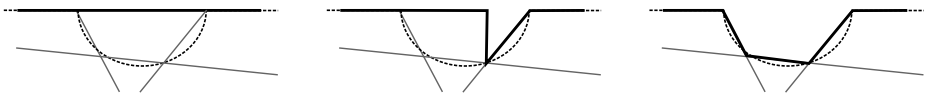
That is, the penalty V_{pq} measures the truncated 2nd derivative in the direction $q-p$. Note that the energy in (1) is symmetric since there is also a term V_{qp}. In contrast, Woodford et.al. [17] only assign a disparity to each pixel, and therefore require triple cliques to estimate the 2nd derivative. It is shown in [10] that the 3D-label formulation gives simpler (often submodular) interactions and therefore each fusion move becomes easier to solve.

Binary fusion moves are limited in that they can only fuse two proposals at a time. Therefore in case of many proposals the end result may depend on the order the proposals are fused. The focus of this paper is to go beyond binary fusion moves and simultaneously fuse hundreds of proposals by solving,

$$\min_{\boldsymbol{z} \in \{0,\dots,k\}^n} E\left(\sum_{i=1}^{n} (\boldsymbol{I}_{\boldsymbol{z}=i} \cdot \boldsymbol{x}_i)\right) \tag{6}$$

where $\boldsymbol{I}_{\boldsymbol{z}=i}$ is the indicator function for proposal i. We will refer to solving (6) as performing *simultaneous* fusion. We will show that this is possible using message-passing methods since generalized distance functions can be efficiently computed for our energy.

Why? One natural question to ask is: Is it not sufficient to consider binary fusion? One reason for simultaneous fusion is that we are able to avoid local optima, as illustrated in Figure 1. Another advantage is that we can get guarantees on our solution. That is, given n proposals we know how far away from the optimal fusion of the n proposals our solution is. If we perform binary fusion iteratively we can only get guarantees on each fusion step but we do not know if the order of the fusion has led us into a local optima.



(a) Start solution (b) One plane (c) Three planes

Fig. 1. Motivational example for simultaneously fusion. a) Suppose a real surface is represented by the dashed curve, our current best solution is given by the thick black line and three proposals are shown as the three gray lines. b) A possible solution after binary fusion with one of the proposals, note that this solution incurs a large regularization cost. Because of the cost this proposal might never be successfully fused. c) If we fuse all three planes simultaneously we can jump directly to a better solution.

1.2 Optimization

Belief propagation (BP) minimizes functions such as (1) by sending *messages* between the nodes of the underlying graph. The message from node p to q contains information about the optimal assignment of p given the current assignments in p's neighborhood. If the neighborhood structure forms a tree then BP is guaranteed to give the optimal solution after one pass of messages. If the graph contains cycles the algorithm is no longer guaranteed to even converge.

The property of optimality on trees is exploited in Tree Reweighted Message Passing (TRW) methods [16]. Given a general graph with cycles TRW decomposes the graph into trees and optimize these separately. If all sub-problems provide the same solution then this is also the optimal solution to the original graph. Otherwise the sub-problems provide a lower bound on the original problem. By changing the decomposition TRW aims at maximizing this lower bound. In [5] a variant of TRW, called TRW-S, which is guaranteed to converge is presented. The decomposition idea was further expanded in [7] where it is shown that modifying the TRW approach slightly results in Lagrangian dual problem, which can be efficiently solved using projected subgradients, with better theoretical properties than TRW-S.

The core of all these methods is efficient message passing. Each message is a vector of the same dimension as the number of possible labels. Let m_{pq}^t denote the message that node p sends to node q at iteration t. The message is generally defined by

$$m_{pq}^t(x_q) = \min_{x_p} \big(V_{pq}(x_p, x_q) + h(x_p)\big), \tag{7}$$

where

$$h(x_p) = U_p(x_p) + \sum_{s \in \mathcal{N}(p) \setminus q} m_{sp}^{t-1}(x_p). \tag{8}$$

The messages updates for TRW-S are similar to this. Straight forward computation of messages using the formula (7) takes $\mathcal{O}\big(n^2\big)$ evaluations, where n is the number of labels. However for special types of pairwise interactions (7) can be evaluated more efficiently using the generalized distance transform [3]. For example, in the case of an ℓ_1 penalty

$$V_{pq}(x_p, x_q) = |x_p - x_q|, \tag{9}$$

the function $m_{pq}^t(x)$ is

$$m_{pq}^t(x) = \min_{x_p} \left(|x_p - x| + h(x_p)\right), \tag{10}$$

which can be seen as the lower envelope of a set of cones, see Figure 3. It was demonstrated in [3] that computation of the lower envelope can be done in linear time, thus reducing the overall time required to compute messages to $\mathcal{O}(n)$. In Section 2 we show how to compute messages in $\mathcal{O}(n)$ for 3D-label stereo.

2 Simultaneous Fusion

Earlier approaches to minimize (1), in the context of 2nd order stereo regularizers, have consisted of iteratively fusing one proposal after another in different order until the energy have stopped decreasing [10,17]. That is, solving (2) over and over. We call this approach *iterative binary fusion*. Binary fusion guarantees to generate a sequence of solutions with decreasing energy and under the assumption of submodularity each fusion move is solved optimally. In the case of multiple proposals there are however little or no guarantees about the quality of the end result.

In this paper we propose to fuse all proposals at the same time, that is solving (6). We call this *simultaneous fusion*. In order to to minimize (6) we use TRW-S [5]. Such an approach will, in contrast to iterative binary fusion, provide both a good solution and a lower bound on the best energy possible to obtain with the available proposals. In the experiments section we show that the solutions given by simultaneous fusion are often very close to the lower bound indicating that the solutions are near optimal.

To minimize (6) we need to compute update messages. If the number of possible labels are small solving (7) in each iteration is no problem. Using a brute force, all messages can be found in $\mathcal{O}(n^2)$ where n is the number of proposals. As the number proposals increase this becomes intractable. To reduce this complexity we will evaluate (7) using generalized distance transforms. In principle our labels belong to a 3D-dimensional space and computing distance transforms in this space may seem difficult. However due to the fact that the interaction is defined using distances along the viewing ray we can apply regular one dimensional distance transforms with little modification. The only extra step required is sorting the labels according to their disparity as the disparity of a given label can vary between neighbors.

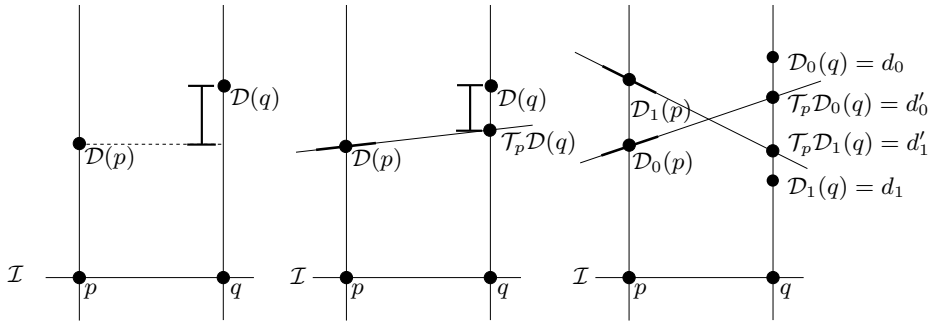
2.1 Efficient Distance Transform

In this section we will cover efficient distance transform for ℓ_1 and ℓ_2 regularization of 3D-label stereo [10]. If pixel p is assigned the tangent of function $\mathcal{D}_i : \mathcal{I} \to \mathbb{R}$ at p and pixel q is assigned the tangent of $\mathcal{D}_j : \mathcal{I} \to \mathbb{R}$ at q, then we formulate the ℓ_1 regularization interaction (from p to q) as

$$V_{pq}(\mathcal{D}_i, \mathcal{D}_j) = \alpha |\mathcal{T}_p \mathcal{D}_i(q) - \mathcal{D}_j(q)|, \tag{11}$$

where α is a regularization weight. For simplicity we postpone the introduction of the threshold term τ to a later stage. The weight α can vary between pixel pairs (in [17] α depends on an underlying segmentation of the image). For ease of notation we only denote the coefficient α. The interaction measures the difference between the disparity assignment at q and the tangent function $\mathcal{T}_p \mathcal{D}_i$ evaluated at q, see Figure 2 (b). Note that there is a similar term V_{qp} (from q to p) and therefore the interaction is symmetric. In contrast, a standard first order method directly measures the distance between the two neighboring disparity assignments $\mathcal{D}_i(p)$ and $\mathcal{D}_j(q)$, see Figure 2 (a). Similarly the ℓ_2 cost is defined as

$$V_{pq}(\mathcal{D}_i, \mathcal{D}_j) = \alpha \left(\mathcal{T}_p \mathcal{D}_i(q) - \mathcal{D}_j(q)\right)^2. \tag{12}$$



(a) Disparity (b) 3D-label stereo (c) Simultaneous fusion

Fig. 2. The difference between standard and 3D-label stereo for rectified cameras. The image plane is denoted by $\mathcal{I}$ and the viewing ray of pixel p and q are marked. a) Standard stereo, the marked distance is $|\mathcal{D}(p) - \mathcal{D}(q)|$ which is used for regularization. b) 3D-label stereo, the tangent plane is indicated by a slanted line. The indicated distance is $|\mathcal{T}_p\mathcal{D} - \mathcal{D}(q)|$ which is used for regularization. c) Taking it one step further given two disparity proposal functions $\mathcal{D}_0$ and $\mathcal{D}_1$ we see that all points needed to calculate the regularization costs are located along $q's$ viewing ray.

Our messages will be of the form

$$m^t_{pq}(\mathcal{D}_j) = \min_{i=0,\dots,n} \big(V_{pq}(\mathcal{D}_i, \mathcal{D}_j) + h(\mathcal{D}_i)\big), \tag{13}$$

where

$$h(\mathcal{D}_i) = U_p(\mathcal{D}_i) + \sum_{s \in \mathcal{N}(p) \setminus q} m^{t-1}_{sp}(\mathcal{D}_i). \tag{14}$$

Here U_p is a data term that can depend on the tangent plane at the pixel p. In general such a functional would require using distance transforms in $\mathbb{R}^3$ (since our labels are 3 dimensional). However, due to the particular form of our pairwise interaction it is enough to consider 1-dimensional distance transforms on the viewing rays of the pixels.

To make further discussion easier we introduce some notation. Suppose that we would like to fuse $n+1$ proposals $\mathcal{D}_0, ..., \mathcal{D}_n$. Each proposal $\mathcal{D}_i$ and tangent $\mathcal{T}_p\mathcal{D}_i$ gives a disparity value at q (the intersection with the viewing ray), see Figure 2 (c). We denote these points

$$\boldsymbol{Q} = \{\mathcal{D}_0(q), \dots, \mathcal{D}_n(q)\} = \{d_0, \dots, d_n\}, \tag{15}$$

$$\boldsymbol{Q}' = \{\mathcal{T}_p\mathcal{D}_0(q), \dots, \mathcal{T}_p\mathcal{D}_n(q)\} = \{d'_0, \dots, d'_n\}. \tag{16}$$

Furthermore we let $h_i = h(\mathcal{D}_i)$. To compute the message m^t_{pq}, in case of our ℓ_1 interaction, we need to evaluate the function

$$f(d) = \min_{i=0,\dots,n} \alpha |d'_i - d| + h_i \tag{17}$$

for the disparities $d \in \boldsymbol{Q}$. If the labels are sorted according to disparity this can be done in linear time (as we will outline below). The order at which the points in $\boldsymbol{Q}'$ appear

along the viewing ray at q depends on the tilt of the tangent functions, see Figure 2 (c). Therefore the same set of tangents at p will have different orderings for different neighbors q of p. To be able to compute the distance transform and evaluate $f(d)$ in linear time we thus need to maintain an ordering of the labels for each neighbor. Since labels are fixed the sorting can be done once at start up and maintained during the optimization, to avoid the $\mathcal{O}(n \log n)$ sorting complexity. When the order at q is available we can apply the algorithm from [3] with some minor modifications.

2.2 Algorithm

In [3] a distance transform is introduced that can handle standard stereo regularization. For 3D-label stereo we propose slight modification to the distance transform in [3] which will handle both the ℓ_1 and ℓ_2 regularization. For ℓ_1 regularization each disparity label at p can be though of as representing a cone rooted at (d'_i, h_i) with slope $\alpha|d'_i - d|$. If we construct the lower envelope of all these cones then we can solve (7) efficiently, since $m^t_{pq}(\mathcal{D}_j)$ is the height of the lower envelope at disparity d_j.

ℓ_1 Regularization. The distance transform sequentially constructs the lower envelope of the cones by first sorting $\boldsymbol{Q}'$ and then going through each point $d' \in \boldsymbol{Q}'$. A important observation is that $\boldsymbol{Q}'$ needs only to be sorted once before the message passing starts since the labels do not change during the optimization. In what follows let i be the index of point d'_i and let j be the last point added to the the lower envelope. For each point d'_i drawn from $\boldsymbol{Q}'$ one of the following three cases may occur

1. h_i is above the current lower envelope at d'_i.
2. $h_i + \alpha|d'_i - d'_j|$ is below the current lower envelope at d'_i.
3. Neither 1 nor 2.

For case 1) we can just skip point i as it will never be part of the lower envelope. For case 2) the previously added interval to the lower envelope is removed and we repeat the comparison for interval $j - 1$. In case 3) the lower envelope intersects the cone associated with point i. For ℓ_1 regularization two cones intersect when

$$d = \frac{h_i - h_j + \alpha(d'_i + d'_j)}{2\alpha}. \tag{18}$$

The intersection is calculated and the new interval is added to the lower envelope. An example lower envelope is constructed in Figure 3.

After the lower envelope is calculated the message cost for $m^t_{pq}(\mathcal{D}_j)$ is just the height of the lower envelope at position d_j. When using standard stereo regularization the set of disparities are usually the same for all pixels. In our case $\boldsymbol{Q}$ and $\boldsymbol{Q}'$ most likely differ. However the evaluation of the message values for all the disparities in $\boldsymbol{Q}$ is easily seen to be linear since both the linear interval of the lower envelope and the set $\boldsymbol{Q}$ are ordered according to disparity.

A final thing to consider is when we add truncation to the regularization term such as

$$V_{pq}(\mathcal{D}_i, \mathcal{D}_j) = \alpha \min\left(|\mathcal{T}_p\mathcal{D}_i(q) - \mathcal{D}_j(q)|, \tau\right), \tag{19}$$

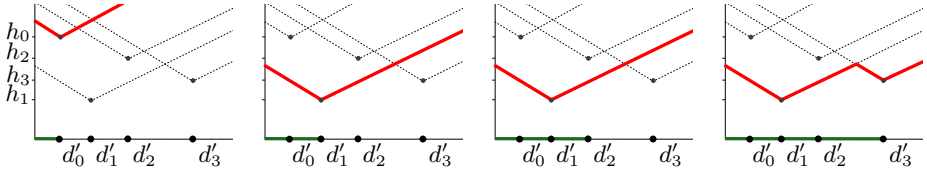


Fig. 3. Example construction of lower envelope for ℓ_1 regularization assuming the points are ordered in ascending order. The dashed lines indicate the cost $\alpha|d'_i - d| + h_i$. The current estimated lower envelope is given by the thick red line and the progress is indicated by the green line. a) The algorithms is initialized. b) $\alpha|d'_1 - d'_0| + h_1 < h_0$. The cone belonging to d'_0 will never be part of the lower envelope. c) $\alpha|d'_2 - d'_1| + h_1 < h_2$. The cone belonging to d'_2 will never be part of the lower envelope. Finally in (d) the current estimate of the lower bound is intersecting with d'_3's cone. The lower envelope is divided into two intervals. One belonging to d'_1 and one to d'_3.

where τ is some truncation level. The highest cost any message can have is

$$\min_i h_i + \alpha\tau, \tag{20}$$

and can be calculated in $\mathcal{O}(n)$. The truncation is then added to the message by taking the minimum of the non-truncated message value $m^t_{pq}(\mathcal{D}_i)$ and the largest message cost. Pseudocode for the algorithm is given in Algorithm 1.

ℓ_2 Regularization. The only modification needed is related to the construction of the lower envelope, which is described in [3]. The rest is carried over from the ℓ_1 regularization algorithm. For ℓ_2 regularization the lower envelope consists of parabolas of the form $\alpha(d'_i - d)^2 + h_i$. The lower envelope is sequentially constructed by considering intersection of the parabolas, which for parabolas i, j is given by

$$d = \frac{\left(h_i + \alpha(d'_i)^2\right) - \left(h_j + \alpha(d'_j)^2\right)}{2\alpha\left(d'_i - d'_j\right)}. \tag{21}$$

Let j be the last added point to the lower envelope and let z_j be the first point in the interval where it is part of the lower envelope. For each point i drawn from $\boldsymbol{Q}'$ and as long as $d'_i \neq d'_j$, consider the intersection of parabolas i and j, Denote the intersection d, one of the following two cases occur

1. $d > z_j$. The intersection is added to the lower envelope.
2. $d \leq z_j$. Repeat intersection with parabola $j - 1$.

When $d'_i = d'_j$, the parabolas do not intersect and the parabola with smallest h_i is kept in the lower envelope. This case is not covered in [3] and correspond to several proposals intersecting the q-ray at the same point. This occurs frequently for 3D-label stereo.

Complexity for ℓ_1 and ℓ_2. Let n be the number of labels. The sorting of points is done only once and outside of the message update. For the lower envelope, the maximum number of intervals is $n + 1$. Each interval is visited at most 3 times. Once when it

is added, once when it is intersected and once when it is removed. All other parts of the algorithm are at most single nested loops of size n, hence the complexity for each message update is $\mathcal{O}(n)$.

Algorithm 1. Message update for ℓ_1 regularization.

function DISTANCE(x, y)
  return $\alpha|x - y|$

function UPDATE_INTERVALS(interval, start, intersection)
  v[interval] $\leftarrow$ start ▷ Label belonging to this interval.
  z[interval] $\leftarrow$ intersection ▷ Lower limit of this interval.
  z[interval + 1] $\leftarrow \infty$ ▷ Upper limit of this interval.

Precondition: $\boldsymbol{Q}$ and $\boldsymbol{Q}'$ are sorted.
function MESSAGE_UPDATE($\boldsymbol{Q}, \boldsymbol{Q}', \alpha, \tau$)
  UPDATE_INTERVALS$(0, 0, -\infty)$
  `maxVal` $\leftarrow \infty$
  for $i \leftarrow 1$ **to** $n - 1$ **do**
    `maxVal` $\leftarrow \min($`maxVal`$, h_i)$
    $j \leftarrow 0$ ▷ Number of intervals in the lower envelope.
    for $l = i$ **to** 0 **do**
      if DISTANCE $\left(d'_i, d'_j\right) + h_{v[j]} \leq h_i$ **then** ▷ Case 1
        break
      else if DISTANCE $\left(d'_i, d'_j\right) + h_i < h_{v[j]}$ **then** ▷ Case 2
        if j = 0 **then**
          UPDATE_INTERVALS$(0, i, -\infty)$
        else
          $j \leftarrow j - 1$
      else ▷ Case 3
        $j \leftarrow j + 1$
        intersection $\leftarrow \left(h_i - h_j + \alpha(d'_i + d'_j)\right)/2\alpha$
        UPDATE_INTERVALS $(j, i,$ intersection$)$
        break
  $j \leftarrow 0$
  `maxVal` $\leftarrow$ `maxVal` $+ \alpha\tau$. ▷ Largest message cost.
  for $i \leftarrow 1$ **to** $n - 1$ **do**
    while $z[j + 1] < d_i$ **do**
      $j \leftarrow j + 1$
    $\boldsymbol{m}[i] \leftarrow \min\left(\text{DISTANCE}\left(d_i, d'_j\right) + h_{v[j]},$ `maxVal`$\right)$
  return $\boldsymbol{m}$

3 Experiments

For all experiments we adopt the energy function used in [17] with the appropriate modifications described in [10]. The occlusion term is removed for simplicity. The energy is defined as $E(\mathcal{D}) = E_{\text{photo}}(\mathcal{D}) + E_{\text{smooth}}(\mathcal{D})$. E_{photo} is defined as:

$$E_{\text{photo}}(\mathcal{D}) = -\log\left(1 + \exp(-\|\mathcal{D}(x) - I_0(x)\|^2/90)\right), \tag{22}$$

where I_0 is the reference image. We adopt the regularization from [17] just like in [10]:

$$E_{\text{smooth}}(\mathcal{D}) = \sum_p \sum_{q \in \mathcal{N}(p)} W(\{p,q\})\, \tau \min\left(\frac{V_{pq}(\mathcal{D},\mathcal{D})}{\tau}, 1\right)^{\gamma}, \tag{23}$$

where $\gamma = i$ gives the ℓ_i reguralization. Futhermore $W(\{p,q\})$ is a weight depending on a segmentation of the image. If p and q lie in different segments it takes a low value, if they belong the same segment it takes a high value. $\mathcal{N}$ is normal 4-connectivity and the truncation $\tau = 0.02$. See [17] for further details.

We also make use of the *SegPln* proposals introduced in [17]. The SegPln proposals are generated by segmenting the image with 14 different parameter settings. For each segmentation local window matching is used to find a disparity estimate. A piecewise planar function is fit to the estimated disparity using LO-RANSAC.

3.1 Simultaneous Fusion

To quantitatively evaluate simultaneous fusion we use the full Middlebury stereo dataset [4,12,13,14], consisting of 40 stereo image pairs using the previously described energy function with ℓ_1 regularization. For every stereo pair we generate the 14 SegPln proposals and 300 equally distanced fronto parallel plane proposals.

TRW-S will for each iteration produce a lower bound (l) and a current best solution (e). We define the relative duality gap as: $|(e - l)/e|$. The iterative binary fusion is performed in random order until no proposal changes the current disparity estimate.

Each binary fusion is optimized using roof duality [11]. The simultaneous fusion is solved using TRW-S for either 3000 iterations or until the relative duality gap is less than 0.001.

In general simultaneous fusion is expected to be slower than iterative binary fusion since it solves a much harder problem. To make the comparison more equal we spend the time difference fusing random fronto parallel planes to the iterative solution. We call the final solution *extra*.

Quantitative result for ℓ_1 regularization on the Middlebury data set can be seen in Table 1. An example highlighting the quality difference for simultaneous and iterative binary fusion is given in Figure 4.

The difference is larger when the number of proposals is low. When the number of proposals is very large the binary fusion approach performs surprisingly well for many of the data sets. The resulting disparity estimate is almost as good as that of simultaneous fusion which can be verified to be near optimal due to the lower bound being almost tight. For some data sets the lower energy clearly improved the resulting disparity map, Figure 5. For other data sets the improvement was more subtle, Figure 6.

We also perform some experiments using ℓ_2 regularization. See Figures 7 and 8. In all examples we only show the part of the image visible in both views.

Table 1. Complete experiments on the middlebury dataset

Sequence	Iterative (Iter)		Simultaneous (Sim)			Sim / Iter		Sim/extra
	Iterations	Time (m)	Iterations	Time (m)	Rel. gap	Energy	Time	Energy
Aloe	3278	16.4	3000	358.8	0.001	0.994	21.83	0.996
Art	3546	23.1	3000	384.5	0.002	0.992	16.67	0.994
Baby1	2458	13.0	3000	342.3	0.036	1.006	26.25	1.006
Baby2	2862	14.8	3000	342.1	0.016	0.988	23.11	0.989
Baby3	3045	17.3	3000	361.7	0.021	0.976	20.94	0.976
Barn1	4353	22.2	3000	370.2	0.000	0.999	16.66	0.999
Barn2	2897	14.6	3000	366.5	0.001	0.997	25.06	0.997
Books	3683	21.3	3000	383.7	0.014	0.996	18.00	0.998
Bowling1	2126	14.8	3000	345.4	0.009	0.992	23.30	0.993
Bowling2	3028	17.0	3000	356.1	0.002	0.995	20.94	0.996
Bull	3896	19.2	1302	163.8	0.000	0.997	8.52	0.997
Cloth1	3614	17.3	3000	341.0	0.001	0.989	19.76	0.991
Cloth2	3356	18.7	3000	352.4	0.020	0.988	18.87	0.990
Cloth3	2967	14.0	3000	341.3	0.001	0.978	24.36	0.980
Cloth4	3940	20.6	3000	353.9	0.001	0.988	17.20	0.990
Computer	3382	20.1	3000	363.1	0.002	0.992	18.09	0.994
Cones	4180	23.6	3000	373.9	0.007	1.002	15.83	1.002
Dolls	3219	19.4	3000	375.7	0.003	0.993	19.33	0.995
Drumsticks	2844	17.9	3000	379.6	0.014	1.007	21.26	1.009
Dwarves	3388	19.3	3000	369.2	0.026	1.012	19.14	1.014
Flowerpots	3103	16.8	3000	356.3	0.023	0.993	21.21	0.994
Lampshade1	2498	15.9	1425	173.7	0.000	0.976	10.91	0.976
Lampshade2	1935	13.1	3000	352.7	0.011	0.991	26.87	0.992
Laundry	4047	24.1	3000	365.4	0.006	0.995	15.17	0.998
Map	2704	4.5	3000	133.9	0.001	0.999	29.64	0.999
Midd1	3627	36.0	3000	378.4	0.013	0.972	10.52	0.973
Midd2	3082	35.8	3000	371.2	0.002	0.980	10.38	0.981
Moebius	4261	26.7	3000	377.3	0.003	0.990	14.12	0.992
Monopoly	3577	47.3	3000	359.9	0.009	1.004	7.60	1.004
Plastic	2522	20.3	1551	181.6	0.000	0.980	8.93	0.980
Poster	2798	15.1	3000	375.3	0.008	1.003	24.83	1.003
Reindeer	3708	23.3	3000	369.0	0.034	1.009	15.82	1.011
Rocks1	2578	13.6	3000	351.5	0.011	0.989	25.86	0.990
Rocks2	3365	16.1	3000	353.3	0.001	0.965	21.93	0.966
Sawtooth	3302	16.2	3000	369.8	0.003	0.999	22.79	1.000
Teddy	4386	23.8	3000	379.6	0.012	1.006	15.97	1.007
Tsukuba	2562	9.0	3000	246.9	0.001	0.995	27.41	0.996
Venus	2697	14.2	3000	373.3	0.000	0.996	26.21	0.997
Wood1	2555	16.2	914	123.5	0.000	0.989	7.63	0.989
Wood2	2958	21.0	3000	355.8	0.000	1.000	16.98	1.000
Geom. mean	3150.9	18.1	2753.6	322.9	0.002	0.993	17.83	0.994

Fig. 4. Surface view of a toy example on the `Baby2` sequence using ℓ_1 regularization. Fusing only the 14 SegPln proposals.

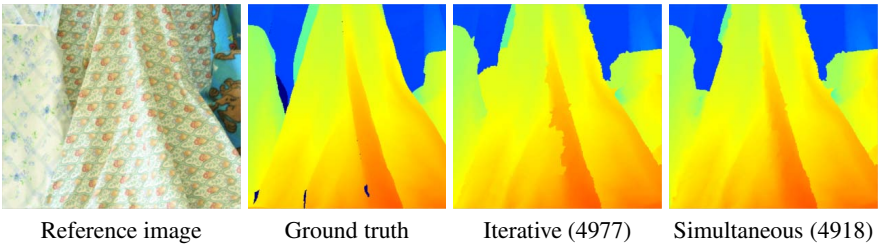


Fig. 5. Disparity map for `Cloth4` sequence using ℓ_1 regularization with (energy)

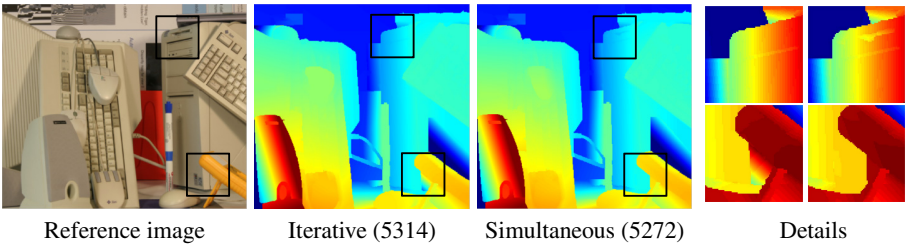


Fig. 6. Disparity map for `Computer` sequence using ℓ_1 regularization with (energy)

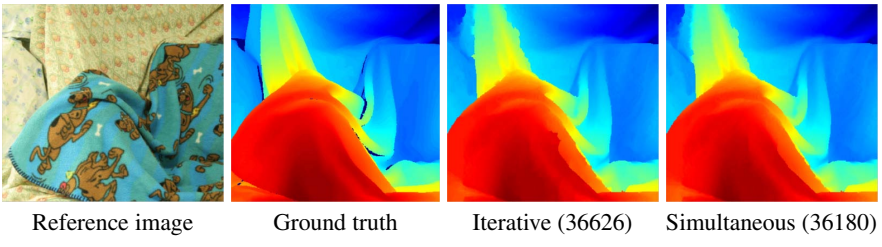


Fig. 7. Disparity map for `Cloth3` sequence using ℓ_2 regularization with (energy)

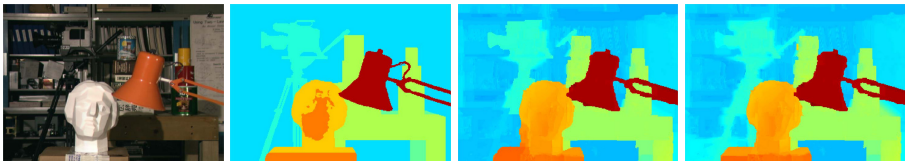
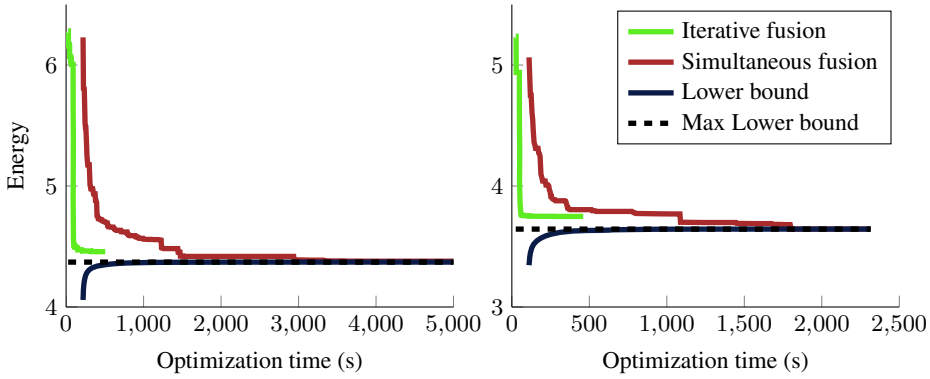


Fig. 8. Disparity map for `Tsukuba` sequence using ℓ_2 regularization with (energy)



(a) `Rocks1` sequence with 200 proposals. (b) `Baby1` sequence with 100 proposals.

Fig. 9. Energy as a function of time (s) using ℓ_1 regularization

3.2 Energy over Time

The performance of iterative and simultaneous fusion is also evaluated by considering energy of the optimization as a function of time, two examples are given in Figure 9. In general iterative fusion is faster to achieve low energies but in the long run simultaneous fusion produces solutions with lower energies.

4 Conclusions

In this paper we show how the 3D-label stereo regularization [9,10] can be optimized using simultaneous fusion, where all proposals are fused at the same time. We identify that the 3D-label pairwise interaction only occur along 1-dimensional viewing rays. The order of the labels along each viewing ray can be precomputed before the optimization starts enabling us to formulate an efficient $\mathcal{O}(n)$ message update algorithm.

As opposed to previously used binary fusion we both get optimality bounds on the full fusion problem and solutions with better quality. As a bonus we show that iterative binary fusion for ℓ_1 regularization achieves solution surprisingly close to global optima.

Source code available at `http://www.maths.lth.se/~ulen/stereo.html`.

References

1. Birchfield, S., Tomasi, C.: Multiway cut for stereo and motion with slanted surfaces. In: Int. Conf. Computer Vision (1999)
2. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Analysis and Machine Intelligence (2001)
3. Felzenszwalb, P., Huttenlocher, D.: Efficient belief propagation for early vision. In: Int. Conf. Computer Vision, vol. 70(1), pp. 41–54 (2006)
4. Hirschmüller, H., Scharstein, D.: Evaluation of cost functions for stereo matching. In: Conf. Computer Vision and Pattern Recognition (2007)
5. Kolmogorov, V.: Convergent tree-reweighted message passing for energy minimization. IEEE Trans. Pattern Analysis and Machine Intelligence 28(10), 1568–1583 (2006)
6. 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)
7. Komodakis, N., Paragios, N., Tziritas, G.: Mrf optimization via dual decomposition: Message-passing revisited. In: Int. Conf. Computer Vision. IEEE (2007)
8. Lempitsky, V.S., Rother, C., Roth, S., Blake, A.: Fusion moves for markov random field optimization. IEEE Trans. Pattern Analysis and Machine Intelligence 32(8), 1392–1405 (2010)
9. Li, G., Zucker, S.: Differential geometric inference in surface stereo. IEEE Trans. Pattern Analysis and Machine Intelligence 32(1), 72–86 (2010)
10. Olsson, C., Ulén, J., Boykov, Y.: In Defense of 3D-Label Stereo. In: Conf. Computer Vision and Pattern Recognition (2013)
11. Rother, C., Kolmogorov, V., Lempitsky, V., Szummer, M.: Optimizing binary mrfs via extended roof duality. In: Conf. Computer Vision and Pattern Recognition (2007)
12. Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: Conf. Computer Vision and Pattern Recognition (2007)
13. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. Journal Computer Vision 47(1), 7–42 (2002)
14. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: Conf. Computer Vision and Pattern Recognition (2003)
15. Veksler, O.: Multi-label moves for MRFs with truncated convex priors. In: Cremers, D., Boykov, Y., Blake, A., Schmidt, F.R. (eds.) EMMCVPR 2009. LNCS, vol. 5681, pp. 1–13. Springer, Heidelberg (2009)
16. Wainwright, M., Jaakkola, T., Willsky, A.: Map estimation via agreement on trees: message-passing and linear programming. IEEE Trans. Information Theory (2005)
17. Woodford, O., Torr, P., Reid, I., Fitzgibbon, A.: Global stereo reconstruction under second-order smoothness priors. IEEE Trans. Pattern Analysis and Machine Intelligence 31(12), 2115–2128 (2009)

Efficient Convex Optimization for Minimal Partition Problems with Volume Constraints

Thomas Möllenhoff, Claudia Nieuwenhuis, Eno Töppe, and Daniel Cremers

Department of Computer Science, Technical University of Munich, Germany

Abstract. Minimal partition problems describe the task of partitioning a domain into a set of meaningful regions. Two important examples are image segmentation and 3D reconstruction. They can both be formulated as energy minimization problems requiring minimum boundary length or surface area of the regions. This common prior often leads to the removal of thin or elongated structures. Volume constraints impose an additional prior which can help preserve such structures. There exist a multitude of algorithms to minimize such convex functionals under convex constraints. We systematically compare the recent Primal Dual (PD) algorithm [1] to the Alternating Direction Method of Multipliers (ADMM) [2] on volume-constrained minimal partition problems. Our experiments indicate that the ADMM approach provides comparable and often better performance.

1 Introduction

Both segmentation and 3D reconstruction approaches aim at partitioning a two or three dimensional domain into a set of 'meaningful' regions. For segmentation such regions usually correspond to the projections of three-dimensional objects in an image, whereas for 3D reconstruction the regions indicate the three dimensional objects themselves, e.g. in a voxel grid. Such minimal partition problems come with regularization assumptions which either promote a minimal boundary length [3–5] or a minimal surface area [6, 7]. These constraints often over-smooth the object boundaries. In segmentation the removal of thin or elongated structures is the consequence. In 3D reconstruction the resulting objects tend to be flat due to the minimal surface assumption. Therefore, constraints restricting the object volume for specific regions are useful as an additional regularization assumption to help preserve thin structures and obtain plastic object shapes, see Figure 1. Such volume constraints can be adapted interactively if the corresponding optimization problem can be solved in an efficient manner. Minimal partition problems

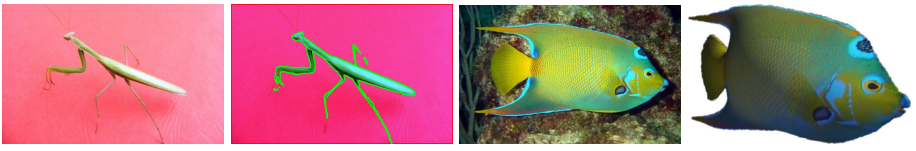


Fig. 1. Results of minimal partition problems (2D segmentation and 3D reconstruction from a single image) with additional volume prior

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 94–107, 2013.
© Springer-Verlag Berlin Heidelberg 2013

can be formulated as energy minimization problems over a continuous domain based on the commonly used total variation functional [8–10] . A bunch of algorithms exists for optimization of the resulting functionals. In this paper the focus will especially lie on the identification of efficient optimization algorithms with respect to the integration of additional constraints such as the volume priors.

2 Related Work

The PD and ADMM algorithms have been previously applied to optimize energies resulting from minimal partition problems. In particular, Goldstein et al. [11] used ADMM to solve the two region segmentation problem amongst other geometrical problems. Recently, ADMM was also applied within several multi-region segmentation approaches [12, 13]. Niethammer et. al. [14] solved the two region segmentation problem with volume constraints using a branch and bound method. They employ ADMM to solve a specific sub–problem in each iteration.

The PD algorithm [15, 16, 10, 1] has become a standard algorithm for solving continuous constrained convex optimization problems with saddle-point structure. It was applied to multi-region segmentation without volume constraints [10, 5] as well as to single view 3D reconstruction [6].

3 Contributions

Both the PD and ADMM algorithm are well known and thoroughly studied, and they have been applied to minimal partition problems before. In this paper we make the following contributions:

- We impose volume constraints on the standard n region minimal partition problem and describe how to apply the ADMM and PD algorithm to minimize the resulting energies. We show that especially ADMM is able to handle projections onto the resulting constraint sets without closed form solution for their combined projection.
- We systematically compare the performance of both algorithms. While ADMM and PD have been compared on standard problems in Computer Vision before, (see [10]) we compare them on a lesser studied problem and also provide performance comparisons on a recent GPU.

4 Minimal Partition Problems with Volume Constraints

Let $\Omega \subset \mathbb{R}^d$ denote the partition domain, i.e. $d = 2$ for segmentation of 2D images, and $d = 3$ for 3D reconstruction. The task of partitioning the domain Ω into a set of n pairwise disjoint regions Ω_i, with $\Omega = \bigcup_{i=1}^{n} \Omega_i, \quad \Omega_i \cap \Omega_j = \emptyset \ \ \forall i \neq j$, can be solved by computing a labeling $l : \Omega \to \{1, \ldots, n\}$ indicating which of the n regions each pixel/voxel belongs to: $\Omega_i = \{x \in \Omega \, : \, l(x) = i\}$.

Recently, minimal partition problems have been formulated as energy minimization problems on the basis of functions of bounded variation (BV), i.e. functions for which

the total variation (TV) is finite. The key idea is to encode the regions Ω_i by their indicator function $u \in \mathrm{BV}(\Omega, \{0,1\})^n$

$$u_i(x) = \begin{cases} 1, & \text{if } l(x) = i \\ 0, & \text{otherwise} \end{cases} \qquad \forall i = 1, \dots, n \tag{1}$$

and to solve for u_i using convex relaxation techniques. For segmentation and 3D reconstruction based on volumetric constraints we can formulate the following convex relaxed optimization problem:

$$\min_{u \in \mathcal{BV}(\Omega,[0,1])^n} \sum_{l=1}^{n} \int_\Omega \|Du_l(x)\| + \int_\Omega f_l(x) u_l(x) \,\mathrm{d}x \tag{2}$$

$$\text{s.t. } u_l(x) \geq 0, \ \sum_{l=1}^{n} u_l(x) = 1 \ \ \forall x \in \Omega, \ \int_\Omega u_l(x) \,\mathrm{d}x = \mathcal{V}_l, \ l \in \{1, \dots, n\}. \tag{3}$$

The total variation of u in (2) measures the length of the region boundaries or the size of the region surfaces, respectively, and the data term $f : \Omega \to \mathbb{R}^n$ indicates how strongly a pixel is associated with each of the n regions. The first two constraints in (3) together form a simplex constraint, which ensures that each pixel is assigned to exactly one label. The linear volume constraints impose a specific volume $\mathcal{V}_l$ on each label. To obtain a binary solution after solving the relaxed problem each pixel is assigned to the label with maximum indicator function value $u_i(x)$. In the two label case, i.e. $n = 2$ this operation will yield a globally optimal solution of the binary problem [17]. Due to the simplex constraint we obtain $u_2 = 1 - u_1$ for this case and thus

$$\min_{u_1 \in \mathcal{BV}(\Omega,[0,1])} \int_\Omega \|Du_1(x)\| + \int_\Omega (f_1(x) - f_2(x)) u_1(x) \,\mathrm{d}x \tag{4}$$

$$\text{s.t. } 0 \leq u_1(x) \leq 1, \ \int_\Omega u_1(x) \,\mathrm{d}x = \mathcal{V}_1, \tag{5}$$

which is the original foreground/background segmentation and single view 3D reconstruction functional of [6]. In this case, the data term $f_1 - f_2$ is chosen so the projection of the segmentation onto the image plane corresponds to the silhouette of the object.

The main problem in the optimization of these functionals are the constraints in (3), which must all be fulfilled simultaneously. One of the simplest optimization strategies for constrained optimization is *Projected Gradient Descent*. However, since the total variation is non-smooth, this method cannot be applied directly. Furthermore, there exists no closed form projection onto the feasible set in (3). Projecting the primal variable onto the intersection of several feasible sets by Dykstra's projection algorithm [18] in every iteration is inexact and inefficient.

A well-known algorithm for non-smooth problems involving the total variation is the *Fast Iterative Shrinkage-Thresholding Algorithm (FISTA)* [19]. However, it turns out that this algorithm also requires a projection onto the feasible set in (3) in every iteration, which makes the algorithm slow and inexact.

Therefore, algorithms which can handle constraints simultaneously without closed form solution for their combined projection are required to obtain fast and accurate

solutions. ADMM can handle such constraints by separate projections on each feasible set. Apart from ADMM there is the Primal Dual algorithm [1, 10], which can handle constraints by the method of Lagrange multipliers.

5 Efficient Convex Optimization

In the following we will present the ADMM and the PD method in detail and see how to apply them to the volume constrained minimal partition task. In general, these algorithms solve problems of the form

$$\min_{x\in\mathbb{R}^n} F(Kx) + G(x) \tag{6}$$

where $K \in \mathbb{R}^{n\times m}$ is a linear operator, and $F : \mathbb{R}^m \to \mathbb{R} \cup \{\infty\}$, $G : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ are proper, closed and convex functions.

Alternating Direction Method of Multipliers. We consider a slightly generalized version of ADMM in *scaled form*, which is given by the update scheme

$$\begin{cases} x^{k+1} = \arg\min_{x\in\mathbb{R}^n} G(x) + \frac{\tau}{2}\|Kx - y^k + \lambda^k\|_2^2 \\ r^{k+1} = \alpha K x^{k+1} + (1-\alpha) y^k \\ y^{k+1} = \arg\min_{y\in\mathbb{R}^m} F(y) + \frac{\tau}{2}\|r^{k+1} - y + \lambda^k\|_2^2 \\ \lambda^{k+1} = \lambda^k + r^{k+1} - y^{k+1} \end{cases} \tag{7}$$

where $\alpha \in (0, 2)$ is an over– or under-relaxation factor, and τ is a step size parameter. For a derivation of this algorithm we refer the interested reader to [20]. A possible interpretation of this algorithm is, that it alternatingly optimizes the Augmented Lagrangian L_τ of the primal problem (6) in the variables x, y and λ:

$$L_\tau(x, y, \lambda) = F(y) + G(x) + \langle \lambda, Kx - y\rangle + \frac{\tau}{2}\|Kx - y\|_2^2$$

This algorithm was first studied by Eckstein et. al. in [2], a proof of convergence of the algorithm is given by Theorem 8 of that paper.

Preconditioned Primal Dual Algorithm. We also consider the recent algorithm by Chambolle et al. [16, 10, 1] for finding a saddle point of the primal-dual formulation of the problem (6)

$$\min_{x\in\mathbb{R}^n} \max_{y\in\mathbb{R}^m} \langle Kx, y\rangle + G(x) - F^*(y) \tag{8}$$

where F^* denotes the convex conjugate of F. The update steps of the algorithm are

$$\begin{cases} x^{k+1} = (\mathrm{id} + T\partial G)^{-1}(x^k - TK^T y^k) \\ \bar{x}^{k+1} = x^{k+1} + \theta(x^{k+1} - x^k) \\ y^{k+1} = (\mathrm{id} + \Sigma\partial F^*)^{-1}(y^k + \Sigma K(\bar{x}^{k+1})) \end{cases} \tag{9}$$

where T and Σ are preconditioning matrices that can be seen as generalized step sizes. This algorithm alternatingly performs gradient descent and ascent steps on (8) in x and y. For further information on the algorithm we refer the interested reader to [1]. Furthermore, it has been shown in [10] that the ADMM algorithm (with $\alpha = 1$) and the PD algorithm are identical, if K is equal to the identity matrix. However, in the general case it is often necessary to solve a least squares problem involving the matrix K in the ADMM algorithm, while the PD algorithm has a closed for solution for the corresponding step.

5.1 Notation and Discretization

In the following we descretize the continuous domain into a regular grid of width W, height H and in the 3D case depth D. To make notation easier we introduce the index set $\Omega = [1 \dots W] \times [1 \dots H]$ to address the individual grid elements of $x \in \mathbb{R}^{WH}$ as x_i for some $i \in \Omega$ (and we define everything analogously in the 3D case).

For the discretization of the gradient operator ∇ we use forward differences and von Neumann boundary conditions. Due to the duality of the gradient and divergence operator the divergence operator is defined as $\operatorname{div} = -\nabla^T$, in order to satisfy the identity $\langle \nabla x, y \rangle = -\langle x, \operatorname{div} y \rangle$. We define the discrete Laplacian as $\Delta = -\nabla^T \nabla$.

Furthermore, we denote the isotropic discretization of the weighted TV

$$TV_g(x) = \int_\Omega g(t) \|\nabla x(t)\|_2 \, \mathrm{d}t \qquad \text{as} \qquad \|\nabla x\|_g = \sum_{i \in \Omega} g_i \|(\nabla x)_i\|_2.$$

We will also make frequent use of the indicator function ι_C of a set C

$$\iota_C(x) := \begin{cases} 0, & x \in C \\ \infty, & x \notin C \end{cases}$$

5.2 Application to Single View 3D Reconstruction

In the discrete setting we can write the continuous formulation of the single view 3D reconstruction problem (4) as

$$\min_{u \in U \cap V} \|\nabla u\|_g + \langle u, f \rangle \tag{10}$$

where the set U corresponds to the simplex constraint and the set V to the volume constraint in (3). In particular, we define the constraints as

$$U = \{u \in X : 0 \leq u_i \leq 1\} \text{ and } V = \left\{ u \in X : \sum_{i \in \Omega} u_i = \mathcal{V} \right\}$$

with $X = \mathbb{R}^{WHD}$ for a domain of width W, height H and depth D.

Alternating Direction Method of Multipliers. We will now apply ADMM to problem (10). We set $K : X \to Y$ as $Ku = \begin{pmatrix}\nabla u & u\end{pmatrix}^T$ and write the constraints as indicator functions ι_U and ι_V to arrive at the following equivalent formulation:

$$\min_{u \in X} \underbrace{\|\nabla u\|_g + \iota_V(u)}_{F(Ku)} + \underbrace{\langle u, f\rangle + \iota_U(u)}_{G(u)} \tag{11}$$

Applying the ADMM scheme (7) yields the following algorithm:

$$\begin{cases} u^{k+1} = \arg\min_{u \in X} \langle u, f\rangle + \iota_U(u) + \frac{\tau}{2}\|\begin{pmatrix}\nabla u \\ u\end{pmatrix} - \begin{pmatrix}v^k \\ w^k\end{pmatrix} + \begin{pmatrix}\lambda_1^k \\ \lambda_2^k\end{pmatrix}\|_2^2 \\ \begin{pmatrix}r_1^{k+1} \\ r_2^{k+1}\end{pmatrix} = \alpha\begin{pmatrix}\nabla u^{k+1} \\ u^{k+1}\end{pmatrix} + (1-\alpha)\begin{pmatrix}v^k \\ w^k\end{pmatrix} \\ \begin{pmatrix}v^{k+1} \\ w^{k+1}\end{pmatrix} = \arg\min_{(v\ w)^T \in Y} \|v\|_g + \iota_V(w) + \frac{\tau}{2}\|\begin{pmatrix}r_1^{k+1} \\ r_2^{k+1}\end{pmatrix} - \begin{pmatrix}v \\ w\end{pmatrix} + \begin{pmatrix}\lambda_1^k \\ \lambda_2^k\end{pmatrix}\|_2^2 \\ \begin{pmatrix}\lambda_1^{k+1} \\ \lambda_2^{k+1}\end{pmatrix} = \begin{pmatrix}\lambda_1^k \\ \lambda_2^k\end{pmatrix} + \begin{pmatrix}r_1^{k+1} \\ r_2^{k+1}\end{pmatrix} - \begin{pmatrix}v^{k+1} \\ w^{k+1}\end{pmatrix} \end{cases} \tag{12}$$

The update steps in r_1, r_2, λ_1 and λ_2 are just simple arithmetic operations. It remains to show how to solve the sub-optimization problems in u, v and w. Note that the coupled optimization problem in v and w is separable into two independent sub-problems.

Solving the sub-problem in u. Ignoring the constraint ι_U which corresponds to $u \in U$, the minimization problem is differentiable. The required optimality conditions are given by the Euler-Lagrange equation:

$$\begin{aligned} \frac{\partial E}{\partial u} &= \frac{f}{\tau} + (u - w^k + \lambda_2^k) - \operatorname{div}(\nabla u - v^k + \lambda_1^k) = 0 \\ \Leftrightarrow \quad u &= (\mathcal{I} - \Delta)^{-1}(-\frac{f}{\tau} + w^k - \lambda_2^k - \operatorname{div} v^k + \operatorname{div} \lambda_1^k) \end{aligned}$$

This smooth, constrained problem can be approximately solved by the *Projected Gradient Descent* method:

$$\hat{u}^{i+1} = \operatorname{proj}_U(\hat{u}^i - \gamma\frac{\partial E}{\partial u})$$

with stepsize γ for $i = 1 \dots N-1$ and $\hat{u}^1 = u^k$, and finally setting $u^{k+1} = \hat{u}^N$.

However, we chose to approximately solve for u by alternating Jacobi iterations with point-wise projection to the constraint set U in the same fashion as [11]. In our experiments we found this to be slightly faster than Projected Gradient Descent, while producing the same final result. We noticed that using only two iterations to solve this sub-problem seem to suffice to make the ADMM algorithm converge.

Solving the sub-problem in v. The closed-form solution for the minimizer of the optimization problem

$$v^{k+1} = \arg\min_v \|v\|_g + \frac{\tau}{2}\|v - (r_1^k + \lambda_1^k)\|_2^2$$

is given by the *coupled shrinkage* formula

$$v^{k+1} = \mathrm{shrink}_{\mathrm{g}}(r_1^k + \lambda_1^k, \tau)$$

which we define similarly to [11, 21] as

$$(\mathrm{shrink}_{\mathrm{g}}(x,\tau))_i = \begin{cases} 0, & \text{if } \|x_i\|_2 \leq \frac{g_i}{\tau} \\ x_i - \frac{g_i}{\tau}\frac{x_i}{\|x_i\|_2}, & \text{otherwise.} \end{cases}$$

Solving the sub-problem in w. The closed form solution to the optimization problem in w is given by the orthogonal projection of $r_2^k + \lambda_2^k$ onto the convex set V.

$$w^{k+1} = \arg\min_w \ \iota_V(w) + \frac{\tau}{2}\|w - (r_2^k + \lambda_2^k)\|_2^2 = \mathrm{proj}_V(r_2^k + \lambda_2^k).$$

Furthermore, a short calculation shows that

$$\mathrm{proj}_V(x) = x - \frac{1}{|\Omega|}\left(\sum_{i\in\Omega} x_i - \mathcal{V}\right)$$

where the set V is defined as above.

Preconditioned Primal Dual Algorithm. We formulate the single view 3D reconstruction functional as the saddle–point problem

$$\min_{u\in X}\max_{\xi\in Y_1,\lambda\in Y_2} \underbrace{\langle \nabla u, \xi\rangle + \langle \lambda, \mathbf{1}^T u\rangle}_{\langle Ku,\,(\xi\ \lambda)^T\rangle} + \underbrace{\langle u, f\rangle + \iota_U(u)}_{G(u)} - \underbrace{(\iota_P(\xi) + \langle \lambda, \mathcal{V}\rangle)}_{F^*((\xi\ \lambda)^T)} \tag{13}$$

where $P = \{\xi \in Y_1 : \|\xi\|_\infty \leq g\}$. Applying scheme (9) yields:

$$\begin{cases} u^{k+1} = \mathrm{proj}_U(u^k - T(-\operatorname{div}\xi^k + \lambda^k - f)) \\ \bar{u}^{k+1} = u^{k+1} + \theta(u^{k+1} - u^k) \\ \xi^{k+1} = \mathrm{proj}_P(\xi^k + \Sigma_1 \nabla \bar{u}^{k+1}) \\ \lambda^{k+1} = \lambda^k + \Sigma_2(\sum_i \bar{u}_i^{k+1} - \mathcal{V}) \end{cases} \tag{14}$$

For a detailed derivation of these update equations, we refer the interested reader to [1], where the algorithm is applied to similar problems without the additional volume constraint. We pick T and $\Sigma = \left(\begin{smallmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{smallmatrix}\right)$ according to the diagonal preconditioning scheme [1] (with $\alpha = 1$), note that in our case K has full rank.

5.3 Application to Volume Constrained Multi-region Segmentation

In the general multi-region case for $N \geq 2$, the discretization of the continuous problem formulation (2) is given by

$$\min_{u\in U^N\cap S\cap V} \sum_{l=1}^{N} \|\nabla u_l\|_g + \langle u_l, f_l\rangle \tag{15}$$

where $u = (u_l)_{l=1}^N$. The constraints are implemented by the three sets $U = \{u \in X : u_i \geq 0\}$, $V = V_1 \times ... \times V_N$ and

$$S = \left\{ u \in X^N : \sum_{l=1}^{N} (u_l)_i = 1 \right\}, \text{ where } V_l = \left\{ u_l \in X : \sum_{i \in \Omega} (u_l)_i = \mathcal{V}_l \right\}$$

with $X = \mathbb{R}^{WH}$. The intersection $U \cap S$ is the simplex constraint and the set V the volume constraint. Furthermore the target volumes for the individual regions are given by $(\mathcal{V}_l)_{l=1}^N$.

Alternating Direction Method of Multipliers. We again rewrite the constraints as indicator functions and arrive at

$$\min_{u \in X^N} \underbrace{\iota_{V \cap S}(u) + \sum_{l=1}^{N} \|\nabla u_l\|_g}_{F(Ku)} + \underbrace{\sum_{l=1}^{N} \langle u_l, f_l \rangle + \iota_U(u_l)}_{G(u)} \tag{16}$$

where $K : X^N \to Y$, $Y := Y_1^N \times Y_2^N$ with $Ku = \begin{pmatrix} \nabla u_1 & ... & \nabla u_N & u_1 & ... & u_N \end{pmatrix}^T$. By applying the ADMM scheme (7) to this formulation we arrive at the following algorithm for the multi-region segmentation problem. Each step is performed for $l = 1..N$, since the minimization problem can be solved separately for each u_l and v_l.

$$\begin{cases} u_l^{k+1} = \arg\min_{u_l \in X} \langle u_l, f_l \rangle + \iota_U(u_l) + \frac{\tau}{2} \| \begin{pmatrix} \nabla u_l \\ u_l \end{pmatrix} - \begin{pmatrix} v_l^k \\ w_l^k \end{pmatrix} + \begin{pmatrix} \lambda_{1,l}^k \\ \lambda_{2,l}^k \end{pmatrix} \|_2^2 \\ \begin{pmatrix} r_{1,l}^{k+1} \\ r_{2,l}^{k+1} \end{pmatrix} = \alpha \begin{pmatrix} \nabla u_l^{k+1} \\ u_l^{k+1} \end{pmatrix} + (1-\alpha) \begin{pmatrix} v_l^k \\ w_l^k \end{pmatrix} \\ v_l^{k+1} = \arg\min_{v_l \in Y_1} \|v_l\|_g + \frac{\tau}{2} \|r_1^{k+1} - v_l + \lambda_{2,l}^k\|_2^2 \\ w^{k+1} = \arg\min_{w \in Y_2^N} \iota_{V \cap S}(w) + \frac{\tau}{2} \|r_2^{k+1} - w + \lambda_2^k\|_2^2 \\ \begin{pmatrix} \lambda_{1,l}^{k+1} \\ \lambda_{2,l}^{k+1} \end{pmatrix} = \begin{pmatrix} \lambda_{1,l}^k \\ \lambda_{2,l}^k \end{pmatrix} + \begin{pmatrix} r_{1,l}^{k+1} \\ r_{2,l}^{k+1} \end{pmatrix} - \begin{pmatrix} v_l^{k+1} \\ w_l^{k+1} \end{pmatrix} \end{cases} \tag{17}$$

Solving the sub-problems in u_l and v_l. These minimization problems are solved in the same way as above.

Solving the sub-problem in w. Solving the minimization problem in w amounts to finding the orthogonal projection onto the convex set $V \cap S$:

$$\begin{aligned} w^{k+1} &= \arg\min_{w} \ \iota_{V \cap S}(w) + \frac{\tau}{2} \|w - (r_2^k + \lambda_2^k)\|_2^2 \\ &= \text{proj}_{V \cap S}(r_2^k + \lambda_2^k) \end{aligned}$$

For the projection of $x = (x_l)_{l=1}^N \in X^N$ onto this convex set we find the following analytical solution

$$\text{proj}_{V \cap S}(x) = \left(x_l - \frac{1}{N} \sum_{j=1}^{N} x_j - \frac{1}{|\Omega|} \left(\sum_{i \in \Omega} (x_l)_i - \frac{1}{N} \sum_{j=1}^{N} \sum_{i \in \Omega} (x_j)_i - \mathcal{V}_l \right) \cdot \mathbf{1}_X \right)_{l=1}^{N}$$

which can be evaluated very efficiently using only basic arithmetic operations. This closed form projection looks somewhat daunting, but can be obtained in a straightforward manner by writing the problem in w as an unconstrained optimization problem with Lagrange multipliers.

Preconditioned Primal Dual Algorithm. Rewriting (15) as a saddle point problem with Lagrange multipliers yields

$$\min_{u\in X}\max_{(\xi,\lambda,\mu)\in Y} \underbrace{\sum_{l=1}^{N}\langle\nabla u_l,\xi_l\rangle + \langle\lambda_l,\mathbf{1}^T u_l\rangle + \langle\mu,\sum_{l=1}^{N}u_l\rangle}_{\langle Ku,(\xi\ \lambda\ \mu)^T\rangle} + \underbrace{\sum_{l=1}^{N}\langle u_l,f_l\rangle + \iota_U(u_l)}_{G(u)} \\ \underbrace{-\big(\iota_P(\xi)+\langle\lambda,\mathcal{V}\rangle+\langle\mu,\mathbf{1}\rangle\big)}_{F^*((\xi\ \lambda\ \mu)^T)} \tag{18}$$

with $P = \big\{\xi\in Y_1^N : \|\xi_l\|_\infty \le g\big\}$, where $Y := Y_1^N \times Y_2^N \times Y_3$. Applying the primal-dual algorithm (9) yields the following update scheme:

$$\begin{cases} u_l^{k+1} = \mathrm{proj}_U(u_l^k - T(-\operatorname{div}\xi_k^k + \lambda_l^k + \mu^k - f)) \\ \bar{u}_l^{k+1} = u_l^{k+1} + \theta(u_l^{k+1} - u_l^k) \\ \xi_l^{k+1} = \mathrm{proj}_P(\xi_l^k + \Sigma_1\nabla\bar{u}_l^{k+1}) \\ \lambda_l^{k+1} = \lambda_l^k + \Sigma_2(\sum_i(\bar{u}_l^{k+1})_i - \mathcal{V}_l) \\ \mu^{k+1} = \mu^k + \Sigma_3(\sum_{l=1}^{N}(\bar{u}_l^{k+1}) - \mathbf{1}_X) \end{cases} \tag{19}$$

Again, we omit the detailed derivation of the update equations and refer to [1] to see how to apply the PD algorithm to saddle point problems. We again chose $\Sigma = \left(\begin{smallmatrix}\Sigma_1 & 0 & 0\\ 0 & \Sigma_2 & 0\\ 0 & 0 & \Sigma_3\end{smallmatrix}\right)$ and T according to the diagonal preconditioning scheme with $\alpha = 1$.

6 Numerical Experiments

For evaluating the ADMM and the PD algorithm with respect to runtime for minimal partition problems under additional volume constraints we apply these algorithms to two important problems: 3D reconstruction as formulated in [6] and image segmentation based on the formulation in [5] with additional volume constraints. Both algorithms were implemented according to the update rules described above and run until the RMSE error $\|u-u^*\|_2/\sqrt{|\Omega|}$ dropped below a threshold of $\varepsilon = 5.0 * 10^{-3}$. The optimal solution u^* of the energy minimization problem was computed by letting the algorithms run for a very long time.

We chose the ADMM parameters as $\tau = 1$, $\alpha = 1.5$ and we used 2 Jacobi iterations for the multi-region segmentation problem and 5 for the 3D reconstruction problem. For the PD algorithm we set $\theta = 1$ and chose T and Σ as described above.

Fig. 2. 3D reconstruction: Results from a single image, first row: original image, second row: reconstructed geometry, third row: textured result. The results look identical for PD and ADMM.

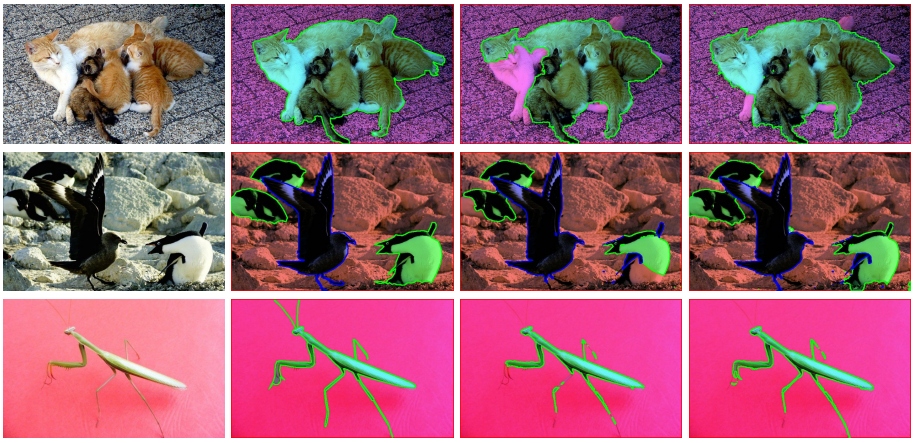


Fig. 3. Segmentation: a) original input images, b) ground truth, c) segmentation results without volume constraints and d) with volume constraints. The results look identical for PD and ADMM.

The algorithms were implemented in C++/CUDA and run on graphics processing units (GPUs). We used 32–bit floating point precision. All numerical experiments were carried out on a PC with a 3.4GHz Intel i7-3770 CPU with 32GB RAM and a NVIDIA GeForce GTX680 graphics card on a Linux distribution.

6.1 Results for 3D Reconstruction

For 3D reconstruction we used a total number of 18 test images and computed the reconstruction for each of them using PD and ADMM, see Figure 2. To examine how the algorithms scale with the resolution of the reconstruction domain Ω we used three different resolutions. A rather coarse resolution with few ($\approx 3.4 \cdot 10^4$) voxels (sm), an intermediate ($\approx 8.0 \cdot 10^5$) resolution (med) and a fine resolution with a large number ($\approx 3.6 \cdot 10^6$) of voxels (big).

For each resolution the average runtime and standard deviation over all test images is given in Table 1. Figure 4 shows an example for the different performances of PD and ADMM on the giraffe image in Figure 2. The horizontal line indicates the termination criterion where the desired accuracy of the solution is reached. Results on the other test images look similar. From the results we can conclude that the ADMM algorithm converges faster than the PD algorithm for the 3D reconstruction problem instance of the minimal partition problems with volume constraints.

Table 1. 3D reconstruction: Average and standard deviation of the runtime and number of iterations for the PD and ADMM algorithm over 18 different single view 3D Reconstruction examples. The results show that ADMM performs better.

	Runtime in Seconds		Number of Iterations	
$\lvert\Omega\rvert$	PD	ADMM	PD	ADMM
sm	0.15 ($\pm$ 0.10)	0.10 ($\pm$ 0.05)	1015 ($\pm$ 618)	300 ($\pm$ 152)
med	13.47 ($\pm$ 11.64)	8.35 ($\pm$ 7.32)	5716 ($\pm$ 5664)	1471 ($\pm$ 1399)
big	115.30 ($\pm$ 113.60)	79.16 ($\pm$ 91.80)	11724 ($\pm$ 12136)	3171 ($\pm$ 3776)

Table 2. Segmentation: Average and standard deviation of the runtime and iteration numbers for the PD and ADMM algorithms on the Graz benchmark for interactive segmentation containing 262 test examples. We can conclude that ADMM outperforms the PD method.

Runtime in Seconds		Number of Iterations	
PD	ADMM	PD	ADMM
4.17 ($\pm$ 6.41)	3.52 ($\pm$ 5.95)	6843 ($\pm$ 8257)	2950 ($\pm$ 3822)

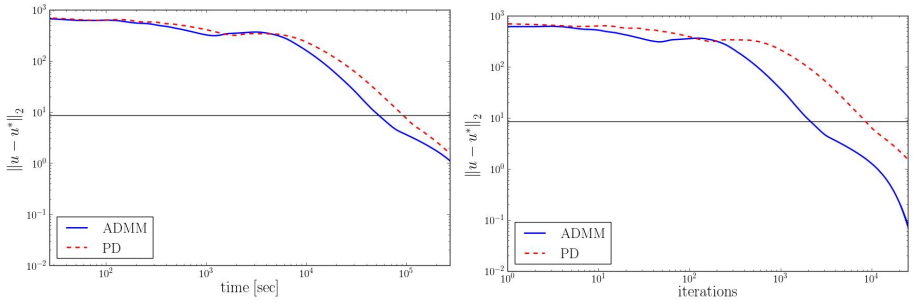


Fig. 4. 3D reconstruction: Exemplary convergence of the ADMM and the PD algorithm. The horizontal line indicates the termination criterion based on the accuracy of the algorithm (vertical axis). The ADMM algorithm converges significantly faster than the PD algorithm.

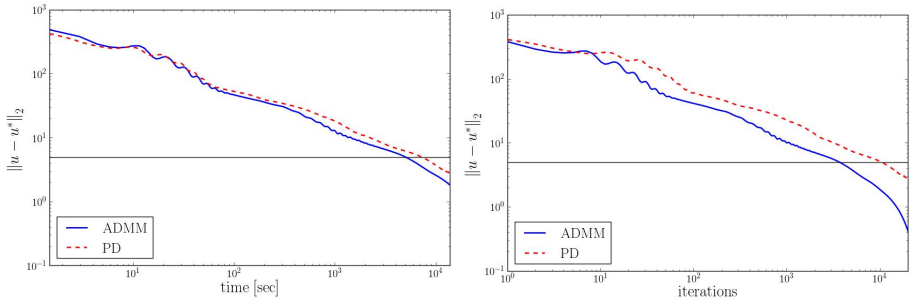


Fig. 5. Segmentation: Exemplary convergence of the ADMM and the PD algorithm for a single image from the Graz benchmark for interactive segmentation. The horizontal line indicates the termination criterion based on the accuracy of the algorithm (vertical axis). For the multi-region segmentation problem the ADMM algorithm converges slightly faster than the PD algorithm.

6.2 Results for Volume Constrained Multi-region Segmentation

For the performance evaluation of the multi-region segmentation approach we use the Graz interactive segmentation benchmark proposed by Santner et al. [4]. This benchmark contains 262 hand labeled pairs of user scribbles with ground truth. Since this benchmark is only used for evaluating the performance of the algorithm the volume constraints are computed from the ground truth segmentations for each image. The data terms f_i in (2) are computed based on [5], but are not part of the performance evaluation.

Figure 5 exemplarily shows the performance in terms of runtime and number of iterations for the first image of the Graz database, which contains four labels. The other performance plots look similar. On average we obtain the results in Table 2 over all images of the benchmark. From the results we can conclude that ADMM is also faster for the multi-region segmentation problem as an instance of the minimal partition problems with volume constraint. Figure 3 shows some qualitative segmentation results based on an optimized volume constraint.

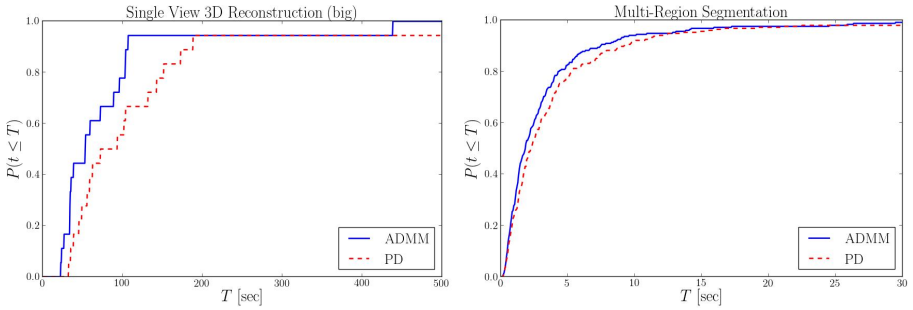


Fig. 6. Cumulative distribution functions over the runtimes for the 18 3D reconstruction examples and the 262 segmentation examples. The plots indicate for each runtime T in seconds (horizontal axis) the ratio of examples with runtimes below T seconds $P(t \leq T)$ (vertical axis). The faster the curve grows the more efficient is the algorithm. For both instances of partition problems the ADMM algorithm yields better performance results.

6.3 Cumulative Distribution Function of Runtimes

Finally, we show the cumulative distribution functions (CDF) of the runtimes for both partition problems instances, i.e. 3D reconstruction and multi-region segmentation, in Figure 6. The vertical axis shows the ratio of samples with runtimes below T seconds on the horizontal axis ($P(t \leq T)$). The 3D reconstruction results are based on 18 sample images, which causes the staircase pattern in the plot. For the segmentation results we had 262 sample images. The CDF plots confirm that the ADMM algorithm in general converges faster than the PD algorithm.

7 Conclusion

We have compared two algorithms for solving the minimal partition problem with additional volume constraints, PD and ADMM. In the PD algorithm the constraints are handled by Lagrange multipliers, and every iteration step has a closed form solution. Contrary, the ADMM algorithm requires an approximate solution of a least squares problem in every iteration, and the constraints are handled by orthogonal projections onto the corresponding sets. The results suggest that both algorithms can be extended in a straightforward manner in order to handle additional convex constraints.

We conducted several experiments on 3D reconstruction examples and on the Graz interactive segmentation benchmark. Our experiments indicate that the ADMM approach provides comparable and often better performance. On average, it yields shorter runtimes than the PD algorithm and requires significantly less iterations, while the quality of the results is identical.

References

1. Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In: Int. Conf. Comp. Vis. (2011)
2. Eckstein, J., Bertsekas, D.P.: On the Douglas-Rachford Splitting Method and the Proximal Point Algorithm for Maximal Monotone Operators. J. Math. Program. 55, 293–318 (1992)

3. Unger, M., Pock, T., Cremers, D., Bischof, H.: TVSeg - Interactive Total Variation Based Image Segmentation. In: Brit. Mach. Vis. Conf. (2008)
4. Santner, J., Pock, T., Bischof, H.: Interactive Multi-Label Segmentation. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 397–410. Springer, Heidelberg (2011)
5. Nieuwenhuis, C., Cremers, D.: Spatially Varying Color Distributions for Interactive Multi-Label Segmentation. IEEE Trans. Pattern Anal. Mach. Intell. (2012)
6. Töppe, E., Oswald, M.R., Cremers, D., Rother, C.: Image-based 3D Modeling via Cheeger Sets. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part I. LNCS, vol. 6492, pp. 53–64. Springer, Heidelberg (2011)
7. Kolev, K., Pock, T., Cremers, D.: Anisotropic Minimal Surfaces Integrating Photoconsistency and Normal Information for Multiview Stereo. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part III. LNCS, vol. 6313, pp. 538–551. Springer, Heidelberg (2010)
8. Zach, C., Gallup, D., Frahm, J.M., Niethammer, M.: Fast Global Labeling for Real-Time Stereo Using Multiple Plane Sweeps. In: Vision, Modeling and Visualization Workshop (2008)
9. Lellmann, J., Schnörr, C.: Continuous Multiclass Labeling Approaches and Algorithms. J. Imaging Sci. 4, 1049–1096 (2010)
10. Chambolle, A., Pock, T.: A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. J. Math. Imaging Vis. 40, 120–145 (2011)
11. Goldstein, T., Bresson, X., Osher, S.: Geometric Applications of the Split Bregman Method: Segmentation and Surface Reconstruction. Technical report, UCLA (2009)
12. Paul, G., Cardinale, J., Sbalzarini, I.: An Alternating Split Bregman Algorithm for Multi-Region Segmentation. In: IEEE Asilomar Conf. Signals, Systems, and Computers, pp. 426–430 (2011)
13. Häuser, S., Steidl, G.: Convex Multiclass Segmentation with Shearlet Regularization. Int. J. Comp. Math. 90, 62–81 (2013)
14. Niethammer, M., Zach, C.: Segmentation with area constraints. Medical Image Analysis 17, 101–112 (2013)
15. Chambolle, A., Cremers, D., Pock, T.: A Convex Approach for Computing Minimal Partitions. Technical report, University of Bonn (2008)
16. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An Algorithm for Minimizing the Piecewise Smooth Mumford-Shah Functional. In: Int. Conf. Comp. Vis. (2009)
17. Chan, T., Esedoḡlu, S., Nikolova, M.: Algorithms for Finding Global Minimizers of Image Segmentation and Denoising Models. J. Appl. Math. 66, 1632–1648 (2006)
18. Boyle, J.P., Dykstra, R.L.: A method for finding projections onto the intersection of convex sets in Hilbert spaces. Lecture Notes in Statistics, vol. 37, pp. 28–47
19. Beck, A., Teboulle, M.: A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. J. Img. Sci. 2, 183–202 (2009)
20. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. J. Found. Trends Mach. Learn. 3, 1–122 (2011)
21. Setzer, S.: Split Bregman Algorithm, Douglas-Rachford Splitting and Frame Shrinkage. In: Tai, X.-C., Mørken, K., Lysaker, M., Lie, K.-A. (eds.) SSVM 2009. LNCS, vol. 5567, pp. 464–476. Springer, Heidelberg (2009)

Discrete Geodesic Regression in Shape Space

Benjamin Berkels[1], P. Thomas Fletcher[2], Behrend Heeren[1], Martin Rumpf[1], and Benedikt Wirth[3]

[1] Institute for Numerical Simulation, Universität Bonn
{benjamin.berkels,behrend.heeren,martin.rumpf}@ins.uni-bonn.de
[2] Scientific Computing and Imaging Institute, University of Utah
fletcher@sci.utah.edu
[3] Courant Institute of Mathematical Sciences, New York University
Benedikt.Wirth@cims.nyu.edu

Abstract. A new approach for the effective computation of geodesic regression curves in shape spaces is presented. Here, one asks for a geodesic curve on the shape manifold that minimizes a sum of dissimilarity measures between given two- or three-dimensional input shapes and corresponding shapes along the regression curve. The proposed method is based on a variational time discretization of geodesics. Curves in shape space are represented as deformations of suitable reference shapes, which renders the computation of a discrete geodesic as a PDE constrained optimization for a family of deformations. The PDE constraint is deduced from the discretization of the covariant derivative of the velocity in the tangential direction along a geodesic. Finite elements are used for the spatial discretization, and a hierarchical minimization strategy together with a Lagrangian multiplier type gradient descent scheme is implemented. The method is applied to the analysis of root growth in botany and the morphological changes of brain structures due to aging.

1 Introduction

Geodesic paths in shape space are the natural generalization of straight lines in Euclidian space. Applications include shape modeling in computer vision and graphics [1,2] or in computational anatomy [3,4], and shape clustering [5] as well as shape statistics [6]. As a generalization of linear regression in Euclidian space we investigate in this paper geodesic paths in shape space which best approximate given time indexed sets of input shapes in a least squares sense. Fig. 1 shows a discrete geodesic regression path in the space of 3D objects representing the growth process of sugar beet roots over a vegetation period.

Time-dependent shape statistics and shape regression has already been investigated in [7], where the regression curve is obtained via a simultaneous kernel weighted averaging in time and on shape space. In the application to brain images the kernel on shape space is linked to the Sobolev metric from the group of diffeomorphisms approach [8]. A variational formulation of geodesic regression is given in [9], where for given input shapes $\mathbf{S}_i$ at times t_i the (in a least squares sense) best approximating geodesic is computed as the minimizer of the energy

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 108–122, 2013.
© Springer-Verlag Berlin Heidelberg 2013

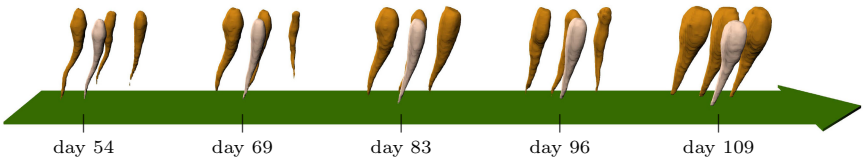


Fig. 1. Discrete regression curve (bright) for sugar beet input shapes (dark) at 5 different days in the vegetation period

$E[\mathbf{S}, v] = \frac{1}{2}\sum_i \mathrm{dist}^2(\exp_p(t_i\, v), \mathbf{S}_i)$ over the initial shape $\mathbf{S}$ of the geodesic path and its initial velocity or momentum v. Here, $\mathrm{dist}(\cdot,\cdot)$ is the Riemannian distance and exp the exponential map. A computationally efficient method in the group of diffeomorphisms shape space is based on duality calculus in constrained optimization and presented in [10]. In [11], a generalization allowing for image metamorphosis—simultaneous diffeomorphic image deformation and image intensity modulation—is proposed. In contrast to these approaches, we here do not minimize over the initial data of geodesic shooting but directly over the shapes along a time discrete geodesic. The classical energy minimization property of cubic splines exploited in the shape space context in [12] is related to the penalty used in our approach, which is defined via the L^2 integral of the covariant derivative of the shape velocity along shape curves.

There is a rich diversity of underlying Riemannian structures in the shape space context. On the space of planar curves the L^2-metric on direction and curvature functions is proposed in [1], and the L^2-metric on stretching and bending variations in [13], as well as curvature-weighted L^2- or Sobolev-type metrics in [14,15], some of which allow closed-form geodesics [16,17]. In the flow of diffeomorphism approach [18] the metric $g(v,v) = \int_D Lv \cdot v \,\mathrm{d}x$ is defined on Eulerian motion fields v for a higher order elliptic operator L on some computational domain $D \subset \mathbb{R}^d$. Fuchs et al. [19] propose a viscous-fluid based Riemannian metric related to the approach considered here. Here, we take up the time discrete concept proposed in [20] and define time discrete geodesics as minimizers of a time discrete path energy. In the concrete application, the path energy consists of a sum of matching energies, whose Hessian at the identity coincides with the rate of viscous dissipation generated by the shape deformation.

The paper is organized as follows. Based on a brief review in Section 2 of the concept of discrete geodesics proposed in [20] we develop a general variational model for shape regression in Section 3. In Section 4 this model is illustrated for the case of a finite dimensional manifold embedded in Euclidian space. Then, in Section 5 we investigate the application to the space of viscous fluidic objects. The algorithmic ingredients required to solve the underlying optimization problem are studied in detail in Section 6. The application of the method to shape statistics in anatomy and botany is presented in Section 7. Finally, in Section 8 we draw conclusions.

2 Discrete Geodesics in Shape Space

In this section, we briefly review the notion of continuous geodesics on shape spaces considered as Riemannian manifolds and adopt a recently introduced variational time discretization [20]. Let $\mathcal{M}$ denote the space of shapes as a Riemannian manifold with a metric $g_{\mathbf{S}}$ acting on variations $\dot{\mathbf{S}}$ of shapes $\mathbf{S}$ which are considered as tangent vectors on the manifold $\mathcal{M}$. A curve $\mathbf{S} : [0,1] \to \mathcal{M}$ with $\mathbf{S}(0) = \mathbf{S}_A$ and $\mathbf{S}(1) = \mathbf{S}_B$ is a geodesic if it is a local minimizer of the path energy $\mathcal{E}[(\mathbf{S}(t))_{t\in[0,1]}] = \int_0^1 g_{\mathbf{S}(t)}(\dot{\mathbf{S}}(t), \dot{\mathbf{S}}(t))\,\mathrm{d}t$. Such curves solve $\nabla_{\dot{\mathbf{S}}(t)}\dot{\mathbf{S}}(t) = 0$, where ∇ denotes the Levi-Civita connection, and in addition $g_{\mathbf{S}(t)}(\dot{\mathbf{S}}(t), \dot{\mathbf{S}}(t)) = \text{const}$. The associated distance $\mathrm{dist}(\mathbf{S}_A, \mathbf{S}_B)$ is the minimal path length $\mathcal{L}[(\mathbf{S}(t))_{t\in[0,1]}] = \int_0^1 (g_{\mathbf{S}(t)}(\dot{\mathbf{S}}(t), \dot{\mathbf{S}}(t)))^{\frac{1}{2}}\,\mathrm{d}t$ and minimizers of the energy also minimize the path length. If a continuous path $(\mathbf{S}(t))_{t\in[0,1]}$ is sampled at times $t_k = k\tau$ for $k = 0, \ldots, K$ and $\tau := \frac{1}{K}$, Jensen's inequality implies the estimate $\mathcal{E}[(\mathbf{S}(t))_{t\in[0,1]}] \geq \frac{1}{\tau}\sum_{k=1}^{K} \mathrm{dist}^2(\mathbf{S}_{k-1}, \mathbf{S}_k)$ with $\mathbf{S}_k = \mathbf{S}(t_k)$. Here, equality holds for geodesic paths due to the constant speed property of energy minimizing paths. This observation allows us to derive the fundamental notion of a discrete geodesic: Approximating the local squared Riemannian distance $\mathrm{dist}^2(\cdot,\cdot)$ by a functional $\mathbf{W} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}, (\mathbf{S}_1, \mathbf{S}_2) \mapsto \mathbf{W}[\mathbf{S}_1, \mathbf{S}_2]$ in such a way that $\mathrm{dist}^2(\mathbf{S}, \tilde{\mathbf{S}}) = \mathbf{W}[\mathbf{S}, \tilde{\mathbf{S}}] + O(\mathrm{dist}^3(\mathbf{S}, \tilde{\mathbf{S}}))$, we are naturally led to the *discrete path energy* $\mathbf{E}$

$$\mathbf{E}[\mathcal{S}_K] = K \sum_{k=1}^{K} \mathbf{W}[\mathbf{S}_{k-1}, \mathbf{S}_k] \tag{1}$$

for discrete paths $\mathcal{S}_K = (\mathbf{S}_0, \ldots, \mathbf{S}_K)$. A *discrete geodesic* (of order K) is then defined as a minimizer of $\mathbf{E}[\mathcal{S}_K]$ for fixed end points $\mathbf{S}_0, \mathbf{S}_K$. In this discrete model $\mathbf{W}[\mathbf{S}, \tilde{\mathbf{S}}]$ can be interpreted as the cost required to deform the shape $\mathbf{S}$ into the shape $\tilde{\mathbf{S}}$. The required approximation property already implies $\mathbf{W}[\mathbf{S}, \mathbf{S}] = 0$ and that for the first and second variation of $\mathbf{W}$ with respect to the second shape we have $\partial_{\mathbf{S}_2}\mathbf{W}[\mathbf{S}, \mathbf{S}] = 0$ and $\frac{1}{2}\partial^2_{\mathbf{S}_2}\mathbf{W}[\mathbf{S}, \mathbf{S}] = g_{\mathbf{S}}$ for smooth g and $\mathbf{W}$ (cf. the exposition in [20]).

3 Derivation of the Discrete Regression Model

Based on the concept of discrete geodesics revisited in the last section, we are now in the position to derive our discrete shape regression model from a corresponding model of continuous geodesic regression. Let us consider sets of input shapes $\{\mathbf{S}_k^i\}_{i=1,\ldots,i_k}$ for $k = 0, \ldots, K$, which represent sets of statistical measurements at times $t_k \in [0,1]$ on the shape manifold $\mathcal{M}$. As a notational simplification let us suppose already here that all times t_k at which input shapes are given are multiples of the time step size $\tau = \frac{1}{K}$ of the discrete model to be introduced later. Indeed, this is no severe restriction because a generalization to discrete geodesics with non-constant time step sizes is straightforward (actually, a non-constant time step has been used in the computation underlying Fig. 1).

Now we ask for a geodesic curve $\mathcal{S} = (\mathbf{S}(t))_{t\in[0,1]}$ on $\mathcal{M}$ which minimizes a sum of dissimilarity measures between the input shapes $\mathbf{S}_k^i$ and the associated shapes $\mathbf{S}(t_k)$ on the geodesic curve. More precisely, we aim at minimizing

$$\mathcal{F}[\mathcal{S}] = \sum_{k=0}^{K} \sum_{i=1}^{i_k} \mathrm{diss}[\mathbf{S}(t_k), \mathbf{S}_k^i] \tag{2}$$

under the constraint that $\mathcal{S}$ is actually a geodesic, namely that $\nabla_{\dot{\mathbf{S}}(t)}\dot{\mathbf{S}}(t) = 0$. In analogy to the standard linear regression model in $\mathbb{R}^n$ the measure $\mathrm{diss}[\cdot,\cdot]$ might be the squared (geodesic) distance on $\mathcal{M}$ (cf. [9]) or another in general nonlinear measure of shape dissimilarity. Based on the above discussed approximation properties $\mathrm{diss}[\mathbf{S}(t_k), \mathbf{S}_k^i] = \mathbf{W}[\mathbf{S}(t_k), \mathbf{S}_k^i]$ is a natural choice.

Instead of enforcing the strict constraint that $\mathcal{S}$ is a geodesic, we might alternatively consider a penalty approach with

$$\mathcal{F}^\epsilon[\mathcal{S}] = \mathcal{F}[\mathcal{S}] + \frac{C}{\epsilon}\int_0^1 g_{\mathbf{S}(t)}(\nabla_{\dot{\mathbf{S}}(t)}\dot{\mathbf{S}}(t), \nabla_{\dot{\mathbf{S}}(t)}\dot{\mathbf{S}}(t))\,\mathrm{d}t \tag{3}$$

for a small penalty parameter ϵ. Obviously, on geodesic curves $\mathcal{S}$ the two energies $\mathcal{F}^\epsilon[\mathcal{S}]$ and $\mathcal{F}[\mathcal{S}]$ coincide. As a scaling factor C we choose $\mathcal{F}[\mathcal{S}^{\mathrm{ref}}]$, where $\mathcal{S}^{\mathrm{ref}}$ is a curve in shape space with $\mathbf{S}^{\mathrm{ref}}(t_k)$ being the shape mean of the input shapes at time t_k in the sense of [21], i. e. $\mathbf{S}^{\mathrm{ref}}(t_k) = \mathrm{argmin}_{\tilde{\mathbf{S}}} \sum_{i=1}^{i_k} \mathrm{diss}[\tilde{\mathbf{S}}, \mathbf{S}_k^i]$.

Let us now derive a discrete analog of the above continuous variational problem and specifically ask for a discrete geodesic regression curve. Now, we consider discrete curves $(\mathbf{S}_0, \ldots, \mathbf{S}_K)$ in shape space and assume (potentially after reindexing) that $\{\mathbf{S}_k^i\}_{i=1,\ldots,i_k}$ is the set of input shapes attached to the time $t_k = k\tau$. If $i_k = 0$ then the corresponding set is empty, and in what follows the associated sums over i are defined to be zero. With this notation at hand the discrete geodesic regression problem reads as follows:

Find a discrete path $\mathcal{S}_K = (\mathbf{S}_0, \ldots, \mathbf{S}_K)$ *such that* $\mathbf{F}[\mathcal{S}_K] = \sum_{k=0}^{K} \sum_{i=1}^{i_k} \mathrm{diss}[\mathbf{S}_k, \mathbf{S}_k^i]$ *is minimal under the constraint that* $\mathcal{S}_K$ *is a discrete geodesic, i. e.* $\mathcal{S}_K$ *minimizes* $\mathbf{E}[(\mathbf{S}_0, \ldots, \mathbf{S}_K)]$ *among all discrete paths with same end shapes* $\mathbf{S}_0$ *and* $\mathbf{S}_K$.

To derive a discrete penalty approach we take into account a suitable approximation of the integrand of the continuous penalty energy. Indeed, for g and $\mathbf{W}$ sufficienty smooth $g_{\mathbf{S}(t_k)}(\nabla_{\dot{\mathbf{S}}(t_k)}\dot{\mathbf{S}}(t_k), \nabla_{\dot{\mathbf{S}}(t_k)}\dot{\mathbf{S}}(t_k)) = 4K^4\mathbf{W}(\tilde{\mathbf{S}}_k, \mathbf{S}(t_k)) + O(K^{-2})$ for each $k = 1, \ldots, K-1$, where $\tilde{\mathbf{S}}_k$ is the middle shape of the discrete geodesic $(\mathbf{S}_{k-1} = \mathbf{S}(t_{k-1}), \tilde{\mathbf{S}}_k, \mathbf{S}_{k+1} = \mathbf{S}(t_{k+1}))$, i. e.

$$\tilde{\mathbf{S}}_k = \mathrm{argmin}_{\mathbf{S}}\left(\mathbf{W}[\mathbf{S}_{k-1}, \mathbf{S}] + \mathbf{W}[\mathbf{S}, \mathbf{S}_{k+1}]\right). \tag{4}$$

This is a straightforward consequence of the convergence theory presented in [22]. Then, writing $\tilde{\mathcal{S}}_K = (\tilde{\mathbf{S}}_1, \ldots, \tilde{\mathbf{S}}_{K-1})$ and using the simple quadrature $\int_0^1 f(t)\mathrm{d}t = K\sum_{k=1}^{K-1} f(t_k) + O(\tau)$ we obtain a discrete penalty approach

$$\mathbf{F}^\epsilon[\mathcal{S}_K] = \mathbf{H}^\epsilon[\mathcal{S}_K, \tilde{\mathcal{S}}_K] = \mathbf{F}[\mathcal{S}_K] + \frac{4CK^3}{\epsilon}\sum_{k=1}^{K-1} \mathbf{W}[\tilde{\mathbf{S}}_k, \mathbf{S}_k] \tag{5}$$

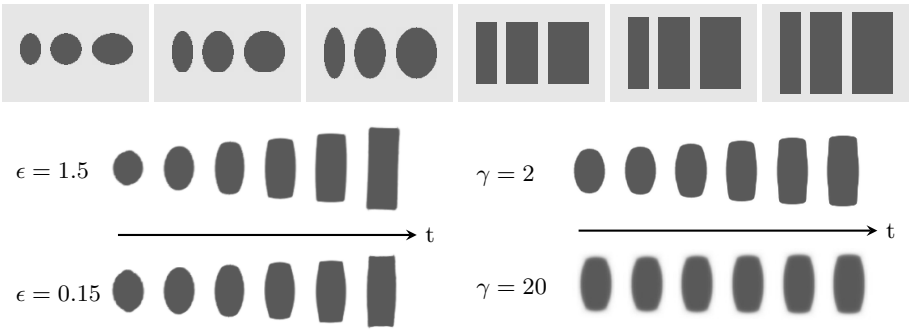


Fig. 2. Results of the discrete geodesic regression for given input objects at 6 timesteps (top row) are shown for different ϵ (left) and compared to the results obtained by using a simple path length penalty $\gamma \mathbf{E}[\mathcal{S}_K]$ (right)

and seek discrete paths which minimize this functional. The approach developed here is fairly general and will in Section 5 be applied to a concrete shape space, namely the space of viscous fluidic objects. Fig. 2 shows results of the discrete geodesic regression approach for different penalty parameters ϵ. For decreasing ϵ one observes an increased rounding effect towards the right of the curve, which reflects the global impact of the round input shapes on the resulting approximate discrete geodesic. These results are compared to regression curves obtained when replacing the proposed penalty by the simpler penalty $\gamma\, \mathbf{E}[\mathcal{S}_K]$, where $\mathbf{E}$ is the discrete path energy defined in (1). The latter leads to a collapse of the regression curve to a global shape average (cf. [21]).

4 Regression on a Finite Dimensional Manifold

At first, as a simple example and to further motivate our approach let us briefly demonstrate the discrete geodesic regression for the much simpler case of an m-dimensional surface $\mathcal{M}$ embedded in $\mathbb{R}^d$. We suppose $\mathcal{M}$ to be parametrized via a smooth parametrization $Y : \omega \to \mathbb{R}^d;\ \theta \mapsto Y(\theta)$ over a parameter domain ω. Furthermore, we consider the simple energy $\mathbf{W}[\tilde{\theta}, \theta] = |Y(\tilde{\theta}) - Y(\theta)|^2$, which reflects the stored elastic energy in a spring spanned between points $Y(\theta)$ and $Y(\tilde{\theta})$ through the ambient space of $\mathcal{M}$ in $\mathbb{R}^d$. Thus, the discrete path energy of a path $(\theta_0, \dots, \theta_K)$ is given by $\mathbf{E}[(\theta_0, \dots, \theta_K)] = K \sum_{k=1}^{K} |Y(\theta_k) - Y(\theta_{k-1})|^2$. Now, we suppose measurements θ_k^i to be given for $k = 0, \dots, K$, $i = 1, \dots, i_k$, i. e. $\{Y(\theta_k^i)\}_{i=1,\dots,i_k}$ are the corresponding input points on $\mathcal{M}$ at time $t_k = \tau k$. (As above $i_k = 0$ indicates an empty set of input points, and in what follows the associated sums over i are assumed empty.) Then the discrete regression problem reads as follows:

Find a discrete path $\Theta = (\theta_0, \dots, \theta_K)$ *and associated points* $\tilde{\Theta} = (\tilde{\theta}_1, \dots, \tilde{\theta}_{K-1})$ *such that* $\mathbf{H}^\epsilon[\Theta, \tilde{\Theta}] = \sum_{k=0}^{K} \sum_{i=1}^{i_k} |Y(\theta_k) - Y(\theta_k^i)|^2 + \frac{4K^3 C}{\epsilon} \sum_{k=1}^{K-1} |Y(\tilde{\theta}_k) - Y(\theta_k)|^2$ *is minimal subject to the constraint that* $\tilde{\theta}_k$ *for* $k = 1, \dots, K-1$ *minimizes*

$\mathbf{G}(\theta_{k-1}, \theta, \theta_{k+1}) := |Y(\theta) - Y(\theta_{k-1})|^2 + |Y(\theta) - Y(\theta_{k+1})|^2$ *over all* $\theta \in \omega$.

Let us emphasize that the chosen parametrization influences the numerical solution process but not the resulting regression curves which solely depend on the manifold $\mathcal{M}$, the input data and the penalty parameter ϵ. Fig. 3 shows discrete regression curves on the sphere. In the implementation a parametrization via polar and azimuth angle was used. One clearly observes that for small ϵ the regression curve is close to a great circle so that sets of input points at a particular time might be located completely on one side of the regression curve. For larger values of ϵ the method still computes a smooth regression curve which better matches the input points at the expense of being far from a geodesic.

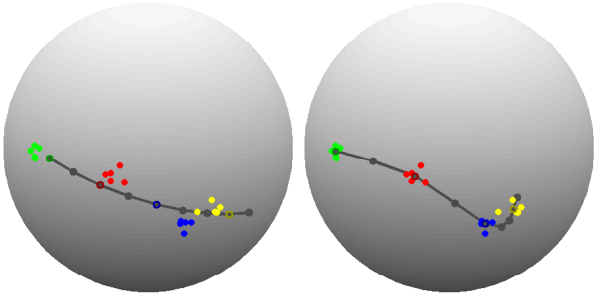


Fig. 3. Regression curve (black) for 20 input points on the unit sphere for a strong penalty weight with $\epsilon = 0.028$ (left) and just a mild curve smoothing with $\epsilon = 28$ (right)

The optimization algorithm. Denote by $\tilde{\Theta}[\Theta] = (\tilde{\theta}_1, \ldots, \tilde{\theta}_{K-1})$ the vector of minimizers of $\mathbf{G}(\theta_{k-1}, \cdot, \theta_{k+1})$, and define $\mathbf{F}^\epsilon[\Theta] = \mathbf{H}^\epsilon[\Theta, \tilde{\Theta}[\Theta]]$. The necessary condition for $\tilde{\theta}_k$ to minimize $\mathbf{G}(\theta_{k-1}, \cdot, \theta_{k+1})$ is $(2Y(\tilde{\theta}_k) - Y(\theta_{k-1}) - Y(\theta_{k+1}))^T DY(\tilde{\theta}_k) = 0$. To perform a gradient descent for the energy $\mathbf{F}^\epsilon$ we apply the duality techniques from constrained optimization [23] and define the dual solution $p_k \in \mathbb{R}^m$ as the solution of $D^2_\theta \mathbf{G}(\theta_{k-1}, \tilde{\theta}_k, \theta_{k+1}) p_k = \big(\partial_{\tilde{\theta}_k} \mathbf{H}^\epsilon\big)[\Theta, \tilde{\Theta}[\Theta]]$. Then we obtain $\partial_{\theta_k} \mathbf{F}^\epsilon[\Theta] = \big(\partial_{\theta_k} \mathbf{H}^\epsilon\big)[\Theta, \tilde{\Theta}(\Theta)] - \big(\partial_{\theta_k} \partial_{\tilde{\theta}_k} \mathbf{H}^\epsilon\big)[\Theta, \tilde{\Theta}[\Theta]]\, p_k$. The required derivatives of $\mathbf{G}$ and $\mathbf{H}^\epsilon$ are evaluated as follows,

$$D^2_\theta \mathbf{G}(\theta_{k-1}, \theta, \theta_{k+1})_{ij} = \sum_{n=1}^{d} \Big(2Y_{n,i}(\theta) Y_{n,j}(\theta) + (2Y_n(\theta) - Y_n(\theta_{k-1}) - Y_n(\theta_{k+1})) Y_{n,ij}(\theta) \Big) ,$$

$$\partial_{\tilde{\theta}_k} \mathbf{H}^\epsilon[\Theta, \tilde{\Theta}]_i = \frac{8CK^3}{\epsilon} \sum_{n=1}^{d} (Y(\tilde{\theta}_k) - Y(\theta_k))_n Y_{n,i}(\tilde{\theta}_k) ,$$

$$\partial_{\theta_k} \partial_{\tilde{\theta}_k} \mathbf{H}^\epsilon[\Theta, \tilde{\Theta}]_{ij} = -\frac{8CK^3}{\epsilon} \sum_{n=1}^{d} Y_{n,j}(\theta_k) Y_{n,i}(\tilde{\theta}_k) .$$

5 Shape Regression in the Space of Viscous Fluidic Objects

Now we apply the general approach of discrete geodesic regression to a physically motivated shape space of viscous fluidic objects with a metric induced by the

notion of viscous dissipation. Here, we assume that a viscous flow model at least captures some characteristics of the usually much more complex underlying processes such as e. g. plant growth. To this end, one considers objects $\mathbf{S}$ which are homeomorphic to a reference object $\hat{\mathbf{S}} \subset \mathbb{R}^d$ $(d = 2, 3)$. In general, one is not interested in point to point correspondences between two different objects but represents the actual shape $\mathbf{S}$ by an equivalence class of deformations $[\phi] = \{\tilde{\phi} \,|\, \tilde{\phi}(\hat{\mathbf{S}}) = \mathbf{S}\}$. A family $(\phi(t))_{t\in[0,1]}$ of such deformations is associated with an Eulerian velocity field $v(t) = \dot{\phi}(t) \circ \phi^{-1}(t)$, and shape variations are equivalence classes of such motion fields with $\tilde{v} \sim v$ if $\tilde{v} \cdot n = v \cdot n$ on $\partial\mathbf{S}$, where n is the outer normal on $\partial\mathbf{S}$. Now, a physically motivated metric on shape variations is given by the minimal rate of dissipation in a Newtonian fluid occupying $\mathbf{S}$ when its free boundary moves according to the shape variation,

$$g_{\mathbf{S}}(v, v) = \min_{\{\tilde{v} \,|\, \tilde{v}\cdot n = v\cdot n \text{ on } \partial\mathbf{S}\}} \int_{\mathbf{S}} \tfrac{\lambda}{2}(\mathrm{tr}\epsilon[\tilde{v}])^2 + \mu \mathrm{tr}(\epsilon[\tilde{v}]^2)\,\mathrm{d}x\,, \tag{6}$$

where $\lambda,\ \mu > 0$, $\epsilon[v] = \frac{1}{2}(\nabla v + \nabla v^T)$ ($\lambda = \mu = 1$ in the examples). In this context, a matching functional $\mathbf{W}$ to approximate the resulting squared Riemannian distance locally can be defined via the minimization of a deformation energy $\hat{\mathbf{W}}[\mathbf{S}, \phi] = \int_{\mathbf{S}} W(D\phi)\,\mathrm{d}x$ over all matching deformations ϕ with $\phi(\mathbf{S}) = \tilde{\mathbf{S}}$, i. e. $\mathbf{W}[\mathbf{S}, \tilde{\mathbf{S}}] = \min_{\phi(\mathbf{S})=\tilde{\mathbf{S}}} \hat{\mathbf{W}}[\mathbf{S}, \phi]$ for some particular deformation energy density W. There are primarily two options to choose a W which both ensure the requested consistency of the functional $\mathbf{W}$ and the metric g, i. e. $\frac{1}{2}D^2_\phi \hat{\mathbf{W}}[\mathbf{S}, \mathrm{id}] = g_{\mathbf{S}}$. One could consider a simple linearized model with

$$W^l(D\phi) = \tfrac{\lambda}{2}(\mathrm{tr}\epsilon[\phi - \mathrm{id}])^2 + \mu \mathrm{tr}(\epsilon[\phi - \mathrm{id}]^2)\,. \tag{7}$$

The advantage of W^l is that it is quadratic in ϕ so that the Euler–Lagrange equations for a functional composed of such energy densities will be linear. However, this is at the expense of the resulting energy being rigid body motion invariant only in an infinitesimal sense so that a relatively large number K of time steps is required in (1) to obtain an approximate frame indifference of discrete geodesic paths. Full rigid body motion invariance for large deformations can be guaranteed only for certain classes of nonlinear models $W = W^{nl}$. A specifically useful example is the energy density

$$W^{nl}(D\phi) = \frac{\mu}{2}\mathrm{tr}(D\phi^T D\phi) + \frac{\lambda}{4}\det(D\phi)^2 - \left(\mu + \frac{\lambda}{2}\right)\log\det D\phi - \frac{d\mu}{2} - \frac{\lambda}{4}\,. \tag{8}$$

We refer to [20] for further details on this approach.

In our application of discrete geodesic regression the computationally most demanding task is the continual computation of the shapes $\tilde{\mathbf{S}}_k$ for the penalty terms $\mathbf{W}[\tilde{\mathbf{S}}_k, \mathbf{S}_k]$. Therefore it turns out to be favorable to use W^l in the definition of the $\tilde{\mathbf{S}}_k$ and the penalty functional, i. e. we choose

$$\mathbf{W}[\mathbf{S}, \tilde{\mathbf{S}}] := \min_{\phi(\mathbf{S})=\tilde{\mathbf{S}}} \hat{\mathbf{W}}^l[\mathbf{S}, \phi] \quad \text{with} \quad \hat{\mathbf{W}}^l[\mathbf{S}, \phi] = \int_{\mathbf{S}} W^l(D\phi)\,\mathrm{d}x \tag{9}$$

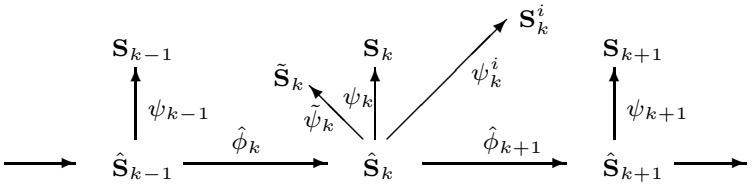


Fig. 4. A diagram illustrating the parametrization of the shapes $\mathbf{S}_k$ via deformations over reference shapes $\hat{\mathbf{S}}_k$. Here, $\hat{\phi}_k$ are given deformations such that $\hat{\mathbf{S}}_k = \hat{\phi}_k(\hat{\mathbf{S}}_{k-1})$.

in (4) and (5) and a large number of time steps K leading to a sufficiently small time step size $\tau = \frac{1}{K}$. The condition on $\tilde{\mathbf{S}}_k$ in (4) thus reads as

$$\tilde{\mathbf{S}}_k = \operatorname*{argmin}_{\mathbf{S}} \left(\min_{\phi(\mathbf{S}_{k-1})=\mathbf{S}} \hat{\mathbf{W}}^l[\mathbf{S}_{k-1}, \phi] + \min_{\phi(\mathbf{S})=\mathbf{S}_{k+1}} \hat{\mathbf{W}}^l[\mathbf{S}, \phi] \right) . \tag{10}$$

The treatment of the dissimilarity measure is computationally less critical and we would like to treat large scale variations between shapes of the discrete geodesic $\mathbf{S}_k$ and corresponding input shapes $\mathbf{S}_k^i$. Hence, here we take into account the nonlinear deformation model and define

$$\operatorname{diss}[\mathbf{S}_k, \mathbf{S}_k^i] := \min_{\phi(\mathbf{S}_k)=\mathbf{S}_k^i} \hat{\mathbf{W}}^{nl}[\mathbf{S}_k, \phi] \;\text{ with }\; \hat{\mathbf{W}}^{nl}[\mathbf{S}, \phi] = \int_{\mathbf{S}} W^{nl}(D\phi)\,\mathrm{d}x\,. \tag{11}$$

Finally, to render the method computationally feasible, we assume all deformations to be defined on a computational domain D ($D = [0,1]^d$ in the examples) containing all shapes under consideration, and we suppose that the material properties outside any of the shapes are by a factor δ softer than inside the shapes ($\delta = 0.01$ in the examples). Thus, we replace $\hat{\mathbf{W}}^{nl/l}[\mathbf{S}, \phi]$ by $\hat{\mathbf{W}}_\delta^{nl/l}[\mathbf{S}, \phi] := \int_D ((1-\delta)\chi_{\mathbf{S}} + \delta) W^{nl/l}(D\phi)\,\mathrm{d}x$ (where the superscript l or nl identifies the linear and the nonlinear model, respectively).

6 The Optimization Algorithm

Here we briefly describe the main algorithmic ingredients to minimize (5) in the context of the space of viscous fluidic objects. The presented variational approach for shape regression is based on functionals which depend on shapes and deformations defined on these shapes. To render the method computationally feasible, we parametrize shapes via deformations acting on reference shapes and work solely with deformations as unknowns. Then, we consider a gradient descent algorithm for the PDE constrained variational approach. In detail we proceed as follows:

Parametrizing shapes via deformations. The shapes $\mathbf{S}_k$ of the discrete geodesic as well as the constraint shapes $\tilde{\mathbf{S}}_k$ are represented via deformations ψ_k and $\tilde{\psi}_k$ of fixed reference shapes $\hat{\mathbf{S}}_k$, i. e. $\mathbf{S}_k = \psi_k(\hat{\mathbf{S}}_k)$ and $\tilde{\mathbf{S}}_k = \tilde{\psi}_k(\hat{\mathbf{S}}_k)$ (cf. Fig. 4). In addition, we consider deformations ψ_k^i on $\hat{\mathbf{S}}_k$ with $\mathbf{S}_k^i = \psi_k^i(\hat{\mathbf{S}}_k)$ for $i = 1, \ldots, i_k$.

We can now rephrase the deformations $\tilde{\phi}_k$ from the $\mathbf{S}_k$ onto the $\tilde{\mathbf{S}}_k$ (which occur in (5) via (9)) and the deformations $(\phi_k^i)_{k=1,\dots,K}^{i=1,\dots,i_k}$ from the shapes $\mathbf{S}_k$ onto the input shapes (which are introduced in (11) and used in $\mathbf{F}[\mathcal{S}_K]$) in terms of the above parametrizing deformations, i. e. $\phi_k^i = \psi_k^i \circ \psi_k^{-1}$, $\tilde{\phi}_k = \tilde{\psi}_k \circ \psi_k^{-1}$.

Approximation of the matching condition. To ensure that ϕ_k^i matches (at least approximately) $\mathbf{S}_k$ onto the input shape $\mathbf{S}_k^i$ we employ a penalty functional $P[\hat{\mathbf{S}}_k, \mathbf{S}_k^i, \psi_k^i] := \int_D |\chi_{\hat{\mathbf{S}}_k} - \chi_{\mathbf{S}_k^i} \circ \psi_k^i|^2 \,\mathrm{d}\hat{x}$ weighted with $\frac{C}{\eta}$ ($\eta = 0.1$ in the examples) and added to the total energy for all $k = 0, \dots, K$ and $i = 1, \dots, i_k$.

Expressing the energy in terms of the parametrizing deformations. To simplify notation, denote by $\Psi := (\psi_k, (\psi_k^i)_{i=1,\dots,i_k})_{k=0,\dots,K}$ the vector of all deformations which are considered as our actual degrees of freedom and by $\tilde{\Psi} := (\tilde{\psi}_1, \dots, \tilde{\psi}_{K-1})$ the vector of all constraint deformations. The energy $\mathbf{F}^\epsilon[\mathcal{S}_K]$ over which we minimize in (5) is rewritten as an energy $\hat{\mathbf{H}}^\epsilon_{\delta\eta}[\Psi, \tilde{\Psi}]$ of Ψ and $\tilde{\Psi}$,

$$\begin{aligned}\hat{\mathbf{H}}^\epsilon_{\delta\eta}[\Psi, \tilde{\Psi}] = &\sum_{k=0}^{K} \left(\sum_{i=1}^{i_k} \left(\hat{\mathbf{W}}^{nl}_\delta[\psi_k(\hat{\mathbf{S}}_k), \psi_k^i \circ \psi_k^{-1}] + \frac{C}{\eta} \mathbf{P}[\hat{\mathbf{S}}_k, \mathbf{S}_k^i, \psi_k^i] \right) \right) \\ &+ \frac{4CK^3}{\epsilon} \sum_{k=1}^{K-1} \hat{\mathbf{W}}^l_\delta[\psi_k(\hat{\mathbf{S}}_k), \tilde{\psi}_k \circ \psi_k^{-1}] \,. \end{aligned} \tag{12}$$

By the transformation rule we obtain the following computationally more efficient reformulation of the involved deformation energies

$$\hat{\mathbf{W}}^{nl/l}_\delta[\psi_k(\hat{\mathbf{S}}_k), \psi \circ \psi_k^{-1}] = \int_D ((1-\delta)\chi_{\hat{\mathbf{S}}_k} + \delta) W^{nl/l}\big(D\psi (D\psi_k)^{-1}\big) \det D\psi_k \,\mathrm{d}\hat{x} \tag{13}$$

for $\psi = \psi_k^i$ or $\psi = \tilde{\psi}_k$, respectively.

Realization of the constraint. We aim at minimizing $\mathbf{H}^\epsilon_{\delta\eta}$ subject to constraint (10). I. e. for all $k = 1, \dots, K-1$ the components $\tilde{\psi}_k$ of the vector of deformations $\tilde{\Psi}$ describe the middle shape of a discrete 3-shape geodesic with end shapes $\mathbf{S}_{k-1} = \psi_{k-1}(\hat{\mathbf{S}}_{k-1})$ and $\mathbf{S}_{k+1} = \psi_{k+1}(\hat{\mathbf{S}}_{k+1})$. We now slightly modify this constraint by requiring $\tilde{\psi}_k$ to be the minimizer of $\hat{\mathbf{W}}^l_\delta[\psi_{k-1}(\hat{\mathbf{S}}_{k-1}), \psi \circ \hat{\phi}_k \circ \psi_{k-1}^{-1}] + \hat{\mathbf{W}}^l_\delta[\psi(\hat{\mathbf{S}}_k), \psi_{k+1} \circ \hat{\phi}_{k+1} \circ \psi^{-1}]$ over all deformations ψ, where $\hat{\phi}_k$ are given matching deformations between the reference shapes with $\hat{\mathbf{S}}_k = \hat{\phi}_k(\hat{\mathbf{S}}_{k-1})$ (cf. Fig. 4). This is not quite the same as (10), not just because of the additional regularization, but also since here the point-to-point correspondence $\psi_{k+1} \circ \hat{\phi}_{k+1} \circ \hat{\phi}_k \circ \psi_{k-1}^{-1}$ between $\mathbf{S}_{k-1}$ and $\mathbf{S}_{k+1}$ is imposed along the 3-shape geodesic, which was not the case originally. Nevertheless, in the limit for $K \to \infty$ and $\epsilon \to 0$ the discrete path experimentally converges to a continuous geodesic (cf. Fig. 6 and 9).

A computationally advantageous symmetrization of the constraint. To fully exploit the quadratic deformation energy in the context of the above-mentioned constraint it is advantageous to further replace $\hat{\mathbf{W}}^l_\delta[\psi(\hat{\mathbf{S}}_k), \psi_{k+1} \circ \hat{\phi}_{k+1} \circ \psi^{-1}]$ by

$$\hat{\mathbf{W}}_\delta^{-l}[\psi_{k+1}(\hat{\mathbf{S}}_{k+1}), \psi \circ \hat{\phi}_{k+1}^{-1} \circ \psi_{k+1}^{-1}]$$
$$= \int_D ((1-\delta)\chi_{\hat{\mathbf{S}}_k} + \delta) W^l(D\psi(D\hat{\phi}_{k+1})^{-1}(D\psi_{k+1})^{-1} \circ \hat{\phi}_{k+1}) \cdot \det D\hat{\phi}_{k+1}(\det D\psi_{k+1}) \circ \hat{\phi}_{k+1} \mathrm{d}\hat{x}$$

which is quadratic in ψ and in effect replaces the relaxation of the energy $\min_{\phi(\mathbf{S})=\mathbf{S}_{k+1}} \hat{\mathbf{W}}^l[\mathbf{S}, \phi]$ in (10) by the relaxation of a similar energy based on the inverse deformation $\min_{\phi(\mathbf{S})=\mathbf{S}_{k+1}} \hat{\mathbf{W}}^l[\mathbf{S}_{k+1}, \phi^{-1}]$. For our applications (e. g. Fig. 6) we experimentally validated that for this computationally motivated modification the resulting discrete curves converge towards discrete geodesics. Altogether we obtain the following variational definition,

$$\tilde{\psi}_k := \operatorname*{argmin}_{\psi} \mathbf{G}[\psi_{k-1}, \psi, \psi_{k+1}] \tag{14}$$

for all $k = 1, \ldots, K-1$, where

$$\mathbf{G}[\psi_{k-1}, \psi, \psi_{k+1}] := \hat{\mathbf{W}}_\delta^l[\psi_{k-1}(\hat{\mathbf{S}}_{k-1}), \psi \circ \hat{\phi}_k \circ \psi_{k-1}^{-1}] + \hat{\mathbf{W}}_\delta^{-l}[\psi_{k+1}(\hat{\mathbf{S}}_{k+1}), \psi \circ \hat{\phi}_{k+1}^{-1} \circ \psi_{k+1}^{-1}] + \nu \hat{\mathbf{W}}^l[D, \psi]\,.$$

Here, we added $\nu\hat{\mathbf{W}}^l[D, \psi]$ as regularizer with $\nu \sim 10^{-2}h$ to ensure that not only the compositions of deformations are regular but also the deformation ψ.

A gradient descent scheme. We apply a standard Fletcher–Reeves nonlinear conjugate gradient descent to the above minimization problem, which at each step requires evaluation of the functional $\hat{\mathbf{F}}_{\delta\eta}^\epsilon[\Psi] := \hat{\mathbf{H}}_{\delta\eta}^\epsilon[\Psi, \tilde{\Psi}[\Psi]]$ and its gradient. For the functional evaluation, the quadratic optimization problem (14) is solved for each $k = 1, \ldots, K-1$ by a preconditioned linear conjugate gradient iteration. Using the standard adjoint method in constrained optimization [23] as in Section 4 we obtain the Gâteaux derivative of $\hat{\mathbf{F}}_{\delta\eta}^\epsilon[\Psi]$ (which incorporates constraint (14)) as

$$\partial_{\psi_k} \hat{\mathbf{F}}_{\delta\eta}^\epsilon[\Psi] = \left(\partial_{\psi_k} \hat{\mathbf{H}}_{\delta\eta}^\epsilon\right)[\Psi, \tilde{\Psi}[\Psi]] - \left(\partial_{\psi_k}\partial_{\tilde{\psi}_k} \hat{\mathbf{H}}_{\delta\eta}^\epsilon\right)[\Psi, \tilde{\Psi}[\Psi]]\, p_k\,, \tag{15}$$

where p_k is defined as the solution of the dual problem

$$D_\psi^2 \mathbf{G}[\psi_{k-1}, \tilde{\psi}_k, \psi_{k+1}]\, p_k = \left(\partial_{\tilde{\psi}_k} \hat{\mathbf{H}}_{\delta\eta}^\epsilon\right)[\Psi, \tilde{\Psi}[\Psi]]\,. \tag{16}$$

Different from the nonlinear energy $\hat{\mathbf{W}}^{nl}$, which is strictly rigid body motion invariant, the quadratic energy $\hat{\mathbf{W}}^l$ is rigid body motion invariant solely in an infinitesimal sense. Thus, in the case of large shape variability in the input data it turned out to be appropriate to enforce the preservation of the center of mass, $\int_D ((1-\delta)\chi_{\hat{\mathbf{S}}} + \delta)\psi \,\mathrm{d}\hat{x} = 0$, and the preservation of the angular momentum, $\int_D ((1-\delta)\chi_{\hat{\mathbf{S}}} + \delta)\left(D\psi - D\psi^T\right) \mathrm{d}\hat{x} = 0$, assuming the input shapes are co-aligned with respect to the zero moment and the direction of the first moment. This is implemented as an additional set of linear constraints in the minimization in (14) and in the outer minimization with respect to the energy term $\mathbf{W}_\delta^{nl}[\hat{\mathbf{S}}_k, \psi_k]$. The latter is realized by a projective gradient descent

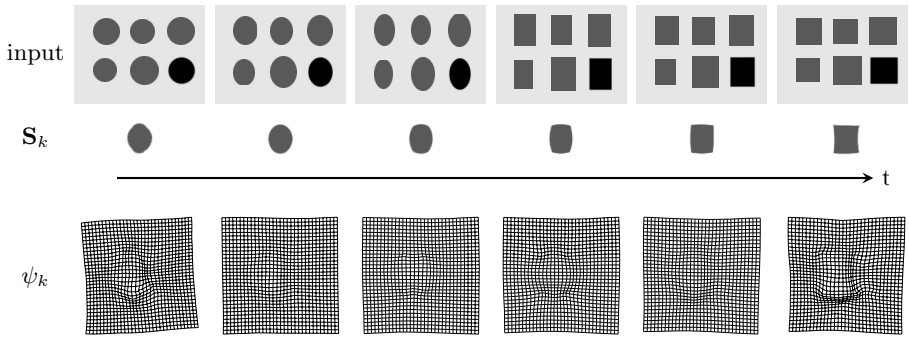


Fig. 5. Top: regression input shapes $(\mathbf{S}_k^i)_{k=0,\dots,5}^{i=1,\dots,5}$ (grey) and reference shapes $(\hat{\mathbf{S}}_k)_{k=0}^5$ (black), middle: discrete regression curve $(\mathbf{S}_k)_{k=0}^5$, bottom: the deformations $(\psi_k)_{k=0}^5$

scheme, the former by a Lagrange multiplier approach. To emphasize the qualitative behavior of geodesic regression we have computed the regression curve for a very basic test case. Fig. 5 displays a discrete geodesic and the associated deformations acting on the reference shapes, and Fig. 6 underlines that the resulting regression curves are actually very close to discrete geodesics. For the spatial discretization we employ multilinear finite elements on the computational domain $D = [0,1]^d$, which is overlaid with a regular square grid of $2^n + 1$, $n \in \mathbb{N}$, nodes in each direction. Energies are computed via Simpson quadrature on each element. Furthermore, we apply a cascadic approach, first computing the regression curve with a coarse spatial resolution for all involved deformations and then proceeding iteratively from coarse to fine. In this cascadic approach one can also adopt the reference shapes $\hat{\mathbf{S}}_k$ and the associated reference deformations $\hat{\phi}_k$ starting on the coarsest level with a single reference shape chosen as one of the coarsely resolved end shapes.

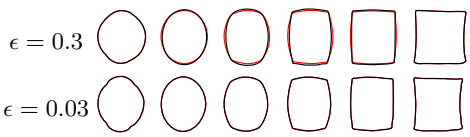


Fig. 6. Difference between the discrete regression curve (black) and the true discrete geodesic (red) connecting its end shapes for different ϵ.

7 Applications in Botany and Anatomy

We applied the geodesic regression approach to the statistical analysis of the aging of the human corpus callosum and to the growth of sugar beet roots over a vegetation period. Fig. 7 shows the discrete regression curve for the corpus callosum input shapes, which clearly reflects a substantial thinning of the structure with increasing age (cf. the results in [9] on a similar data set). Fig. 8 presents results obtained for 2D slices of sugar beet roots. Here, we also show the effect of an increasing number of intermediate shapes along the regression curve.

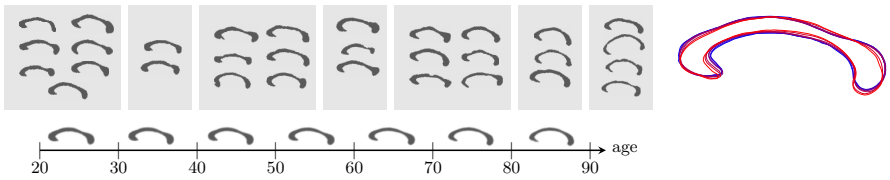


Fig. 7. Discrete geodesic regression curve for 31 shapes representing slices of the corpus callosum of humans at different age (2^{nd} to 8^{th} decade). On the right the 7 contours are overlayed showing a thinning process with increasing age (blue to red).

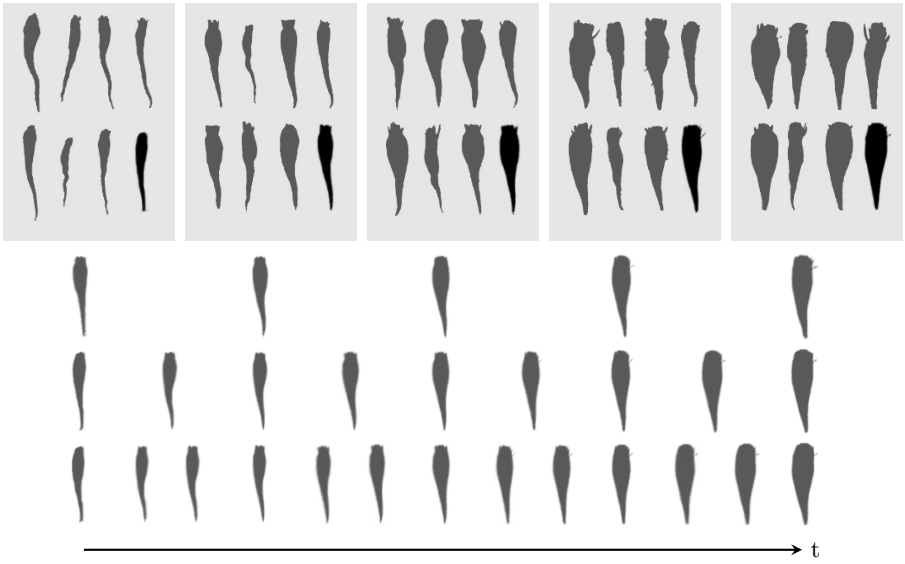


Fig. 8. From top to bottom: 2D input slices of sugar beets at five time points (grey) and shape averages for each time (black), regression curves for $\epsilon = 0.08$ and $K = 4, 8, 12$

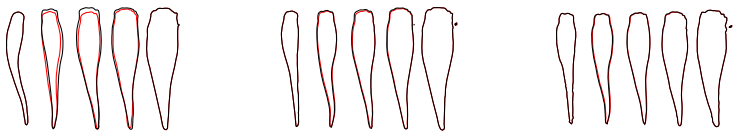


Fig. 9. Difference between the discrete regression curve (black) and the true discrete geodesic (red) connecting the end shapes for three different values of ϵ

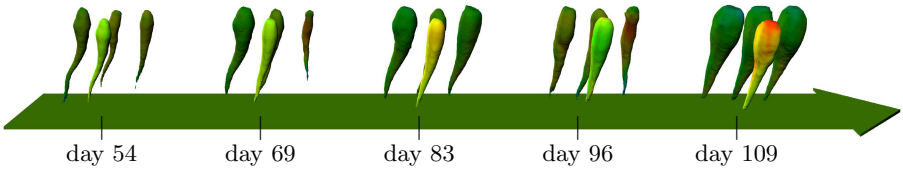


Fig. 10. For the discrete sugar beet regression curve (cf. Fig. 1) the shapes $\mathbf{S}_k$ in the front are color coded according to the signed distance from the shape average at each time point and the input shapes in the back according to the distance from the regression shape in front (colorcode: -0.127 — 0.061, maximal root height ≈ 0.75).

Fig. 9 once more demonstrates that already for moderately small penalty parameter ϵ the resulting curves are actually very close to a discrete geodesic. Finally, Fig. 10 shows some quantitative analysis of the regression curve for sugar beets in 3D.

8 Conclusions

We have described a time discrete geodesic regression approach on manifolds and applied it to the shape space of viscous fluidic objects. The method requires the solution of a PDE constrained optimization problem for deformations defined on a family of reference shapes. Applications in the context of plant growth and anatomical brain structures demonstrate the method's potential in time dependent shape statistics. A thorough convergence analysis for $K \to \infty$ and $\epsilon \to 0$ following the direction in [22] is still open. The current implementation based on a gradient descent approach requires several hours to compute a regression geodesic for 2D shapes and several days for 3D shapes. There is a great potential for faster energy relaxation using Newton type methods and more efficient parallel implementations. Furthermore, it would be interesting to investigate the generalization to more general classes of regression curves and the application to other shape spaces.

Acknowledgement. The 3D volumetric data of sugar beets acquired by 4.7T MRI are courtesy of R. Metzner (Forschungszentrum Jülich). The corpus callosum shape data was derived from MRI from the OASIS brain database (`www.oasis-brains.org`).

References

1. Klassen, E., Srivastava, A., Mio, W., Joshi, S.H.: Analysis of planar shapes using geodesic paths on shape spaces. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(3), 372–383 (2004)
2. Kilian, M., Mitra, N.J., Pottmann, H.: Geometric modeling in shape space. ACM Transactions on Graphics 26, #64, 1–8 (2007)

3. Beg, M.F., Miller, M., Trouvé, A., Younes, L.: Computational anatomy: Computing metrics on anatomical shapes. In: Proceedings of 2002 IEEE ISBI, pp. 341–344 (2002)
4. Miller, M.I., Trouvé, A., Younes, L.: The metric spaces, Euler equations, and normal geodesic image motions of computational anatomy. In: Proceedings of the 2003 International Conference on Image Processing, vol. 2, pp. 635–638. IEEE (2003)
5. Srivastava, A., Klassen, E., Joshi, S.H., Jermyn, I.H.: Shape analysis of elastic curves in euclidean spaces. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(7), 1415–1428 (2011)
6. Fletcher, P., Lu, C., Pizer, S., Joshi, S.: Principal geodesic analysis for the study of nonlinear statistics of shape. IEEE Transactions on Medical Imaging 23(8), 995–1005 (2004)
7. Davis, B., Fletcher, P.T., Bullitt, E., Joshi, S.: Population shape regression from random design data. In: Proceedings of IEEE International Conference on Computer Vision (2007)
8. Miller, M., Trouvé, A., Younes, L.: On the metrics and Euler-Lagrange equations of computational anatomy. Annual Review of Biomedical Enginieering 4, 375–405 (2002)
9. Fletcher, P.T.: Geodesic regression on Riemannian manifolds. In: MICCAI Workshop on Mathematical Foundations of Computational Anatomy (MFCA), pp. 75–86 (2011)
10. Niethammer, M., Huang, Y., Vialard, F.-X.: Geodesic regression for image time-series. In: Fichtinger, G., Martel, A., Peters, T. (eds.) MICCAI 2011, Part II. LNCS, vol. 6892, pp. 655–662. Springer, Heidelberg (2011)
11. Hong, Y., Joshi, S., Sanchez, M., Styner, M., Niethammer, M.: Metamorphic geodesic regression. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) MICCAI 2012, Part III. LNCS, vol. 7512, pp. 197–205. Springer, Heidelberg (2012)
12. Trouvé, A., Vialard, F.X.: Shape splines and stochastic shape evolutions: a second order point of view. Quart. Appl. Math. 70(2), 219–251 (2012)
13. Srivastava, A., Jain, A., Joshi, S., Kaziska, D.: Statistical shape models using elastic-string representations. In: Narayanan, P.J., Nayar, S.K., Shum, H.-Y. (eds.) ACCV 2006. LNCS, vol. 3851, pp. 612–621. Springer, Heidelberg (2006)
14. Michor, P.W., Mumford, D.: Riemannian geometries on spaces of plane curves. J. Eur. Math. Soc. 8, 1–48 (2006)
15. Sundaramoorthi, G., Yezzi, A., Mennucci, A.: Sobolev active contours. International Journal of Computer Vision 73(3), 345–366 (2007)
16. Younes, L., Michor, P.W., Shah, J., Mumford, D.: A metric on shape space with explicit geodesics. Atti Accad. Naz. Lincei Cl. Sci. Fis. Mat. Natur. Rend. Lincei (9) Mat. Appl. 19(1), 25–57 (2008)
17. Sundaramoorthi, G., Mennucci, A., Soatto, S., Yezzi, A.: A new geometric metric in the space of curves, and applications to tracking deforming objects by prediction and filtering. SIAM Journal on Imaging Sciences 4(1), 109–145 (2011)
18. Dupuis, D., Grenander, U., Miller, M.: Variational problems on flows of diffeomorphisms for image matching. Quarterly of Applied Mathematics 56, 587–600 (1998)
19. Fuchs, M., Jüttler, B., Scherzer, O., Yang, H.: Shape metrics based on elastic deformations. J. Math. Imaging Vis. 35(1), 86–102 (2009)

20. Wirth, B., Bar, L., Rumpf, M., Sapiro, G.: A continuum mechanical approach to geodesics in shape space. International Journal of Computer Vision 93(3), 293–318 (2011)
21. Rumpf, M., Wirth, B.: A nonlinear elastic shape averaging approach. SIAM Journal on Imaging Sciences 2(3), 800–833 (2009)
22. Rumpf, M., Wirth, B.: Variational time discretization of geodesic calculus (2012), http://de.arxiv.org/abs/1210.2097
23. Fletcher, R.: Practical Methods of Optimization, 2nd edn. John Wiley & Sons (1987)

Object Segmentation by Shape Matching with Wasserstein Modes

Bernhard Schmitzer and Christoph Schnörr

Image and Pattern Analysis Group, Heidelberg University

Abstract. We gradually develop a novel functional for joint variational object segmentation and shape matching. The formulation, based on the Wasserstein distance, allows modelling of local object appearance, statistical shape variations and geometric invariance in a uniform way. For learning of class typical shape variations we adopt a recently presented approach and extend it to support inference of deformations *during* segmentation of new query images. The resulting way of describing and fitting trained shape variations is in style reminiscent of contour-based variational shape priors, but does not require an intricate conversion between the contour and the region representation of shapes. A well-founded hierarchical branch-and-bound scheme, based on local adaptive convex relaxation, is presented, that provably finds the global minimum of the functional.

Keywords: Wasserstein distance, non-convex optimization, convex relaxation.

1 Introduction

Object segmentation and shape matching are fundamental problems in image processing and computer vision that underlie many high-level approaches to understanding the content of an image. They are intimately related: segmentation of the foreground object is a prerequisite for shape matching in a sequential analysis of an image. On the other hand, if performed simultaneously, matching can guide the segmentation process by supplying additional information about the object shape in a noisy environment where unsupervised segmentation would fail. Naturally, joint application is a more complicated problem.

We propose a new functional for simultaneous segmentation and statistical model-based shape matching within a single variational approach. The mathematical framework allows to *combine* key concepts - appearance modelling, modelling and description of deformable regions or contours, geometric invariance - in a uniform way. We rely on convex relaxation and a hierarchical branch-and-bound scheme for global optimization.

1.1 Related Literature

Wasserstein Distance and Image Registration. Optimal transport is a popular tool for object matching and image registration [3,10] due to its

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 123–136, 2013.
© Springer-Verlag Berlin Heidelberg 2013

favourable properties: Choosing the cost function to be the squared Euclidean distance between pixels gives access to the rich theory of the Monge formulation of optimal transport. Also, thanks to the linear programming relaxation due to Kantorovich such problems usually involve convex functionals. On the other hand, directly converting grey values to mass densities, as often done [3], is not robust to noise. A naïve extension to noise handling will only work if the query and the reference image are aligned properly (see e.g. [7] for an attempt to alleviate these restrictions). Additionally, when measuring the similarity between two objects via plain optimal transport, their distance will exclusively be determined by the resulting optimal transport cost. *During* the registration process there is no way to benefit from prior knowledge to distinguish common and uncommon types of deformations. However, *after* having computed the registrations, there are promising ideas how to extract and analyze information on the deformations from the optimal registrations [10]. The observed deformation fields are viewed as elements of the tangent space of a reference shape where then standard machine learning techniques (e.g. PCA) can be applied.

Variational Image Segmentation and Contour Based Shape Priors. For object segmentation variational approaches with shape priors, based on contour spaces have received a lot of attention [1,2]. The manifold of shapes, described by closed contours has been studied extensively [8,5]. Again, here working in the tangent space of a reference shape enables application of machine learning tools to learn object typical deformations from training data.

However, the map between the contour representation of a shape and the region representation by its indicator function is mathematically complex. Thus, when combined with region based variational segmentation functionals, the contour based shape priors tend to yield highly non-convex functionals that rely on a good initialization to give useful, only locally optimal, results (e.g. [2]). There are approaches to model shape statistics directly on the set of indicator functions, yielding overall convex functionals [4,6]. But due to required convexity, these shape representations are rather simplistic and lack important features such as geometric invariance.

1.2 Contribution

In this paper we propose a new functional for noise robust joint object segmentation and shape matching based on the Wasserstein distance. We start with a functional for variational segmentation where we regularize the segmentation with the Wasserstein distance to a reference measure. This functional has several limitations, hinted at above and further discussed in Section 3.1. To overcome these, we enhance the functional by additional degrees of freedom, obtaining an advantageous new approach:

(i) The optimal segmentation & matching become invariant under translations and approximately invariant w.r.t. rotations and scale transformations.
(ii) Prior knowledge on object-typical deformations can be learned from training samples and exploited *during* the registration process. Although the

mathematical representation is different, the scope of our approach is similar to that of contour-based shape priors.

(iii) The overall functional is non-convex. Yet, instead of heuristic local optimization we propose a hierarchical branch-and-bound scheme to obtain global optimizers. We show how bounds can be obtained by adaptive convex relaxation that becomes tighter as the hierarchy-scale becomes finer and how successive refinement of the bound computations converges towards the global optimum (Propositions 1 and 2).

Organization. After a brief review of the mathematical background in the next section, we will gradually motivate and develop the full functional in Sec. 3. Global optimization of the non-convex functional is discussed in Sec. 4, key properties of the approach are illustrated with experiments in Sec. 5, before we reach a short conclusion at the end.

2 Mathematical Background: Wasserstein Distance

For any measurable space A denote by $\mathcal{P}(A)$ the set of non-negative measures thereon. For two measurable spaces A, B, a measure μ on A and a measurable map $f : A \to B$, the push-forward $f_\sharp\,\mu$ of μ onto B via f is defined by $f_\sharp\,\mu(\sigma) = \mu(f^{-1}(\sigma))$ for all measurable $\sigma \subset B$.

Let X be a measurable space with measures $\mu, \nu \in \mathcal{P}(X)$ of the same total mass. Then the set of *couplings* $\mathit{\Pi}(\mu, \nu)$ between μ and ν is given by

$$\begin{aligned}\mathit{\Pi}(\mu,\nu) = \{&\pi \in \mathcal{P}(X \times X) :\ \pi(\sigma \times X) = \mu(\sigma),\ \pi(X \times \sigma) = \nu(\sigma) \\ &\text{for all measurable } \sigma \subset X\}\,. \end{aligned} \tag{1}$$

For a metric $d_X : X \times X \to \mathbb{R} \cup \{\infty\}$ the Wasserstein distance is determined by

$$D(\mu,\nu) = \left(\inf_{\pi \in \mathit{\Pi}(\mu,\nu)} \left\{ \int_{X\times X} d_X^2(x,y)\, d\pi(x,y) :\ \pi \in \mathit{\Pi}(\mu,\nu) \right\} \right)^{1/2}. \tag{2}$$

The minimizing coupling π, determining the Wasserstein distance D is called *optimal transport* in the literature [9].

For absolutely continuous measures on $X = \mathbb{R}^n$, metrized with the Wasserstein distance w.r.t. the Euclidean metric, the optimal $\pi \in \mathit{\Pi}(\mu, \nu)$ is induced by a unique map $\varphi : X \to X$, i.e. $\pi = (\mathrm{id}, \varphi)_\sharp\,\mu$. That is, at any point x, the mass of μ is transported to the unique location $\varphi(x)$. Further, these measures constitute a Riemannian manifold. When φ is the optimal transport map between μ and ν then the vector field $t(x) = \varphi(x) - x$ corresponds to a vector in the tangent space of μ. For two vectors t_1, t_2 in the tangent space the inner product is given by $\langle t_1, t_2 \rangle_\mu = \int_X \langle t_1(x), t_2(x) \rangle_{\mathbb{R}^n}\, d\mu(x)$.

We adopt the idea from [10] to use PCA on the tangent space to learn typical object deformations. However, in our approach we will be able to benefit from this learned knowledge during segmentation/matching of new images.

3 Variational Approach

3.1 Problem Setup and Naïve Approach

We will now, step by step, describe the problem to be solved, set out the notation and develop the final form of our proposed functional.

Let Y be the image domain which we want to separate into fore- and background. We can describe the separation by an indicator function $u : Y \to \{0, 1\}$. To obtain feasible optimization problems, one typically relaxes the constraint that u must be binary to the interval $[0, 1]$. In this paper we use optimal transport as a regularizer. Therefore we interpret the relaxed function u as the density of a measure ν. For simplicity we will define our functionals directly over the set of measures, drop u and translate the $[0, 1]$-constraint appropriately.

To find the optimal segmentation ν of Y we want to combine local information with prior knowledge on the shape of the sought-after object. This information is given by a reference measure μ on a template space X. Let both X and Y be embedded into $\mathbb{R}^2$, i.e. $X, Y \subset \mathbb{R}^2$. Considering the literature, one might be tempted to optimize ν w.r.t. a local data term and regularize by its Wasserstein distance to μ in $\mathbb{R}^2$. The optimal coupling between μ and the optimal ν can then be interpreted as a registration between the template and its counterpart in the image. A corresponding functional could look like this:

$$E_0(\nu) = \frac{1}{2} D(\mu, \nu)^2 + G(\nu) = \frac{1}{2} \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} \|x - y\|^2 \, d\pi(x, y) + G(\nu) \quad (3)$$

where G is the function that encodes local appearance information. An illustration of the functional is given in Fig. 1a. For the remainder of the paper we choose G to be linear in ν:

$$G(\nu) = \int_Y g(y) \, d\nu(y) \quad (4)$$

Here g is a function of local weights, $g(y) < 0$ ($g(y) > 0$) indicating foreground (background) affinity of point $y \in Y$. For the applicability of the framework presented in this paper, G can be any 1-homogeneous convex function.

An optimal segmentation is then described by an optimizer of E_0 w.r.t. the following feasible set:

$$\mathcal{S}(M) = \{\nu \in \mathcal{P}(Y) : \nu(Y) = M, \nu(\sigma) \leq \mathcal{L}(\sigma) \text{ for all measurable } \sigma \subset Y\}. \quad (5)$$

Here $\mathcal{L}$ denotes the Lebesgue measure on Y. This constraint is equivalent to the density of ν being a relaxed indicator function. $M = \mu(X)$ is the total mass of μ to ensure that the Wasserstein distance $D(\mu, \nu)$ is well defined for all feasible ν.

Limitations. In addition to the mass constraint, as discussed in the introduction, functional (3) has two major shortcomings. The first is the dependence of the optimal ν on the relative embedding of X and Y into the $\mathbb{R}^2$ plane. Assume both μ and ν were fixed. Then any optimal coupling $\pi \in \Pi(\mu, \nu)$ would be still be optimal after relative translation of X and Y (of course taking into account

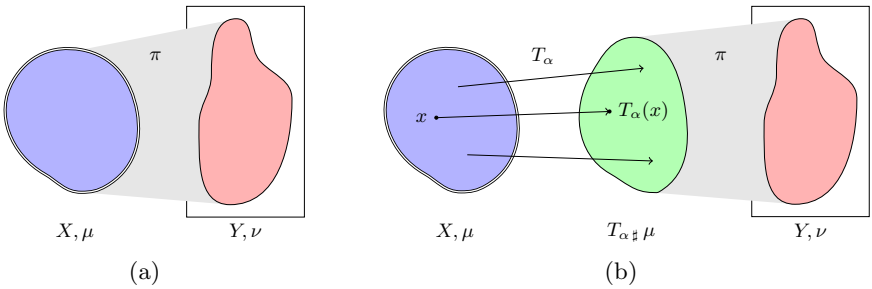


Fig. 1. Illustration of functionals $E_0(\nu)$, eq. (3), and $E_1(\alpha, \nu)$, eq. (8): (a) The segmentation in Y is described by measure ν which is regularized by the Wasserstein distance to a template measure μ, living on X. This simple approach introduces strong bias, depending on the relative location of X and Y, and lacks the ability to explicitly model typical object deformations. (b) In the enhanced functional the template measure μ is deformed by the map T_α, resulting in the push-forward $T_\alpha \sharp\, \mu$. The segmentation ν is then regularized by its Wasserstein distance to $T_\alpha \sharp\, \mu$. The corresponding optimal coupling π gives a registration between the foreground part of the image and the deformed template.

the coordinate transformation caused by the translation). However, since we do not consider ν to be fixed, as in other approaches, but optimize over ν, we cannot exploit this quasi-invariance. Any fixed embedding of X and Y will always introduce a bias, encouraging ν to have its mass close to μ, breaking translation invariance, which is clearly not what we want.

The second problem is that any deformation between μ and ν is uniformly penalized by its transportation distance. No information on more or less common deformations can be encoded.

To overcome these restrictions we propose to additionally optimize over the embedding of X into $\mathbb{R}^2$.

3.2 Wasserstein Modes

Let $T_\alpha : X \to \mathbb{R}^2$ be a family of functions, parametrized by some vector $\alpha \in \mathbb{R}^n$, used to adjust the position of X to obtain better matches between template and query image. We choose:

$$T_\alpha(x) = x + \sum_{i=1}^{n} \alpha_i \cdot t_i(x) \tag{6}$$

This linear decomposition will give us enough flexibility to deform X while keeping the resulting functionals easy to handle. We refer to the functions t_i as *modes*. They can be used to make the approach invariant w.r.t. translation, approximately invariant under rotation and scale and introduce prior knowledge

on learned object deformations into the functional. The enhanced version of (3) that we consider in this paper is:

$$\begin{aligned} E_1(\alpha,\nu) &= \frac{1}{2} D(T_\alpha \sharp \mu, \nu)^2 + F(\alpha) + G(\nu) \\ &= \frac{1}{2} \inf_{\pi \in \Pi(T_\alpha \sharp \mu, \nu)} \int_{X\times Y} \|x-y\|^2 \, d\pi(x,y) + F(\alpha) + G(\nu) \end{aligned} \tag{7}$$

Note that by a standard argument from measure theory we can rewrite this as

$$E_1(\alpha,\nu) = \frac{1}{2} \inf_{\pi \in \Pi(\mu,\nu)} \int_{X\times Y} \|T_\alpha(x)-y\|^2 \, d\pi(x,y) + F(\alpha) + G(\nu)\,. \tag{8}$$

F is a function that can introduce statistical knowledge on the distribution of the coefficients α. The functional E_1 is illustrated in Fig. 1b. For simplicity, in the course of this paper we choose

$$F(\alpha) = \frac{1}{2} \alpha^\top \Sigma^{-1} \alpha\,, \tag{9}$$

for some symmetric, positive semi-definite Σ^{-1}. We choose a basis in which $\Sigma^{-1} = \operatorname{diag}(\{\Sigma_i^{-1}\}_{i=1}^n)$ is diagonal. So coefficients with $\Sigma_i^{-1} = 0$ can move freely and and coefficients $\Sigma_i^{-1} > 0$ model i.i.d. Gaussian distributions $\alpha_i \sim \mathcal{N}(0, \Sigma_i^2)$.

The functional E_1 has an intuitive and transparent interpretation: With the coefficients α we describe a finite dimensional submanifold of known shapes, F modelling their plausibility. ν is the segmentation-measure, its local plausibility measured by G. $D(T_\alpha \sharp \mu, \nu)$ allows the optimal segmentation to be more flexible than the finite-dimensional submanifold given by the modes would allow, by measuring the distance of ν from the most plausible point on the manifold, and actually carrying out the corresponding assignment.

We will now discuss the choices for the modes t_i to model different types of variation in position and shape of X.

3.3 Geometric Invariance and Statistical Shape Deformation

Euclidean Isometries. If one chooses

$$t_{\mathrm{t}1}(x) = (1,0)^\top, \quad t_{\mathrm{t}2}(x) = (0,1)^\top \tag{10}$$

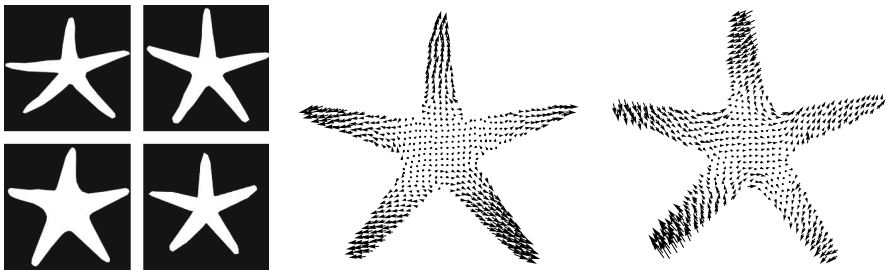
one can use the corresponding coefficients $\alpha_{\mathrm{t}1,\mathrm{t}2}$ to translate the template X and thus reintroduce translation invariance, that the simple approach (3) lacks. Further, let $R(\phi)$ be the 2-dimensional rotation matrix by angle ϕ. Then choose

$$t_{\mathrm{r}}(x) = \frac{d}{d\phi} R(\phi)\bigg|_{\phi=0} x = (-x_2, x_1)^\top \tag{11}$$

to approximately model rotation of X up to angles of about $\pm 30°$. For explicit invariance under these transformations one chooses $\Sigma_{\mathrm{t}1}^{-1} = \Sigma_{\mathrm{t}2}^{-1} = \Sigma_{\mathrm{r}}^{-1} = 0$.

Learning Class Typical Deformations. In this section we describe how modes $t_{\mathrm{d}i}$ can be learned that model class-typical shape variations from a set of training samples. These modes can then be used to allow X to be deformed in the learned way, to prefer known deformations over unknown deformations during the segmentation process.

Let $\{\mu_i\}_{i=1}^m$ be a set of training segmentations, given as indicator-measures: the support of μ_i marks the foreground of the corresponding segmentation. Assume that all μ_i have the same mass. We arbitrarily choose μ_1 to be the reference segmentation and compute the optimal transport couplings $\{\pi_{1,i}\}_{i=1}^k$ between the reference and the other segmentations, *optimized over rotation.* As discussed earlier, for fixed measures (μ_1, μ_i) the optimal coupling does not depend on the relative translation. It is easy to see that the relative translation with smallest cost is the one where the centers of mass coincide [10]. Note that the optimal coupling $\pi_{1,1}$ simply transports mass from all pixels onto themselves. The relative transportation maps that underlie the optimal couplings $\pi_{1,i}$ are then elements of the tangent space of the manifold of measures at μ_1. As in [10], we can then perform a classical principal component analysis (PCA) on the set of tangent vectors to obtain the *mean deformation* t_{m}, a set of *principal deformation modes* $\{t_{\mathrm{d}i}\}_i$ and the corresponding parameters Σ_i^{-1} for the statistical term. Together the pairs $(t_{\mathrm{d}i}, \Sigma_i^{-1})$ act like the well-known contour based shape priors. *However, in our approach no difficult conversion between different mathematical shape representations is necessary.*



(a) (b) (c)

Fig. 2. Learning class-typical deformation modes for starfish: (a)four ground-truth segmentations for learning (b) first principal component: modelling elongation of arms(c) second principal component: modelling angles between arms

The choice of a reference template is arbitrary and rather heuristic. In our application it is however not possible to take the average of all given samples as reference (as done in [10]) since we work with indicator measures and the mean would no longer be an indicator measure. For a proof of concept however, we consider this choice sufficient and it should be noted that the arbitrary choice of the reference measure is somewhat alleviated by the PCA where the mean of all observed transportation maps is determined. During segmentation the template μ will then be generated by the reference template μ_1, shifted by this mean.

Scale Transformation. The presented framework of deformation modes can, with some slight extensions, also be used to approximately model scale variations of the template.

Let us first have a look at how the map T_α transforms the measure μ via push-forward: μ has uniform mass density 1 within its support since it is an indicator measure (i.e. its density is an indicator function). At $T_\alpha(x)$ the density of the measure $T_{\alpha\,\sharp}\mu$ will depend on the Jacobian determinant $J_{T_\alpha}(x)$. For the decomposition (6) and small coefficients α we find:

$$J_{T_\alpha}(x) = 1 + \sum_{i=1}^{n} \alpha_i \cdot \operatorname{div} t_i(x) + \mathcal{O}(\alpha^2)$$

The rotation mode has zero divergence and since the deformation modes learned from the training samples map indicator measures onto similar indicator measures their divergence should also be small. Since the translation modes are constant, they have no influence on the Jacobian. Hence, the presented method to deform the reference template, is in fact a reasonable approximate description of a set of 'allowed' indicator measures.

Now for rescaling, the corresponding mode is $t_{\mathrm{s}}(x) = x$. The Jacobian determinant of $x \mapsto x + \alpha_{\mathrm{s}} \cdot t_{\mathrm{s}}(x)$ is $(1 + \alpha_{\mathrm{s}})^2$, hence to keep $T_{\alpha\,\sharp}\mu$ an indicator measure, we must multiply it by $(1+\alpha_{\mathrm{s}})^2$. To make the resulting functional scale invariant, we choose $\Sigma_{\mathrm{s}}^{-1} = 0$ and to rescale the functional by $(1 + \alpha_{\mathrm{s}})^2$. One then gets:

$$E_{1,\mathrm{s}}(\alpha, \nu) = (1 + \alpha_{\mathrm{s}})^{-2} \left(\frac{1}{2} D((1 + \alpha_{\mathrm{s}})^2 \cdot T_{\alpha\,\sharp}\,\mu, \nu)^2 + F(\alpha) + G(\nu) \right) \tag{12}$$

This functional will have to be optimized subject to the modified condition $\nu \in \mathcal{S}\big(\mu(X) \cdot (1 + \alpha_{\mathrm{s}})^2\big)$, with the mass of ν adjusted according to α_{s}. Obviously the terms depending on the mass of the measures should be rescaled. But it is reasonable to also rescale the term F since on a larger scale also the deformation modes need to have higher coefficients to obtain the same relative deformation. Since we chose F to be quadratic, the chosen renormalization is exactly the one to cancel that effect.

The scale invariance is only approximate near $\alpha_{\mathrm{s}} = 0$ because of the rasterization applied for practical implementation. When the grid sizes of X and Y become too different, one will expect numerical artifacts.

4 Optimization

Eliminating ν. The functional E_1 is convex in ν for fixed α and vice versa. But instead of a heuristic alternating optimization scheme we propose a hierarchical branch and bound approach that yields global optimizers and also applies to the scale-invariant version $E_{1,\mathrm{s}}$.

For some fixed α let

$$E_2(\alpha) = \min_{\nu \in \mathcal{S}(\mu(X))} E_1(\alpha, \nu)\,, \qquad E_{2,\mathrm{s}}(\alpha) = \min_{\nu \in \mathcal{S}((1+\alpha_\mathrm{s})^2 \cdot \mu(X))} E_{1,\mathrm{s}}(\alpha, \nu)\,. \tag{13}$$

These can be computed by solving the convex optimization problem in ν for fixed α. The remaining problem is to optimize E_2 w.r.t. α. Let now $\mathcal{A}$ be a set of α-parameters and define a functional over such sets:

$$E_3(\mathcal{A}) = \inf_{\nu \in \mathcal{S}(\mu(X))} \frac{1}{2} \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times Y} \left(\min_{\alpha \in \mathcal{A}} \|T_\alpha(x) - y\|^2 \right) d\pi(x,y) + \min_{\alpha \in \mathcal{A}} F(\alpha) + G(\nu)\,. \tag{14}$$

This is an adaptive convex relaxation of minimizing E_2 over a given interval. The relaxation becomes tighter as the interval becomes smaller.

For our global optimization scheme the following proposition is required:

Proposition 1. *The functional E_3 has the following properties:*

$$\text{(i)}\ E_3(\mathcal{A}) \le \min_{\alpha \in \mathcal{A}} E_2(\alpha), \qquad \text{(ii)}\ \lim_{\mathcal{A} \to \{\alpha_0\}} E_3(\mathcal{A}) = E_2(\alpha_0),$$
$$\text{(iii)}\ \mathcal{A}_1 \subset \mathcal{A}_2 \Rightarrow E_3(\mathcal{A}_1) \ge E_3(\mathcal{A}_2)\,. \tag{15}$$

Proof. For the lower bound property (i) note that for any feasible ν and $\pi \in \Pi(\mu, \nu)$:

$$\int_{X \times Y} \left(\min_{\alpha \in \mathcal{A}} \|T_\alpha(x) - y\|^2 \right) d\pi(x,y) \le \min_{\alpha \in \mathcal{A}} \int_{X \times Y} \|T_\alpha(x) - y\|^2 \, d\pi(x,y)$$

This inequality holds also for the infimum of $\pi \in \Pi(\mu, \nu)$ and $\nu \in \mathcal{S}\big(\mu(X)\big)$. So the first term in E_3 is bounded by $\min_{\alpha \in \mathcal{A}} 1/2\, D(T_\alpha \sharp\, \mu, \nu)^2$. The separate minimization of the D and F term is obviously smaller than the joint minimization, so the bound property holds.

For the limit property (ii) note that $\|T_\alpha(x) - y\|^2$ and $F(\alpha)$ are continuous functions in α. Hence, when $\mathcal{A} \to \{\alpha_0\}$ all involved minimizations will converge towards the respective function value at α_0 and E_3 converges as desired.

For the hierarchical bound property (iii) just note that for fixed π and ν minimization over the larger set will never yield the larger result for all occurrences of α. This relation will then also hold after minimization. □

With slight modifications E_3 can be extended to the case $E_{3,\mathrm{s}}$ with a scaling-mode involved: the additional rescaling factor will also be independently optimized over and the feasible sets $\mathcal{S}(\mu(X)), \Pi(\mu, \nu)$ in the initial optimization must be replaced by $\mathcal{S}(\mu(X), (1+\alpha_{\mathrm{s,l}})^2, (1+\alpha_{\mathrm{s,u}})^2), \Pi(\mu, \nu, (1+\alpha_{\mathrm{s,l}})^2, (1+\alpha_{\mathrm{s,u}})^2)$

where $\alpha_{\mathrm{s,l}}$ and $\alpha_{\mathrm{s,u}}$ are lower and upper bound for the scale coefficient for all $\alpha \in \mathcal{A}$. The modified feasible sets are defined by

$$\mathcal{S}(M, s_1, s_2) = \bigcup_{s_1 \le s \le s_2} \mathcal{S}(s \cdot M) \tag{16}$$

$$\begin{aligned} \mathit{\Pi}(\mu, \nu, s_1, s_2) = &\{\pi \in \mathcal{P}(X \times X) : \; s_1 \cdot \mu(\sigma) \le \pi(\sigma \times Y) \le s_2 \cdot \mu(\sigma), \\ &\pi(X \times \sigma) = \nu(\sigma) \text{ for all measurable } \sigma \subset X\}\,. \end{aligned} \tag{17}$$

Since from any π the measure ν can be obtained by marginalization, the nested optimization over ν and π can actually be performed at once, by only minimizing π and transferring the constraints of ν onto π.

So E_3, $E_{3,\mathrm{s}}$ can be computed by solving linear programs, which can also be rewritten as optimal transport problems with suitable dummy nodes, to use specialized, more efficient solvers.

Branch and Bound in α. Let $L = \{(\mathcal{A}_i, b_i)\}_i$ be a list of α-parameter intervals $\mathcal{A}_i$ and lower bounds b_i on E_2 on these respective intervals. For such a list consider the following refinement procedure:

`refine`**(L):**
find the element $(\mathcal{A}_{\hat{\imath}}, b_{\hat{\imath}}) \in L$ with the smallest lower bound $b_{\hat{\imath}}$
let $\texttt{subdiv}(\mathcal{A}) = \{\mathcal{A}_{\hat{\imath},j}\}_j$ be a subdivision of the interval $\mathcal{A}_{\hat{\imath}}$ into smaller intervals
compute $b_{\hat{\imath},j} = E_3(\mathcal{A}_{\hat{\imath},j})$ for all $\mathcal{A}_{\hat{\imath},j} \in \texttt{subdiv}(\mathcal{A})$
remove $(\mathcal{A}_{\hat{\imath}}, b_{\hat{\imath}})$ from L and add $\{(\mathcal{A}_{\hat{\imath},j}, b_{\hat{\imath},j})\}_j$ for $\mathcal{A}_{\hat{\imath},j} \in \texttt{subdiv}(\mathcal{A})$

This allows the following statement:

Proposition 2. *Let L be a list of finite length. Let the subdivision in* `refine` *be such that any interval will be split into a finite number of smaller intervals, and that any two points will eventually be separated by successive subdivision.* $\texttt{subdiv}(\{\alpha_0\}) = \{\{\alpha_0\}\}$. *Then repeated application of* `refine` *to the list L will generate an adaptive piecewise underestimator of E_2 throughout the union of the intervals $\mathcal{A}$ appearing in L. The sequence of smallest lower bounds will converge to the global minimum of E_2.*

Proof. Obviously the sequence of smallest lower bounds is non-decreasing (see Proposition 1 (iii)) and never greater than the minimum of E_2 throughout the considered region. So it must converge to a value which is at most this minimum. Assume that $\{\mathcal{A}_i\}_i$ is a sequence with $\mathcal{A}_{i+1} \in \texttt{subdiv}(\mathcal{A}_i)$ such that $E_3(\mathcal{A}_i)$ is a subsequence of the smallest lowest bounds of L (there must be such a sequence since L is finite). Since `subdiv` will eventually separate any two distinct points, this sequence must converge to a singleton $\{\alpha_0\}$ and the corresponding subsequence of smallest lowest bounds converges to $E_3(\{\alpha_0\}) = E_2(\alpha_0)$. Since the sequence of smallest lowest bounds converges, and the limit is at most the minimum of E_2, $E_2(\alpha_0)$ must be the minimum. □

In practice we start with a coarse grid of hypercubes covering the space of reasonable α-parameters (translation throughout the image, rotation and scale

within bounds where the approximation is valid and the deformation-coefficients in ranges adjusted to the corresponding Gaussian covariances) and the respective E_3-bounds. Any hypercube with the smallest bound will then be subdivided into equally sized smaller hypercubes, leading to an adaptive 2^n-tree cover on the considered parameter range.

We stop the refinement, when the interval with the lowest bound has edge-lengths that correspond to an uncertainty in $T_\alpha(x)$ which is in the range of the pixel discretization of X and Y. Further refinement would only reveal structure determined by rasterization effects.

5 Experiments

Implementation Details. Computation of the E_3-lower bound requires local optimization w.r.t. α for the cost function entries of the optimal transport term. Given the linear decomposition of T_α these are low-dimensional constrained quadratic programs that can quickly be solved. For a given α-interval $\mathcal{A}$ the locally minimized cost function $\min_{\alpha\in\mathcal{A}} \|T_\alpha(x) - y\|^2$ has low values where α values in $\mathcal{A}$ allow $T_\alpha(x)$ to be close to y and rises quickly elsewhere. Exploiting this, we only consider a sparse subset of the full product space $X \times Y$ to speed up computation. To ensure that we still obtain the global optimum, we add overflow variables. As long as no mass is transported onto these dummy variables, the global optimum is attained. Otherwise, more coupling combinations need to be added.

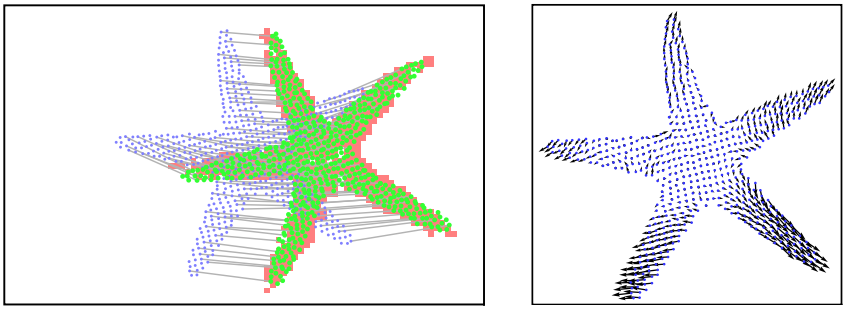


Fig. 3. Left: Illustration of the approach on example data, analogous to Fig. 1b. Small blue dots indicate the arbitrary position of X relative to the image Y (bounding box). Large green dots give the position of $T_\alpha(X)$, the map is indicated by grey lines. The optimal segmentation ν is given by the red shaded region in the image. As intended, the *modes* model the Euclidean isometries (the true object position is not known beforehand and is not relevant for the result), and the major deformations. The Wasserstein distance handles the remaining degrees of freedom, guided by the local data term. Right: The deformation of X by the non-Euclidean modes. Length and relative orientation of the arms are adjusted.

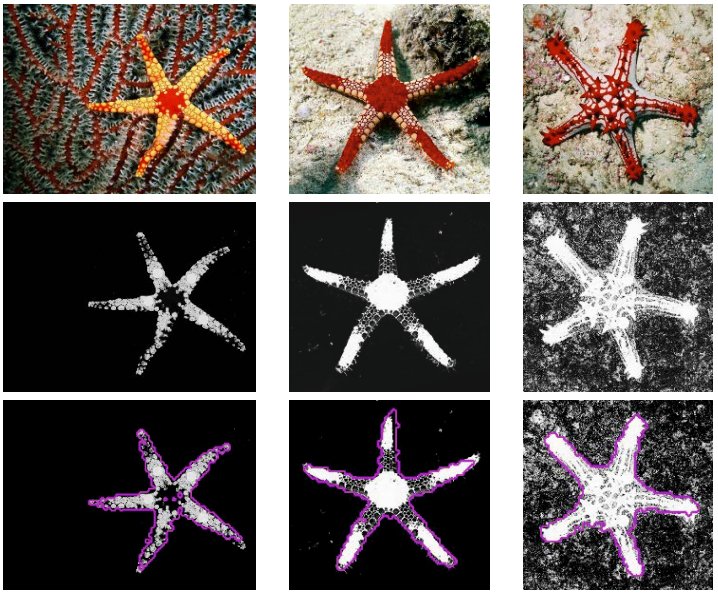


Fig. 4. Segmentation Results with Starfish-Prior. First row: original images. Second row: affinity coefficients g, based on a primitive local color model. There is false-positive clutter, foreground parts are poorly detected or missing. Third row: optimal segmentations, based on joint segmentation and matching. The objects are correctly located, clutter is ignored, missing parts are 'filled in'. Different variants are segmented with the same prior, due to statistical deformation modelling with modes.

On a small scale fluctuations may appear, although the underlying matching is smooth (cp. Fig. 3). These could be handled by enhancing the functional G to locally regularize the boundary of the segmentation.

Setup and Numerical Results. For learning of the object class 'starfish' we used about 20 ground truth segmentations. We took the first four principal components as modes, capturing about 70% of the variance in the training set. Together with translation and rotation this amounts to seven modes to be optimized over.

To the test images we applied a simple local color model, based on seeds, to obtain affinity coefficients g, eq. (4). Note that we specifically chose test images with inhomogeneously colored foreground objects and insufficient seeds for color model training, to obtain coefficients on which a purely local segmentation would fail and the benefit of shape-modelling can be demonstrated.

Fig. 3 illustrates the approach for a typical example. Position and pose of the sought-after object are correctly estimated by the modes, *independent of the position of X, i.e. without requiring a good initial guess.* Figure 4 gives original image, affinity coefficients g and the resulting segmentation for the example in Fig. 3 in column 1 and for further examples. The segmentations in Fig. 4 sometimes exhibit small holes or fluctuations along the boundary, even though

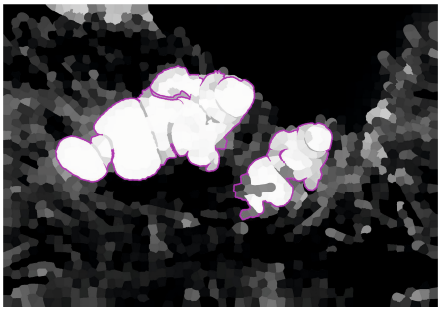


Fig. 5. Scale Invariance and Superpixels. Left: Foreground-affinities g, eq. (4), of a superpixel over-segmentation of an image with two fish of different size, both of which induce strong local minima of our approach. By artificially breaking scale invariance through modelling the scale coefficient α_s to be Gaussian, one can choose which one shall be segmented by setting the mean scale. Other than that, the setup was absolutely identical. One obtains similar segmentation results with a free-moving scale coefficient, if one, in turn, erases one of the fish from the g-coefficients. Right: Template X for fish-experiment. To prevent that the small fish is simply immersed in the big one, one must explicitly model a region of background around the fish, by reversing the affinity coefficients for this region of X. Black (grey) dots indicate fore-(back-)ground.

the underlying object position and pose are very accurately determined (see Fig. 3) and the computed matching is smooth. These irregularities on the pixel level are induced by noise in g and could be removed by local regularization of the boundary of ν (e.g. total variation). As long as such an extension yields a convex functional G, it is still compatible with our approach. To make the acting of the presented Wasserstein-regularization as transparent as possible, however, we chose to omit such fine-tuning.

Scale Invariance and Representation Flexibility. In this section we demonstrate two further important properties of our approach: scale invariance and flexibility in application. Due to the general formulation of optimal transport, adaption to superpixels is straightforward, which facilitates application to large images. In a discrete implementation Y need not be a regular grid (pixel-level) in $\mathbb{R}^2$, but can be any set of points.

We illustrate both aspects in Fig. 5. Our approach, equipped with a prior trained on fish, is run on an image with a large and a small fish. We demonstrate scale invariance by actually artificially breaking it: by modelling the scale-coefficient α_s to be Gaussian, through the choice of the mean scale $\alpha_{s,m}$ we can trigger which of the two fish is segmented, while the wrong sized one is ignored. Except for the mean scale, no modifications in the approach were made.

6 Conclusion

We developed a novel variational approach for joint image segmentation and shape matching. The formulation, based on the Wasserstein distance, allows to

combine modelling of appearance, statistical shape deformation and geometric invariance in a uniform way, by allowing the reference template to be moved and deformed. We extended previous work on analyzing observed deformation fields for object classification to be applicable already during matching of new query images. The resulting way of describing and fitting trained shape variations is in style reminiscent of contour-based variational shape priors, but does not require an intricate conversion between the contour and the region representation of shapes. A well-founded hierarchical branch-and-bound scheme, based on local adaptive convex relaxation, is presented, that provably finds the global minimum.

At some points, development of the presented approach is not yet complete (e.g. modelling rotation beyond linear approximation, a more satisfying way to find a reference template, adding local boundary regularization of ν to suppress fluctuations). Yet, as experiments demonstrated, the functional is able to perform robust segmentation and matching in a noisy environment, which, due to geometric invariance, does not depend on a proper initialization. Additionally, the scale invariance property and the natural portability onto superpixel images have been illustrated.

Acknowledgement. This work was supported by the DFG grant GRK 1653.

References

1. Charpiat, G., Faugeras, O., Keriven, R.: Shape statistics for image segmentation with prior. In: Computer Vision and Pattern Recognition (CVPR 2007), pp. 1–6 (2007)
2. Cremers, D., Kohlberger, T., Schnörr, C.: Shape statistics in kernel space for variational image segmentation. Patt. Recognition 36(9), 1929–1943 (2003)
3. Haker, S., Zhu, L., Tannenbaum, A., Angenent, S.: Optimal mass transport for registration and warping. Int. J. Comput. Vision 60, 225–240 (2004)
4. Klodt, M., Cremers, D.: A convex framework for image segmentation with moment constraints. In: ICCV (2011)
5. Michor, P., Mumford, D.: Riemannian geometries on spaces of plane curves. Journal of the European Mathematical Society 8(1), 1–48 (2006)
6. Schmitzer, B., Schnörr, C.: Weakly convex coupling continuous cuts and shape priors. In: Bruckstein, A.M., ter Haar Romeny, B.M., Bronstein, A.M., Bronstein, M.M. (eds.) SSVM 2011. LNCS, vol. 6667, pp. 423–434. Springer, Heidelberg (2012)
7. Schmitzer, B., Schnörr, C.: Modelling convex shape priors and matching based on the Gromov-Wasserstein distance. Journal of Mathematical Imaging and Vision 46(1), 143–159 (2013)
8. Sundaramoorthi, G., Mennucci, A., Soatto, S., Yezzi, A.: A new geometric metric in the space of curves, and applications to tracking deforming objects by prediction and filtering. SIAM Journal on Imaging Sciences 4(1), 109–145 (2011)
9. Villani, C.: Optimal Transport: Old and New. Springer (2009)
10. Wang, W., Slepčev, D., Basu, S., Ozolek, J.A., Rohde, G.K.: A linear optimal transportation framework for quantifying and visualizing variations in sets of images. International Journal of Computer Vision 101, 254–269 (2012)

Learning a Model for Shape-Constrained Image Segmentation from Weakly Labeled Data

Boris Yangel and Dmitry Vetrov

Lomonosov Moscow State University

Abstract. In the paper we address a challenging problem of incorporating preferences on possible shapes of an object in a binary image segmentation framework. We extend the well-known conditional random fields model by adding new variables that are responsible for the shape of an object. We describe the shape via a flexible graph augmented with vertex positions and edge widths. We derive exact and approximate algorithms for MAP estimation of label and shape variables given an image. An original learning procedure for tuning parameters of our model based on unlabeled images with only shape descriptions given is also presented. Experiments confirm that our model improves the segmentation quality in hard-to-segment images by taking into account the knowledge about typical shapes of the object.

Keywords: MRF, image segmentation, shape priors, weakly-labeled data, part-based models

1 Introduction

Image segmentation is a well-studied problem in computer vision. It can be solved well (see, for example, [13]) when objects differ in color and texture from background significantly. However, in case of non-discriminative color models and weak object boundaries some high-level knowledge about the scene is required to make segmentation more robust. For example, one can have some clues about the shape of the object being segmented, and use those as a segmentation prior.

Taking such knowledge into account while segmenting an image is not an easy task. Introduction of high-level constraints into state-of-the-art approaches to image segmentation such as conditional random fields (CRF) [9] leads either to oversimplified models, or to complex models with high-order terms, which are hard to infer solutions from and even harder to learn. As a result, such models are tweaked manually and tend to perform much worse than they could have.

In this paper we aim to introduce a model for shape-constrained binary segmentation (i.e. segmentation to object and background) that is powerful enough to describe complex shapes, and yet has tractable inference and learning procedures. Our model is built upon a popular way (see, for example, [8,10,16]) of introducing global constraints into CRF, which uses unary terms of the CRF energy to constrain labeling together with an additional term as a prior for high-level clues, which in our case are shape descriptions. We describe object shape

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 137–150, 2013.
© Springer-Verlag Berlin Heidelberg 2013

via a graph augmented with vertex positions and edge widths, a way that is similar to part-based models [4,3] but seems to be more rich. One interesting property of our segmentation model is that it can be seen as a shape fitting model that uses pixel labels as latent variables. It allows us to come up with a training formulation that does not require pixel labels to learn from.

1.1 Related Work

The description of object shape proposed in this work can be seen as an application of part-based object modeling, technique that is well-developed in the context of the object part detection problem. Furthermore, both exact inference and learning procedures proposed in this paper are build on top of techniques developed for this kind of models. In [4] one such model, namely pictorial structure, is presented together with an efficient inference algorithm based on dynamic programming. A follow-up work [5] introduces a max-margin semi-supervised learning procedure for it. Another example is [3], which uses a trained mixture of part-based models for pose estimation and action classification.

Part-based models were used to constrain segmentation before. One example is [16], which proposes a description of object shape that is quite similar to ours. In fact, it is even more powerful since we don't allow width to vary along the edges of graph representing shape. However, this limitation makes it possible to build an algorithm for exact inference in our model. Another similar work is [12], which proposes a two-step algorithm for human segmentation. The proposed algorithm first tries to find the most plausible configuration of a part-based human model given an image using MCMC, and then uses it to constrain the segmentation. We think, however, that pixel labelings induced by shapes have lots of information about the correctness of shape fitting, and, therefore, fitting should not be decoupled from the segmentation process. Another example of a work that uses part-based models for image segmentation is [8], which represents object shape via a layered pictorial structure and finds segmentation by using Monte Carlo EM combined with loopy BP, since the model is too complicated for exact inference.

Other approaches to shape-constrained segmentation that do not involve modeling of object parts are known in literature. One notable class of such techniques includes star-shape prior [14] and tightness prior [11]. Both come with an efficient segmentation procedure that has good optimality guarantees, but impose quite weak restrictions on object shape and, therefore, can be of limited utility when segmenting certain kinds of objects. Another group of works includes those trying to describe shape via a hard mask [15,2], representation that we find to be improper for classes of objects with high shape variability. One notable member of this group is [10], which models object shape via a huge set of different masks and also provides a framework for exact inference in CRF with high-order terms resulting from such constraints. We've used this framework as a basis for the exact inference algorithm we propose in this paper.

1.2 Contribution

Main contributions of this paper are

- a flexible model of object shape, which is invariant to translation and rotation and allows to describe classes of forms with high variability;
- two inference procedures for the segmentation constrained with the proposed shape model. The first one is fast but approximate, while the other one is slower but capable of obtaining exact solutions;
- a new formulation of learning from weakly labeled data problem for adjusting the parameters of our shape-constrained segmentation model. As a weak labeling we only use object shape descriptions for each image without pixel-wise image labeling.

2 Shape-Constrained Binary Segmentation

We state the problem of image segmentation with shape constraints as the problem of finding minimum of the energy function

$$\begin{aligned} E(L,s) = f(s) + w_c^0 \sum_i (1-L_i)\phi_i^0 + w_s^0 \sum_i (1-L_i)\psi_i^0(s) + \\ + w_c^1 \sum_i L_i \phi_i^1 + w_s^1 \sum_i L_i \psi_i^1(s) + w_p \sum_{(i,j)\in\mathcal{N}} |L_i - L_j| \phi_{ij} \end{aligned} \tag{1}$$

w.r.t. variables $L_i \in \{0,1\}$ representing pixel labels and variable s describing object shape. Variable i indexes all pixels and $\mathcal{N}$ stands for the set of pairs of indices of neighboring image pixels. Pairwise terms ϕ_{ij}, which we require to be non-negative for reasons explained later, can be used to move object boundary towards the areas with high color gradient magnitude. There are two types of unary terms: constant $\phi_i^{0,1}$ that can be used to encode known color distributions of object and background, and $\psi_i^{0,1}$ that depend on s and, thus, allow us to relate shape descriptions with labeling configurations. Energy term $f(s)$ is used to penalize improbable shape descriptions. We give more information about terms involving variable s in the following sections of the paper. Term weights $w_c^{0,1}$, $w_s^{0,1}$ and w_p act as model parameters.

2.1 The Model of Object Shape

We describe object shape via a graph augmented with vertex positions and edge width values. One example of such description, which we would call *shape graph* from now, is shown in Fig. 1(a). By varying angles, lengths, and widths in graphs we can obtain different shape variations. However incidence relation in the graph is fixed so all shape variations have similar structure.

More formally, shape graph is a tuple $(\mathcal{E}_s, \mathcal{V}_s)$, where

$$\begin{aligned} \mathcal{E}_s &= \{e_k^s\}, \quad e_k^s = \left(f_k^s, t_k^s, b_k^s\right), \\ \mathcal{V}_s &= \{v_l^s\}, \quad v_l^s = \left(x_l^s, y_l^s\right). \end{aligned}$$

Here $\mathcal{E}_s$ stands for the set of edges of shape graph s, in which every edge e_k^s has width b_k^s and connects vertices with indices f_k^s t_k^s. The set of graph vertices is denoted as $\mathcal{V}_s$. Vertex v_l^s has coords (x_l^s, y_l^s) on the image.

Each shape graph has an associated value of the shape energy

$$\begin{aligned} f(s) = w_r\left(\frac{\|e_r^s\|}{l_r} - 1\right)^2 + \sum_k w_k^b\left(\frac{b_k^s}{\|e_k^s\|} - \rho_k^b\right)^2 + \\ + \sum_{(k_1,k_2)\in\mathcal{M}} w_{k_1,k_2}^l\left(\|e_{k_1}^s\| - \rho_{k_1,k_2}^l\|e_{k_2}^s\|\right)^2 + \\ + \sum_{(k_1,k_2)\in\mathcal{M}} w_{k_1,k_2}^\alpha \min\Big\{\big(\angle(e_{k_1}^s, e_{k_2}^s) - \alpha_{k_1,k_2} + 2\pi\big)^2, \\ \big(\angle(e_{k_1}^s, e_{k_2}^s) - \alpha_{k_1,k_2} - 2\pi\big)^2, \big(\angle(e_{k_1}^s, e_{k_2}^s) - \alpha_{k_1,k_2}\big)^2\Big\}, \end{aligned} \tag{2}$$

where $\mathcal{M}$ stands for the set of pairs of edge indices involved in pairwise constraints,

$$\|e_k^s\| = \|v_{f_k}^s - v_{t_k}^s\| \tag{3}$$

is the length of edge e_k^s, and

$$\angle(e_{k_1}^s, e_{k_2}^s) = \angle\big((v_{f_{k_1}}^s, v_{t_{k_1}}^s), (v_{f_{k_2}}^s, v_{t_{k_2}}^s)\big) \tag{4}$$

is the directed angle between edges with indices k_1 and k_2. Parameter ρ_{k_1,k_2}^l stands for the mean length ratio between edges with indices k_1 and k_2, while w_{k_1,k_2}^l measures softness of the constraint. Parameters α_{k_1,k_2}, w_{k_1,k_2}^α, ρ_k^b, w_k^b have analogous meaning for angles between edges and width-to-length ratios respectively. Parameters l_r and w_r are used to specify the scale of the so-called *root* edge e_r^s, thus enforcing the scale of the whole model through pairwise edge length constraints.

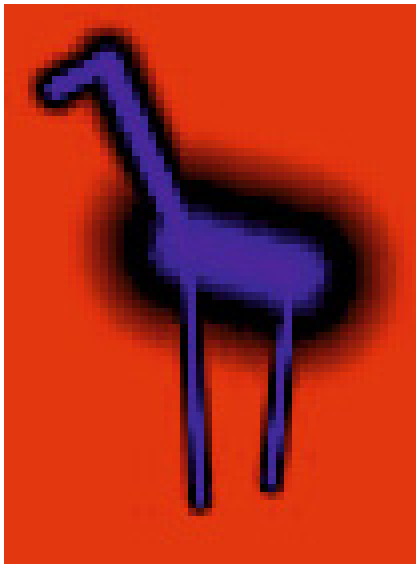
The form of angle penalty terms is justified by the fact that the difference of two angles inside the $[-\pi, \pi]$ range can lie outside this range and, thus, can have an alternative representation with lower penalty inside.

The idea behind this energy function is to penalize uncommon shapes while allowing different object parts to have different variability. For example, a shape model aiming to describe both running and standing horses can enforce quite soft constraints on the angle between horse body and legs compared to the angle between its neck and head.

Since the shape energy terms depend only on edge length, width and angles between edges, the energy is invariant to both rotation and translation. However such information, if available, can be easily incorporated into the energy function as a constraint on the root edge, as it was done for scale.

The reason this model is not invariant to scale lies in the form of pairwise terms enforcing constraints on lengths of neighboring edges. These terms could have been made scale-invariant by replacing length difference by ratio difference as it was done for terms involving edge width, but then we won't be able to apply the exact inference algorithm described in Sect. 2.4 to our problem.

(a) A photo of a giraffe with a shape graph modeling its shape



(b) $\psi_i^1(s) - \psi_i^0(s)$ induced by the graph on the left. Positive values are colored red.

Fig. 1. An illustration of a graph-based shape model

2.2 Relation to Pixel Labeling

In order to complete the description of the proposed shape model, we need to specify the way it affects image labeling. It is natural to assume that pixels located near some edge of the shape graph will probably belong to object, while pixels that are far from any edge will most likely belong to background. Based on this assumption we define shape-based unary terms for pixel i as

$$\psi_i^1(s) = \frac{1}{|I|} \min_{e_k^s \in \mathcal{E}_s} \beta(b_k^s) d^2[p_i, (v_{f_k}^s, v_{t_k}^s)], \tag{5}$$

$$\psi_i^0(s) = \frac{1}{|I|} \max_{e_k^s \in \mathcal{E}_s} -\log\Big[1 - \exp\big(-\beta(b_k^s) d^2[p_i, (v_{f_k}^s, v_{t_k}^s)]\big)\Big], \tag{6}$$

$$\beta(b) = \frac{4 \log 2}{b^2}, \tag{7}$$

where $d[p, (v_1, v_2)])$ is the distance from a point p to a segment (v_1, v_2), p_i represents the coordinates of the i-th pixel of an image and $|I|$ is the total number of pixels. Reasons for scaling terms by inverse image size are explained in Sect. 4.2. Form of the coefficient $\beta(b)$ is justified by the requirement that object and background labels should be equally likely at half-width distance from an edge. An example of shape terms induced by a shape graph can be seen in Fig. 1(b).

2.3 Approximate Inference

Energy function (1) has a nice property: if the shape description s is fixed and $\phi_{ij} \geq 0$ for every pair of neighboring pixels, its minimum can be efficiently computed via graph cuts [1]. It allows us to cast the problem of minimizing $E(L, s)$ to the problem of finding

$$\min_s F(s) = \min_s \min_L E(L, s). \tag{8}$$

Function $F(s)$ has quite a few parameters compared to the original energy function, can be efficiently computed in each point, and, thus, can be minimized using some derivative-free optimization technique.

In this work we use simulated annealing as a minimization technique. It was selected mostly due to its natural applicability to the graph-based shape description. We use the following transition moves:

- randomly changing the length of a random edge e_k^s by sampling l^{new} from the truncated Gaussian $\mathcal{N}_{l \geq l^{min}}(l; \|e_k^s\|, \sigma_l^2 T^2)$ and shifting $v_{t_k^s}$ so that l^{new} is the new length of e_k^s (which, in turn, changes the configuration of all other edges incident to $v_{t_k^s}$);
- randomly changing the width of a random edge e_k^s by sampling b_k^s from $\mathcal{N}_{b \geq b^{min}}(b; b_k^s, \sigma_b^2 T^2)$;
- randomly changing the angle between a random edge e_k^s and a fixed direction by sampling a new angle from $\mathcal{N}(\alpha; \angle e_k^s, \sigma_\alpha^s T^2)$ and shifting $v_{t_k^s}$ accordingly (again, it will change the configuration of all neighboring edges);
- applying random amount of translation, rotation or scale to the whole shape graph. Amounts of transformation were sampled from Gaussian distributions (truncated Gaussian in the case of scale) in the same fashion as above, with standard deviation being proportional to T. Rotation and scale transforms use the mean of vertex coordinates as the origin.

The current annealing temperature, T, was set to $\frac{1}{\log_2(n+1)}$ on the n-th annealing iteration. We've used $\min\left[1, \exp\left(\frac{F(s_{best}) - F(s)}{T}\right)\right]$ as the acceptance probability for the proposed shape graph s, where s_{best} is the graph with the lowest value of $F(s)$ found so far.

We were able to obtain quite good local minima starting the optimization process from a *mean* shape graph automatically fitted into image bounds. It was produced by first building a graph with the lowest possible energy in which root edge has the specified direction, and then shifting the graph so that its center corresponds to the image center. If the expected object orientation was known, we used it as the orientation of a fitted graph. Otherwise, we tried to start the process from a number of mean shape graphs fitted with different orientations.

2.4 Exact Inference

The exact inference procedure for the discussed model is built upon the branch-and-mincut framework [10]. This framework aims to minimize an energy

function of form (1) via a breadth-first branch-and-bound procedure, which uses the expression

$$\min_L \Big[f^S + w_c^0 \sum_i (1 - L_i)\phi_i^0 + w_s^0 \sum_i (1 - L_i)\psi_i^{0,S} + \\ + w_c^1 \sum_i L_i \phi_i^1 + w_s^1 \sum_i L_i \psi_i^{1,S} + w_p \sum_{(i,j)\in\mathcal{N}} |L_i - L_j| \phi_{ij} \Big], \tag{9}$$

where

$$f^S = \min_{s\in S} f(s), \quad \psi_i^{0,S} = \min_{s\in S} \psi_i^0(s), \quad \psi_i^{1,S} = \min_{s\in S} \psi_i^1(s), \tag{10}$$

to bound below the minimum of the energy when s is constrained to be in S. This lower bound can be efficiently computed for any set S via graph cuts if the *aggregated potentials* f^S, $\{\psi_i^{0,S}\}$ and $\{\psi_i^{1,S}\}$ are known. In order to apply this framework to our model we need to provide a way to describe a set of shape graphs S together with a subdivision scheme for it. We also need to provide an efficient algorithm for computing aggregated potentials for any given set S.

We choose to represent S as a set of axis-aligned bounding boxes (AABB) limiting possible positions of shape graph vertices, together with a set of one-dimensional ranges limiting the width of each edge. More formally,

$$s \in S \iff \forall k\ b_k^s \in B_k^S,\ \forall l\ v_l^s \in V_l^S, \\ B_k^S = [b_k^{S,min}, b_k^{S,max}], V_l^S = [x_l^{S,min}, x_l^{S,max}] \times [y_l^{S,min}, y_l^{S,max}]. \tag{11}$$

A natural subdivision scheme for this representation is to either split one of the vertex constraints in four, or split some edge width constraint in two until the constraints become singletons.

In order to compute $\psi_i^{1,S}$ we first note that

$$\min_{s\in S} \psi_i^1(s) = \frac{1}{|I|} \min_k \min_{b_k^s \in B_k^S} \beta(b) \min_{v_{f_k}^s \in V_{f_k}^S} \min_{v_{t_k}^s \in V_{t_k}^S} d^2\big(p_i, (v_{f_k}^s, v_{t_k}^s)\big). \tag{12}$$

So, in order to compute this aggregated potential we need to find the closest location to the pixel i for every edge of s w.r.t. constraints on S, and then take the minimum across all edges using the maximum possible width for every edge. Similar considerations apply to $\psi_i^{0,S}$, but in this case every edge should be taken away from pixel as far as possible, while its width should be made as small as possible. When edge constraints are given by AABB, finding closest (or farthest) location of an edge from a pixel is a simple problem that can be solved in constant time.

In order to compute f^S we first note that angles $\angle(e_{k_1}^s, e_{k_2}^s)$ in (2) can be rewritten as $\angle e_{k_2}^s - \angle e_{k_1}^s$, where $\angle e$ is an angle between an edge and some fixed direction. Then all pairwise terms of (2) will be quadratic w.r.t. values $\|e_{k_1}^s\|$, $\|e_{k_2}^s\|$, $\angle e_{k_1}^s$, $\angle e_{k_2}^s$. Thus, given $\mathcal{M}$ corresponds to a tree, the energy can be efficiently minimized via dynamic programming accelerated using generalized distance transform (GDT) technique [4], considering these values together

with b_k as variables. Feasible sets for edge lengths and angles are given by V_l, while widths should be constrained by B_k. Note that the shape graph itself is not forced to be a tree, only the graph of pairwise edge constraints given by $\mathcal{M}$.

3 Learning from Weakly-Labeled Data

In this section we describe a way to estimate the parameters of the shape-constrained binary segmentation model (1). We first explain our training problem formulation and the way we build training set, and then show a way to estimate non-linear and linear parameters of the model separately.

3.1 Training Set

Since the energy (1) is a function of shape graph and pixel labeling, it is natural to use a set of images augmented with both ground truth labelings and graphs as a training set. However, true labelings are much more expensive to obtain in terms of human labor amount required to label one image compared to shape graphs. We aim to solve this issue by proposing a novel formulation of the segmentation model training problem. We note that the segmentation energy minimum, if expressed in a form (8), can be seen as the most plausible configuration of a shape graph on a given image, with pixel labels acting as latent variables of the model. We therefore state the problem of training our model as the problem of training shape graph fitting model given shape graph ground truth only. The hope is that the best labeling associated with the fitted graph corresponds to a meaningful segmentation. Experimental results shown later confirm that this is actually the case.

We denote the training set for our training problem as

$$(I^1, s^1), \ldots, (I^m, s^m), \tag{13}$$

where I^m is m-th training image, s_m is a ground truth shape graph for it and M is the total number of images in the training set.

3.2 Learning Non-linear Parameters

Our model has a number of non-linear parameters, r, l_r, $\{\rho^l_{k_1,k_2}\}$, $\{\alpha_{k_1,k_2}\}$, $\{\rho^b_k\}$, which we will together denote as θ from now. We propose to train these parameters before the rest of the model in order to simplify the procedure. Training objective is to choose θ that minimizes the total energy of the training set

$$\sum_{m=1}^{M} f(s^m; \theta), \tag{14}$$

where $f(s;\theta)$ is given by (2). While the shape energy also depends on the values of term weights, the optimal parameter values do not and are given by the following expressions:

$$l_k = \frac{\sum_{m=1}^{M} |e_k^{s^m}|^2}{\sum_{m=1}^{M} |e_k^{s^m}|}, \tag{15}$$

$$r = \arg\min_k \sum_{m=1}^{M} \left(\frac{|e_k^{s_m}|}{l_k} - 1 \right)^2, \tag{16}$$

$$\rho_k^b = \frac{1}{M} \sum_{m=1}^{M} \frac{b_k^{s_m}}{|e_k^{s_m}|}, \tag{17}$$

$$\rho_{k_1,k_2}^l = \sum_{m=1}^{M} \frac{|e_{k_1}^{s_m}||e_{k_2}^{s_m}|}{|e_{k_2}^{s_m}|^2}. \tag{18}$$

The estimation procedure for α_{k_1,k_2} is also trivial, but hard to write in a closed form since multiple angle representations should be considered when averaging angles.

3.3 Learning Linear Parameters

Discussed model has w_r, $\{w_{k_1,k_2}^l\}$, $\{w_{k_1,k_2}^\alpha\}$, $\{w_k^b\}$, $w_c^{0,1}$, $w_s^{0,1}$ and w_p as its linear parameters. Let us denote the vector containing all these parameters as w and introduce the vector $\Phi(I, L, s)$ containing negated energy terms corresponding to weights in w for a given image I with an associated labeling L and a shape graph s. Now the problem of segmenting image I can be stated as the problem of finding

$$\arg\min_{L,s} E(L, s) = \arg\max_{L,s} w^T \Phi(I, L, s). \tag{19}$$

Such a reformulation allows us to use the latent structural SVM formulation [17] for weight training. Thus, we want to find

$$\arg\min_w \left[\frac{1}{2} w^T w + \frac{C}{M} \sum_{m=1}^{M} \left\{ \max_{L,s} [w^T \Phi(I^m, L, s) + \Delta(s, s^m)] - \right. \right. \\ \left. \left. - \max_L w^T \Phi(I^m, L, s^m) \right\} \right]. \tag{20}$$

The idea behind this objective is to enforce a ground truth shape graph together with its best possible labeling have lesser energy value than any other shape graph and labeling. Loss function $\Delta(s_1, s_2)$ is used to rescale the energy margin depending on how close the two graphs are: graphs similar to the ground truth should also have low energy, while those significantly distinct from the ground truth should have much higher energy values. The loss function we use is of form

$$\Delta(s_1, s_2) = \sum_l \min\{\|v_l^{s_1} - v_l^{s_2}\|, t_v\} + \lambda \sum_k \min\{|b_k^{s_1} - b_k^{s_2}|, t_e\}. \tag{21}$$

The motivation behind truncating the loss is that when some graph is too much apart from the ground truth, one should not really care about the value of margin as soon as it's large.

Convex-concave procedure (CCCP) [18] can be used to minimize (20), which would result in an iterative procedure with two-step iterations. On each CCCP iteration the expression $w^T \Phi(I^m, L, s^m)$ is first maximized w.r.t. L for each m in order to obtain $\tilde{L}^m$, new optimal values of the latent variables. For the model considered in this paper it can be done via a graph cut. Obtained $\tilde{L}^m$ are then substituted into (20), which results in a regular structural SVM problem

$$\frac{1}{2} w^T w + \frac{C}{M} \sum_{m=1}^{M} \Big\{ \max_{L,s} [w^T \Phi(I^m, L, s) + \Delta(s, s^m)] - \\ - w^T \Phi(I^m, \tilde{L}^m, s^m) \Big\} \to \min_w \tag{22}$$

that should be solved in order to update weights.

In order to solve (22) we employ the cutting plane algorithm [7] in which most violated constraints $\arg\max_{L,s} [w^T \Phi(I^m, L, s) + \Delta(s, s^m)]$ are found by the inference algorithm described in Sect. 2.3. Loss function can be easily incorporated into it, since the annealing only requires the ability to evaluate the objective function at a point. We've found it useful to start the annealing from both a mean shape graph fitted into image and a ground truth shape graph and then choose the best solution among two. While we are unable to obtain exact solutions to the SSVM problem this way due to the usage of an approximate procedure for finding most violated constraints, weights we've found on each iteration were close enough to global optima for CCCP to converge.

4 Experiments

4.1 Datasets

In order to validate the model proposed in this paper, we've trained it on two different datasets. The first dataset was build from a specific subset of ETHZ shape classes dataset [6] that contained images of giraffes. It featured images that are hard to segment using low-level cues only due to weak boundaries and significantly overlapping color distributions of object and background pixels. Meanwhile, giraffe shape has a simple structure and can be described with a graph-based model well. Another dataset was build from synthetic images of capital "E" letter that feature totally non-discriminative object and background color distributions, and, so, the only information segmentation model can rely on is object shape and boundaries. All the images were downscaled to size of about 140×140 pixels in order to speedup the training process.

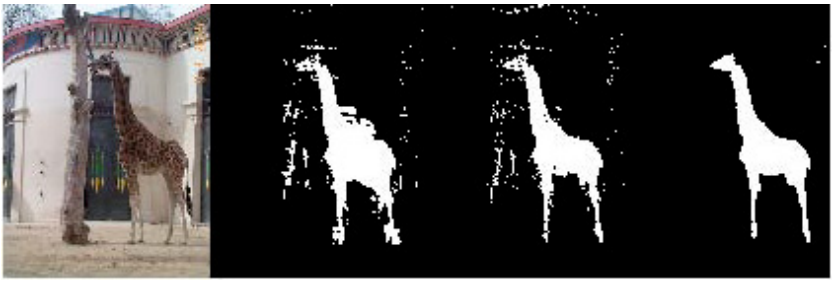


Fig. 2. Convergence of latent variables during training

4.2 Color-Based Terms

In our experiments we used unary color terms of the form

$$\phi_i^l = -\frac{1}{|I|}\log P(I_i \mid L_i = l),\ l \in \{0,1\}. \tag{23}$$

Color distributions for object and background were represented by 3-component GMMs in RGB space, which were learned from seeds placed on a few training images. Pairwise terms we used were of the form

$$\phi_{ij} = \frac{1}{|I|^{\frac{1}{2}}} e^{-\alpha \frac{(\|I_i\| - \|I_j\|)^2}{D^2}}, \tag{24}$$

where $\|I_i\|$ is brightness value of i-th pixel of image I and D is the mean brightness difference value for all the pixel pairs from $\mathcal{N}$ in that image. Value of α was set to 0.2. Color-based terms were scaled according to image size in the same way as (5) and (6) to make model features independent of image size.

4.3 Model Training

Weight vector with $w_c^0 = w_c^1 = 1$, $w_s^0 = w_s^1 = 0.3$, $w_p = 0.001$ and all other components set to 0 was used to initialize the training process. Constant C in latent SSVM objective (20) was set to 300. Parameters of the loss function (21) were set as follows: $\lambda = 10$, $t_v = 0.25|I|^{\frac{1}{2}}$, $t_e = 0.1|I|^{\frac{1}{2}}$. Training process usually converged in 5-7 iterations, each iteration requiring about 70-100 cutting planes.

The most interesting observation about the training process is that the latent variables $\tilde{L}^k$ found on each iteration of CCCP tend to converge to ground truth labelings (see Fig. 2 for an example). Thus, true pixel labels are actually not required for learning since they can be closely approximated during the training process.

4.4 Segmentation

We've then applied trained models to images not involved in the training process using the proposed approximate inference algorithm. Some images together with the best found shape graph and labeling configuration can be seen in Fig. 3. Segmentations based solely on color-based terms (both unary and pairwise) are provided for comparison. It can be seen that our algorithm is able to fit shape graph

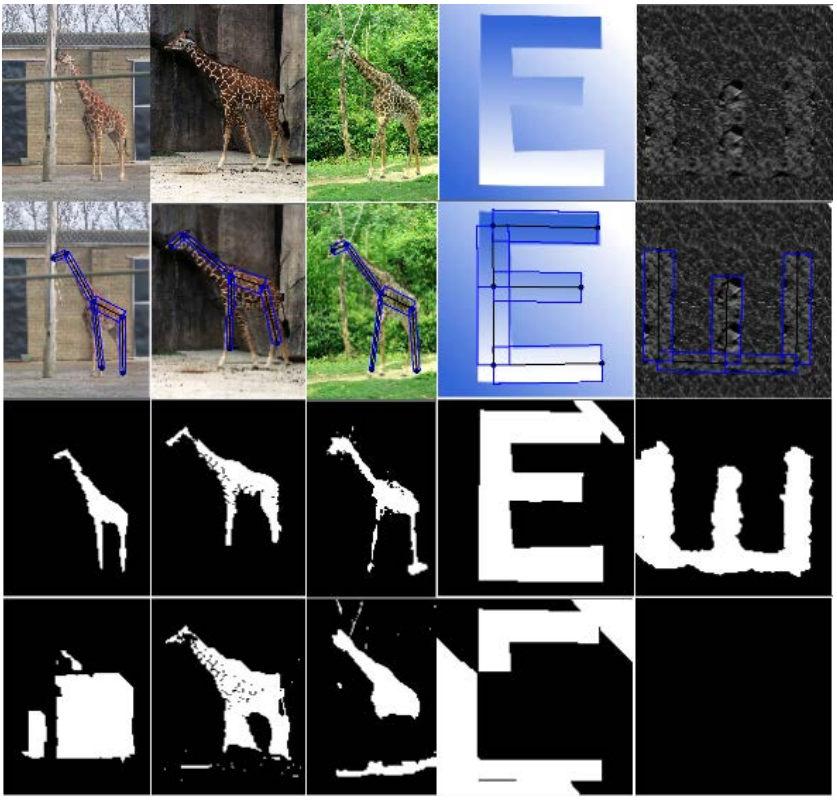


Fig. 3. Top to bottom: image, found shape graph, shape-constrained segmentation, color-based segmentation

into an image correctly. Fitted graph induces quite reasonable segmentation of an image, much better than the one obtained from color-based terms only.

In order to compare the approximate inference algorithm with the exact one, we've applied the latter to same images. As a result, we found out that in many cases approximate algorithm is able to find solutions very close to global optima. However, it's not always the case. One example where approximate inference fails while exact succeeds is shown in Fig. 4. Unfortunately, our exact algorithm may took a lot of time (hours sometimes) to converge despite its very efficient implementation (utilization of GPU for calculation of aggregated potentials, extensive caching). The reason is that the bound (9) is not tight, and, thus, lots of lower bound computations are needed before a good subset of solution space would be discovered. Many of images in the test set require a few millions of lower bound computations. There are also some images that require too many lower bound computations to discover the solution, so we were not able to segment those using our exact inference algorithm. At the same time, the proposed approximate algorithm is quite fast. It usually takes less than a minute per image

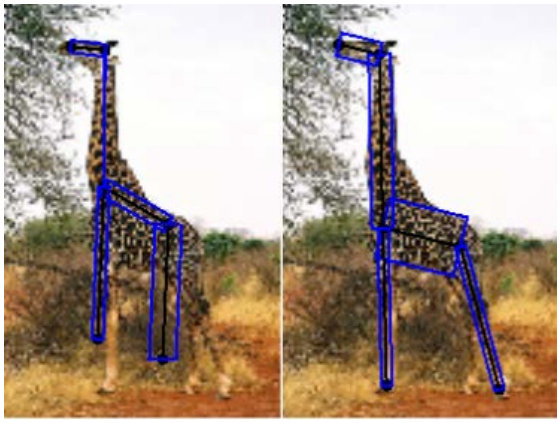


Fig. 4. Solution found by simulated annealing (left) vs branch-and-bound

and can be further accelerated by reducing the number of annealing iterations or reannealing attempts.

5 Conclusion

In this paper we've proposed a model for shape-constrained binary segmentation of an image. The model emerges from combining regular CRF for binary segmentation with high-order terms based on a specific form of object shape description, namely graph-based shape representation. This representation is invariant to object rotation and translation and can describe classes of objects with complex shapes and high in-class variability. We present two inference algorithms for the proposed model. One, which is based on simulated annealing, is fast but approximate, while the other can obtain exact solutions via the branch-and-bound procedure.

We've also proposed a novel training formulation for our model, which requires only ground truth shape descriptions but no pixel-wise labelings to learn its parameters. Training procedures are provided for both linear and non-linear parameters of the model. Experiments on artificial as well as real-world images confirm that ground truth image labelings are indeed not required to learn a well-performing model.

One interesting direction of future work would be to combine the model proposed in this paper with state-of-the-art object part detectors, which output can be used as additional clues about possible positions of shape graph vertices. Another question is whether the exact inference algorithm we've presented can be significantly accelerated by, for example, tightening the lower bound (9).

References

1. Boykov, Y., Jolly, M.: Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In: Proceedings of the 8th IEEE International Conference on Computer Vision, vol. 1, pp. 105–112. IEEE (2001)

2. Cremers, D., Schmidt, F., Barthel, F.: Shape priors in variational image segmentation: Convexity, lipschitz continuity and globally optimal solutions. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–6. IEEE (2008)
3. Desai, C., Ramanan, D.: Detecting actions, poses, and objects with relational phraselets. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part IV. LNCS, vol. 7575, pp. 158–172. Springer, Heidelberg (2012)
4. Felzenszwalb, P., Huttenlocher, D.: Pictorial structures for object recognition. International Journal of Computer Vision 61(1), 55–79 (2005)
5. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE (2008)
6. Ferrari, V., Jurie, F., Schmid, C.: From images to shape models for object detection. International Journal of Computer Vision 87(3), 284–303 (2010)
7. Joachims, T., Finley, T., Yu, C.: Cutting-plane training of structural svms. Machine Learning 77(1), 27–59 (2009)
8. Kumar, M., Torr, P., Zisserman, A.: Obj cut. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 18–25. IEEE (2005)
9. Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the 18th International Conference on Machine Learning, pp. 282–289. Morgan Kaufmann Publishers Inc. (2001)
10. Lempitsky, V., Blake, A., Rother, C.: Image segmentation by branch-and-mincut. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 15–29. Springer, Heidelberg (2008)
11. Lempitsky, V., Kohli, P., Rother, C., Sharp, T.: Image segmentation with a bounding box prior. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 277–284. IEEE (2009)
12. Rauschert, I., Collins, R.T.: A generative model for simultaneous estimation of human body shape and pixel-level segmentation. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 704–717. Springer, Heidelberg (2012)
13. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. ACM Transactions on Graphics (TOG) 23, 309–314 (2004)
14. Veksler, O.: Star shape prior for graph-cut image segmentation. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 454–467. Springer, Heidelberg (2008)
15. Vu, N., Manjunath, B.: Shape prior segmentation of multiple objects with graph cuts. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE (2008)
16. Yangel, B., Vetrov, D.: Image segmentation with a shape prior based on simplified skeleton. In: Boykov, Y., Kahl, F., Lempitsky, V., Schmidt, F.R. (eds.) EMMCVPR 2011. LNCS, vol. 6819, pp. 247–260. Springer, Heidelberg (2011)
17. Yu, C., Joachims, T.: Learning structural svms with latent variables. In: Proceedings of the 26th Annual International Conference on Machine Learning, pp. 1169–1176. ACM (2009)
18. Yuille, A., Rangarajan, A.: The concave-convex procedure. Neural Computation 15(4), 915–936 (2003)

An Optimal Control Approach to Find Sparse Data for Laplace Interpolation

Laurent Hoeltgen, Simon Setzer, and Joachim Weickert

Mathematical Image Analysis Group,
Faculty of Mathematics and Computer Science, Campus E1.7
Saarland University, 66041 Saarbrücken, Germany
{hoeltgen,setzer,weickert}@mia.uni-saarland.de

Abstract. Finding optimal data for inpainting is a key problem in the context of partial differential equation-based image compression. We present a new model for optimising the data used for the reconstruction by the underlying homogeneous diffusion process. Our approach is based on an optimal control framework with a strictly convex cost functional containing an L_1 term to enforce sparsity of the data and non-convex constraints. We propose a numerical approach that solves a series of convex optimisation problems with linear constraints. Our numerical examples show that it outperforms existing methods with respect to quality and computation time.

Keywords: Laplace Interpolation, Optimal Control, Inpainting, Non-convex Optimisation.

1 Introduction

A major challenge in data analysis is to reconstruct a function, for example a 1-D signal or an image from a few data points. In image processing this interpolation problem is called inpainting [2,18]. In most applications one has no influence on the given data and thus improvements can only be made by introducing more powerful reconstruction models. In some interesting applications however, one has the freedom to choose the interpolation data. For instance, in recent approaches to image compression [12,16,22] the authors choose suitable interpolation data for reconstructions via partial differential equations (PDEs). A related approach can also be found in [8,9].

In this paper we present an energy-based approach to optimise the interpolation data. Since this is a very challenging task, we restrict ourselves to a simple reconstruction method, namely homogeneous diffusion inpainting [1,17] which is also called Laplace interpolation. Further, we will solely focus on the optimisation with respect to its reconstruction quality. Whether the methods presented herein are competitive to other image compression algorithms will be the subject of future work. Homogeneous diffusion inpainting is a fast and very flexible way to

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 151–164, 2013.
© Springer-Verlag Berlin Heidelberg 2013

reconstruct large scattered datasets, especially in higher dimensions. It considers the following boundary value problem:

$$\begin{aligned} -\Delta u &= 0, \quad \text{on } \Omega \setminus \Omega_K \\ u &= f, \quad \text{on } \Omega_K \\ \partial_n u &= 0, \quad \text{on } \partial\Omega \setminus \partial\Omega_K \end{aligned} \tag{1}$$

where f is a smooth function on some bounded domain $\Omega \subset \mathbb{R}^n$ with a sufficiently regular boundary $\partial\Omega$. The set Ω_K is assumed to be a subset of Ω with positive measure. It represents the known data that will be interpolated by the underlying diffusion process. Finally $\partial_n u$ denotes the derivative of u in outer normal direction. Following [17], we use a binary valued mask c to rewrite (1) in the form

$$\begin{aligned} c(x)\,(u(x) - f(x)) - (1 - c(x))\,\Delta u(x) &= 0, \quad \text{on } \Omega \\ \partial_n u(x) &= 0, \quad \text{on } \partial\Omega \setminus \partial\Omega_K \end{aligned} \tag{2}$$

where c is the characteristic function of the set Ω_K, i.e. $c(x) = 1$ if $x \in \Omega_K$ and 0 otherwise. It was shown in [17] that a careful tuning of the data can lead to tremendous quality gains in the reconstruction. Related investigations can also be found in [12], where the authors present subdivision strategies that exploit non-linear PDEs. We also note that the problem considered in this paper is closely related to the domain of shape, or more generally, topology optimisation. We refer to [14,23] for general introductions to this topic and to [1] for a more specific analysis in the context of image inpainting with homogeneous inpainting.

We will use (2) as a starting point and relax the restrictions placed upon the function c by allowing it to take a continuous range of values. Secondly, we will add a strictly convex energy to the PDE to penalise poor reconstructions and non-sparse sets of interpolation data. Our complete framework can then be regarded as an optimal control problem and presents a large-scale optimisation task with a strictly convex but non-differentiable objective and non-convex constraints. The concrete formulation of the model will be derived in Section 2. In Section 3 we will present a strategy to handle the occurring difficulties in this optimisation problem. The underlying idea of our solver will be to replace the original problem by a series of simpler convex optimisation problems that can be solved efficiently. Section 4 will provide some insight in our framework by stating optimality conditions and requirements for a monotonic convergence towards a solution. Finally, in Section 5, we will present experiments that show the general usefulness of our model both in the 1-D and 2-D setting.

2 A New Model for Finding Optimal Interpolation Data

If $c(x) \in \{0, 1\}$ for all $x \in \Omega$, then (1) and (2) represent equivalent formulations of the same boundary value problem. Interestingly, the latter equation makes also sense if c is allowed to take a continuous range of values such as $\mathbb{R}$. One may

regard continuously-valued functions c as a relaxation of the initial formulation which is much easier to handle. Our goal will be to optimise such masks c with respect to the accuracy of the reconstruction and to the sparsity of the interpolation data. Note that these two objectives cannot be perfectly fulfilled at the same time. If $c\,(x) \equiv 1$, then the reconstruction obtained by solving (2) is perfect. On the other hand, the sparsest possible choice would be $c\,(x) \equiv 0$ which does not allow any reconstruction at all. Therefore, we suggest to complement (2) by an energy that reflects exactly this trade-off between the quality of the reconstruction and the amount of interpolation data. The constrained optimisation problem that we will consider is

$$\begin{aligned} \underset{u,c}{\arg\min}\ E_{\lambda,\varepsilon}\,(u,c) &:= \int_\Omega \frac{1}{2}\,(u\,(x) - f\,(x))^2 + \lambda\,|c\,(x)| + \frac{\varepsilon}{2} c\,(x)^2\,\mathrm{d}x \\ \text{such that:}\quad c\,(x)\,(u\,(x) - f\,(x)) - (1 - c\,(x))\,\Delta u\,(x) &= 0 \quad \text{on } \Omega \\ \partial_n u &= 0 \quad \text{on } \partial\Omega \setminus \partial\Omega_K \end{aligned} \tag{3}$$

with positive parameters λ and ε. The first term of the energy penalises deviations of the reconstruction from the original data f. As in many other imaging applications such as image segmentation [20], we encourage a sparse mask by penalising the L_1 norm of c. The choice of λ lets us steer the sparsity of the mask. For $\lambda = 0$, $c(x) \equiv 1$ is the optimal solution. If λ increases, the mask will become sparser. Finally, $\lambda \to \infty$ will require $c(x) \equiv 0$.

As we will see in the forthcoming section, our numerical solver will require us to solve intermediate problems with a linear instead of non-convex constraint. These problems are related to optimal control problems of the form

$$\begin{aligned} \underset{u,c}{\arg\min}\ \int_\Omega \frac{1}{2}\,(u\,(x) - g\,(x))^2 + \lambda\,|c\,(x)| + \frac{\varepsilon}{2} c\,(x)^2\,\mathrm{d}x \\ \text{such that:}\quad Lu = f + c \end{aligned} \tag{4}$$

with a second-order elliptic and linear differential operator L, a state u, a control variable c and given data g and f. Existence and regularity of such formulations is analysed in [7,24,26]. The problem in (4) may not necessarily have a solution c if $\varepsilon = 0$, unless one resorts to measures [7]. In order to avoid such ill-posed formulations, one has to fix ε at a small positive value. A convergence analysis for $\varepsilon \to 0$ is presented in [26]. Although the analytical discussion of our model in (3) is out of the scope of this paper, we remark that we include the penaliser on the squared L_2 norm of c for the same regularity reasons. Alternatively one could also introduce box constraints of the form $a \leqslant u(x) \leqslant b$ on u as discussed in [24].

3 A Solution Strategy

Our optimal control model proposed in (3) is challenging for two reasons. First of all, the energy contains a non-differentiable term and secondly, the occurring mixed products $c(x)u(x)$ and $c(x)\Delta u(x)$ render the set of points (u, c) that fulfil

the PDE non-convex. In order to devise a solution strategy, we opt for a discretise-first-then-optimise approach. We discretise the data on a regular grid and reorder it into a vector of length n. The discrete analogue of (3) is given by

$$\begin{aligned} &\underset{(u,c)\in\mathbb{R}^n\times\mathbb{R}^n}{\arg\min} \quad \frac{1}{2}\|u-f\|_2^2 + \lambda\|c\|_1 + \frac{\varepsilon}{2}\|c\|_2^2 \\ &\text{such that: } \operatorname{diag}(c)\,(u-f) - (I-\operatorname{diag}(c))\,Du = 0 \end{aligned} \tag{5}$$

where the vectors u, f and c from $\mathbb{R}^n$ denote our reconstruction, the initial data and the corresponding discretisation of the confidence function, respectively. The operator $\operatorname{diag}(\cdot)$ represents the diagonal matrix in $\mathbb{R}^{n\times n}$ with its argument as the main diagonal. Finally, I is the identity matrix and D the discrete version of the Laplace operator with Neumann boundary conditions.

In order to tackle (5) numerically, we will replace it by a series of simpler *convex* optimisation problems. This idea is related to several well-known methods from the literature. One of the simplest strategies is known as *sequential linear programming (SLP)* [13]. SLP methods replace a single nonlinear optimisation problem by a sequence of linear programs. These linear programs are obtained through a first-order Taylor approximation of the objective and the constraints. Even though this method sounds appealing because it significantly reduces the complexity of the problem, it has a major drawback. In order to achieve an accurate result, the solution must lie at a vertex of the linearised constraint space. This requirement is usually not fulfilled. As an alternative, one may consider *linearly constrained Lagrangian* methods *(LCL)* [11,19]. They differ from SLP formulations by the fact that they do not linearise the objective function. They only consider a linear approximation of the constraints and try to minimise the (augmented) Lagrangian of the original problem. In [21] it is shown that under suitable conditions one can achieve quadratic convergence rates with LCL methods.

The main difference between these methods and ours will be the treatment of the objective function. We keep the original energy and merely augment it by an additional penalty term. This way we can circumvent the need to differentiate the objective and provide an alternative approach to LCL methods that often require the involved data to be differentiable. A similar strategy to ours is also briefly mentioned in [25] as a possibility to derive optimality conditions for nonlinear optimal control problems. Furthermore, similar approaches that exploit a partial linearisation of the considered optimisation problem have recently been analysed in [27].

As already mentioned, our goal is to replace the problem in (5) by a series of convex problems that are easier to solve. Therefore, we will replace the constraints by linear counterparts that approximate the original conditions. We define a mapping T which evaluates the constraints for given vectors u and c.

$$T : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n, \quad (u,c) \mapsto \operatorname{diag}(c)\,(u-f) - (I-\operatorname{diag}(c))\,Du \tag{6}$$

Its first-order approximation around some point $(\bar{u},\bar{c})$ can be written as

$$T(u,c) \approx T(\bar{u},\bar{c}) + D_u T(\bar{u},\bar{c})\,(u-\bar{u}) + D_c T(\bar{u},\bar{c})\,(c-\bar{c}) \tag{7}$$

where $D_u T(\bar{u}, \bar{c})$ and $D_c T(\bar{u}, \bar{c})$ describe the Jacobi matrices for the differentiation with respect to u and c at position $(\bar{u}, \bar{c})$. It is easy to check that

$$D_u T(\bar{u}, \bar{c}) = \operatorname{diag}(\bar{c}) - (I - \operatorname{diag}(\bar{c}))\, D \tag{8}$$

$$D_c T(\bar{u}, \bar{c}) = \operatorname{diag}(\bar{u} - f + D\bar{u}) \tag{9}$$

If $(\bar{u}, \bar{c})$ is a feasible solution of the constraints in (5), then $T(\bar{u}, \bar{c}) = 0$ and our initial problem is approximated by

$$\begin{aligned} &\underset{(u,c)\in\mathbb{R}^n\times\mathbb{R}^n}{\arg\min} \ \frac{1}{2}\|u - f\|_2^2 + \lambda\|c\|_1 + \frac{\varepsilon}{2}\|c\|_2^2 \\ &\text{such that: } D_u T(\bar{u}, \bar{c})\,(u - \bar{u}) + D_c T(\bar{u}, \bar{c})\,(c - \bar{c}) = 0 \end{aligned} \tag{10}$$

However, the previous formulation is only reliable for pairs (u, c) in a neighbourhood of $(\bar{u}, \bar{c})$. Therefore, we additionally penalise large deviations from $(\bar{u}, \bar{c})$ by adding an additional proximal term $\frac{\mu}{2}\|c - \bar{c}\|_2^2$ with a positive weight μ.

$$\begin{aligned} &\underset{(u,c)\in\mathbb{R}^n\times\mathbb{R}^n}{\arg\min} \ \frac{1}{2}\|u - f\|_2^2 + \lambda\|c\|_1 + \frac{\varepsilon}{2}\|c\|_2^2 + \frac{\mu}{2}\|c - \bar{c}\|_2^2 \\ &\text{such that: } D_u T(\bar{u}, \bar{c})\,(u - \bar{u}) + D_c T(\bar{u}, \bar{c})\,(c - \bar{c}) = 0 \end{aligned} \tag{11}$$

Note that u depends continuously on c. Therefore, there is no need to introduce an additional proximal term for this variable. For the sake of brevity, we define $A := D_u T(\bar{u}, \bar{c})$ and $B := D_c T(\bar{u}, \bar{c})$. A simple computation yields $A\bar{u} + B\bar{c} = \operatorname{diag}(\bar{c})\,(I + D)\,\bar{u} =: g$. This leads us to the final form of our discrete approximation of (3):

$$\begin{aligned} &\underset{(u,c)\in\mathbb{R}^n\times\mathbb{R}^n}{\arg\min} \ \frac{1}{2}\|u - f\|_2^2 + \lambda\|c\|_1 + \frac{\varepsilon}{2}\|c\|_2^2 + \frac{\mu}{2}\|c - \bar{c}\|_2^2 \\ &\text{such that: } Au + Bc = g \end{aligned} \tag{12}$$

Equation (12) is a constrained optimisation problem with a strictly convex cost and linear constraints. Such problems are well studied and many highly efficient algorithms exist. Note that A is the system matrix one would obtain by discretising (2) with known and fixed c and sought u. In [16] it was shown that this matrix is invertible if all the c_i take only the values 0 or 1 with at least one $c_i = 1$. However, it is easy to see that the same argument carries over to our case with the requirement that $c_i > 0$ for at least one i.

In order to solve the problem occurring in (12), we use a primal-dual algorithm for convex problems from Esser et al. [10] and Chambolle and Pock [5] where it is referred to as PDHGMu and Algorithm 1, respectively. For convex functions F, G and a linear operator K this algorithm solves

$$\underset{x}{\arg\min} \ G(x) + F(Kx) \tag{13}$$

by computing iteratively

$$\begin{aligned} y^{k+1} &= \arg\min_y \frac{1}{2}\|y - \left(y^k + \sigma K \hat{x}^k\right)\|_2^2 + \sigma F^*(y) \\ x^{k+1} &= \arg\min_x \frac{1}{2}\|x - \left(x^k - \tau K^\top y^{k+1}\right)\|_2^2 + \tau G(x) \\ \hat{x}^{k+1} &= x^{k+1} + \theta\left(x^{k+1} - x^k\right) \end{aligned} \tag{14}$$

where F^* is the Fenchel conjugate of F. It was shown in [5] that if $\tau\sigma\|K\|_2^2 < 1$, $\theta \in [0,1]$ and if the primal problem from (13) as well as its dual have a solution, the sequences $(x^k)_k$, $(y^k)_k$ generated by the above algorithm converge to a solution of the primal and dual problem, respectively. We apply this algorithm with

$$G(u,c) := \frac{1}{2}\|u - f\|_2^2 + \lambda\|c\|_1 + \frac{\varepsilon}{2}\|c\|_2^2 + \frac{\mu}{2}\|c - \bar{c}\|_2^2 \tag{15}$$

$$F(Au + Bc) := \iota_g(Au + Bc) \tag{16}$$

where the indicator function ι_g is given by

$$\iota_g(x) := \begin{cases} 0, & \text{if } x = g, \\ \infty, & \text{else.} \end{cases} \tag{17}$$

We note that $F^*(x) = \langle x, g\rangle$. This gives us the algorithm depicted in Algorithm 1. Note that the updates of y^k and u^k only require matrix-vector products and can be carried out in a very fast way. The update of c^k can also be expressed in

Algorithm 1. Minimisation strategy for solving (12).

Input : N number of iterations.
Output: Vectors u and c solving (12)

Choose τ, $\sigma > 0$ such that $\sigma\tau\|\begin{pmatrix} A & B \end{pmatrix}\|_2^2 < 1$ and $\theta \in [0,1]$.
Set u^0, c^0, y^0 arbitrary and set $\hat{u}^0 = u^0$ and $\hat{c}^0 = c^0$.
for k **from** *1* **to** N **do**
  $y^{k+1} = \arg\min_z \ \frac{1}{2}\|z - \left(y^k + \sigma\left(A\hat{u}^k + B\hat{c}^k\right)\right)\|_2^2 + \sigma\langle z, g\rangle$
  $u^{k+1} = \arg\min_z \ \frac{1}{2}\|z - \left(u^k - \tau A^\top y^{k+1}\right)\|_2^2 + \frac{\tau}{2}\|z - f\|_2^2$
  $c^{k+1} = \arg\min_z \ \frac{1}{2}\|z - \left(c^k - \tau B^\top y^{k+1}\right)\|_2^2 + \tau\left(\frac{\varepsilon}{2}\|z\|_2^2 + \frac{\mu}{2}\|z - \bar{c}\|_2^2 + \lambda\|z\|_1\right)$
  $\hat{u}^{k+1} = u^{k+1} + \theta\left(u^{k+1} - u^k\right)$
  $\hat{c}^{k+1} = c^{k+1} + \theta\left(c^{k+1} - c^k\right)$
end
Set $c \leftarrow c^N$ and $u \leftarrow u^N$.
Return *Optimal solutions* c *and* u.

closed form using the soft shrinkage operator shrink_α given by

$$\text{shrink}_\alpha(x) := \text{sgn}(x) \cdot \max(|x| - \alpha, 0) \tag{18}$$

For vector-valued arguments, the shrinkage is performed componentwise. With this definition, updating c comes down to computing

$$c^{k+1} = \text{shrink}_{\frac{\tau\lambda}{1+\tau\varepsilon+\tau\mu}} \left(\frac{c - \tau B^\top y^{k+1} + \tau\mu\bar{c}}{1 + \tau\varepsilon + \tau\mu} \right) \tag{19}$$

which again consists only of matrix vector multiplications and can be carried out in a very fast way.

Algorithm 2. Minimisation strategy for solving (5).

Input : Initial mask c, parameters λ, ε, μ and N the number of iterations.
Output: Vectors u and c solving (5)

for k **from** *1* **to** N **do**
  Set $\bar{c}^k \leftarrow c$.
  Solve $\left(\text{diag}(\bar{c}^k) - \left(I - \text{diag}(\bar{c}^k)\right) D\right) \bar{u}^k = \text{diag}(\bar{c}^k) f$ for $\bar{u}^k$.
  Compute first-order approximation of $T\,(u, c)$ around the $\left(\bar{u}^k, \bar{c}^k\right)$.
  Obtain c by solving (12) with Algorithm 1.
end
Compute u from c.
Return *Optimal solutions* c *and* u.

It is important to note that the pair (u, c) obtained from Algorithm 1 is in general not a feasible point for the problem stated in (3). It only represents a solution of (12). Therefore, we discard the vector u and plug c into the constraint of (3) to obtain a linear system of equations which can be solved to obtain a new feasible pair $(\bar{u}, \bar{c})$. Using these vectors, we compute a new first-order approximation of the previously defined function T and repeat all the steps. This iterative scheme is repeated until convergence. The complete algorithm to solve (5) is given in Algorithm 2. Let us remark at this point that matrix factorisation and completion problems have a similar non-convex structure as the problems presented in this work. Hence, similar methods as ours were recently proposed in [15,27,28].

4 Theoretical Properties of Our New Scheme

4.1 Optimality Conditions

The linearised problem from (12) seeks the minimiser of a strictly convex energy under linear constraints. It is a well known fact that for such problems the Karush-Kuhn-Tucker conditions yield necessary and sufficient optimality conditions. They are given by the following lemma.

Lemma 1. *The Lagrangian associated with the optimisation problem from* (12) *is given by*

$$L\,(u, c, p) := \frac{1}{2}\|u - f\|_2^2 + \lambda\|c\|_1 + \frac{\varepsilon}{2}\|c\|_2^2 + \frac{\mu}{2}\|c - \bar{c}\|_2^2 + \langle p, Au + Bc - g\rangle\ . \tag{20}$$

The corresponding first-order optimality conditions are given by

$$\begin{aligned} u - f + A^\top p &= 0 \\ \lambda \partial \left(\|\cdot\|_1 \right)(c) + \varepsilon c + \mu \left(c - \bar{c} \right) + B^\top p &\ni 0 \\ Au + Bc - g &= 0 \end{aligned} \tag{21}$$

where $\partial \left(\|\cdot\|_1 \right)(c)$ *denotes the subdifferential of* $\|\cdot\|_1$ *at point* c*.*

In a similar manner, one can also derive necessary optimality conditions for (5). We refer to [3,6,25] for more details.

Lemma 2. *The following system of equations yields necessary optimality conditions for* (5)*. For a solution* $(u, c) \in \mathbb{R}^n \times \mathbb{R}^n$ *of* (5) *there must exist a* $p \in \mathbb{R}^n$ *such that the following relations are fulfilled.*

$$\begin{aligned} u - f + D_u T(u, c)^\top p &= 0 \\ \lambda \partial \left(\|\cdot\|_1 \right)(c) + \varepsilon c + D_c T(u, c)^\top p &\ni 0 \\ T(u, c) &= 0 \end{aligned} \tag{22}$$

Using the previous two lemmas we are able to show that fixed points of our Algorithm 2 fulfil the necessary optimality conditions.

Proposition 1. *If Algorithm 1 has reached a fixed point with respect to its input (e.g.* $u = \bar{u}^k$ *and* $c = \bar{c}^k$ *in lines 5 and 7 of Algorithm 2), then* u *and* c *also fulfil the necessary optimality conditions from Lemma 2.*

Proof. By construction, u *and* c *fulfil the optimality conditions from* (21)*. Since* $c = \bar{c}^k$ *and* $u = \bar{u}^k$*, it follows that*

$$A = D_u T\left(\bar{u}^k, \bar{c}^k\right) = D_u T(u, c) \quad \text{and} \quad B = D_c T\left(\bar{u}^k, \bar{c}^k\right) = D_c T(u, c) \tag{23}$$

and hence, the first two equations from (21) *and* (22) *coincide. Further,* $\bar{u}^k$ *and* $\bar{c}^k$ *are by construction feasible, thus it follows that* $T(u, c) = T\left(\bar{u}^k, \bar{c}^k\right) = 0$*.* □

The following proposition states under which condition the iterates of Algorithm 2 will monotonically decrease the energy.

Proposition 2. *If we denote the solutions obtained from Algorithm 1 by* (u, c) *and let* $\tilde{u}$ *fulfil* $T(\tilde{u}, c) = 0$*, then* $(\tilde{u}, c)$ *will be feasible iterates that decrease the energy in every iteration as long as the following condition is valid.*

$$\frac{1}{2}\left(\|\tilde{u} - f\|_2^2 - \|\bar{u}^k - f\|_2^2 \right) \leqslant \lambda \|\bar{c}^k\|_1 + \frac{\varepsilon}{2}\|\bar{c}^k\|_2^2 - \left(\lambda \|c\|_1 + \frac{\varepsilon}{2}\|c\|_2^2 \right) \tag{24}$$

Proof. By definition of $(\hat{u}, c)$ *and* $\left(\bar{u}^k, \bar{c}^k\right)$ *we have*

$$\begin{aligned} \frac{1}{2}\|\hat{u} - f\|_2^2 + \lambda \|c\|_1 + \frac{\varepsilon}{2}\|c\|_2^2 &\leqslant \frac{1}{2}\|\hat{u} - f\|_2^2 + \lambda \|c\|_1 + \frac{\varepsilon}{2}\|c\|_2^2 + \frac{\mu}{2}\|c - \bar{c}^k\|_2^2 \\ &\leqslant \frac{1}{2}\|\bar{u}^k - f\|_2^2 + \lambda \|\bar{c}^k\|_1 + \frac{\varepsilon}{2}\|\bar{c}^k\|_2^2 \,. \end{aligned} \tag{25}$$

Replacing $\hat{u}$ *by* $\tilde{u}$ *and reordering the terms yields the sought expression.* □

Equation (24) allows an interesting interpretation. The left-hand side can be seen as the loss in accuracy whereas the right-hand side can be considered as the gain in sparseness. Thus, the energy will be decreasing as long as the gain in sparseness outweighs the loss in precision.

5 Experiments

5.1 Signals in 1-D

To demonstrate the performance of our approach we have chosen the piecewise polynomial and non-continuous signal Piece-Polynomial from the WAVELAB 850 Toolbox [4] and normalised it to the interval $[0, 1]$ to ease the simultaneous visualisation of signal, reconstruction and mask. The result is shown in Figure 1. Note that the obtained mask is sparse and the non-zero entries are placed at positions where one would naturally expect them, e.g. two mask points are used to encode a step in the signal. Also note the excellent quality of the reconstruction.

We initialise our method with u being the original signal and a full mask, i.e. $c_i = 1$ for all i, and use the parameters $\varepsilon = 10^{-9}$, $\mu = 1.0$, $\lambda = 0.02$. For Algorithm 1 we set $\theta = 1$ and $\tau = 0.25$. In order to fulfil the step length constraint $\tau\sigma L^2 < 1$ where $L = \|(A\ B)\|$ we approximate L through power iterations and set $\sigma = \left(\left(L^2 + 0.1\right)\tau\right)^{-1}$. The method aborted when the distance between two consecutive iterates dropped below $3 \cdot 10^{-16}$. In order to reach this precision we required about 225000 iterations of Algorithm 1. After 630 iterations of Algorithm 2 the distance between two consecutive iterates c had dropped below 10^{-15} at which point the algorithm stopped. The whole algorithm was implemented Matlab with Algorithm 1 as a mex function written in ANSI C. All the tests were done on a desktop PC with an Intel Xeon processor (3.2GHz) and 24GB of memory. The total runtime was roughly 10 minutes. The squared Euclidean distance between the input signal and the reconstruction is 0.0377.

5.2 Images in 2-D

To show that our approach performs as well on 2-D data sets as it does on 1-D signals, we apply our algorithm to three different test images and compare our method to the state-of-the-art approach from [17]. In [17] the authors proposed a greedy method, called stochastic sparsification, that iteratively selects a set of candidate points and discards those pixels that yield the smallest increase in the error when removed from the mask. This step is repeated until a desired density is reached. In a second step, called non-local pixel exchange, mask and non-mask pixels are swapped. If the reconstruction error increases, the swap is undone, otherwise it is kept. This latter step is repeated until the desired error or the maximal number of swaps is reached.

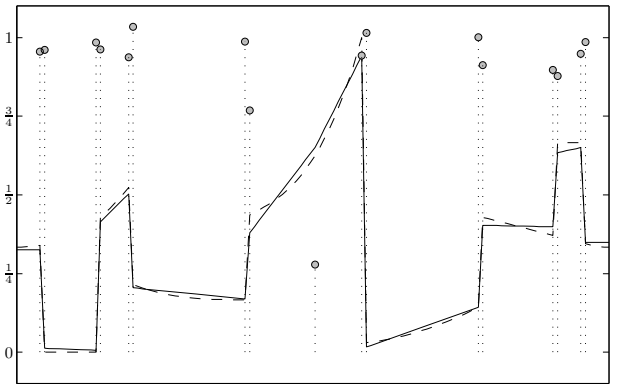


Fig. 1. The original signal (dashed line), the reconstruction (solid line) as well as the used mask (grey dots). As expected, the mask is sparse (17 non-zero entries out of 128) and not binary-valued. The mask point in the middle of the signal with the smallest value allows to better adapt to the curvature of the input signal by blending the diffusion result with the data. Also note that the mask entries neatly align with the discontinuities of the signal.

The results of our method are depicted in Fig. 2 and a summary with a comparison to the approach from [17] is given in Tab. 1. As an error measure we use the mean squared error (MSE) which is computed by

$$\mathrm{MSE}\,(f,u) := \frac{1}{n}\sum_{i=1}^{n}(f_i - u_i)^2 \tag{26}$$

where the two images f and u to be compared have been reshaped into vectors of length n. For the computation of the MSE we assume that the image values lie in the interval $[0, 255]$.

The third column in Tab. 1 represents the MSE by using the data as it is returned by Algorithm 2. Binarising the mask by thresholding it at 0.01 significantly increases the reconstruction error as we can see in the fourth column of Tab. 1. In [17] an additional optimisation step on the interpolation data was presented. The idea is to decrease the reconstruction error by optimising the image values for a fixed mask. This so called grey value optimisation (GVO) can clearly be used as a postprocessing step in combination with our masks.

The GVO is a least squares formulation to minimise the reconstruction error. It solves

$$\underset{(u,g)\in\mathbb{R}^n\times\mathbb{R}^n}{\arg\min} \|u-f\|_2^2 \quad \text{such that} \quad \mathrm{diag}\,(c)\,(u-g) - (I - \mathrm{diag}\,(c))\,Du = 0 \tag{27}$$

with a fixed c and known image f. Let us define the matrix $M := \mathrm{diag}\,(c) - (I - \mathrm{diag}\,(c))\,D$. Note that this matrix is identical to $D_u T\,(\bar{u}, \bar{c})$ given in (8) for which we know that it is an invertible matrix as long as at least one entry in

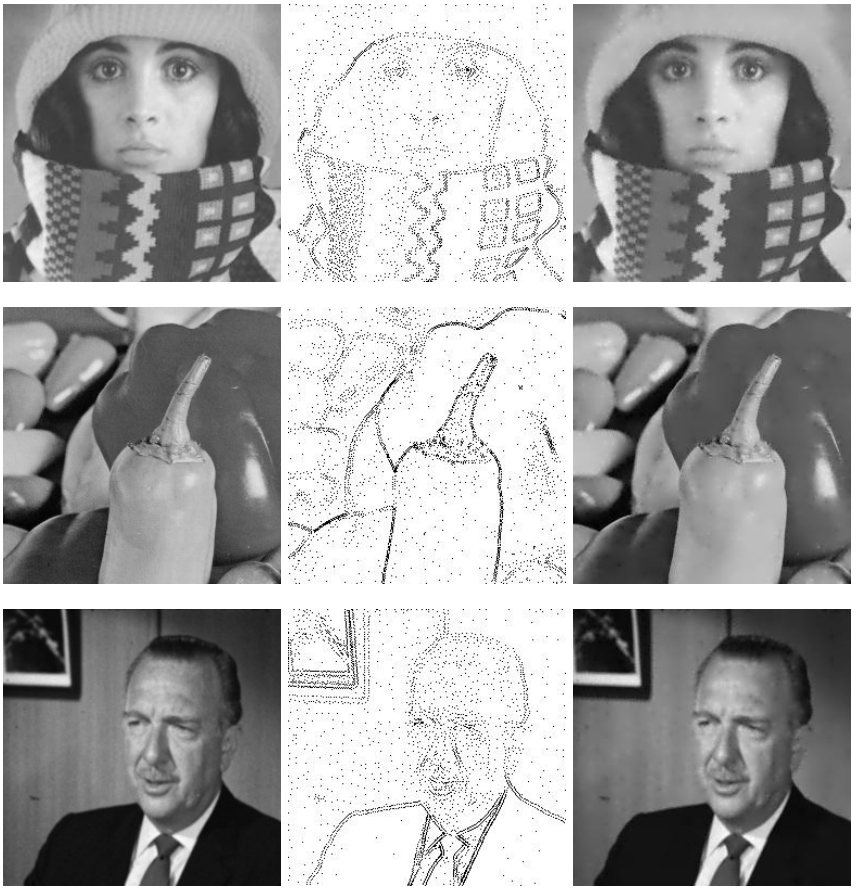


Fig. 2. First row (from left to right): Test image *Trui* (256×256), mask with the locations of the interpolation points (marked in black) with 4.95% of non-zero entries, reconstruction with grey-value optimisation and binarised mask (MSE = 16.95). Note that the bright spots visible in the face are an artifact stemming from the fact that the Laplace operator is used. Parameters: $\lambda = 0.0036$, $\mu = 1.25$, $\varepsilon = 10^{-4}$, 3600 outer iterations and 25000 iterations of Algorithm 1. The parameter choices for the Algorithm 1 are identical to those mentioned in the caption of Figure 1. As initialisation we used $u = f$ and $c \equiv 1$. **Second row (from left to right):** Test image *Peppers* (256×256), mask with the locations of the interpolation points (marked in black) with 5.02% of non-zero entries, reconstruction with grey-value optimisation and binarised mask (MSE = 18.44). Parameters: $\lambda = 0.0034$, $\mu = 0.01$, $\varepsilon = 10^{-9}$, 130 outer iterations and 100000 iterations of Algorithm 1. The remaining parameters are identical as in the experiment with *Trui*. **Third row (from left to right):** Test image *Walter* (256×256), mask with the locations of the interpolation points (marked in black) with 5.00% of non-zero entries, reconstruction with grey-value optimisation and binary mask (MSE = 7.59). Parameters: $\lambda = 0.0018$, 430 outer iterations and 100000 iterations of Algorithm 1. All the other parameters are identical to those used for the *Peppers* experiment.

Table 1. MSE for our experimental results. If combined with a grey value optimisation, our binarised mask is always better than the one obtained in [17]. Note that the mask density in [17] was exactly 5% for each image. For our experiments we had a density of 4.95% for *Trui*, 5.02% for *Peppers* and 5.00% for *Walter*. The best results for each image are marked in boldface. Unavailable results are marked with a —.

Image	Algorithm	Continuous c	Binary c	Continuous c with GVO	Binary c with GVO
Trui	Our method	17.41	47.89	16.98	**16.95**
	Method of [17]	—	23.21	—	17.17
Peppers	Our method	18.91	30.51	**18.44**	**18.44**
	Method of [17]	—	—	—	19.38
Walter	Our method	07.81	19.61	**07.59**	**07.59**
	Method of [17]	—	—	—	08.14

c is non-zero. Using that $u = M^{-1}\operatorname{diag}(c)g$, it follows that the GVO can be formulated as

$$g = \operatorname*{arg\,min}_{x\in\mathbb{R}^n} \frac{1}{2}\|M^{-1}\operatorname{diag}(c)\,x - f\|_2^2\,. \tag{28}$$

Applying the GVO to the continuous masks only yields minimal gains. However, it completely compensates for the loss of accuracy by binarising the mask. This is a remarkable observation and greatly increases the potential usefulness of this method for image compression tasks since it cuts the necessary amount of data to be stored for the reconstruction in half. Let us emphasise that neither the binarised nor the results with additional GVO represent solutions of our optimal control formulation. Even though the binarised mask fulfils the constraints of our minimisation problem it yields a higher energy. The results with GVO do not, in general, fulfil the constraints of (5). The results from Tab. 1 clearly show the superiority of our approach. Except for *Trui*, our continuous mask already improves the quality from [17]. With the additional GVO we can further reduce our error measures and are able to outperform the results from [17] in every test case.

A Greedy Speed-Up Strategy. In the above experiments we used very large numbers of iterations in order to find the optimal error. This results in computation times of several hours, similarly as the methods in [17]. For practical purposes one can significantly reduce the number of iterations in Algorithm 1. Moreover, the results from Tab. 1 suggest an interesting heuristic to further speed up the computation of the mask. Our algorithm, as it is presented in this paper, starts with a full mask which is made gradually sparser. Since the smallest errors are obtained with a binarised mask combined with a GVO, it is not really necessary to know the exact value of the mask at every position. Only if it will be 0 or not. Therefore, one can threshold the mask during the iterations and check whether this thresholded version differs significantly from the thresholded mask of the previous iterate. If not, one aborts the iteration. Using this heuristic with

$\lambda = 0.00325$, $\mu = 0.01$, $\varepsilon = 10^{-9}$, 1250 iterations of Algorithm 1 and 50 outer iterations we obtain a binary mask with a density of 5.01%. In combination with GVO, the MSE for the Peppers image was 19.38, which is identical to the result from [17]. The total runtime was 272 seconds. Even though the obtained mask yields a slightly larger error when compared to the results from Tab. 1, the runtime was reduced from 15 hours down to less than 5 minutes. This corresponds to a speed-up factor of 180.

6 Conclusion

We present a new model for the determination of optimal interpolation data for homogeneous diffusion inpainting. The formulation contains basically only a single parameter with a very intuitive meaning. An iterative strategy to solve the corresponding discrete optimisation problem is also derived. Numerical experiments show the usefulness of the suggested method. We note that the our method is not restricted to the Laplacian as differential operator. An extension using more powerful operators is possible. Therefore, a thorough theoretical analysis of the framework presented in this work as well as the extension to other PDE-based interpolation methods for image compression tasks will be the subject of future work. We also note that our method is capable of handling non-differentiable penalties on the reconstruction, for example the L_1 norm. This will also be the topic of future work.

References

1. Belhachmi, Z., Bucur, D., Burgeth, B., Weickert, J.: How to choose interpolation data in images. SIAM Journal on Applied Mathematics 70(1), 333–352 (2009)
2. Bertalmío, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: Proc. 27th Annual Conference on Computer Graphics and Interactive Techniques, pp. 417–424. ACM Press/Addison-Wesley Publishing Company (2000)
3. Bonnans, J.F., Shapiro, A.: Perturbation Analysis of Optimization Problems. Springer Series in Operations Research. Springer (2000)
4. Buckheit, J., Chen, S.S., Donoho, D., Huo, X., Johnstone, I., Levi, O., Scargle, J., Yu, T.: WAVELAB 850 toolbox for matlab (2012), `http://www-stat.stanford.edu/~wavelab/Wavelab_850/download.html`
5. Chambolle, A., Pock, T.: A first order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision 40(1), 120–145 (2011)
6. Clarke, F.H.: Optimization and Nonsmooth Analysis. SIAM (1990)
7. Clason, C., Kunisch, K.: A duality-based approach to elliptic control problems in non-reflexive banach spaces. ESAIM: Control, Optimisation and Calculus of Variations 17(1), 243–266 (2011)
8. Demaret, L., Dyn, N., Iske, A.: Image compression by linear splines over adaptive triangulations. Signal Processing 86(7), 1604–1616 (2006)
9. Demaret, L., Iske, A.: Advances in digital image compression by adaptive thinning. Annals of the MCFA 3, 105–109 (2004)

10. Esser, E., Zhang, X., Chan, T.F.: A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. SIAM Journal on Imaging Sciences 3(4), 1015–1046 (2010)
11. Friedlander, M.P., Saunders, M.A.: A globally convergent linearly constrained lagrangian method for nonlinear optimization. SIAM Journal on Optimization 15(3), 863–897 (2005)
12. Galić, I., Weickert, J., Welk, M., Bruhn, A., Belyaev, A., Seidel, H.P.: Image compression with anisotropic diffusion. Journal of Mathematical Imaging and Vision 31(2-3), 255–269 (2008)
13. Griffith, R.E., Stewart, R.A.: A nonlinear programming technique for the optimization of continuous processing systems. Management Science 7(4), 379–392 (1961)
14. Haslinger, J., Mäkinen, R.A.E.: Introduction to Shape Optimization: Theory, Approximation, and Computation. SIAM (1987)
15. Lin, C.J.: Projected gradient methods for nonnegative matrix factorization. Neural Computation 19(10), 2756–2779 (2007)
16. Mainberger, M., Bruhn, A., Weickert, J., Forchhammer, S.: Edge-based image compression of cartoon-like images with homogeneous diffusion. Pattern Recognition 44(9), 1859–1873 (2011)
17. Mainberger, M., Hoffmann, S., Weickert, J., Tang, C.H., Johannsen, D., Neumann, F., Doerr, B.: Optimising spatial and tonal data for homogeneous diffusion inpainting. In: Bruckstein, A.M., ter Haar Romeny, B.M., Bronstein, A.M., Bronstein, M.M. (eds.) SSVM 2011. LNCS, vol. 6667, pp. 26–37. Springer, Heidelberg (2012)
18. Masnou, S., Morel, J.M.: Level lines based disocclusion. In: Proc. of the International Conference on Image Processing, vol. 3, pp. 259–263. IEEE (1998)
19. Murthagh, B.A., Saunders, M.A.: A projected lagrangian algorithm and its implementation for sparse nonlinear constraints. Mathematical Programming Study 16, 84–117 (1982)
20. Pock, T., Schoenemann, T., Graber, G., Bischof, H., Cremers, D.: A convex formulation of continuous multi-label problems. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 792–805. Springer, Heidelberg (2008)
21. Robinson, S.M.: A quadratically-convergent algorithm for general nonlinear programming problems. Mathematical Programming 3, 145–156 (1972)
22. Schmaltz, C., Weickert, J., Bruhn, A.: Beating the quality of JPEG 2000 with anisotropic diffusion. In: Denzler, J., Notni, G., Süße, H. (eds.) DAGM 2009. LNCS, vol. 5748, pp. 452–461. Springer, Heidelberg (2009)
23. Sokolowski, J., Zolesio, J.P.: Introduction to Shape Optimization. Springer (1992)
24. Stadler, G.: Elliptic optimal control problems with L_1-control cost and applications for the placement of control devices. Computational Optimization and Applications 44(2), 159–181 (2009)
25. Tröltzsch, F.: Optimale Steuerung Partieller Differentialgleichungen: Theorie, Verfahren und Anwendungen, 2nd edn. Vieweg+Teubner (2009)
26. Wachsmuth, G., Wachsmuth, D.: Convergence and regularization results for optimal control problems with sparsity functional. ESAIM: Control, Optimisation and Calculus of Variations 17(3), 858–886 (2011)
27. Xu, Y., Yin, W.: A block coordinate descent method for multi-convex optimization with applications to nonnegative tensor factorization and completion. Rice CAAM Technical Report TR12-15, Rice University (2012)
28. Xu, Y., Yin, W., Wen, Z., Zhang, Y.: An alternating direction algorithm for matrix completion with nonnegative factors. Frontiers of Mathematics in China 7(2), 365–384 (2012)

Curvature Regularization for Resolution-Independent Images

John MacCormick[1] and Andrew Fitzgibbon[2]

[1] Department of Computer Science, Dickinson College, USA
[2] Microsoft Research, Cambridge, UK

Abstract. A resolution-independent image models the true intensity function underlying a standard image of discrete pixels. Previous work on resolution-independent images demonstrated their efficacy, primarily by employing regularizers that penalize discontinuity. This paper extends the approach by permitting the *curvature* of resolution-independent images to be regularized. The main theoretical contribution is a generalization of the well-known elastica energy for regularizing curvature. Experiments demonstrate that (i) incorporating curvature improves the quality of resolution-independent images, and (ii) the resulting images compare favorably with another state-of-the-art curvature regularization technique.

Keywords: curvature, elastica, regularization.

1 Introduction and Related Work

Viola *et al.* [19,20] introduced the notion of a *resolution-independent* latent image to model the true intensity function underlying a standard image of discrete pixels. Figure 1 gives an example of the approach: the true intensity function is approximated by a piecewise linear function u, whose linear patches are defined on a triangle mesh. The crucial feature is that the mesh's vertices are positioned with arbitrary precision, which frees the model from any notion of discrete pixels. The vertex positions and patch intensities are determined by minimizing an energy that includes a regularizer term, which models the prior expectations of resolution-independent images in general.

Previous work on resolution-independent images employed a regularizer based primarily on the *discontinuities* in u. The main contribution in this paper is to extend the regularizer to incorporate the *curvature* of u. Starting from the well-known *elastica* energy [2], we derive explicit expressions for computing the elastica energy on the smooth and non-smooth regions of the image domain. The non-smooth region includes *steps* and *corners* (defined rigorously later), leading to separate step energy and corner energy terms in the energy functional. The paper also includes practical experiments demonstrating the benefits of the approach and a favorable comparison with another state-of-the-art curvature regularizer.

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 165–179, 2013.
© Springer-Verlag Berlin Heidelberg 2013

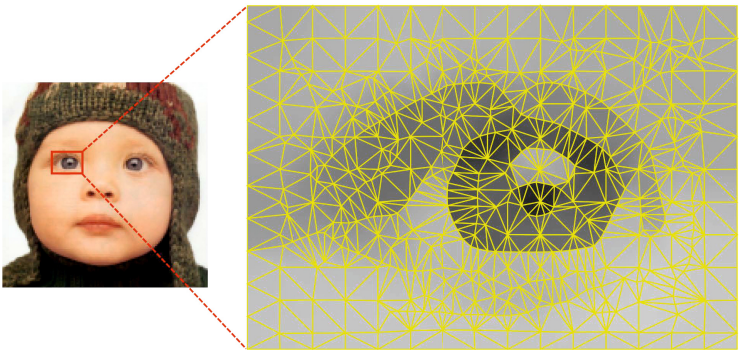


Fig. 1. In this paper, resolution-independent images are modeled as in Viola *et al.* [20], employing a piecewise linear intensity function defined on a triangle mesh whose vertices are positioned with arbitrary precision

There is a considerable amount of related work on curvature regularization, including a long history of variational and level set methods (e.g. [9,12,16]), methods derived from the elastica energy (e.g. [2,10]), and other approaches such as total curvature [4] and Gaussian derivatives [5]. The work of Schoenemann *et al.* [15] and Strandmark and Kahl [17] is most closely related to the present paper. These approaches regularize curvature based on a fixed [15] or adaptive [17] mesh, employing linear programming relaxations for optimization. However, the meshes are restricted to a fixed small set of edge angles, so lines not at those orientations must be jagged. In our work, all angles are equally treated (ignoring floating point issues). The curvature term contrasts with this paper in that it applies to binary images and to corners with exactly two prongs (as defined in Section 4.1); the approach here permits resolution-independent images with arbitrary intensities and multi-pronged corners. Hence, we believe the novel theoretical contribution of the paper is twofold: first, the well-known approach of regularizing curvature by minimizing an elastica energy is reformulated so that it can be applied explicitly to resolution-independent images (Sections 3 and 4); second, this reformulation leads to a corner energy that has not, to our knowledge, been studied previously (Section 4.1).

2 The Set of Resolution-Independent Images

At the core of our approach is the concept of a *resolution-independent image*. Formal mathematical definitions are given in our technical report [6]. Here, we rely primarily on intuition to convey the essential concepts. A resolution-independent image is produced by an idealized camera with infinite resolution, infinite color depth, infinite depth of field, and zero noise. The resulting image u is defined on a connected subset Ω of $\mathbb{R}^2$, with intensities in the continuous range $[0, 1]$. We assume the world consists of piecewise smoothly-varying objects, giving rise

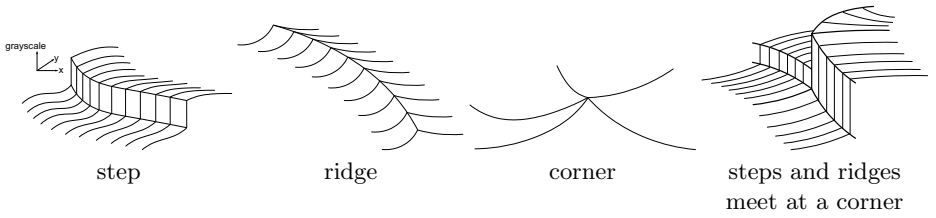


Fig. 2. Taxonomy of the jump set of a resolution-independent image. Each panel shows part of a 2D grayscale resolution-independent image, visualized as a surface. The image plane coincides with the horizontal x-y plane, and the grayscale intensity of the image is plotted on the vertical z axis, as indicated by the set of axes on the left.

to resolution-independent images that are also piecewise smooth. So Ω is partitioned into a *differentiable* region D (where u is continuously differentiable) and its complement J, termed the *jump set* (where u or its derivative is discontinuous). As shown in Figure 2, it is useful to further partition J into *steps*, *ridges*, and *corners*—so $\Omega = D \cup J_{\text{step}} \cup J_{\text{ridge}} \cup J_{\text{corner}}$. Some mild assumptions [6] guarantee that u is well-behaved near the jump set. In particular, $\lim_{x \to x_0} u(x)$ exists for any $x_0 \in J$, and is independent of the path used to approach x_0, provided the path remains in the differentiable region D.

3 Regularizers for Resolution-Independent Images

We are interested in imposing a prior on resolution-independent images u. This will be done via a real-valued regularizer $E(u)$, with the usual interpretation of E as an energy functional, so that u-functions with low values of E have high prior probability.

The *elastica energy* is a commonly-used regularizer for curvature in computer vision applications. The one-dimensional version of this energy, derived from the physical energy required to bend a thin pliable rod into a given smooth shape, was considered as early as 1744 by Euler ([3]; see this paper's appendix for details). For a smooth curve Γ parameterized by arc length s, it is given by

$$E_{\text{1D-elastica}}(\Gamma) = \int_s (a + b\kappa(\Gamma, s)^p)\, ds. \tag{1}$$

Here, $a, b, p \geq 0$ are constants and $\kappa(\Gamma, s)$ is the (unsigned) curvature of Γ at s, as defined in elementary geometry. Physics (and Euler) say that $p = 2$, but other values may give good results in computer vision applications.

Of more direct interest here is the generalization of elastica energy to two dimensions, as proposed by Masnou and Morel [8], and employed by many others (e.g. [15]). This two-dimensional elastica energy is given by

$$E_{\text{elastica}}(u) = \int_{\mathbf{x} \in \Omega} (a + b\kappa_{\text{LL}}(\mathbf{x})^p) |\nabla u(\mathbf{x})|\, d\mathbf{x}. \tag{2}$$

Here $\kappa_{\mathrm{LL}}(\mathbf{x})$ is the (unsigned, 1D) curvature of the level line of u passing through $\mathbf{x} \in \Omega$. Chan *et al.* [2] provide a detailed and illuminating derivation of the 2D elastica energy (2) from the 1D pliable-rod definition (1). The basic idea is to integrate the 1D version over levels l; the extra weight of $|\nabla u(\mathbf{x})|$ in (2) then appears as the Jacobian when transforming from height and arc-length parameters (l, s) to image plane parameters $\mathbf{x} = (x, y)$.

As with the one-dimensional elastica energy, the two-dimensional energy (2) has an intuitive physical interpretation: it is the total amount of energy that would be expended to build the image out of thin, horizontal, pliable rods, assuming the energy of each individual rod is given by Equation (1) multiplied by the height spacing δl between rods. Note that for this physical analogy to be appropriate, the rods must be horizontal (so that they correspond to level sets), and they should be placed at equally-spaced heights separated by δl. As we will be repeatedly appealing to this physical interpretation of the elastica energy later, let us call it the *pliable rod analogy*. As our first practical example, the next subsection calculates the elastica energy of a step in the image.

3.1 Computation of the Step Contribution

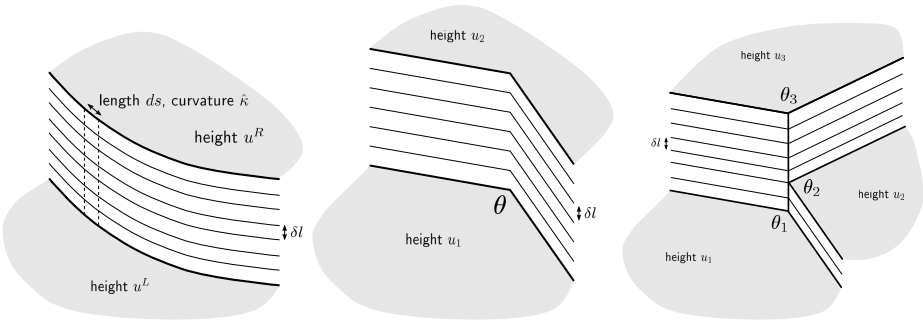
Consider a small portion ds of J_{step} shown in Figure 3(a), where the portion is small enough that we can approximate u^L and u^R as constant. To build this part of the image requires stacking horizontal rods directly on top of each other. Each individual rod has energy $\delta l(a + b\hat{\kappa}^p)\, ds$, by definition. The total height of the stack is just $|u^L - u^R|$, so the contribution of this stack is $(a + b\hat{\kappa}^p)|u^L - u^R|\, ds$. Integrating over all elements of the step set, this is equivalent to stating that the contribution of the entire step set to the elastica energy is

$$\int_{J_{\mathrm{step}}} (a + b\hat{\kappa}^p)|u^L - u^R|\, ds. \tag{3}$$

Obviously, the above argument is based on physical intuition rather than mathematical rigor, which may trouble some readers. In this particular case, it is relatively easy to give a more rigorous calculation, based on smoothing u with a small unit-volume kernel, applying the definition of elastica energy (2) that is valid for smooth u, then taking the limit as the width of the kernel tends to zero. However, we prefer the approach based on physical intuition because it is easier to understand, and our final goal does not require mathematical rigor. We need to construct an energy whose minimization results in pleasing resolution-independent images; constructing that energy via plausible physical reasoning is a perfectly acceptable approach. Hence, in the remainder of the paper, we will appeal to physical intuition whenever necessary without attempting to inject additional rigor.

4 Curvature-Related Extensions of the Elastica Energy

This section describes the main theoretical contributions of the paper. It first gives details of how to compute the contribution to the elastica energy due to



(a) step (b) 2-prong corner (c) 3-prong corner

Fig. 3. Computing the elastica energy of steps and corners

corners (Sections 4.1). Section 4.2 then unifies the preceding calculations into a single generalized elastica energy. Finally, Section 4.3 describes the variant of the generalized elastica energy appropriate for the triangle meshes used in the present paper. To the best of our knowledge, all three subsections present primarily novel material.

4.1 Computation of the Corner Contribution

For the reasons given in Section 3.1, we use the pliable rod model to compute the energy of a sharp corner point, such as the one in the right-most panel of Figure 2. It can be shown [6] that every corner lies at the endpoint of two or more *prongs*: smooth curves in the jump set. The right-most panel of Figure 2 has five prongs, for example. The image intensity is, by definition, smooth between prongs. Some mild additional assumptions (essentially Lipschitz conditions, as described in the technical report [6]) guarantee that the intensity tends to sensible limits as we approach the prongs and the corner point itself. Thus, by considering a sufficiently small neighborhood of the corner, the intensity function can be approximated arbitrarily well by using prongs that are straight lines and an intensity that is constant between prongs.

Let us first examine the simplest possible case: a two-prong corner (see Figure 3(b)), whose prongs meet with interior angle θ, with constant intensity values u_1 and u_2 on the inside and outside of the corner respectively. How would we build this geometric shape using horizontal pliable rods? As shown in Figure 3(b), each rod must be bent through angle $|\pi - \theta|$, and the rods are stacked vertically (using our standard vertical spacing, δl).

Here we encounter an apparent difficulty: the construction requires sharp corners in the rods, but this leads to infinite energies if we insist on an elastica energy of the form (1). Two easy solutions to suggest themselves. One solution is to take $p = 1$. In this particular case, the energy remains finite (and is easily seen to be $b|\pi - \theta|$). The other solution is to adopt a more general physical model of our rods: simply declare that the rods are made of some material that

can be bent into a sharp corner using finite energy. For example, this energy could be $b|\pi - \theta|^p$ for some p, or the energy could also incorporate robustness by employing, say, $b\min(\tau, |\pi - \theta|^p)$ for some threshold τ. Later experiments use the non-robust version, which performs well for our applications.

Adopting this (literally) more flexible definition of a rod, the energy of each rod is $b|\pi - \theta|^p\,\delta l$, and the total height of the stack of rods is $|u_2 - u_1|$. Integrating over l, we obtain the total corner energy for a two-pronged corner as $b|\pi - \theta|^p|u_2 - u_1|$.

Now let us turn to the general case of a multi-pronged corner, with N prongs. (See Figure 3(c) for a 3-prong example.) For concreteness, label the prongs from 1 to N in an anti-clockwise direction. As before, we may assume the prongs are straight lines and the image takes on constant values $u_1, u_2, \ldots, u_N$ on the *wedges* between each consecutive pair of prongs. So u_1 is the value of the image on wedge 1 between prongs 1 and 2, and so on up to u_N, which is the value on wedge N between prong N and prong 1. The angle of wedge i is θ_i. In what follows, subscripts are computed modulo N. In particular, u_{N+1} means the same thing as u_1, and similarly for θ_{N+1}.

We can think of this simplified geometry as a circular pie cut into wedges, where each wedge happens to be of a different height and angle. We need to calculate the energy required to build this multi-level pie out of pliable rods, as in Figure 3(c). One simple approach uses recursion: find the lowest wedge, and build up the sides of the wedge to the height of the lowest adjacent wedge. At this point, the lowest wedge has effectively been removed from the structure, and the problem has been reduced to building a new pie that possesses one fewer wedge. The recursion can bottom out at two wedges, which is the two-prong case considered above. Alternatively, we can make our final formula (4) a little more elegant by bottoming out at one wedge, which is a degenerate "corner" of zero energy.

More formally, let i^* be the index of the lowest wedge, so $i^* = \arg\min_{i\in\{1,\ldots,N\}} u_i$. Let j^* be the index of the lowest wedge adjacent to i^*, so $j^* = \arg\min_{i\in\{i^*-1, i^*+1\}} u_i$. Let $C = (u_i, \theta_i)_{i=1}^N$ denote the N-prong corner, and $E_{\text{cnr}}(C)$ the desired elastica energy of this corner. Write $\hat{C}$ for the $(N-1)$-prong corner that results from filling in wedge i^* up to height u_{j^*}. Then compute $E_{\text{cnr}}(C)$ recursively according to

$$E_{\text{cnr}}(C) = \begin{cases} 0 & \text{if } N = 1, \\ E_{\text{cnr}}(\hat{C}) + b|\pi - \theta_{i^*}|^p|u_{i^*} - u_{j^*}| & \text{if } N > 1. \end{cases} \tag{4}$$

Later, we will consider a generalization of this formula in which $|\cdot|$ is replaced by a robust function $\rho_\tau(\cdot) \equiv \min(\tau, |\cdot|)$ and raised to a power α. We also make explicit the dependence on parameters b, p by writing

$$E_{\text{cnr}}(C; b, p, \alpha) = \begin{cases} 0 & \text{if } N = 1, \\ E_{\text{cnr}}(\hat{C}; b, p, \alpha, \tau) & \\ \quad + b|\pi - \theta_{i^*}|^p \rho_\tau(u_{i^*} - u_{j^*})^\alpha & \text{if } N > 1. \end{cases} \tag{5}$$

4.2 A Generalized Elastica Energy for All Resolution-Independent Images

The elastica energy (2) decomposes into integrals over four regions: differentiable (D), step (J_{step}), ridge (J_{ridge}), and corner (J_{corner}). Our technical report [6] gives details for J_{ridge}, and this term is omitted here and in the remainder of the paper, since our focus is on the corner energy. Hence, by substituting (3) and (5) into (2), (and making some further generalizations described shortly) we obtain a generalized elastica energy E_G (where "G" stands for "generalized"):

$$\begin{aligned} E_G(u) = \lambda_1 &\int_{\mathbf{x}\in D} (a + b\kappa_{\text{LL}}(\mathbf{x})^{p_1})|\nabla u(\mathbf{x})|^{\alpha_1}\, d\mathbf{x} \\ + \lambda_2 &\int_{J_{\text{step}}} (a + b\hat{\kappa}^{p_2})|u^L - u^R|^{\alpha_2}\, ds + \lambda_3 \sum_{\text{corners } C} E_{\text{cnr}}(C; b, p_3, \alpha_3) \end{aligned} \tag{6}$$

Here we have allowed an arbitrary exponent α_i for the gradient factor, an arbitrary coefficient λ_i, and an arbitrary curvature exponent p_i in each term. The generalization to arbitrary α_i, λ_i, p_i is not justified by any physical or theoretical reasoning. Rather, we appeal to the fact that we are seeking a regularizer that works well in practice. The generalization is justified if, by generalizing a physically realistic expression to one that is not physically realistic, we can obtain better performance when analyzing real images. As we shall soon see, numerous previous authors have taken exactly the same approach. But it should be noted that the natural (i.e. physically realistic, according to the pliable rod model) values for the α_i, λ_i are all 1, and for the p_i the natural value is 2.

Let us now examine how the generalized elastica energy (6) relates to previous work. By taking $\lambda_1 = \lambda_2 = \alpha_1 = \alpha_2 = 1$, and $\lambda_3 = b = 0$, we recover the total variation [21], up to a constant multiplier. By taking $\lambda_3 = b = 0$, we obtain an expression similar to the regularizer used by Viola *et al.* [20]—which, as previously noted, is the direct inspiration for the present work. Hence, the high-level claim that the present work adds a notion of curvature to Viola *et al.* can now be made more explicit: this paper incorporates the corner energy, by permitting $\lambda_3 \neq 0$ in (6).

4.3 Elastica Energy for Images on a Triangle Mesh

We are particularly interested in computing the generalized elastica energy E_G for images that are piecewise linear on a triangle mesh. These images have zero curvature on the interiors of all the triangles, so $\kappa_{\text{LL}} \equiv 0$ on D. Moreover, the mesh edges (which are all straight lines between triangle vertices) have zero curvature too, so $\hat{\kappa} \equiv 0$ on J_{step}. It is easy to see that this renders irrelevant the values of p_1, p_2, a, b in (6). These observations result in a simplified form of the elastica energy for triangle meshes, E_T (where the "T" stands for "triangle"):

$$\begin{aligned} E_{\text{T}}(u) = \lambda_1 &\int_{\mathbf{x}\in D} |\nabla u(\mathbf{x})|^{\alpha_1}\, d\mathbf{x} \\ + \lambda_2 &\int_{J_{\text{step}}} |u^L - u^R|^{\alpha_2}\, ds + \lambda_3 \sum_{\text{corners } C} E_{\text{cnr}}(C; b, p_3, \alpha_3, \tau) \end{aligned} \tag{7}$$

Previous work [19] has shown some benefits from taking $\alpha_1 = 2, p_i = 1$. The experiments in this paper also adopt these settings, and set all other constants (λ_i, α_i) to their physically realistic value (1.0), except where stated otherwise. The robustness parameter τ is set to 10% of the dynamic range in the input image.

5 Algorithmic Details

The algorithm used here for computing resolution-independent images is modeled closely on Viola's work [19,20], where the reader can find many additional details. First, we need a data term that expresses the affinity between u and some input image I. This input I is a standard, discrete set of grayscale pixel values denoted I_i. We assume pixel i of I was formed by blurring the true (continuous) intensity function with some kernel κ_i. This leads to a data term $\mathcal{D}(u, I)$ of the form

$$\mathcal{D}(u, I) = \lambda_0 \sum_i \|I_i - \int_\Omega \kappa_i(x) u(x)\, dx\|. \tag{8}$$

Here, λ_0 is the *data gain* expressing the relative importance of the data and regularization terms. Experiments in this paper take $\lambda_0 = 10$ (unless stated otherwise), a value that was determined by trial and error to yield reasonable performance on a variety of inputs. For the norm $\|\cdot\|$, we use the square of the standard Euclidean norm. Ideally, the kernel functions κ_i would be estimated from the point spread function of the camera used to capture I, but this lies outside the scope of the present paper. We take the pragmatic and simple choice of setting κ_i to be a 2D square box function, equal to 1 on the unit square centered at pixel i and zero elsewhere.

The computation of a resolution-independent image $\hat{u}$ is achieved by minimizing the total energy $\mathcal{E}(u, I)$, which combines the triangle mesh energy (7) and data term (8):

$$\hat{u} = \arg\min_u \mathcal{E}(u, I) = \arg\min_u (E_T(u) + \mathcal{D}(u, I)). \tag{9}$$

Recall that u is a piecewise linear triangle mesh, parameterized by: (i) the 2D locations of each vertex in the mesh (two real parameters per vertex); and (ii) the height and slope of each triangle in the mesh (three real parameters per triangle). The average density of the mesh is application-dependent. In experiments reported here, the typical distance between neighboring vertices is 1–3 pixels. Even on modest-sized images, this leads to tens of thousands of triangles and vertices, and hundreds of thousands of parameters. For example, the 256×256 input of the segmentation result in Figure 7 leads to a mesh with over 42,000 triangles, 21,000 vertices and a resulting total of 170,000 parameters.

All experiments in this paper perform the minimization (9) over these parameters by first initializing the mesh to a reasonable estimate, then applying an off-the-shelf nonlinear optimizer. Specifically, the initialization is done by using a regular grid of vertices spaced 1.5 pixels apart, augmented by further vertices

placed at subpixel locations identified as edgels by a Canny edge detector. The intensity values are initialized by assigning each triangle the constant intensity obtained by integrating I over the triangle.

Minimization is performed in Matlab via Schmidt's `minFunc`[1], using the LBFGS algorithm [11] with default options. Note that LBFGS is a quasi-Newton method, requiring the objective function's derivative but not its Hessian. The derivative of non-corner terms is taken from Viola [19]; the derivative of the corner energy (5), although tedious to implement and debug, requires only elementary geometry and calculus.

The experiments described here require hundreds or thousands of iterations to reach convergence (as defined by the default `minFunc` criteria). The approach is thus rather computationally expensive. The computational cost of the experiments reported here, all employing Matlab implementations on a 2012 desktop PC with an Intel Core2 Q9400 CPU, range from several CPU-core-minutes (for the results of Figure 5) to nearly 50 CPU-core-hours (for the results of Figure 7).

As previously stated, the above approach follows Viola in many respects. There are two important differences, however. First, we perform joint optimization over all parameters simultaneously. This contrasts with Viola's approach, which alternates between optimizations over the vertex location variables and the intensity height/slope variables, and also employs so-called *N/Z flip* moves, which make global changes to the mesh structure.

Second, we take a simpler approach to the problem of degenerate triangles—triangles that become excessively narrow slivers as the optimization proceeds. If the mesh contains one or more problematic slivers, we remove a vertex from each sliver, and re-triangulate the resulting hole using a constrained Delauney triangulation [13]. Each new triangle's intensities must then be initialized based on the nearest undisturbed triangle, and the entire minimization restarted. In principle, this could lead to extremely slow convergence. In practice, however, we find that running sliver-removal just once before beginning any minimization is typically sufficient.

Figure 6, discussed in more detail below, demonstrates that our *joint* minimization approach has similar performance to the more elaborate *alternating* approach of previous work. Moreover, the joint approach is simpler to implement and appears to encounter fewer problems with degenerate triangles. (Note that this discussion compares optimization approaches only. So in this experiment, both the joint and alternating approaches incorporated the corner energy, and therefore required the corner energy derivative also.)

6 Results

6.1 Qualitative Assessment of Incorporating Curvature

Figure 4 demonstrates the main qualitative result of this paper: incorporating curvature into the energy functional leads to modest improvements in the

[1] Mark Schmidt, `http://www.di.ens.fr/~mschmidt/Software/minFunc.html`, 2012

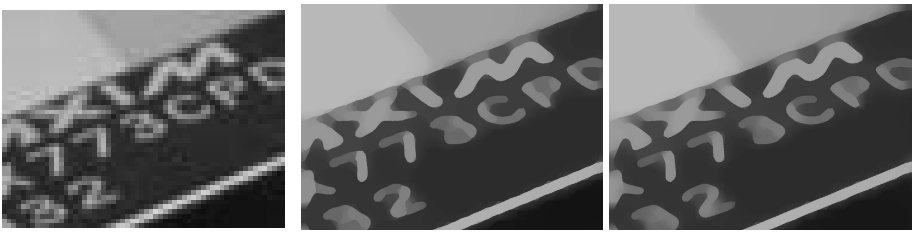
quality of resolution-independent images. Given the input (a), we run our experiments twice with identical settings—except for the parameter λ_3 of (7) which is switched from 0 (corresponding to "without curvature") to 1 (corresponding to "with curvature") between experimental runs. Panels (b) and (c) are renderings of the resolution-independent images produced by the two runs. To the human eye, panels (b) and (c) outputs appear extremely similar, with excellent reconstructions in some regions (e.g. the *M*, the two *7*s, and the *C*) and imperfect ones in others (e.g. the *P* and the *3* have their interiors incorrectly filled).

But as shown in panels (d)–(g), which zoom in on some particular regions of interest, there are some subtle but important differences between the two outputs. (Note that these panels represent extreme super-resolution, showing regions that are 11×7 pixels in the input image.) Specifically, panels (d) and (e) show the letter *C* derived from the input, without and with curvature respectively. In both cases, the curved shape of the *C* has been recovered surprisingly well, albeit imperfectly. More importantly, the output computed with curvature shows some improvement over the without-curvature output: in panel (e), the outline of the *C* represents a smoother curve, and the grayscale values in the interior of the *C* are also smoother. Panels (f) and (g) show a portion of a straight specular edge. Again, the extreme super-resolution performs well in both cases, recovering boundaries that are nearly straight despite the blocky, staircase-like input. And we again see artifacts of the triangle mesh in both outputs: a few triangles with incorrectly-inferred grayscale values protrude from the main strip of high intensity. But the more important point is that panel (g), computed with curvature, produces a straighter boundary for the high-intensity strip, when compared with panel (f) (which was computed without curvature).

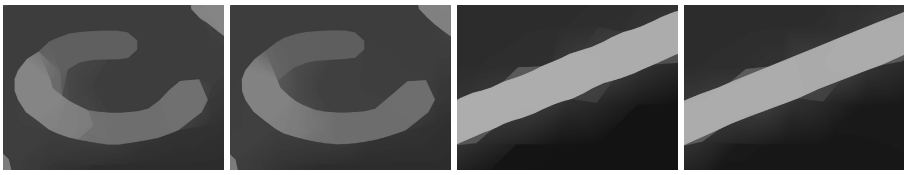
Although we have shown outputs for only one image here, the results are typical. It is reasonable to conclude that incorporating curvature produces modest improvements in the detailed structure of resolution-independent images.

6.2 Quantitative Assessment of Incorporating Curvature

In this subsection, we confirm the previous qualitative results with a quantitative assessment based on peak signal-to-noise ratio (PSNR). The experiment involves the task of simultaneous denoising and upsampling, as shown in Figure 5. The ground truth image in panel (a) is a 64×64 detail of the well-known "peppers" image. Panel (b) is a blurred, noisy version of the ground truth. It was created by first averaging 4×4 blocks of (a), then adding Gaussian noise with standard deviation equal to 5% of the image's dynamic range. This results in a 16×16 image to be used as input to the algorithm for estimating resolution-independent images (Section 5). As with the previous experiment, outputs were produced without and with curvature energy, shown in panels (c) and (d) respectively. A subjective assessment seems to confirm the previous experiment, since the with-curvature result appears to have smoother object boundaries, and smoother grayscale values within objects.



(a) 60×46 input image (b) output without curvature energy (c) output with curvature energy

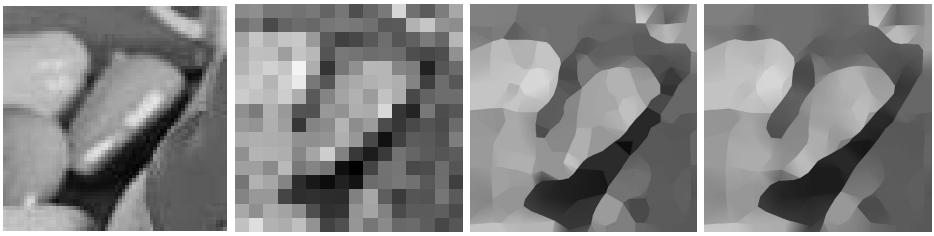


(d) detail without curvature energy (e) detail with curvature energy (f) detail without curvature energy (g) detail with curvature energy

Fig. 4. Comparison of resolution-independent images computed with and without curvature energy

Because we have the ground truth for this experiment, we can also assess these outputs quantitatively, by computing their PSNR with respect to the ground truth. The left panel of Figure 6 shows the results. This graph also demonstrates the sensitivity of the computation to the data gain parameter, λ_0, in Equation (8). The data gain is varied on the horizontal axis, with the corresponding PSNRs for the computations with and without curvature shown on the vertical axis. A higher PSNR corresponds to a higher-quality reconstruction, so it is clear that the with-curvature results are superior to the without-curvature results for each value of the data gain. A further experiment demonstrated that these results are typical, on average. A 64×64 patch was selected uniformly at random from each of the first 40 images of the Berkeley Segmentation Dataset [7], and the above experiment was run with identical settings on all 40 patches. The mean improvement in PSNR after switching on curvature energy was 0.17 dB; further details are in the technical report [6].

As discussed at the end of Section 5, this paper employs a simple joint optimization approach, contrasting with the more elaborate alternating approach of previous work. The right panel of Figure 6 shows the computational expense of these two approaches for the experiment described above (i.e. simultaneously denoising and upsampling the "peppers" image). It is clear that the energy minimization proceeds at roughly the same rate for both approaches, but the alternating approach is less smooth since it encounters degenerate triangles more often. The resulting retriangulation can also lead to an increase in the energy value.



(a) ground truth image (64×64)

(b) input image: noisy, subsampled, 16×16 version of (a)

(c) output without curvature energy

(d) output with curvature energy

Fig. 5. Simultaneous denoising and super-resolution

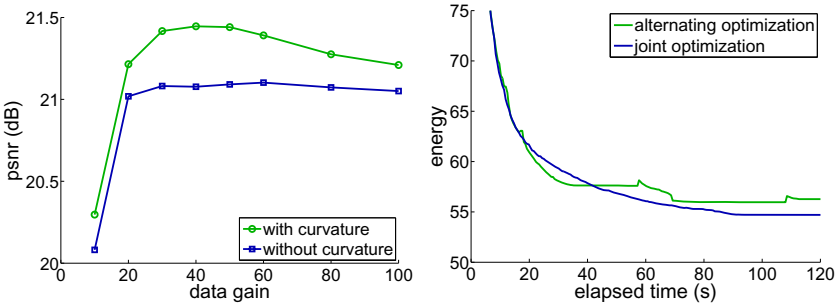
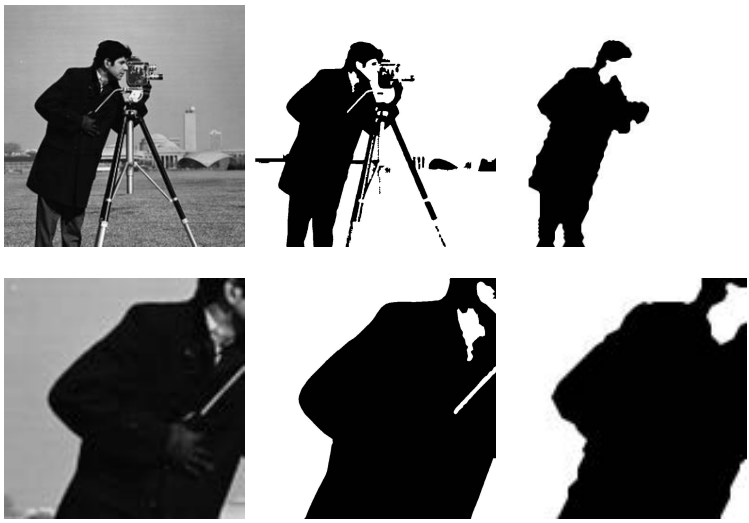


Fig. 6. Left: Estimating a resolution-independent image with curvature energy produces superior PSNR. Right: The simple joint optimization approach of this paper has similar computational expense to the more elaborate alternating approach of previous work.

6.3 Comparison with Alternative Curvature Regularization

A comprehensive comparison with other curvature regularization techniques is beyond the scope of this paper, since our main objective is to enhance the theory and practice of resolution-independent images. Here, we provide here a comparison with one recent state-of-the-art approach: the technique of Strandmark and Kahl [17]. The Strandmark-Kahl (SK) approach minimizes a particular choice of elastica energy over an adaptive mesh, but the precise form of the energy and the methodology for adapting the mesh are quite different to the present paper.

In addition, the SK approach is targeted at regularizing curvature in *binary* output images, which can therefore be regarded as foreground-background segmentations. So that our results can be compared directly with SK, we obtain a binary image by thresholding a rendering of the resolution-independent output. More precisely, given an input image I, we run the minimization algorithm of Section 5 to obtain a resolution-independent image u. Each triangle in the mesh of u is assigned a new constant intensity of 1 or 0 according to whether the triangle's mean intensity is above a given threshold. The resulting u' is a binary,



input image (256×256) | binarized resolution-independent image | SK output [17]

Fig. 7. Resolution-independent segmentation

resolution-independent image, and can be rendered at any desired resolution for comparison with other algorithms.

Figure 7 shows the results of this comparison on the well-known "cameraman" image. The bottom row comprises details from the top row, and the final column shows the best SK output on this image. (Here, "best" simply means the most visually pleasing result appearing in SK [17]; the image was kindly provided by the first author of that paper.) Comparing the middle and right columns of Figure 7, we see that the approach of this paper has segmented many more thin, elongated regions than SK. But this difference is of little interest—without specifying a particular application and associated error metric, one cannot say whether it is preferable to include elongated regions in the foreground or not. Of much more interest is the qualitative nature of the cameraman's boundary. We see that this paper's approach yields boundaries that are considerably more smooth and visually appealing than the SK output. This is particularly noticeable on the coat, right arm, and legs.

7 Conclusion

The key contribution of the paper was the derivation of a novel corner energy (5), used to regularize curvature in resolution-independent images modeled by piecewise linear triangle meshes. Experiments showed qualitative and quantitative improvements in the accuracy of resolution-independent images inferred using the new curvature regularizer. The technique also compared favorably with a state-of-the-art approach for binary segmentation. The clearest opportunity for

future work is to incorporate an energy term for the ridge set. It may also be possible to reduce the computational expense of the approach by employing different techniques for mesh generation (e.g. [1,14,18]).

Acknowledgments. The first author is grateful to Microsoft Research, UK (where this work was conducted) for support as a Visiting Researcher. We are indebted to Chris Francese for assistance with translating Euler's Latin. We also thank the two anonymous reviewers who provided useful comments.

References

1. Adams, M.D.: An improved content-adaptive mesh-generation method for image representation. In: Proc. ICIP (2010)
2. Chan, T.F., Kang, S.H., Shen, J.: Euler's elastica and curvature-based inpainting. SIAM Journal on Applied Mathematics 63(2), 564–592 (2002)
3. Euler, L.: Methodus inveniendi lineas curvas maximi minimive proprietate gaudentes. Bousquet (1744)
4. Goldluecke, B., Cremers, D.: Introducing total curvature for image processing. In: Proc. ICCV, pp. 1267–1274 (2011)
5. Koenderink, J.J., van Doorn, A.J.: Surface shape and curvature scales. Image and Vision Computing 10(8), 557–564 (1992)
6. MacCormick, J., Fitzgibbon, A.: Curvature regularization for resolution-independent images. Tech. rep., Dickinson College (2013)
7. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. ICCV, pp. 416–423 (2001)
8. Masnou, S., Morel, J.M.: Level lines based disocclusion. In: Proc. ICIP, pp. 259–263 (1998)
9. Mitiche, A., Ben Ayed, I.: Variational and Level Set Methods in Image Segmentation. Springer (2011)
10. Mumford, D.: Elastica and computer vision. In: Bajaj, C. (ed.) Algebraic Geometry and Its Applications, pp. 491–506. Springer (1994)
11. Nocedal, J., Wright, S.J.: Numerical Optimization. Springer (1999)
12. Osher, S., Fedkiw, R.: Level Set Methods and Dynamic Implicit Surfaces. Springer (2003)
13. Paul Chew, L.: Constrained delaunay triangulations. Algorithmica 4, 97–108 (1989)
14. Sarkis, M., Diepold, K.: Content adaptive mesh representation of images using binary space partitions. IEEE Trans. Image Processing 18(5), 1069–1079 (2009)
15. Schoenemann, T., Kahl, F., Cremers, D.: Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In: Proc. ICCV, pp. 17–23 (2009)
16. Sethian, J.A.: Level Set Methods and Fast Marching Methods, 2nd edn. Cambridge University Press (1999)
17. Strandmark, P., Kahl, F.: Curvature regularization for curves and surfaces in a global optimization framework. In: Boykov, Y., Kahl, F., Lempitsky, V., Schmidt, F.R. (eds.) EMMCVPR 2011. LNCS, vol. 6819, pp. 205–218. Springer, Heidelberg (2011)
18. Vasilescu, M., Terzopoulos, D.: Adaptive meshes and shells: Irregular triangulation, discontinuities, and hierarchical subdivision. In: Proc. CVPR, pp. 829–832 (1992)

19. Viola, F.: Resolution-independent image models. Ph.D. thesis, University of Cambridge (2011)
20. Viola, F., Cipolla, R., Fitzgibbon, A.: A unifying resolution-independent formulation for early vision. In: Proc. CVPR (2012)
21. Ziemer, W.P.: Weakly Differentiable Functions: Sobolev Spaces and Functions of Bounded Variation. Springer (1989)

Appendix: Euler's Definition of Elastica

Numerous papers cite Euler's work on elastica, but it is surprisingly difficult to track down the relevant excerpt. It appears in Euler's 1744 publication [3], *Methodus inveniendi lineas curvas ...*, Appendix I ("Additamentum I"), paragraph 2 (p247):

> ... ut inter omnes curvus eiusdem longitudinis, qua non solum per puncta A & B transeant, sed etiam in his punctis a rectis positione datis tangantur, definiatur ea in qua sit valor huius expressionis $\int \frac{ds}{RR}$ minimus.

This can be translated as:

> ... that, of all curves of the same length, which not only pass through points A and B, but also are touched at these points by given tangents, it is defined by that in which the value of the expression $\int \frac{ds}{R^2}$ is the smallest.

PoseField: An Efficient Mean-Field Based Method for Joint Estimation of Human Pose, Segmentation, and Depth

Vibhav Vineet*, Glenn Sheasby*, Jonathan Warrell, and Philip H.S. Torr

Oxford Brookes University
Oxford, UK
{vibhav.vineet-2010,glenn.sheasby,jwarrell,philiptorr}@brookes.ac.uk

Abstract. Many models have been proposed to estimate human pose and segmentation by leveraging information from several sources. A standard approach is to formulate it in a dual decomposition framework. However, these models generally suffer from the problem of high computational complexity. In this work, we propose **PoseField**, a new highly efficient filter-based mean-field inference approach for jointly estimating human segmentation, pose, per-pixel body parts, and depth given stereo pairs of images. We extensively evaluate the efficiency and accuracy offered by our approach on H2View [1], and Buffy [2] datasets. We achieve 20 to 70 times speedup compared to the current state-of-the-art methods, as well as achieving better accuracy in all these cases.

1 Introduction

Human pose estimation and segmentation have long been popular tasks in computer vision, and a large body of research has been developed on these problems [3–7]. Several of these methods model pose estimation and segmentation problems separately, and fail to capture the large variability and deformation in appearance and the structure of humans.

However, when segmentation and pose estimation results are considered together, one can observe discrepancies, for example a foreground region not corresponding to any detected body part, or vice versa. Joining the two problems together, either sequentially or simultaneously, can help to remove these discrepancies. Researchers have thus begun to consider the possibility of jointly estimating these outputs, leveraging the information from several high-level and low-level cues.

A number of methods insert various algorithms into a pipeline, where the result of one algorithm is used to initialize another. For example, Bray et al. tackle the problem of human segmentation by introducing a pose-specific MRF, encouraging the segmentation result to look "human-like" [8]. Similarly, Kumar et al. use layered pictorial structures to generate an object category specific MRF to

* The first two authors contributed to this work equally as joint first author.

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 180–194, 2013.
© Springer-Verlag Berlin Heidelberg 2013

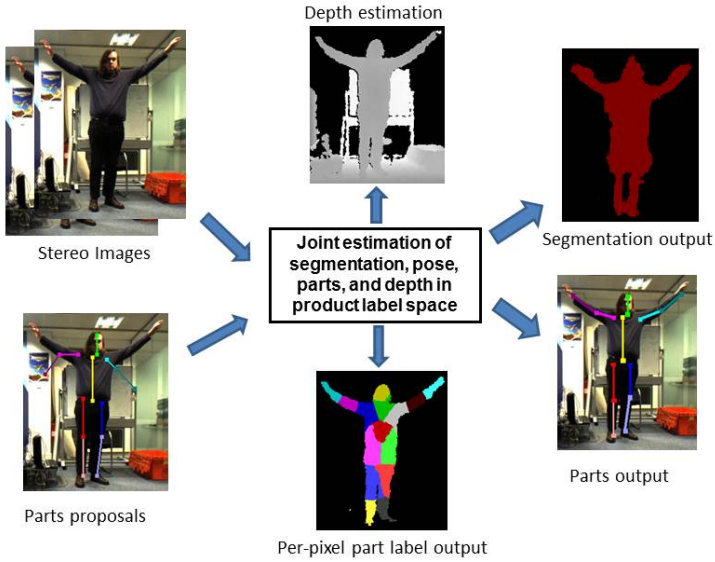


Fig. 1. Given stereo pairs and initial part proposals, our approach jointly estimates the human segmentation, pose, and depth, considering the relationships between per-pixel body part labels and part configurations

improve segmentation [9]. The problem with this kind of approach is that errors in one part of the algorithm can propagate to later stages. Joint inference can be used to overcome this issue; Ladický et al. obtain an improvement in object class segmentation by incorporating global information from object detectors and object co-occurrence terms [10], solving detection and segmentation with one CRF. Further, Ladický et al. frame joint estimation of object classes and disparity as CRF problems in the product label space, and solve the two tasks together [11].

Additionally, in the context of human pose estimation and segmentation, Wang and Koller propose a dual-decomposition based inference method [12] in a multi-level CRF framework to jointly estimate pose and segmentation by introducing variables that capture the coupling between these two problems [13]. Extending their formulation, Sheasby et al. [1] add depth information, thus allowing human pose, segmentation and depth to be solved together [14, 1].

The complexity of such joint frameworks is a serious issue; if the framework is to be used for applications such as security and video gaming, fast output is required. In such situations, it might prove desirable to find an efficiently solvable approximation of the original problem. One such method that can be applied here is mean-field inference [15]. For a certain class of pairwise terms, mean-field inference has been shown to be very powerful in solving the object class segmentation problem, and object-stereo correspondence problems in CRF frameworks, providing an order-of-magnitude speedup [16]. In this paper, we propose a highly efficient filter-based mean-field approach to perform joint

estimation of human segmentation, pose, per-pixel part labels, and disparity in the product label space, producing a significant improvement in speed.

Further, to model the human skeleton, we propose a hierarchical model that captures relations on multiple levels. At the lowest level, we estimate part labels per pixel. Such a representation has been shown to be successful in generating body parts proposals and pose estimation by Shotton et al. [17]. Secondly, the higher level tries to find the best configuration from a set of part proposals. Our framework is represented graphically in Fig 1.

Finally we extensively evaluate the efficiency and accuracy offered by our mean-field approach on two datasets: H2View [14], and Buffy [2]. We show results for segmentation, per pixel part labelling and pose estimation; disparity computation is used to improve these results, but is not quantitatively evaluated as it is not feasible to obtain dense ground truth data. We achieve 20-70 times speedup compared to the current state-of-the-art graph-cuts based dual-decomposition approach [1], as well achieving better accuracy in all cases.

The remainder of the paper is structured as follows: an overview of dense CRF formulation is introduced in the next section, while our body part formulation is discussed in Section 3. We describe our joint inference framework in Section 4 and learning of different parameters is discussed in the Section 5. Results follow in Section 6, and Section 7 concludes the paper.

2 Overview of Dense Random Field Formulation

The goal of our joint optimization framework is to estimate human segmentation and pose, together with part labels at the pixel level, and perform stereo reconstruction given a pair of stereo images. These problems however can be separately solved in a conditional random field (CRF) framework. Thus, before going into the details of the joint modelling and inference, we provide the models for solving them separately. Let $\mathcal{X}^{\mathcal{S}} = \{X_1^S, ..., X_N^S\}$, $\mathcal{X}^{\mathcal{J}} = \{X_1^J, ..., X_N^J\}$, $\mathcal{X}^{\mathcal{D}} = \{X_1^D, ..., X_N^D\}$ be the human segmentation, per-pixel part and disparity variables respectively. We assume each of these random variables is associated with each pixel in the image $\mathcal{N} = \{1, ..., N\}$. Further, each X_i^S takes a label from segmentation label set $\mathcal{L}^S \in \{0, 1\}$, X_i^D takes a label from $\mathcal{L}^D \in \{0...D\}$ disparity labels and X_i^J takes a label from $\mathcal{L}^J \in \{0, 1, ..., M\}$ where 0 represents background and M is the number of body parts.

First, we give details of the energy function for the segmentation variables. Assuming the true distribution of the segmentation variables is captured by the unary and pairwise terms, the energy function takes the following form:

$$E^S(\mathbf{x}^S) = \sum_{i \in V} \psi_u^S(x_i^S) + \sum_{i \in V, j \in N_i} \psi_p^S(x_i^S, x_j^S) \tag{1}$$

where N_i represents the neighborhood of the variable i, $\psi_u^S(x_i^S)$ represent unary terms for human segmentation class and $\psi_p^S(x_i^S, x_j^S)$ are pairwise terms capturing the interaction between a pair of segment variables. The human object specific unary cost $\psi_u^S(x_i^S)$ is computed based on a boosted unary classifier on

image-specific appearance using the model of Shotton et al. [19]. The pairwise terms between human segmentation variables ψ_p^S take the form of Potts models weighted by edge-preserving Gaussian kernels [18] as:

$$\psi_p^S(x_i^S, x_j^S) = \mu(x_i^S, x_j^S) \sum_{v=1}^{V} w^{(v)} k^{(v)}(\mathbf{f}_i, \mathbf{f}_j) \tag{2}$$

where $\mu(.,.)$ is an arbitrary *label compatibility function*, while the functions $k^{(v)}(.,.)$, $v = 1...V$ are Gaussian kernels defined on feature vectors $\mathbf{f}_i$, $\mathbf{f}_j$ derived from the image data at locations i and j (where Krahenbuhl and Koltun [18] form $\mathbf{f}_i$ by concatenating the intensity values at pixel i with the horizontal and vertical positions of pixel i in the image), and $w^{(v)}$, $m = 1...V$ are used to weight the kernels.

Similarly we define the energy functions over the per-pixel part and disparity variables as:

$$E^J(\mathbf{x^J}) = \textstyle\sum_{i \in V} \psi_u^J(x_i^J) + \sum_{i \in V, j \in N_i} \psi_p^J(x_i^J, x_j^J) \tag{3}$$

$$E^D(\mathbf{x^D}) = \textstyle\sum_{i \in V} \psi_u^D(x_i^D) + \sum_{i \in V, j \in N_i} \psi_p^D(x_i^D, x_j^D) \tag{4}$$

where $\psi_u^J(x_i^J)$ and $\psi_u^D(x_i^D)$ represent unary term for the per-pixel part and disparity variables respectively, and $\psi_p^J(x_i^J, x_j^J)$ and $\psi_p^D(x_i^D, x_j^D)$ are pairwise terms capturing the interaction between pairs of per-pixel part and disparity variables respectively. The per-pixel part variable dependent unary cost $\psi_u^J(x_i^J)$ is computed based on a boosted unary classifier on depth image. Further, if we do not have ground truth for the depth map, we can learn the unary cost for the per-pixel parts on image-specific appearance. The unary cost $\psi_u^D(x_i^D)$ for the disparity variables measures the color agreement of a pixel with its corresponding pixel i from the stereo-pair given a choice of disparity x_i^D. The pairwise terms for both these variables ψ_p^J and ψ_p^D take the form of contrast-sensitive Potts models as mentioned earlier.

3 Joint Formulation

The goal of our joint optimization framework is to estimate human segmentation and pose, together with part labels at the pixel level, and also perform stereo reconstruction. We formulate the problem in a conditional random field (CRF) framework as a product label space in a hierarchical framework. At the lower level, we define the random variables $\mathcal{X} = [\mathcal{X}^S, \mathcal{X}^J, \mathcal{X}^D]$, where $\mathcal{X}$ takes a label from the product label space $\mathcal{L} = \{(\mathcal{L}^S \times \mathcal{L}^J \times \mathcal{L}^D)^N\}$. For specifying human pose, we define a second layer, represented by a set of latent variables $\mathcal{Y} = \{Y_1, Y_2, ..., Y_M\}$ corresponding to the M body parts, each taking labels from $\mathcal{L}^P \in \{0, ..., K\}$ where $1, 2, ..., K$ corresponds to the K part proposals generated for each body part, and zero represents the background class. We generate K part proposals using the model of Yang and Ramanan [7]. The graphical model explaining our hierarchical joint model is shown in the Fig 2.

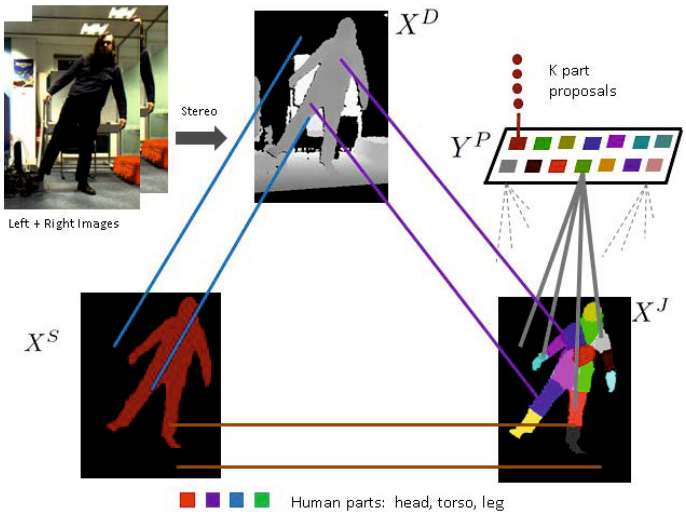


Fig. 2. PoseField model jointly estimates the per-pixel human/background segmentation, body part, and disparity labels. Further, the relationship between per pixel body part label, and part configurations are captured in a hierarchical model with information propagating between these different layers. (Best viewed in color)

3.1 Joint Energy Function

Given the above model, we wish to define an energy function which is general enough to capture sufficient mutual interaction between the variables while still providing scope for efficient inference. For this reason, we assume our energy function to take the following form:

$$E(\mathbf{x},\mathbf{y}) = E^S(\mathbf{x}^S) + E^D(\mathbf{x}^D) + E^J(\mathbf{x}^J) + E^P(\mathbf{y}) \\ + E^{SJ}(\mathbf{x}^S,\mathbf{x}^J) + E^{SD}(\mathbf{x}^S,\mathbf{x}^D) + E^{DJ}(\mathbf{x}^D,\mathbf{x}^J) + E^{JP}(\mathbf{x}^J,\mathbf{y}) \quad (5)$$

Here, our joint model has been factorized into separate layers representing human segmentation, disparity, per pixel part and latent part variables. The individual terms at the layers are captured by E^S, E^D, E^J as defined earlier and E^P, the energy function for the latent part variables, details of which are provided later in this section. Further, in order to incorporate the dependency between these variables, we add pairwise interactions between these CRF layers. E^{SJ}, E^{SD}, E^{DJ} captures the interaction between (*segment*,*per-pixel part*), (*segment*,*disparity*) and (*disparity*, *per-pixel part*) variables. The term E^{JP} captures the mutual interaction between the (*per-pixel part*, *latent part*) variables. We design the forms of these pairwise interactions to allow efficient and accurate inference; details are provided below.

Per-part terms E^P. In our hierarchical model, the top layer corresponds to the human part variables Y, which involve per-part unary cost $\psi_u^P(x_i = k)$ for

associating the i^{th} part to the k^{th} proposal or to the background [1], and the pairwise term $\psi_p^P(y_i, y_j)$ penalizes the case where parts that should be connected are distant from one another in image space. The per-part unary term $\psi_u^P(y_i = k)$ is the score generated by the Yang and Ramanan model [7].

Segment, per-pixel part terms (E^{SJ}). The joint human segmentation and per-pixel body part term, E^{SJ}, encodes the relation between segmentation and per-pixel body part. Specifically, we expect a variable that takes a body part label to belong to the foreground class, and vice versa. We pay a cost of C^{SJ} for violation of this constraint, incorporated through a pairwise interaction between the segmentation and per-pixel part variables; this interaction takes the following form:

$$E^{SJ} = \psi_p^{SJ}(\mathbf{x}^S, \mathbf{x}^J) = \sum_{i=1}^{N} C^{SJ} \cdot [(x_i^S = 1) \wedge (x_i^J = 0)] + \sum_{i=1}^{N} C^{SJ} \cdot [(x_i^S = 0) \wedge (x_i^J \neq 0)] \quad (6)$$

Segment, disparity terms (E^{SD}). Additionally, our joint object-depth cost E^{SD} encourages pixels with a high disparity to be classed as foreground, and pixels with a low disparity to be classified as background. We penalize the violation of this constraint by a cost C^{SD}. Following the formulation of [1], we first generate a segmentation map $\mathcal{F} = \{F_1, F_2, ..., F_N\}$ by thresholding the disparity map, thus each F_i takes a label from L^S. We would expect the prior map $\mathcal{F}$ to agree with the segmentation result, so that pixels taking human label ($f_i = 1$) are classified as human, and vice versa, otherwise we pay a cost C^{SD} for violation of this constraint:

$$E^{SD} = \psi_p^{SD}(\mathbf{x}^S, \mathbf{x}^D) = \sum_{i=1}^{N} C^{SD} \cdot [(x_i^S = 1) \wedge (f_i = 0)] + \sum_{i=1}^{N} C^{SD} \cdot [(x_i^S = 0) \wedge (f_i = 1)] \quad (7)$$

Per-pixel part, disparity terms (E^{JD}). The joint energy term E^{JD} encodes the relationship between the per-pixel body part variables and the disparity variables. As with the cost term E^{SD}, we use a flood fill to generate a segmentation map $\mathcal{F} = \{F_1, F_2, \ldots, F_N\}$ which gives us a prior based on disparity. We expect pixels classed as human by this prior (so $f_i = 1$) to be assigned to a body part label, so $x_i^J > 0$. Conversely, pixels classed as background ($f_i = 0$) should be assigned to the background label ($x_i^J = 0$). Therefore, the energy term has the following form:

$$E^{JD} = \psi_p^{JD}(\mathbf{x}^J, \mathbf{x}^D) = \sum_{i=1}^{N} C^{JD} \cdot [(x_i^J > 0) \wedge (f_i = 0)] + \sum_{i=1}^{N} C^{JD} \cdot [(x_i^J = 0) \wedge (f_i = 1)] \quad (8)$$

Per-pixel part, latent part terms (E^{JP}). E^{JP} enforces the constraint that when a body part l is present in the solution at the pixel level, then the variable Y_l^P corresponding to the part l must be on, otherwise we pay a cost of C^{JP}.

$$E^{JP} = \psi_p^{JP}(\mathbf{x}^J, \mathbf{y}) = C^{JP} \cdot \sum_{l \in M} [(y_l = 0) \wedge (\sum_i [x_i^J = l]) > 0] \tag{9}$$

4 Inference in the Joint Model

Given the above complex hierarchical model, we now propose a new mean-field based inference approach to perform efficient inference for joint estimation. But, before going into details of our approach, we give a general form of mean-field update. We also highlight the work of Krahenbuhl and Koltun [18] for filter-based efficient inference in fully connected pairwise CRFs. This model was later extended by Vineet et.al. [16] to incorporate higher order potentials, and to solve jointly the object-stereo correspondence problems.

Let us consider a general form of energy function:

$$E(\mathbf{Z}|\mathbf{I}) = \sum_{c \in \mathcal{C}} \psi_c(\mathbf{z}_c|\mathbf{I}) \tag{10}$$

where $\mathbf{Z}$ is a joint assignment of the random variables $\mathcal{Z} = \{Z_1, ..., Z_{N_Z}\}$, $\mathcal{C}$ is a set of cliques each consisting of a subset of random variables $c \subseteq \mathcal{Z}$, and associated with a potential function ψ_c over settings of the random variables in c, $\mathbf{z}_c$. In Sec. 2 we have that $\mathcal{Z} = \mathcal{X}^S$, that each X_i takes values in the set $\mathcal{L}^S$ of human labels, and that $\mathcal{C}$ contains unary and pairwise cliques of the types discussed. In general, in the models discussed below we will have that $\mathcal{X}^S \subseteq \mathcal{Z}$, so that $\mathcal{Z}$ may also include other random variables (e.g. latent variables) which may take values in different label sets.

Considering this model, the general form of the mean-field update equation (see [15]) is:

$$Q_i(z_i = \nu) = \frac{1}{\tilde{Z}_i} \exp\{-\sum_{c \in \mathcal{C}} \sum_{\{\mathbf{z}_c | z_i = \nu\}} Q_{c-i}(\mathbf{z}_{c-i}) \cdot \psi_c(\mathbf{z}_c)\} \tag{11}$$

where ν is a value in the domain of the random variable z_i, $\mathbf{z}_c$ denotes an assignment of all variables in clique c, $\mathbf{z}_{c-i}$ an assignment of all variables apart from Z_i, and Q_{c-i} denotes the marginal distribution of all variables in c apart from Z_i derived from the joint distribution Q. $\tilde{Z}_i = \sum_\nu \exp\{-\sum_{c \in \mathcal{C}} \sum_{\{\mathbf{z}_c | z_i = \nu\}} Q_{c-i}(\mathbf{z}_{c-i}) \cdot \psi_c(\mathbf{z}_c)\}$ is a normalizing constant for random variable z_i. We note that the summations $\sum_{\{\mathbf{z}_c | z_i = \nu\}} Q_{c-i}(\mathbf{z}_{c-i}) \cdot \psi_c(\mathbf{z}_c)$ in Eq. 11 evaluate the expected value of ψ_c over Q given that Z_i takes the value ν.

Following this general update strategy, the updates for the densely connected pairwise model in Eq. 1 are derived by evaluating Eq. 11 across the unary and

pairwise potentials defined in Sec. 2 for $z_i = x_{1...N}$ and $\nu = 0...L$. For the densely connected pairwise CRF model, the mean-field update takes the following form:

$$Q_i(z_i = l) = \frac{1}{\tilde{Z}_i} \exp\{-\psi_i(z_i) - \sum_{l' \in \mathcal{L}} \sum_{j \neq i} Q_j(z_j = l')\psi_{ij}(z_i, z_j)\} \tag{12}$$

With this mean-field update, Krahenbuhl and Koltun [18] proposed a filter-based method for performing fast inference thus reducing the complexity from $O(N^2)$ to $O(N)$ under the assumption that the pairwise potentials take the form of a linear combination of Gaussian kernels. They show how the expensive message passing update in the mean-field is approximated by a convolution with a bilateral filter in a high dimensional space. Given this Gaussian convolution, they use a permutohedral lattice based bilateral filtering method [20] for performing efficient inference. They run the update equation for a fixed number of iterations, where each iteration leads to a decrease in the KL-divergence value. To extract a solution, they evaluate the approximate maximum posterior marginal as $z_i^* = \max_{z_i} Q_i(z_i)$.

4.1 Efficient Inference

In our framework, we need to jointly estimate the best possible configurations of the segmentation variables X^S, per-pixel part variables X^J, disparity variable X^D and part variable Y^P by minimizing the energy function $E(\mathbf{x}, y)$ in Eq. 5. We now provide the details of our mean-field updates for efficient joint inference.

Update for segment variables (X^S). Given the energy function detailed in Sec. 3.1, the marginal update for human segmentation variable X_i^S takes the following form:

$$\begin{aligned} Q_i^S(x_{[i,l]}^S) = \frac{1}{Z_i^S} \exp\{-\psi^S(x_i^S) - \sum_{l' \in \mathcal{L}^J} \sum_{j \neq i} Q_j^S(x_{[j,l']}^S)\psi(x_i^S, x_j^S) \\ - \sum_{l' \in \mathcal{L}^D} Q_i^D(x_{[i,l']}^D)\psi(x_i^D, x_i^S) - \sum_{l' \in \mathcal{L}^J} Q_i^J(x_{[i,l']}^J)\psi(x_i^S, x_i^J)\} \end{aligned} \tag{13}$$

where $Q_i^D(x_{[i,l']}^D)\psi(x_i^D, x_i^S)$ and $Q_i^J(x_{[i,l']}^J)\psi(x_i^S, x_i^J)$ are the messages from disparity and per-pixel part variables respectively to the segmentation variables. Thus, these messages enforce the consistency between the segmentation, disparity and per-pixel part term variables. We write $x_{[i,l]}$ for $(x_i = l)$ and the same notation will be followed subsequently.

Update for disparity variables (X^D). Similar to the updates for X_i^S, the marginal update for the per-pixel depth variables X_i^D takes the following form:

$$\begin{aligned} Q_i^D(x_{[i,l]}^D) = \frac{1}{Z_i^D} \exp\{-\psi^D(x_i^D) - \sum_{l' \in \mathcal{L}^D} \sum_{j \neq i} Q_j^D(x_{[j,l']}^D)\psi(x_i^D, x_j^D) \\ - \sum_{l' \in \mathcal{L}^S} Q_i^S(x_{[i,l']}^S)\psi(x_i^D, x_i^S) - \sum_{l' \in \mathcal{L}^J} Q_i^J(x_{[i,l']}^J)\psi(x_i^J, x_i^D)\} \end{aligned} \tag{14}$$

where $Q_i^S(x^S_{[i,l']})\psi(x_i^D, x_i^S)$ and $Q_i^J(x^J_{[i,l']})\psi(x_i^J, x_i^D)$ correspond to the messages from the segmentation and per-pixel part variables to the disparity variables.

Update for per-pixel part variables (X^J). The per-pixel part variable X_i^J takes messages from part configuration in the hierarchy along with the messages from the other per-pixel part variables, segmentation variables and disparity variables. Thus, the marginal update for the per-pixel part variables X_i^J take the following form:

$$Q_i^J(x^J_{[i,l]}) = \frac{1}{Z_i^J}\exp\{-\psi_u^J(x_i^J) - \sum_{l'\in\mathcal{L}^\mathcal{J}}\sum_{j\neq i} Q_j^J(x^J_{[j,l']})\psi(x_i^J, x_j^J) \\ - \sum_{l'\in\mathcal{L}^\mathcal{D}} Q_i^D(x^D_{[i,l']})\psi(x_i^J, x_i^D) - \sum_{l'\in\mathcal{L}^\mathcal{S}} Q_i^S(x^S_{[i,l']})\psi(x_i^J, x_i^S) - \sum_{l'\in\mathcal{L}^\mathcal{P}} Q_i^P(y_{[i,l']})\psi(y_i, x_i^J)\} \tag{15}$$

Here $Q_i^P(y_{[i,l']})\psi(y_i, x_i^J)$ carry messages from the valid part configuration in the hierarchy to the per-pixel part variables, and $Q_i^S(x^S_{[i,l']})\psi(x_i^J, x_i^S)$ and $Q_i^D(x^D_{[i,l']})$ $\psi(x_i^D, x_i^J)$ correspond to the messages from the segmentation and disparity variables to per-pixel part variables.

It is also to be noted that the required expectation update for messages from other joint variables, e.g. messages from segmentation variables to disparity variables, contribute a time complexity of $O(N)$. Thus, the marginal update steps do not increase the overall time complexity.

Update for latent part variables (Y). Finally, the mean-field update for the part variables in the hierarchy corresponds to:

$$Q_i^P(y_{[i,l]}) \propto \exp\{-\psi_u(y_i) - \sum_{j'\in\mathcal{L}^\mathcal{J}}\sum_{j=1}^{N} Q_j^J(x^J_{[j,j']})\psi(y_i, x_j^J) \tag{16}$$

where $Q_j^J(x^J_{[j,j']})\psi(y_i, x_j^J)$ corresponds to the messages from the per-pixel part variables to the part configuration variables. Evaluation of the expectation for part variables contributes $O(N)$ to the overall complexity. Thus, our inference method does not increase the overall complexity of $O(N)$ for fully connected pairwise updates.

5 Learning

The weights $C^{SJ}, C^{SD}, C^{JD}, C^{JP}$ capturing the relationships between variables at different CRF layers are set through cross-validation. Our cross validation step to search for good set of parameters to weight these different terms in Eq. 5 is greedy in the sense that we set them one at a time sequentially. This way of sequential learning ensured an efficient way to search for a good set of the parameters without going through all the possible joint configurations of the parameters. Structured learning [21] provides a possible future direction to learn these parameters, however our focus was efficient inference. Further, the Gaussian kernel parameters are set through cross-validation as well.

6 Experiments

In this section, we demonstrate the efficiency and accuracy provided by our approach on the H2View [1] dataset. Further, to highlight the generalization of our approach, we also conduct experiment on the Buffy [2] dataset where we do not have stereo pairs of images. In all experiments, timings are based on code run on an Intel® Xeon ® 3.33 GHz processor, and we fix the number of full mean-field update iterations to 5 for all models. As a baseline, we compare our approach for the joint estimation of human segmentation, pose, per-pixel part and disparity with the dual-decomposition based model of Sheasby et al. [1]. Further, we compare our joint approach against some other state-of-the-art approaches which do not perform any joint estimation. For example, we compare our human segmentation results against a graph-cuts based AHCRF [22] and the mean-field model of Krähenbühl et al. [18]. We assess human segmentation accuracy in terms of the overall percentage of pixels correctly labelled, the average recall and intersection/union score per class (defined in terms of the true/false positives/negatives for a given class as TP/(TP+FP+FN)). Similarly, for pose estimation, apart from comparing against the dual-decomposition based joint labelling model of Sheasby et.al. [1], we compare the probability of correct pose (PCP) criterion against the models of Yang and Ramanan [7], and Andriluka et al. [23], which do not perform joint labelling. In all these cases, we use the code provided by the authors for the AHCRF, Krähenbühl et al., Yang and Ramanan, Andriluka et al., and Sheasby et al. However we do not quantitatively evaluate the disparity results as we do not have the ground truth data for the disparity.

6.1 H2View Dataset

The H2View dataset [1] comprises 1108 training images and 1598 test images consisting of humans in different poses performing standing, walking, crouching, and gesticulating actions in front of a stereo camera. Ground truth human segmentation, and pose are provided; we augment these with a per-pixel part labels.

We first show the accuracy and efficiency achieved by our method on the human segmentation results. We observe an improvement of almost 3.5% over the dual-decomposition based joint inference model of Sheasby et al. [1], almost 4.5% compared to AHCRF [22] and almost 4% over dense CRF [18] in the I/U score, shown in Tab. 1. Significantly, we observe an order of magnitude of speed up (almost 20×) over the model of Sheasby et al. and a speed up of almost 5× over the AHCRF model. Further as far as pose estimation results are concerned, we achieve an improvement of almost 3.5% over Yang and Ramanan, 7% over Andriluka et al. in the PCP scores. Though these methods do not perform joint inference, we compare to highlight the importance of joint inference. Further compared to the model of Sheasby et al., we perform slightly worse in the PCP score, but we observe a speed up of almost 20× over their model. Here it should be noted that the time to evaluate the model of Yang and Ramanan to generate initial pose proposals is not included in the models of Sheasby et.al. and our model. Quantitative results for pose estimation are as shown in Tab. 2.

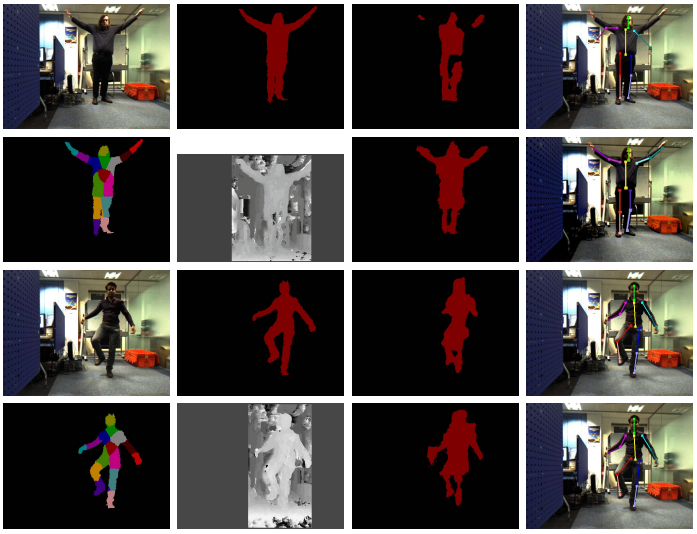


Fig. 3. Qualitative results on two sets of images from H2View dataset. First two rows correspond to the first image, and next two rows to the second image. From left to right: (top row) input image, ground truth for human segmentation, output from [1], pose estimation output from [1]; (second row): our per-pixel part label output, disparity estimation output, segmentation and pose-estimation outputs. Last two rows show the same set of images on the second input image. Our method is able to recover the limbs properly on both the segmentation and pose estimation problems. (Best viewed in color)

Table 1. Quantitative results on H2View dataset for human segmentation. The table compares timing and accuracy of our approach (last 2 lines) against the dual-decomposition model of Sheasby et.al. [1] as well as over other baselines. Note the significant improvement in inference time and class-average performance our approach against the baselines.

Method	Time (s)	Overall	Av.Re	I/U
Unary	0.36	96.12%	85.90%	78.94%
ALE [22]	1.5	96.14%	86.10%	80.14%
Sheasby [1]	25	96.67%	90.48%	81.52%
MF [18]	0.48	96.56%	86.12%	80.57%
Our	1.25	**97.14%**	92.32%	**84.60%**

Additionally we observe qualitative improvement in both the segmentation and pose results, as shown in Fig. 3. As far as per-pixel part label accuracy is concerned, we achieve 94.43% of overall percentage of correctly labelled pixels, compared to 92.63% achieved by the dual-decomposition method of Sheasby et.al. [1], and 89.55% achieved by the graph-cuts based AHCRF method [22].

Fig. 4. Qualitative results on Buffy dataset [2]. From left to right: (first row:) input image, ground truth of segmentation, segmentation output before joint estimation, (second row:) segmentation output after joint estimation, pose output before joint estimation, and pose output after joint estimation. (Best viewed in color)

Table 2. The table compares timing and accuracy of our approach (last line) against the baseline for the pose estimation problem on H2View dataset [1]. Observe that our approach achieves almost 20× speedup against the dual-decomposition model of Sheasby et.al. [1] as well as over other baselines. U/LL represents average of upper and lower legs, and U/FA represents average of upper and fore arms.

Method	T(s)	U/LL	U/FA	TO	Head	Overall
Sheasby [1]	25	83.43	54.56	90.05	89.8	73.18
Yang [7]	10	79.65	49.05	88.5	83.0	69.85
Andriluka [23]	35	74.85	47.7	83.9	76.0	66.03
Ours	1.2	82.86	55.16	89.05	86.20	73.12

6.2 Buffy Dataset

In order to show the generalization and effectiveness of our approach, we also evaluate our model on the Buffy dataset. We select a set of 476 images as training images, and 276 images as test images, using the same split as used in [2]. Since there are no depth images, we evaluate only on joint pose and segmentation problems. For the human segmentation case, our joint approach achieves a speed up of almost 70× compared to the dual-decomposition based method of Sheasby et al. [1], and 3× compared to AHCRF [22]. We also observe an improvement of almost 10% and 1% in the I/U scores respectively on segmentation results, shown in Tab. 3. Further, we observe an improvement of almost 0.4% over the Yang and Ramanan model and almost 7% over the model of Sheasby et al. model in the PCP score for the pose estimation problem, shown in Tab. 4. It should be noted that the results of the Yang and Ramanan model [7] reported in our paper is different than the one in their original paper since they first generate a set of detection windows by running an upper-body detector, and then evaluate pose detection only on these detected windows. Here we evaluate the poses on whole image, thus a good detection of the non-detected person could

Table 3. Quantitative results on Buffy dataset for human segmentation problem. Observe the significant speedup (almost 70×) achieved compared to the dual-decomposition method of Sheasby et.al. [1] and over other approaches. Further, our approach achieves better accuracy than other methods as well.

Method	Time (s)	Overall	Av.Pr	I/U
Sheasby [1]	20	80.85%	85.80%	65.01%
ALE [22]	0.96	87.88%	86.05%	74.16%
MF [18]	0.26	88.40%	**86.47%**	75.01%
Ours	0.28	**88.79%**	86.45%	**75.18%**

Table 4. The table compares timing and accuracy of our approach (last line) against the baseline for pose estimation problem. Observe that our approach achieves almost 70× speedup, and almost 7% improvement in accuracy over the dual-decomposition model of Sheasby et.al. [1].

Method	T(s)	L	R	TO	H	Overall
Sheasby [1]	20	61.3	63.5	81.5	85.1	69.17
Yang [7]	1	66.6	71	87.3	90.5	75.6
Ours	0.28	68.2	71	87.6	90.2	76.05

be penalized. Further improvement through pose estimation within the detected boxes remains a possibility to our approach as well. However, our main goal is to show the efficiency achieved by our joint model without losing any accuracy given the same initial conditions. We also observe an improvement in qualitative results for both the segmentation and pose estimation problems, shown in Fig. 4.

7 Discussion

In this work, we proposed *PoseField*, an efficient mean-field based method for joint estimation of human segmentation, pose, per-pixel part and disparity. We formulated this product label space problem in a hierarchical framework, which captures interactions between the pixel level (human/background, disparity, and body part labels), and the part level (head, torso, arm). Finally we have shown the value of our approach on the H2View and Buffy datasets. In each case, we have shown substantial improvement in inference speed (almost $20 - 70\times$) over the current state-of-the-art dual-decomposition methods, while also observing a good improvement in accuracies for both human segmentation and pose estimation problems. We believe our efficient inference algorithm would provide an alternative to some of the existing computationally expensive inference approaches in many other fields of computer vision where joint inference is required. Future directions include investigating new ways to improve the efficiency through parallelization and learning of the relationships between different layers of the hierarchy in a max-margin framework.

Acknowledgment. The work was supported by the EPSRC and the IST programme of the European Community, under the PASCAL2 Network of Excellence. Professor Philip H.S. Torr is in receipt of a Royal Society Wolfson Research Merit Award.

References

1. Sheasby, G., Valentin, J., Crook, N., Torr, P.: A robust stereo prior for human segmentation. In: Lee, K.M., Matsushita, Y., Rehg, J.M., Hu, Z. (eds.) ACCV 2012, Part II. LNCS, vol. 7725, pp. 94–107. Springer, Heidelberg (2013)
2. Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: CVPR, pp. 1–8 (2008)
3. Sun, M., Kohli, P., Shotton, J.: Conditional regression forests for human pose estimation. In: CVPR, pp. 3394–3401. IEEE (2012)
4. Sigal, L., Black, M.: Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Brown Univertsity TR, 120 (2006)
5. Kumar, M., Zisserman, A., Torr, P.: Efficient discriminative learning of parts-based models. In: CVPR, pp. 552–559 (2009)
6. Winn, J., Shotton, J.: The layout consistent random field for recognizing and segmenting partially occluded objects (pdf) (2006)
7. Yang, Y., Ramanan, D.: Articulated pose estimation with flexible mixtures-of-parts. In: CVPR, pp. 1385–1392 (2011)
8. Bray, M., Kohli, P., Torr, P.: PoseCut: Simultaneous segmentation and 3D pose estimation of humans using dynamic graph-cuts. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part II. LNCS, vol. 3952, pp. 642–655. Springer, Heidelberg (2006)
9. Kumar, M., Torr, P., Zisserman, A.: Objcut: Efficient segmentation using top-down and bottom-up cues. PAMI 32, 530–545 (2010)
10. Ladicky, L., Russell, C., Kohli, P., Torr, P.H.S.: Graph cut based inference with co-occurrence statistics. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 239–253. Springer, Heidelberg (2010)
11. Ladický, L., Sturgess, P., Russell, C., Sengupta, S., Bastanlar, Y., Clocksin, W., Torr, P.: Joint optimisation for object class segmentation and dense stereo reconstruction. BMVC, 104.1–104.11 (2010), doi:10.5244/C.24.104
12. Komodakis, N., Paragios, N., Tziritas, G.: Mrf energy minimization and beyond via dual decomposition. PAMI, 1
13. Wang, H., Koller, D.: Multi-level inference by relaxed dual decomposition for human pose segmentation. In: CVPR, pp. 2433–2440 (2011)
14. Sheasby, G., Warrell, J., Zhang, Y., Crook, N., Torr, P.: Simultaneous human segmentation, depth and pose estimation via dual decomposition. BMVW (2012)
15. Koller, D., Friedman, N.: Probabilistic graphical models: principles and techniques. MIT Press (2009)
16. Vineet, V., Warrell, J., Torr, P.H.S.: Filter-based mean-field inference for random fields with higher-order terms and product label-spaces. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 31–44. Springer, Heidelberg (2012)
17. Shotton, J., Fitzgibbon, A.W., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth images. In: CVPR, pp. 1297–1304 (2011)

18. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In: NIPS, pp. 109–117 (2011)
19. Shotton, J., Winn, J.M., Rother, C., Criminisi, A.: *TextonBoost*: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 1–15. Springer, Heidelberg (2006)
20. Adams, A., Baek, J., Davis, M.A.: Fast high-dimensional filtering using the permutohedral lattice. Comput. Graph. Forum 29, 753–762 (2010)
21. Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector machine learning for interdependent and structured output spaces. In: ICML (2004)
22. Ladický, L., Russell, C., Kohli, P., Torr, P.: Associative hierarchical crfs for object class image segmentation. In: ICCV, pp. 739–746 (2009)
23. Andriluka, M., Roth, S., Schiele, B.: Pictorial structures revisited: People detection and articulated pose estimation. In: CVPR, pp. 1014–1021 (2009)

Semantic Video Segmentation from Occlusion Relations within a Convex Optimization Framework

Brian Taylor[1], Alper Ayvaci[2], Avinash Ravichandran[1], and Stefano Soatto[1]

[1] University of California, Los Angeles
[2] Honda Research Institute

Abstract. We describe an approach to incorporate scene topology and semantics into pixel-level object detection and localization. Our method requires *video* to determine occlusion regions and thence local depth ordering, and any visual recognition scheme that provides a score at local image regions, for instance object detection probabilities. We set up a cost functional that incorporates occlusion cues induced by object boundaries, label consistency and recognition priors, and solve it using a convex optimization scheme. We show that our method improves localization accuracy of existing recognition approaches, or equivalently provides semantic labels to pixel-level localization and segmentation.

1 Introduction

Object detection, recognition, and localization in images and video have occupied a large portion of the Computer Vision community in the last decade. Detection refers to the binary decision as to whether an object (or object class, as represented by a prior model or training set) is present in an image. A positive outcome for a specific object class provides an answer to the recognition (or categorization) task. Given that an object is present, localization, performed at each pixel [1], is the binary decision as to whether it back-projects onto an object in the scene having said object label. This is often referred to as "semantic (image) segmentation."

However, a key challenge in labeling a scene using only a single image lies in obtaining accurate object boundaries. Low-level segmentation schemes fall short when the object of interest is highly textured and semantic classifiers tend to "bleed" across image boundaries due to the support requirement for different features. Motivated by this limitation, existing methods have proposed to incorporate additional cues (e.g. object detectors) into the framework, extend the label consistency using higher order potentials or take advantage of temporal information by using video sequences.

With video data, the temporal consistency between adjacent frames has often been used to "smooth" labeling errors at the individual frame level. However, video sequences can also provide strong cues for object boundaries. When objects or the camera move, they uncover portions of the scene that were previously hidden and obscure other portions that were previously visible from the viewer's perspective. These newly revealed/occluded regions in the image naturally co-occur with object boundaries and

[1] This is unlike "coarse localization" that aims to determine a few location parameters, say position and scale of a bounding box.

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 195–208, 2013.
© Springer-Verlag Berlin Heidelberg 2013

thus provide constraints on the spatial distribution of class labels, since objects generally project to simply-connected regions of the image. With respect to the viewer, occlusion regions also provide local depth ordering relations between the occluder (in front) and the occluded (behind) objects. We exploit these relations, inferred from video, to produce a segmentation of the image domain into semantic class labels and depth ordering labels. We do so within a convex optimization framework that integrates low-level (scene topology), mid-level (label consistency), and high-level (semantic class label) priors in the inference process. The result can be seen as either a method for segmentation that provides class identity, or a method for recognition that provides accurate pixel-level localization (Fig. 1).

2 Related Work

Pixel-level labeling in single images has been well-studied, where conditional random fields (CRFs) have become a popular tool of choice [1,2,3,4]. Significant work has been accomplished since the introduction of TextonBoost [1], including the development of robust P^N potentials [2] to model pixel relationships beyond pairwise constraints and their hierarchical extension [4], leading to considerable visual improvement in the labeling. Inference remains tractable as [2] showed that these higher order potentials can be reduced to pairwise terms and minimized using graph cuts.

A number of methods incorporate additional cues into this framework, for example 3D information from stereo [5,6] or from structure from motion (SfM) [7]. An alternative is to add appearance-based geometric labels (i.e. vertical, horizontal, sky) [8], and a model for camera viewpoint [9], which can provide powerful constraints on the plausible locations of objects in the scene [10]. [11] takes advantage of powerful sliding window detectors [12,13] to facilitate the labeling of objects (e.g. cars, persons) as they project to smaller regions in the image than background classes (e.g. building, road, or sky), an issue our approach also addresses. Incorporating multiple cues facilitates [5,11,8,9] the joint optimization of coupled problems (e.g. semantic labels and depth estimates).

To leverage temporal information from video, most existing work [14,15,6] enforces label consistenty in a temporal window to smooth the output labels. [14] leverages a spatial CRF with temporal information to independently model scene and object motion. Alternatively, label smoothing can be accomplished via a Hidden Markov Model [15] layered on top of classifier output. Recent work [6] enforces temporal consistency on labels by associating their 3D locations, estimated from stereo image-pairs, across frames. This approach works well for static environments, but struggles with reconstructing and labeling moving objects like cars and people.

A related line of work is object segmentation in video, where temporal consistency can be exploited by matching sparse features across images [16]. Recent results [16,17,18] in unsupervised video object segmentation highlight the importance of using motion as a segmentation cue. Such methods can be adopted for semantic labeling by first segmenting the video sequence followed by labeling each segment. However, failures in the first step cannot be recovered at the categorization step.

Our **contributions** can be summarized as follows: first, we propose a unified criterion (5) that incorporates categorical information, label consistency constraints, and

low-level topological cues from occluded regions for the semantic segmentation task (Sect. 3). In addition to pixel-level object categorization, this approach also provides pixel-level layer labels. When objects of the same category occlude each other, instead of merely categorizing the union of their support as a single category, the layer label indicates the presence of multiple objects in that region and accurately localizes each instance at a pixel-level. This scenario is shown in Fig. 1. Second, we show that our optimization criterion can be converted into a linear program and solved efficiently (Sect. 4), once depth ordering constraints induced by occluded regions are provided. Such constraints can only be obtained using video data [19].

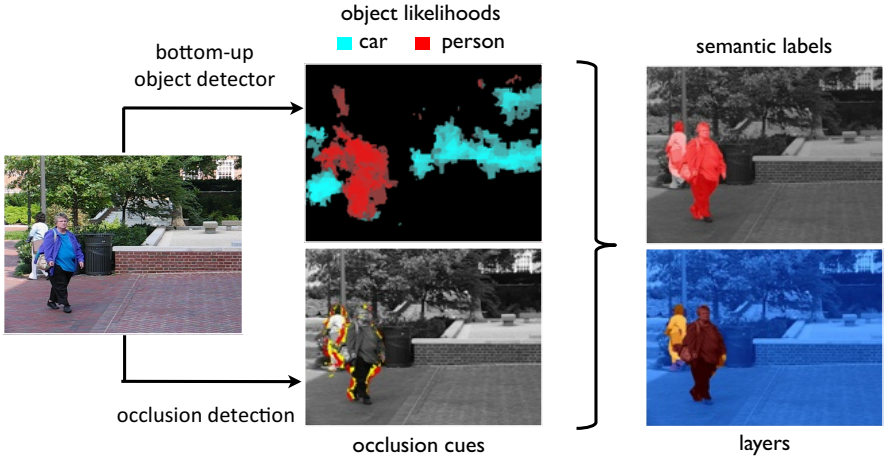


Fig. 1. Object probabilities from class-specific detectors (top) and occlusion cues (bottom), with red and yellow indicating occluded and occluder regions. A single optimization solved efficiently with a linear program solver produces pixel-level classification. Our approach not only provides a semantic label ("person" or "background") for each pixel (top right), but also determines there are *two instances* of the same object (bottom right), one occluding the other, where the person marked by yellow is behind the person marked by brown.

3 Formalization

Given a video sequence $I_t : D \subset \mathbb{R}^2 \to \mathbb{R}^+; x \mapsto I_t(x),\ t = 1 \dots N$, we are interested in determining whether an object of interest is present in the scene, where in the image it projects to, and the spatial relationship between different objects (e.g. multiple instances of the same class) with respect to the viewer. The resulting labeling is represented by a set of binary indicator functions $u^l : D \to \{0, 1\}; x \mapsto u^l(x)$ of the presence/absence of an object of category $l \in \mathcal{L} = \{1, 2, 3, \dots, L\}$ at location $x \in D$.

Let $\mathcal{I}$ be the set of all possible images defined on the domain D and $\mathcal{S} = \{s \mid s \subset \mathbb{R}^2, (0, 0) \in s\}$ be the set of regions centered at the origin. A detector for object l is a function $f^l : D \times \mathcal{I} \times \mathcal{S} \to [0, 1]; (x, I, s) \mapsto f^l(x, I, s)$ that yields the probability that object l projects onto a location x based on statistics of the image restricted to a neighborhood s of x: $I|_{\{x+x' | x' \in s\}}$.

Label Consistency: In principle, we could run a detector for every pixel location $x \in D$ independently of the outcome of detectors at other locations $y \in D$. Clearly, this would miss important correlations and spatial regularities that natural images exhibit. To take these into account, we enforce label consistency between neighboring locations $x, y \in D$ by requiring that $|u^l(x) - u^l(y)|$ be small unless a data-dependent measure $d\mu(x, y)$ provides evidence of the contrary. Such a measure can be written in terms of a kernel $d\mu(x, y) = K(x, y)dxdy$, where $K(x, y) = e^{-\gamma(I_t(x)-I_t(y))^2}$ if $\|x - y\|_2 < \epsilon$ and 0 otherwise, where γ is a regularization parameter. Using the detector responses $\{f^l(x)\}_{l=1}^L$ and the label consistency, instantaneous localization at time t (subscript omitted for simplicity) could be framed as the following optimization problem

$$\begin{aligned}\{\hat{u}^l\}_{l=1}^L = \underset{u^0 \ldots u^L}{\operatorname{argmin}} \alpha \sum_{l=1}^{L} \int_D (1 - f^l(x))u^l(x)dx + \beta \sum_{l=1}^{L} \int_D |u^l(x) - u^l(y)|d\mu(x, y) \\ \text{s. t. } \sum_{l=1}^{L} u^l(x) = 1, \ \ \forall x \in D\end{aligned} \tag{1}$$

where α, β are tuning parameters. $\sum_{l=1}^L u^l(x) = 1$ enforces that only a single object be present at each pixel (opacity).

Topological (Occlusion) Constraints. Multiple images $\{I_t\}_{t=1}^T$ provide more than just repeated trials to compute the likelihood in (1). They provide information on the topology of the surfaces in the scene. Under the assumption of Lambertian reflection, constant illumination and co-visibility, an image frame I_t is related to its immediate neighbors I_{t+dt}, I_{t-dt} by the usual brightness-constancy equation

$$I_t(x) = I_{t\pm dt}(x + v_{\pm t}(x)) + n_{\pm}(x), \ \ x \in D \backslash \Omega_{\pm t} \tag{2}$$

where v_{+t} and v_{-t} are the forward and backward motion fields [19]. However, in general, there are regions, occlusions and disocclusion, in the current image at time t that are not visible in the next and previous frames, Ω_{+t} and Ω_{-t}. Let Ω_t be $\Omega_{+t} \cup \Omega_{-t}$. Then Ω_t^c is the *occluder*, which can be restricted to a local region around the occluded regions Ω_t by using morphological operations. The relation between Ω_t^c be Ω_t induces an ordering between two neighboring surfaces where the *occluder* maintains a higher depth value (closer to camera) then the *occluded* one. One can enforce this ordering as a constraint to estimate a depth layer field $c_t = D \to \mathbb{Z}^+; x \mapsto c_t(x)$ at time t, which maps each image location x to an integer indicating the depth order [20]. This allows us to handle multiple layers (see Fig. 1). Omitting the subscript t for simplicity, we can infer c by solving the minimization problem,

$$\begin{aligned}\hat{c} = \underset{c}{\operatorname{argmin}} \int_D |c(x) - c(y)|d\mu(x, y) \\ \text{s. t. } c(x) > c(y), \ \ x \in \Omega^c, \ y \in \Omega, \ \|x - y\| < \varepsilon.\end{aligned} \tag{3}$$

The super-level set $\{x|c(x) \geq 1\}$ is an indicator for simply-connected regions of the image called "detachable objects" [20]. We adopt this characterization of objects as

label-consistent regions that do not violate occluding boundaries. As Fig. 1 shows, it is possible for an object of a given class to occlude another of the same class. Thus an occlusion boundary does not necessarily force a change of semantic label.

Semantic Priors: While labels obtained by solving (3) are consistent with the topology of the scene, there is nothing that couples them with object identity. Prior knowledge implicit in the training set, which could facilitate detection and localization, is ignored. We need to enable cross-talk between high-level (semantic) priors and low-level (topological) constraints, so that object hypotheses from the probabilistic detectors can inform occlusion-consistent segmentation and vice-versa. To this end, we can relate the label field to the object identity via

$$c(x) \geq \sum_{l \in \mathcal{F}} u^l(x), \;\; \forall\, x \in D, \tag{4}$$

where $\mathcal{F} \subset \mathcal{L}$ denotes the set of "foreground" classes. This constraint not only implies that, when $c(x) = 0$, we obtain $\sum_{l \in \mathcal{F}} u^l(x) = 0$ based on the exclusion constraint $\sum_{l=1}^{L} u^l(x) = 1$, but also encourages $c(x)$ to become nonzero when there is a strong detector response for any object class. Unfortunately, we cannot add additional constraints between the layers and the object labels as any object can be present in any layer. With this constraint, we propose the following optimization problem:

$$\begin{aligned}
\{\hat{u}^l\}_{l=1}^{L}, \hat{c} = \underset{u^0 \ldots u^L, c}{\operatorname{argmin}} \; & \alpha \sum_{l=1}^{L} \int_D (1 - f^l(x)) u^l(x) dx + \beta \sum_{l=1}^{L} \int_D |u^l(x) - u^l(y)| d\mu(x,y) \\
& + \int_D |c(x) - c(y)| d\mu(x,y) \\
\text{s.t.} \; & \sum_{l=1}^{L} u^l(x) = 1, \;\; \forall x \in D, \\
& c(x) > c(y), \; x \in \Omega^c, \; y \in \Omega, \; \|x - y\| < \varepsilon \\
& \boxed{c(x) \geq \sum_{l \in \mathcal{F}} u^l(x).}
\end{aligned} \tag{5}$$

This inference criterion incorporates high-level (category), mid-level (label consistency) and low-level (occlusion) information. In what follows, we show how the above minimization problem can be solved efficiently with *linear programming*.

4 Implementation

The domain D is a discrete lattice Λ of $M \times N$ pixels, that we can represent as a graph $\mathcal{G} = (V, E)$: each pixel is identified with a node i in the vertex set V, and adjacency relationships between pixels i and j are encoded in the edge set $E \subset V \times V$ via $i \sim j$. The location of each pixel i in the image is given by $x_i \in D$. Abusing notation, we define discrete versions of the category indicator functions $\{u^l : V \to \{0,1\}; i \mapsto u_i^l\}_{l=1}^{L}$,

the layers $c : V \to \mathbb{Z}^+; i \mapsto c_i$ and the object detectors $\{f^l : V \to [0,1]; i \mapsto f_i^l\}_{l=1}^L$ where each f_i^l is the probability of vertex i having semantic label l. By introducing auxiliary variables $\{d_{ij}\}_{i\sim j}$ and $\{v_{ij}\}_{i\sim j}$ and relaxing the integer constraints for $\{u^l\}_{l=1}^L$ and c, we can write (5) as a linear program:

$$\begin{aligned}\{\hat{u}^l\}_{l=1}^L, \hat{c} = \underset{u^1 \dots u^L, c}{\operatorname{argmin}} & \underbrace{\alpha \sum_{l=1}^{L} \sum_{i \in V} (1 - f_i^l) u_i^l + \beta \sum_{l=1}^{L} \sum_{i \sim j} v_{ij}}_{\phi_{\text{semantic}}} + \underbrace{\sum_{i \sim j} d_{ij} + \mu \sum_{i \in V} c_i + \lambda \sum_{k=1}^{K} \xi_k}_{\phi_{\text{layers}}} \\ \text{s. t.} & \sum_{l=1}^{L} u_i^l = 1, \ \forall x \in D, \\ & -v \preceq Gu \preceq v, \quad -d \preceq Gc \preceq d \\ & c_i - c_j \geq 1 - \xi_k, \ x_i \in \Omega^c, \ x_j \in \Omega, \ i \sim j, \\ & c_i \geq \sum_{l \in \mathcal{F}} u_i^l. \end{aligned} \tag{6}$$

where G is a $|E| \times |V|$ matrix that acts as a weighted gradient operator on the functions defined on the graph $\mathcal{G}$ which is given by

$$G_{i\sim j,k} = \begin{cases} w_{ij}, & \text{if } k = i \\ -w_{ij}, & \text{if } k = j \\ 0, & \text{otherwise,} \end{cases} \tag{7}$$

and the measure $d\mu(x_i, x_j)$ becomes a symmetric matrix $w_{ij} = K(x_i, x_j)$. Also note that we incorporate model selection via $\sum_i c_i$ in (6) which penalizes spurious object layers. We also introduce a slack variable set $\{\xi_k\}$, one for each of K occluder-occluded constraints on a pair of pixels, allowing us to ignore spurious occlusion constraints by paying a penalty λ. This enables us to deal with inevitable errors in occlusion detection. The formulation of the problem as a linear program enables us to benefit from efficient numerical solvers.

Following common practice to reduce the complexity of the optimization, we partition the domain D into N *non-overlapping* superpixels $\{s_i\}_{i=1}^N$ such that $\bigcup_{i=1}^N s_i = D$, using a statistical multi-cue edge detector [21]. The form of the optimization (6) is unchanged, since it is already written for a general graph. This reduces the size of the linear program to a manageable complexity.

5 Experiments

We tested our model on video sequences from the MOSEG dataset [16] as well as the CamVid dataset [22]. MOSEG contains short sequences (20-60 frames) of moving objects such as cars and people with ground truth annotations for a few frames in each sequence. Since ground truth is only provided for moving objects, we exclude detection

of stationary ones for evaluation purposes. For our framework, we choose cars and people to be the object classes and construct a generic background class.

The CamVid database consists of longer clips of driving sequences taken under different conditions (e.g. day and dusk). The motion between adjacent frames is much larger than in MOSEG as the camera is mounted on a fast-moving platform. Annotations are provided sparsly at 1 Hz for 32 semantic classes. However, we train and test on only the 11 classes shown in Fig. 2 as common practice when evaluating on CamVid [7,11,23]. For our framework, we consider cars, sign-symbol, pedestrian, column-pole, and bicyclist as the object classes and building, tree, sky, road, fence, and sidewalk as the background classes.

While our algorithm is designed to run on video, the categorization component only considers single image-based features. This is convenient as there exist many publicly available image categorization datasets with densely labeled annotations and a number of baseline evaluation methods to choose from. For our experiments with MOSEG, we construct three classes (car, person, background) from the Graz02 database [24]. We chose not to train on MOSEG as testing would then be trivial and limited given the small number of annotated frames. An advantage of our approach is that any single image semantic labeling scheme can be used as our baseline categorization method, so long as it provides per-class probability maps.

For our experiments on MOSEG, we used [25] for our categorization component, a bag-of-features approach shown to perform well for single image recognition and localization on Graz02 and Pascal VOC 2007/2009 [26]. Dense SIFT features [27] are extracted and quantized via K-means ($K = 400$). A histogram of these quantized features is constructed for each superpixel, which aggregates information within a local superpixel neighborhood for training an SVM classifier with an intersection kernel.

For our experiments on CamVid, we followed the dataset's default training and testing split. For the categorization component, we used [4] to run an improved version of TextonBoost that incorporates additional features such as Local Binary Patterns, colour histograms, location, and dense SIFT to obtain per-pixel object class probabilities. Occlusion detections and optical flow estimation are performed by [19], for which source code is available on-line.

5.1 Comparative Evaluation

MOSEG: We compared five approaches: (i) the baseline categorization component [25] applied to each frame of the video; (ii) TextonBoost which partitions still images into regions each labeled with a specific object class considering textural properties of objects [1]; (iii) the work of [4], which augments the pixel-wise graph cut labeling framework with label consistency relationships on image regions and across a segmentation hierarchy, (iv) the two-stage pipeline first partitioning the image into segments and then classifying each segment; and (v) the proposed approach that unifies the segmentation and classification steps into a single energy minimization framework.

We accomplish method (i) by assigning each pixel to the class inferred from the detection probabilities under a CRF model enforcing spatial continuity. We use publicly

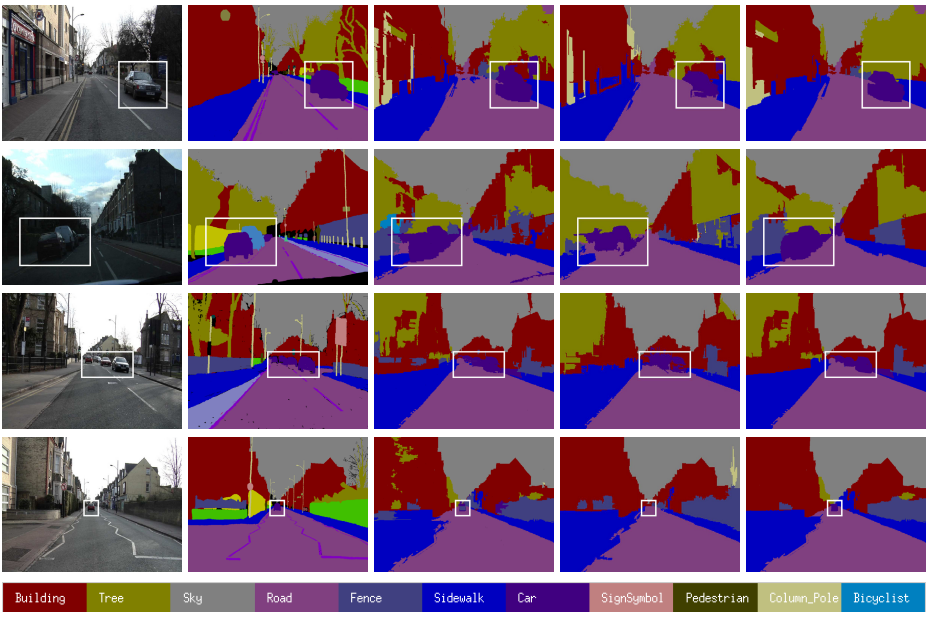


Fig. 2. Results on CamVid sequences. (Left to right) original frame, ground truth annotations (col. 2), labeling results from [1,4] (col. 3, 4), and the output of our proposed method (col. 5). Notice how the proposed method better captures the boundaries of the cars (highlighted with white boxes).

available code for (ii)[2] and (iii)[3] [28]. (iv) is accomplished by image segmentation from the depth layers estimated by [20] and labeling the connected components of all the depth layers using a bag-of-features classifier trained on the Graz dataset. (v) is our method.

CamVid: We compare the baseline (TextonBoost with an augmented feature vector), the single image-labeling approach of [4], and our proposed method. Here we also enforce temporal consistency by connecting the class and layer nodes in each frame to those in the surrounding frames. This amounts to solving equation 6 on a larger graph. Thus we label multiple frames in one minimization. In our experiments, we used a 7 frame temporal window.

5.2 Qualitative Performance

Fig. 3 provides a basic intuition of our system. By combining the output of class-specific bottom-up object detectors and occlusion detectors (2nd and 3rd column, respectively), we can accurately segment and classify the vehicles (5th column), which not only

[2] `http://jamie.shotton.org/work/code.html`
[3] `http://www.robots.ox.ac.uk/~lubor/`

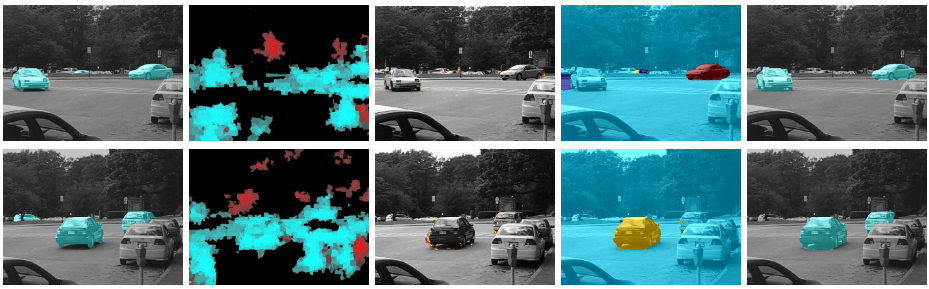


Fig. 3. Our algorithm compared to baseline methods on cars5 (top row) and cars2 (bottom row) from MOSEG. Left to right: original image with groundtruth overlaid (col. 1), detector responses for category car (cyan) and person (red) (col. 2), occluded (red) and occluder (yellow) regions (col. 3), layer segmentation produced from occlusion cues (dark blue is the lowest layer and red is the highest) (col. 4), and finally our classification result (col. 5). Notice the classification baseline labels much of the image as car, while the layer segmentation often misses objects due to errors in occlusion detection. In the top row, the car is in the same layer as the background while the region immediately to its left is separated. In both examples, all approaches struggle with the moving car in the far field.

improves on the baseline classifier in localization (2nd column), but also improves on the segmentation from occlusion cues alone (4th column). In addition, our framework provides an object category label to each segmented image region.

In the first row, the left vehicle is difficult to extract via occlusion cues alone as too few are generated while the car is turning. By incorporating the response of a car detector, our algorithm is able to accurately recover the leftmost vehicle. We are still unable to extract the moving car in the background, however, as the detector response is not significant enough in this region. This shows the interplay between the two object localization approaches. However, in comparison to methods (i) and (ii), our approach improves on the results in the localization and classification of objects.

It should be noted that our method not only produces a pixel-level localization, but also a global depth ordering, which partitions the image into a set of depth layers. Thus, if a person in the scene occludes another one, we do not merely label an amorphous region of pixels that responded to a "person detector." Instead, the combination of the semantic label field and the depth layer distinguishes the individual object instances and indicates which person is in front of the other person in the scene from the viewer's perspective. This is visible in Fig. 1 for the case of people. The work of Yang et al. [29] similarly takes advantage of depth layers. However, their layers are based on the output of single image object detectors and shape priors. We instead used the motion of objects in the scene.

We present qualitative results of our algorithm and the comparison methods evaluated on the MOSEG sequences in Fig. 5. Our algorithm consistently makes improvement over the labeling schemes for single images (the baseline [25], TextonBoost [1], and [4]), which is particularly evident at object boundaries. Thus, occlusion cues can help regularize the probabilities generated by the baseline detection system. Note that in the fourth column (person2), [4] fails around the head of the person. This is likely due to

a failure in the oversegmentation step. With occlusion cues, our approach captures the silhouette of the woman's head. However, in this frame we fail to capture the woman in the background, as the person detection response is weak in that region. Note that where very small or very large motion occurs, occlusion detection fails. Despite these issues, our method still matches or outperforms the two-step pipeline approach, recovering from failures in layer estimation due to weak occlusion cues (the vehicle windshield is mislabeled as background in cars4) or spurious occlusion detections (cars9, cars8).

Qualitative results of our algorithm and the comparison methods evaluated on frames from CamVid are presented in Fig. 2, where we compare the output of the extended version of TextonBoost, [4], and our proposed method. Although it is difficult to declare one approach as superior for labeling the background classes, our approach clearly obtains more accurate boundaries when considering foreground objects (here, cars), which we highlight with white boxes. In these sequences, the significant camera motion generates strong occlusion cues for accurate segmentation, where single image appearance features may lead to a faulty object segmentation (row 2).

5.3 Quantitative Assessment

Table 1 details the quantitative results of the comparison methods and our proposed algorithm. F-measure scores were computed for all of the annotated frames in MOSEG and averaged per sequence. The proposed algorithm outperforms the baseline and performs on par or better than the two-step approach. In a few cases, the recent method [4] performs better, particularly for sequences when the moving car is larger. Our algorithm falls short of the two step approach on person2, where the rear left woman is partially occluded while exiting the camera's field of view. We do not test our scheme on the "Tennis" and "Marple" sequences, since the annotated segmentations capture a range of scale and viewpoint that is not represented in Graz02, our training set.

The results for the CamVid database are shown in Table 2. In most cases, we outperform [1,4], especially on the object categories (e.g. cars). Often, for applications such as autonomous driving, mislabeling a portion of sky as building is of much less consequence than mislabeling a person crossing the street as drive-able road. Given that object classes often have a much smaller spatial support in the image, the addition of occlusion cues help us to more reliably detect and accurately label the objects (e.g. cars, people, bicyclists).

After pre-processing (occlusion detection and probability map generation), the optimization takes 2-3 minutes/frame in CVX [30,31] and approximately 15 seconds with MOSEK, a commercial tool. The entire pipeline including optical flow and occlusion detection, edge detection and categorization scheme followed by the minimization stage runs in 4.35 minutes for a 720×960 frame. We use a GPU implementation of the optical flow estimation and run it on an NVIDIA GeForce GTX 560Ti, while the other stages are performed on a single core 2.5 Ghz CPU.

In order to visualize the effects of the parameters, we obtained the minimizer of $\phi_{\text{semantic}} + \omega\phi_{\text{layers}}$ for different choices of ω. We fixed α and β and used ω to scale the other parameters i.e. ω, $\omega\mu$ and $\omega\lambda$. Fig. 4 illustrates the variation of the pixel-level

Fig. 4. F-measure scores for pixel level classification results obtained by varying ω between 0.2 and 2. The F-measure remains constant for the intervals depicted in red, green and blue. Hence, we only show one exemplar solution from each interval.

classification accuracy as a function of ω. Note that the classification results remain constant over large intervals of ω. This shows that our approach is robust to reasonable perturbations in the parameters.

Failure Modes: Our method does not always work. In the case when the training set is not representative of the conditions to be encountered in the test set, unsurprisingly the semantic prior does not help improve segmentation. Our method is also susceptible to gross failures in occlusion detection, although it compensates for minor errors with the introduction of slack variables. We are also sensitive to failures of the low-level object detection module.

Table 1. Performance results for [25,1,4], the two step segment-then-classify approach, and finally the proposed method on the MOSEG [16] dataset. Reported F-measure scores have been computed for the groundtruth class in each image and averaged over the sequence. In cars4, objects from classes 'People' and 'Car' both appear in the scene.

Sequence	cars1	cars2	cars3	cars4	cars5	cars6	cars7	cars8	cars9	cars10	person1	person2
Fulkerson et al. [25]	0.71	0.51	0.50	0.34	0.41	0.26	0.31	0.57	0.44	0.39	0.38	0.65
TextonBoost [1]	0.37	0.59	0.44	0.57	0.47	0.00	0.51	0.01	0.29	0.18	0.13	0.41
Ladicky [4]	**0.93**	0.73	0.79	0.66	0.74	0.62	0.82	0.73	**0.87**	0.61	**0.83**	0.84
TwoStep	0.86	0.83	0.92	0.88	0.58	**0.95**	0.91	0.88	0.68	0.90	0.03	**0.95**
Our Method	0.90	**0.90**	**0.96**	**0.94**	**0.86**	**0.95**	**0.92**	**0.89**	0.69	**0.92**	0.72	0.85

Table 2. Performance results for [1,4] and the proposed method on CamVid [16]. F-measure scores were computed for each class and 'Average' is the average of the F-measure scores. Bolded numbers show the best performance for the specified category. In general, our method performs better than comparison methods. In particular, note that the use of occlusion cues allows us to outperform the other methods in all but one object category.

Class	Building	Tree	Sky	Road	Fence	Sidewalk	Car	Column-Pole	Pedestrian	Sign-Symbol	Bicyclist	Average
TextonBoost* [1]	70.2	56.7	**93.1**	**87.0**	31.8	76.0	53.9	13.2	-	44.5	24.0	55.0
Ladicky [4]	**72.7**	**62.4**	92.3	86.4	31.9	75.6	59.0	18.0	-	32.9	**30.9**	56.2
Our Method	71.6	57.0	92.9	**87.0**	**37.5**	**76.1**	**60.9**	**19.8**	-	**54.4**	24.3	**58.2**

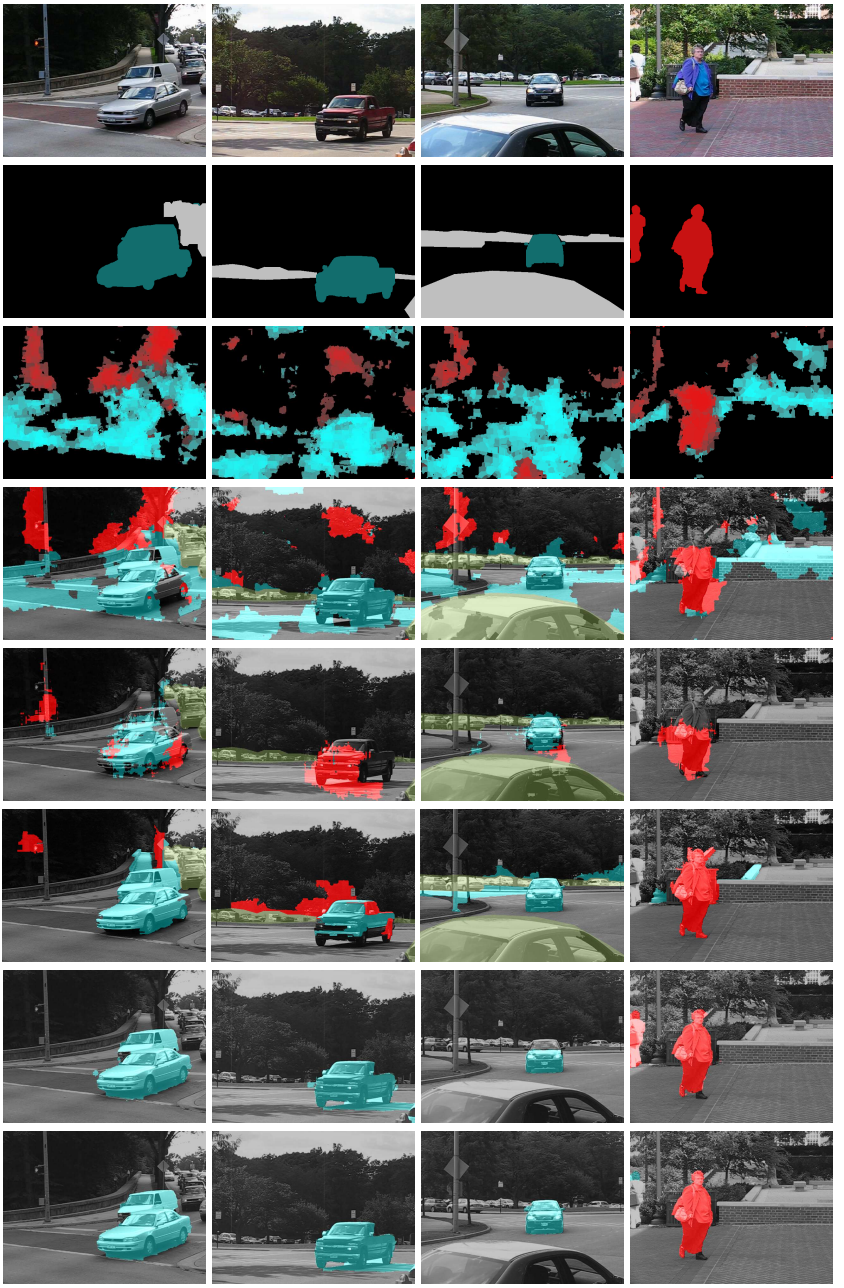


Fig. 5. Results on (left to right) cars{9,10,4}, and person2 from MOSEG. (Row 1) Original frame. (Row 2) Ground truth for cars (blue), person (red), and masked regions (gray). (Row 3) Object probabilities from the baseline [25]. (Rows 4,5,6) Results of [25,1,4], respectively, with masked regions in yellow. (Row 7) Segment-then-classify approach. (Row 8) Proposed method

6 Discussion

We have presented a method to integrate (low-level) local topology information from occlusions, (mid-level) label consistency, and (high-level) semantic information via object class detectors to provide pixel-level localization in object recognition, or equivalently to augment segmentation with object labels. We use existing occlusion detection methods to extract occlusion regions from video and thus provide local depth ordering constraints and existing object detection schemes to provide the probability of correct detection for a region of an image relative to a training set of objects or object classes. We incorporate all of these cues into a cost functional that we discretize and solve using modern combinatorial optimization schemes. Our approach improves localization performance of existing segmentation and recognition schemes, including those enforcing label consistency using conditional random fields or other generic priors.

Acknowledgement. Research supported by ONR N000141110863 and ARO 56765-CI.

References

1. Shotton, J., Winn, J., Rother, C., Criminisi, A.: *TextonBoost*: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 1–15. Springer, Heidelberg (2006)
2. Kohli, P., Ladicky, L., Torr, P.H.S.: Robust higher order potentials for enforcing label consistency. International Journal of Computer Vision (2009)
3. Gould, S., Gao, T., Koller, D.: Region-based segmentation and object detection (2009)
4. Ladický, L., Russell, C., Kohli, P., Torr, P.H.S.: Associative hierarchical CRFs for object class image segmentation. In: Proc. of the International Conference on Computer Vision (2009)
5. Ladický, L., Sturgess, P., Russell, C., Sengupta, S., Bastanlar, Y., Clocksin, W., Torr, P.H.S.: Joint optimization for object class segmentation and dense stereo reconstruction. International Journal of Computer Vision (2011)
6. Floros, G., Leibe, B.: Joint 2d-3d temporally consistent segmentation of street scenes. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (2012)
7. Sturgess, P., Alahari, K., Ladický, L., Torr, P.H.S.: Combining appearance and structure from motion features for road scene understanding. In: British Machine Vision Conference (2009)
8. Tighe, J., Lazebnik, S.: Superparsing: Scalable nonparametric image parsing with superpixels. International Journal of Computer Vision (2012)
9. Gould, S., Fulton, R., Koller, D.: Decomposing a scene into geometric and semantically consistent regions. In: Proc. of the International Conference on Computer Vision (2009)
10. Hoiem, D., Efros, A.A., Hebert, M.: Putting objects in perspective. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (2006)
11. Ladický, Ľ., Sturgess, P., Alahari, K., Russell, C., Torr, P.H.S.: What, where and how many? combining object detectors and CRFs. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 424–437. Springer, Heidelberg (2010)
12. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part based models. IEEE Transactions on Pattern Analysis and Machine Intelligence (2010)
13. Vedaldi, A., Gulshan, V., Varma, M., Zisserman, A.: Multiple kernels for object detection. In: Proc. of the International Conference on Computer Vision (2009)

14. Wojek, C., Schiele, B.: A dynamic conditional random field model for joint labeling of object and scene classes. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 733–747. Springer, Heidelberg (2008)
15. Ess, A., Mueller, T., Grabner, H., van Gool, L.: Segmentation-based urban traffic scene understanding. In: British Machine Vision Conference (2009)
16. Brox, T., Malik, J.: Object segmentation by long term analysis of point trajectories. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 282–295. Springer, Heidelberg (2010)
17. Ochs, P., Brox, T.: Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions. In: Proc. of the International Conference on Computer Vision (2011)
18. Lee, Y.J., Kim, J., Grauman, K.: Key-segments for video object segmentation. In: Proc. of the International Conference on Computer Vision (2011)
19. Ayvaci, A., Raptis, M., Soatto, S.: Occlusion detection and motion estimation with convex optimization. In: Advances in Neural Information Processing Systems (2010)
20. Ayvaci, A., Soatto, S.: Detachable object detection with efficient model selection. In: Boykov, Y., Kahl, F., Lempitsky, V., Schmidt, F.R. (eds.) EMMCVPR 2011. LNCS, vol. 6819, pp. 191–204. Springer, Heidelberg (2011)
21. Martin, D., Fowlkes, C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Transactions on Pattern Analysis and Machine Intelligence (2004)
22. Brostow, G.J., Shotton, J., Fauqueur, J., Cipolla, R.: Segmentation and recognition using structure from motion point clouds. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 44–57. Springer, Heidelberg (2008)
23. Tighe, J., Lazebnik, S.: Understanding scenes on many levels. In: Proc. of the International Conference on Computer Vision (2011)
24. Opelt, A., Pinz, A.: Object localization with boosting and weak supervision for generic object recognition. In: Proc. of the Scandinavian Conference on Image Analysis (2005)
25. Fulkerson, B., Vedaldi, A., Soatto, S.: Class segmentation and object localization with superpixel neighborhoods. In: Proc. of the International Conference on Computer Vision (2009)
26. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge (2011)
27. Lowe, D.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (2004)
28. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Transactions on Pattern Analysis and Machine Intelligence (2004)
29. Yang, Y., Hallman, S., Ramanan, D., Fowlkes, C.C.: Layered object models for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2011)
30. Grant, M., Boyd, S.: CVX: Matlab software for disciplined convex programming, version 1.21 (2011)
31. Grant, M., Boyd, S.: Graph implementations for nonsmooth convex programs. In: Blondel, V.D., Boyd, S.P., Kimura, H. (eds.) Recent Advances in Learning and Control. LNCIS, vol. 371, pp. 95–110. Springer, Heidelberg (2008)

A Co-occurrence Prior for Continuous Multi-label Optimization

Mohamed Souiai, Evgeny Strekalovskiy, Claudia Nieuwenhuis, and Daniel Cremers

Technical University Munich, Germany

Abstract. To obtain high-quality segmentation results the integration of semantic information is indispensable. In contrast to existing segmentation methods which use a spatial regularizer, i.e. a local interaction between image points, the co-occurrence prior [15] imposes penalties on the co-existence of different labels in a segmentation. We propose a continuous domain formulation of this prior, using a convex relaxation multi-labeling approach. While the discrete approach [15] is employs minimization by sequential alpha expansions, our continuous convex formulation is solved by efficient primal-dual algorithms, which are highly parallelizable on the GPU. Also, our framework allows isotropic regularizers which do not exhibit grid bias. Experimental results on the MSRC benchmark confirm that the use of co-occurrence priors leads to drastic improvements in segmentation compared to the classical Potts model formulation when applied .

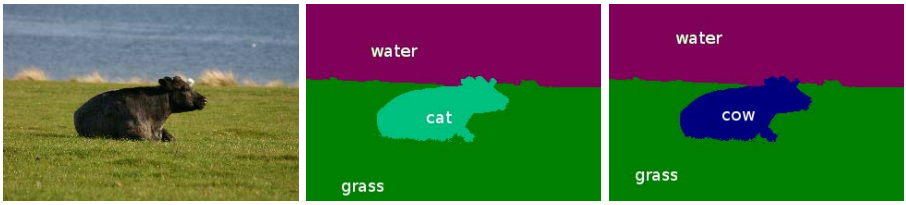
1 Introduction

1.1 Semantic Image Labeling

While traditional image segmentation algorithms have focused on separating regions based on homogeneity of color or texture, more recent methods have aimed at incorporating *semantic* knowledge into what is often called class-based image segmentation. Rather than simply grouping regions of similar color, the goal is to assign to each pixel of an image a semantic label such as “grass”, “sky”, “cow” or “horse”, each of which does not necessarily share the same color model – horses may be white, brown or black for example. Such approaches allow to impose prior knowledge about which pairs of labels are likely to co-occur in a given image [15]. Figure 1 shows semantic labelings computed for the image of a cow on grass with and without a co-occurrence prior: While the color likelihood based data term has a slight preference for cat over cow, the co-occurrence additionally imposes the information that cows are more commonly observed on grass next to the ocean than cats.

A separate line of work has promoted the use of minimum description length (MDL) priors [16,27,5,25] which impose a prior that favors a smaller number of labels in the final segmentation. In practice, the advantage of such MDL priors is that one can preserve a level of regularity while reducing the often over-smoothing boundary length regularization. While many experiments demonstrating the advantage of co-occurrence can often be reproduced with a simple MDL prior (that suppresses the emergence of undesired labels), for the example in Figure 1 co-occurrence is vital since the number of labels is in both cases the same.

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 209–222, 2013.
© Springer-Verlag Berlin Heidelberg 2013



Input image Semantic segmentation with co-occurrence prior

Fig. 1. We propose a convex relaxation for co-occurrence priors in spatially continuous semantic segmentation. Whereas purely data-driven semantic segmentation (middle) assigns the label 'cat' to the cow, co-occurrence priors (right) substantially improve the performance by imposing the knowledge that cows are more commonly encountered next to grass and ocean than cats.

Co-occurrence priors have been applied to the finite-dimensional discrete setting by means of combinatorial problems on a grid [15]. These problems can be cast as large scale integer linear programs which are solved approximatively using α-expansion [1] or the fast primal dual algorithm [14]. However, being defined on the grid these methods are inherently anisotropic. In addition the discrete methods used for solving such problems are sequential and thus only partially parallelizable [13]. In contrast, continuous formulations allow for isotropic regularizers and appropriate discretizations of such functionals do not show grid artefacts. Furthermore, the primal dual algorithm [4] used for solving the continuous saddle-point problem is defined point-wise and can thus be parallelized in a straight-forward manner and run in parallel using modern GPU's or other parallel architectures. For a detailed discussion see [19,11].

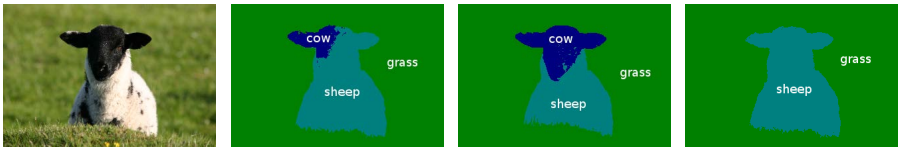
Thus, a major challenge addressed in this paper is how to efficiently integrate co-occurrence priors into a convex continuous optimization approach, which allows for fast solutions independent of the initialization of the algorithm.

In contrast to common segmentation methods we refrain from using super-pixels [18,7,22,15]. Super-pixels prevent pixels with similar colors from being assigned to different labels. As a consequence, elongated structures may be lost or larger chunks may be incorrectly assigned in the final solution – see the head of the sheep in Figure 2 which is assigned to the label 'cow'. To preserve elongated structures we use the non-local total variation formulation [24,8].

1.2 Related Work

The inspiration to this work predominantly draws from two lines of research, namely research on label configuration priors and research on convex relaxation techniques. On the one hand, there are a number of recent advances on label configuration energies for semantic image labeling, including the co-occurrence priors [15], MDL priors [5,27,25,16], and hierarchical label cost priors [5].

On the other hand, there are a number of recent advances on convex relaxation techniques for spatially continuous multi-label optimization. These include relaxations for the continuous Potts model [2,3,17,26], for the non-local continuous Potts model [24], for MDL priors [25], and for vector-valued labeling problems [9,23].



Input no co-occ. Ladicky et al. [15] proposed

Fig. 2. Methods using super-pixel information for improving label consistency can produce mislabeling in the resulting segmentation as can be seen in the case of the approach in [15] which builds on the higher order CRF approach introduces by Kohli et al. [12].

1.3 Contributions

Our contributions are the following:

- We formulate the co-occurrence priors within a spatially continuous approach to semantic multi-label segmentation.
- We propose a convex relaxation of the co-occurrence based segmentation functional which can be solved optimally. This approach yields results independent of intialization and is - in contrast to discrete methods - straighfowardly parallelizable.
- The proposed integration of co-occurrence priors is done on a pixel level and therefore avoids the commonly used pre-segmentation into super-pixels.

2 Convex Multi-label Segmentation

Given a discrete label space $\mathcal{G} = \{1, ..., n\}$ with $n \geq 3$, the multi-labeling problem can be stated as a minimal partition problem. The image domain $\Omega \subset \mathbb{R}^2$ is to be segmented into n pairwise disjoint regions Ω_i which are encoded by the label indicator function $u \in BV(\Omega, \{0,1\})^n$

$$u_i(x) = \begin{cases} 1 & \text{if } x \in \Omega_i, \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

Here BV denotes the space of functions u for which the total variation

$$TV(u) := \sup_{\substack{p \in C_c^1(\Omega;\mathbb{R}^2) \\ |p(x)| \leq 1 \, \forall x}} \int_\Omega u(x) \ \mathrm{div}\, p(x) \, dx \tag{2.2}$$

is bounded allowing for discontinuities. To ensure that each pixel is assigned to exactly one region a point-wise simplex constraint is imposed on u:

$$\sum_{i=1}^{n} u_i(x) = 1 \quad \forall x \in \Omega. \tag{2.3}$$

To find a solution to the minimal partition problem we minimize the general energy E which can be decomposed as follows:

$$E(u) = E_D(u) + E_S(u) + E_C(u), \tag{2.4}$$

The term E_D is called the data term, the expression E_S represents a regularization term usually requiring smoothness of the solution, and the term E_C is the global co-occurrence energy which will be introduced in Section 3.

The data term $E_D(u)$ assigns a cost $\varrho_i(x) : \Omega \to \mathbb{R}$ to each pixel x for belonging to region i (based on its color or texture). It can be written in terms of the indicator functions as

$$E_D(u) = \sum_{i=1}^{n} \int_{\Omega} u_i(x) \varrho_i(x) dx. \tag{2.5}$$

The regularization term $E_S(u)$ imposes a spatial smoothness which can be formulated by means of the Potts model.

The classical total variation based formulation of the Potts model, $\frac{1}{2}\sum_{i=1}^{n} TV(u_i)$, minimizes the length of the interface of each region which leads to an over-smoothing in images exhibiting objects with fine or elongated structures. To improve over this, Werlberger et al. [24] proposed a non-local variant of the Potts model, which improves the labeling quality on the boundaries. The key idea is that pixels x and y are likely to share the same label if they are spatially close *and* have a similar color. For each pair of pixels a weight w is defined, which measures this similarity:

$$w(x,y) = \exp\left[-\left(\frac{d_c(x,y)}{\alpha} + \frac{d_s(x,y)}{\beta} \right) \right]. \tag{2.6}$$

Here d_c and d_s denote the color and spatial distance scaled by the parameters α and β. The regularizer, finally, measures the weighted label differences for each pixel compared to its spatial neighborhood $\mathcal{N}_x$

$$E_S(u) = \sum_{i=1}^{n} \int_{\Omega} \left(\int_{\mathcal{N}_x} w(x,y)\ |u_i(y) - u_i(x)|\ dy \right) dx. \tag{2.7}$$

Introducing a dual variable p transforms the non-differentiable expression (2.7) to a fully differentiable one (for each fixed p):

$$E_S(u) = \sup_{p \in \tilde{\mathcal{K}}} \sum_{i=1}^{n} \int_{\Omega} \left(\int_{\mathcal{N}_x} p_i(x,y)\ (u_i(y) - u_i(x))\, dy \right) dx \tag{2.8}$$

with the convex constraint set $\tilde{\mathcal{K}}$

$$\tilde{\mathcal{K}} := \left\{ p(x,y) \in C^1\left(\Omega \times \Omega, \mathbb{R}\right)^n \ \middle|\ |p_i(x,y)| \le w(x,y) \right\}. \tag{2.9}$$

Thus E_D and E_S above are both convex in u. Note that the above non-local regularizer (2.7), and with it also our energy, is not isotropic. One could also easily use its isotropic version, where the L^1-norm in (2.7) is replaced by $\sqrt{\int_{\mathcal{N}_x} w(x,y)(u_i(y) - u_i(x))^2\, dy}$ as in [8]. However, the results with both versions are almost the same, so that we have chosen the form (2.7) for computational efficiency. In general, our continuous formulation easily allows to incorporate isotropic regularizers, such as the isotropic Potts model

relaxations [2,3]. Additionally, in contrast to graph based methods there is no need for complex graph constructions when solving our optimization problem.

In order to obtain a convex optimization problem, the optimization domain must be convex as well. Therefore, we relax the binary constraints of the indicator function u and — together with (2.3) — obtain the convex set

$$\mathcal{S} := \left\{ u \in BV(\Omega, [0,1])^n \;\middle|\; \sum_{i=1}^{n} u_i(x) = 1 \quad \forall\, x \in \Omega \right\}. \tag{2.10}$$

After optimization of the relaxed problem the final pixel labeling $\mathcal{L} : \Omega \to \mathcal{G}$ can be recovered from the relaxed solution u^* by:

$$\mathcal{L}(x) = \arg \max_{1 \le i \le n} u_i^*(x) \tag{2.11}$$

3 A Continuous Co-occurrence Prior

In this section we introduce the global co-occurrence energy E_C in (2.4), which allows for the integration of semantic scene knowledge, such as for example that sheep and grass often appear together in the same image whereas sheep and wolves are rarely to be found. For each subset of labels $L \subseteq \mathcal{G}$ a specific penalty can be defined or learned from training data. Note that this penalty only depends on the simultaneous occurrence of specific labels in the image, not on their location or the size of their corresponding regions in the segmentation.

3.1 A Convex Formulation of Label Occurrences

In order to devise the co-occurrence prior it is necessary to model the occurrences of specific labels in the image. To this end, we introduce the label indicator function $l : \mathcal{S} \to \{0,1\}^n$,

$$l_i(u) = \begin{cases} 1, & \text{if } \exists x \in \Omega :\ u_i(x) = 1, \\ 0, & \text{otherwise,} \end{cases} \tag{3.1}$$

which indicates for each label $i \in \mathcal{G}$ if it appears in the segmentation given by $u \in \mathcal{S}$. To obtain a convex formulation of the function l, we use the following relation which was already used by Yuan et al. [25]:

$$l_i(u) = \max_{x \in \Omega} u_i(x) \qquad \forall i \in \mathcal{G}. \tag{3.2}$$

where we use $\max$, instead of the formally correct $\operatorname{ess\,sup}$ operator, for readability. Note that the L^∞-norm on the left hand side of (3.2) couples the indicator functions of *all* pixels in the image domain and, thus, represents the key ingredient for the introduction of the *global* co-occurrence prior. Due to the non-differentiability of the L^∞-norm and the non-convex range $\{0,1\}$ of each label indicator function l_i, we relax l to map to the unit intervals $l : \mathcal{S} \to [0,1]^n$ and replace (3.2) by the following convex constraint:

$$l_i(u) \ge u_i(x) \quad \forall x \in \Omega \quad \forall i \in \mathcal{G}. \tag{3.3}$$

The idea of defining a co-occurrence energy is that it should only depend on the label indicator functions l, which capture the global occurrence information:

$$E_C(u) = E_C(l(u)). \tag{3.4}$$

Provided that E_C has the following properties:

1. $E_C(l)$ is convex.
2. $E_C(l)$ is monotonically increasing w.r.t. l i.e.

$$l \preceq \tilde{l} \implies E_C(l) \leq E_C(\tilde{l}), \tag{3.5}$$

it can be easily shown that, replacing (3.2) by (3.3) and minimizing over l, one recovers the optimum of the optimization problem (2.4) with respect to constraint (3.2).

3.2 A Convex Formulation of the Co-occurrence Prior

We will now formulate the continuous *co*-occurrence prior. To this end, the concept of occurrence functions l in (3.2) for single labels is generalized to the occurrence of label *subsets* $L \subseteq \mathcal{G}$ by introducing the label subset indicator function $\delta_L : \mathcal{S} \to \{0, 1\}$:

$$\delta_L(u) = \begin{cases} 1 & \text{if } l_i(u) = 1 \; \forall i \in L, \\ 0 & \text{otherwise.} \end{cases} \tag{3.6}$$

The function δ_L indicates the simultaneous occurrence of *all* labels in the subset L in the image, so that it can be rewritten in the following way:

$$\delta_L(u) = \prod_{i \in L} l_i(u). \tag{3.7}$$

The co-occurrence prior is then defined as the sum over all possible label combinations of $\mathcal{G}$ (the elements of the power set of $\mathcal{G}$) weighted by the associated co-occurrence penalties $C(L) \geq 0$

$$E_C(l(u)) = \sum_{L \in \mathcal{P}(\mathcal{G})} \delta_L(u) \cdot C(L) = \sum_{L \in \mathcal{P}(\mathcal{G})} C(L) \prod_{i \in L} l_i(u). \tag{3.8}$$

Because of the product in (3.7), the term δ_L is not convex in terms of l for $|L| \geq 2$. Thus, a convex relaxation of the product is required to make the energy term convex. We relax the energy (3.8) term-wise for each subset L, i.e. by relaxing each individual addend

$$E_L(u) := C(L) \prod_{i \in L} l_i(u). \tag{3.9}$$

A convex formulation of this kind of energies was given in [23]. Let us briefly recall this approach for the convenience of the reader. The general considered energy is

$$E_0(v) = \sum_{\gamma_1 \in \Lambda_1, \ldots, \gamma_d \in \Lambda_d} c_\gamma \, v^1_{\gamma_1} \cdot \ldots \cdot v^d_{\gamma_d} \tag{3.10}$$

with $d \geq 1$ finite label sets $\Lambda_1, \ldots, \Lambda_d$ and, for each $1 \leq i \leq d$, corresponding indicator variables $(v^i_\gamma)_{\gamma \in \Lambda_i} \in [0,1]$ which satisfy the simplex constraint $\sum_{\gamma \in \Lambda_i} v^i_\gamma = 1$. The costs $c_{\gamma_1,\ldots,\gamma_d} \in \mathbb{R}$ can be arbitrary. The convex relaxation proposed in [23] is

$$E_0^{\text{rel}}(v) = \sup_{q \in \mathcal{Q}} \sum_{\gamma_1 \in \Lambda_1} q^1_{\gamma_1} v^1_{\gamma_1} + \ldots + \sum_{\gamma_d \in \Lambda_d} q^d_{\gamma_d} v^d_{\gamma_d} \tag{3.11}$$

with the convex set

$$\mathcal{Q} = \left\{ (q^i_{\gamma_i})_{1 \leq i \leq d,\, \gamma_i \in \Lambda_i} \,\middle|\, q^i_{\gamma_i} \in \mathbb{R},\ q^1_{\gamma_1} + \ldots + q^d_{\gamma_d} \leq c_\gamma \ \forall \gamma_1 \in \Lambda_1, .., \gamma_d \in \Lambda_d \right\}. \tag{3.12}$$

The convex energy E_0^{rel} is the *tightest possible* relaxation of E_0: In the recent journal version [10] of [23] it is shown that E_0^{rel} is the convex hull of E_0. For instance, this means that E_0^{rel} preserves the minimizers of E_0, i.e. minimizers of E_0 are also minimizers of E_0^{rel}, Furthermore, $E_0^{\text{rel}}(v)$ coincides with $E_0(v)$ for binary v ($v^i_{\gamma_i} \in \{0,1\}$ for all i and $\gamma_i \in \Lambda_i$).

In our case, we have $d = |L|$ and only two labels per factor, i.e. $\Lambda_i = \{0,1\}$ for all $1 \leq i \leq d$. The corresponding indicator variables are $v^i_0 = 1 - l_i$ and $v^i_1 = l_i$ for each i. Finally, the costs c_γ are given by

$$c_\gamma = \begin{cases} C(L), & \text{if } \gamma_1 = \ldots = \gamma_d = 1 \\ 0, & \text{otherwise.} \end{cases} \tag{3.13}$$

Directly applying (3.11), and writing the dual variables as $q^i_0 =: \varphi^i_L$ and $q^i_1 =: \psi^i_L$ for each $1 \leq i \leq |L|$, we obtain the following convex formulation of E_C:

$$E_C(l(u)) = \sum_{L \in \mathcal{P}(\mathcal{G})} \left(\sup_{(\varphi_L, \psi_L) \in \mathcal{Q}_L} \sum_{i \in L} (1 - l_i(u)) \varphi^i_L + l_i(u) \psi^i_L \right) \tag{3.14}$$

with the convex constraint set

$$\begin{aligned} \mathcal{Q}_L := \Big\{ (\varphi_L, \psi_L) \big| \ \forall z \in \{0,1\}^{|L|} \neq \mathbf{1} : \\ \sum_{i \in L} (1 - z_i) \varphi^i_L + z_i \psi^i_L \leq 0, \quad \sum_{i \in L} \psi^i_L \leq C(L) \Big\}, \end{aligned} \tag{3.15}$$

where $\mathbf{1}$ is a vector consisting of all ones. The terms $(1 - z_i)\varphi^i_L + z_i \psi^i_L$ in (3.15) arise from (3.12) by noting that $q^i_{z_i} = (1 - z_i)\varphi^i_L + z_i \psi^i_L$ for all i and $z_i \in \{0,1\}$.

For $C(L) \to \infty$ in (3.8) for some subset L we obtain *hard constraints* on label configurations, i.e. $\delta_L(u) = 0$. Including this constraint prohibits segmentations containing *all* labels of L: The energy will be infinite if the labels from L occur in the image simultaneously. This corresponds to simply dropping the constraint $\sum_{i \in L} \psi^i_L \leq C(L)$ in the above constraint set (3.15).

We can easily prove that $E(l(u))$ is monotonically increasing, i.e. that the requirement (2.5) is fulfilled. First, it is monotonic for binary l: It this case it is represented by the original formula (3.8). Since the label subset indicator functions $\delta_L(l)$ are clearly

monotonous in l and the costs $C(L)$ are nonnegative, it follows that E_C is monotonous for binary l. This is also referred to as the principle of Occam's razor which states that among competing labelings the one with fewer labels should be favoured energetically. Second, we use the fact that (3.14) is the *convex hull* relaxation of $\delta_L(l)$ to the set of possibly non-binary l's. This and the monotonicity for binary l yields the monotonicy of $E_C(l)$ for general $l \in [0,1]^n$.

It follows that by means of the convex relaxation (3.3) we can recover the constraint (3.2) and thus globally minimize the overall energy (2.4). Since the power set of $\mathcal{G}$ is very large, we follow [15] and approximate the true costs $C(L)$ by taking only sets of two labels into account, with fixed costs which best approximate the original costs. For details see [15]. The resulting co-occurrence energy for sets of only two labels then reads as:

$$E_C(l(u)) = \sum_{1\le i<j\le n} \Bigg(\sup_{\{\varphi_{ij},\psi_{ij}\}\in\mathcal{Q}_{i,j}} (1-l_i(u))\varphi_{ij,1} l_i(u)\psi_{ij,1} + (1-l_j(u))\varphi_{ij,2} + l_j(u)\psi_{ij,2} \Bigg), \tag{3.16}$$

where $\mathcal{Q}_{ij}$ is the convex constraint set of dual variables $\varphi_{ij}, \psi_{ij} \in \mathbb{R}^2$:

$$\mathcal{Q}_{ij} := \Big\{ (\varphi_{ij},\psi_{ij}) \big| \quad \psi_{ij,1} + \psi_{ij,2} \le c_{ij}, \quad \varphi_{ij,1} + \varphi_{ij,2} \le 0,\ \varphi_{ij,1} + \psi_{ij,2} \le 0,\ \psi_{ij,1} + \varphi_{ij,2} \le 0 \Big\}. \tag{3.17}$$

Note that in the integer linear program (ILP) given in [15] the label subset indicator function $\delta_L(u)$ is realized by the following constraint ([15] equation (31)):

$$\delta_L(u) \ge \sum_{i\in L} l_i(u) - |L| + 1. \tag{3.18}$$

In contrast to our framework with the tight convex relaxations (3.11) and (3.16), equation (3.18) would introduces additional trivial solutions after relaxing the ILP to allow $\delta_L(u)$ and $l_i(u)$ to be from $[0,1]$. For instance, (3.18) is always fulfilled for $\delta_L(u) = 0$ and any l such that $\sum_{i\in L} l_i \le |L| - 1$, e.g. $l_i = 1/(|L|-1)$. Therefore, our continuous formulation is a tighter relaxation than the linear relaxation given in [15].

4 Implementation

The overall saddle-point formulation of our optimization problem for label subsets containing only two labels can be summarized as follows:

$$\min_{\substack{u\in\mathcal{S}\\ l\in[0,1]^n}} \sup_{\substack{p\in\tilde{\mathcal{K}}\\ \varphi_{ij},\psi_{ij}\in\mathcal{Q}_{i,j}}} \sum_{i=1}^{n}\int_\Omega \left(\int_{\mathcal{N}_x} p(x,y)(u_i(y)-u_i(x))\, dy\right) dx + \sum_{i=1}^{n}\int_\Omega u_i(x)\varrho_i(x)dx + \sum_{1\le i<j\le n}\left((1-l_i)\varphi_{ij,1}+l_i\psi_{ij,1}+(1-l_j)\varphi_{ij,2}+l_j\psi_{ij,2}\right)$$
$$\text{s.t.}\quad l_i \ge u_i(x) \quad \forall x\in\Omega \quad \forall i\in\mathcal{G}. \tag{4.1}$$

In order to solve the above saddle-point problem we use the first order primal dual algorithm [21,4] which is essentially a gradient descent in the primal variables and a gradient ascent in the dual variables with a subsequent computation of the respective proximity operators and an over-relaxation step for the primal variables. For the time steps we use recent preconditioning techniques introduced in [20]. This way there is no need to compute the Lipschitz constant of the underlying linear operator. The details of the implementation can be found in the appendix.

5 Experiments

The key contribution of this paper is the introduction of a co-occurrence prior into the continuous multi-label framework. In this section we evaluate the proposed continuous formulation of the segmentation problem with the co-occurrence prior on the MSRC database [6]. To preserve comparability to [15], we use their data term, which is based on texture boosting. As explained in Section 4 we approximate the exact co-occurrences by considering only binary label interactions. Overall we obtain $\binom{n}{2}$ possible binary label interactions, i.e. 210 for the 21 labels considered in the MSRC dataset. For each interaction we need to keep track of only eight scalar variables (dual variables and Lagrange multipliers), which is almost negligible compared to the remaining part of the optimization approach. Note that we obtain the co-occurrence weights c_{ij} from the training data as in [15].

Some of the obtained segmentation results of the MSRC dataset are shown in Figure 3. A comparison of the segmentation accuracy of the proposed algorithm to the original formulation by Ladicky et al. on the whole benchmark can be found in Figure 4. The table indicates the numbers for each label separately as well as the average on the whole benchmark. The results show that the segmentations obtained with the proposed continuous formulation are comparable to those obtained by Ladicky et al. in terms of global accuracy (which neglects the number of images per label), and even outperforms them in terms of average accuracy over all benchmark images. Note that due to different training and evaluation sets of images the numbers in table 4 differ slightly from the numbers in [15].

In order to evaluate the tightness of the optimization problem we computed the following relative optimality bound, which is an upper bound of the energy difference between the globally optimal binary solution and the computed binarized solution

$$B(u,\tilde{u}) = \frac{E(\tilde{u}) - E(u)}{E(u)}. \tag{5.1}$$

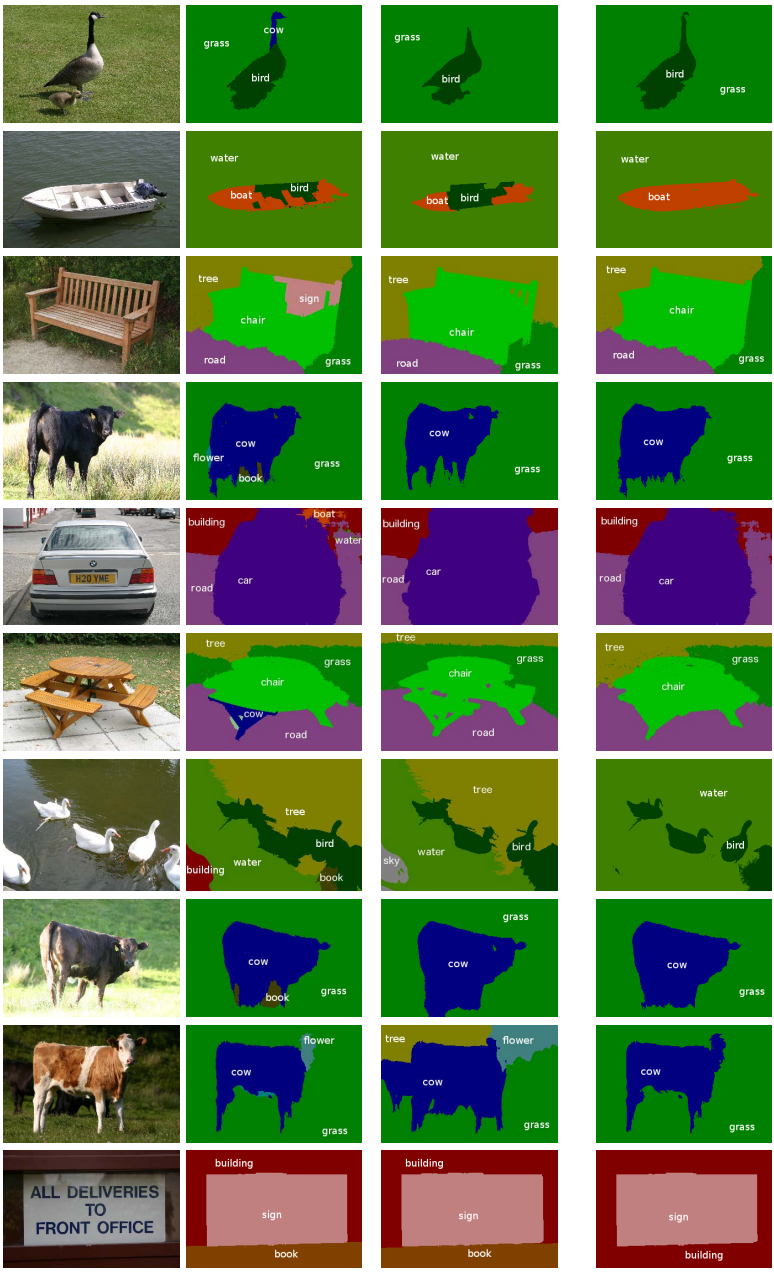


Fig. 3. Co-occurrence prior: Qualitative results on images taken from the MSRC database. The results show (a) the original benchmark image, (b) the segmentation result without co-occurrence prior, (c) the results by Ladicky et al. [15], (d) results from our continuous formulation of the segmentation algorithm incorporating the co-occurrence energy.

Here E is the energy given in (2.4), u the solution of the relaxed saddle point problem (4.1) and $\tilde{u}$ the binarized solution of u used for computing the final labeling, see (2.11). We obtained an average value of $B(u, \tilde{u}) = 0.17\%$ for the optimality bound, which means that our computed binary solutions are very close to the global optimum of the original optimization problem.

	Global	Average	Building	Grass	Tree	Cow	Sheep	Sky	Aeroplane	Water	Face	Car	Bicycle	Flower	Sign	Bird	Book	Chair	Road	Cat	Dog	Body	Boat
dataterm	83.99	77.18	67	97	91	85	86	95	88	81	90	82	94	81	62	42	91	66	86	79	54	72	31
Potts	84.80	77.88	69	97	91	86	86	96	86	82	90	81	93	83	62	42	91	68	86	80	56	72	28
proposed	85.99	**78.95**	72	97	91	87	86	97	86	84	90	83	93	83	64	44	93	73	87	81	59	74	25
[15]	**86.76**	77.78	76	99	90	77	84	99	82	88	88	80	90	90	71	47	94	68	90	73	55	77	15

Fig. 4. Segmentation accuracies of the daterterm, the pure Potts model, our approach using the continuous formulation of the co-occurrence energy, and the results by Ladicky et al. [15]. The scores for each label are defined as $\frac{\textit{True Positives} \cdot 100}{\textit{True Positives} + \textit{False Negatives}}$. While the score of the proposed method is slightly below that of [15] in the global score, it provides a better average performance.

6 Runtime

The proposed algorithm is based on a variational approach which allows for an implementation on graphics hardware. In practice the optimization scheme presented in Section 4 converges within 1000 iterations in terms of the maximal change of two successive iterations. For the experiments we used a NVIDIA Geforce GTX480 GPU and obtained average runtimes per image of 10 seconds for the co-occurrence segmentation. The runtimes are similar to the computation of the pure Potts model, since the overhead of computation compared to the classical multi-labeling problems is marginal as the configuration priors are defined on n scalar indicator variables compared to $\mathcal{O}(|\Omega|)$ variables for the indicator functions u.

7 Conclusion

We proposed a convex framework for continuous multi-label optimization which allows for the integration of semantic scene label information by means of co-occurrence priors. We formulated a variational approach together with a convex relaxation which can be optimized with fast primal-dual schemes. The approach compared favourably with respect to the discrete co-occurrence prior by Ladicky et al. [15] and optimality bounds demonstrate the tightness of our convex relaxations.

A Appendix

In the following we will give some details on the optimization scheme for solving saddle point problem (4.1). We start by introducing the Lagrange multipliers used for handling

the convex constraints. Then we give a list of the update steps performed in our iterative minimization scheme.

In order to impose the label configuration priors, several kinds of constraints have to be implemented. The *simplex constraint* $\sum_i u_i(x) = 1$ in (2.3) can be implemented by introducing Lagrange multipliers $\lambda : \Omega \to \mathbb{R}$. The inequality constraints can also be easily implemented by introducing the following Lagrange multipliers:

Contraints	Lagrange Multipliers
$l_i(u) \geq u_i(x)\ \forall x \in \Omega$	$\alpha_i : \Omega \to \mathbb{R}^+ \quad \forall\ i \in \{1 \cdots n\}$
$\psi_{ij,1} + \psi_{ij,2} \leq c_{ij}$	$\beta_{ij} \in \mathbb{R}^- \quad \forall\ i < j \in \mathcal{G}$
$\varphi_{ij,1} + \varphi_{ij,2} \leq 0$	$\theta_{ij} \in \mathbb{R}^- \quad \forall\ i < j \in \mathcal{G}$
$\varphi_{ij,1} + \psi_{ij,2} \leq 0$	$\eta_{ij} \in \mathbb{R}^- \quad \forall\ i < j \in \mathcal{G}$
$\psi_{ij,1} + \varphi_{ij,2} \leq 0$	$\xi_{ij} \in \mathbb{R}^- \quad \forall\ i < j \in \mathcal{G}$

The update steps are performed point-wise for all $x \in \Omega, i \in \{1, ..., n\}$ and for all label pairs $1 \leq i < j \leq n$ which makes it possible to parallely implement above algorithm on modern graphics cards. Local constraints are tackled by a simple orthogonal projection Π into the respective convex sets which can be performed by simple truncation. Overall the update steps for solving saddle-point problem (4.1) are iterated in the following order:

Updates for dual variables:

$$
\begin{aligned}
p_i^{k+1}(x) &= \Pi_{\bar{\mathcal{K}}}\left(p(x)_i^k + \frac{1}{2}(\nabla_w \bar{u}_i^k(x))\right)\\
\lambda^{k+1}(x) &= \lambda^k(x) + \frac{1}{n}\left(\sum_{i=1}^{n} \bar{u}_i^k(x) - 1\right)\\
\alpha_i^{k+1}(x) &= \Pi_{\mathbb{R}^+}\left(\alpha_i^k(x) + \frac{1}{2}(\bar{u}_i^k(x) - \bar{l}_i^k)\right)\\
\varphi_{ij}^{k+1} &= \varphi_{ij}^k + \frac{1}{3}\begin{pmatrix} 1 - \bar{l}_i^k + \bar{\theta}_{ij}^k + \bar{\eta}_{ij}^k \\ 1 - \bar{l}_j^k + \bar{\theta}_{ij}^k + \bar{\xi}_{ij}^k \end{pmatrix}\\
\psi_{ij}^{k+1} &= \psi_{ij}^k + \frac{1}{3}\begin{pmatrix} \bar{l}_i^k + \bar{\beta}_{ij}^k + \bar{\xi}_{ij}^k \\ \bar{l}_j^k + \bar{\beta}_{ij}^k + \bar{\eta}_{ij}^k \end{pmatrix}
\end{aligned}
$$

Updates for primal variables:

$$
u_i^{k+1}(x) = \Pi_{[0,1]}\left(u_i^k(x) - \frac{1}{2 + |\mathcal{N}(x)|}\left(\varrho_i(x) + \alpha_i^k(x) + \lambda^k(x) - \mathrm{div}_w\, p_i^k(x)\right)\right)
$$

$$
\begin{aligned}
l_i^{k+1} &= l_i^k - \frac{1}{|\Omega|}\left(\int_\Omega \alpha_i^k(x)\, dx\right) \\
\beta_{ij}^{k+1} &= \Pi_{\mathbb{R}^-}\left(\beta_{ij}^k - \frac{1}{2}(\psi_{ij,1}^k + \psi_{ij,2}^k - C_{ij})\right) \\
\theta_{ij}^{k+1} &= \Pi_{\mathbb{R}^-}\left(\theta_{ij}^k - \frac{1}{2}(\varphi_{ij,1}^k + \varphi_{ij,2}^k)\right) \\
\eta_{ij}^{k+1} &= \Pi_{\mathbb{R}^-}\left(\eta_{ij}^k - \frac{1}{2}(\varphi_{ij,1}^k + \psi_{ij,2}^k)\right) \\
\xi_{ij}^{k+1} &= \Pi_{\mathbb{R}^-}\left(\xi_{ij}^k - \frac{1}{2}(\psi_{ij,1}^k + \varphi_{ij,2}^k)\right)
\end{aligned}
$$

Extrapolation steps:

$$
\begin{aligned}
\bar{u}_i(x)^{k+1} &= 2u_i(x)^{k+1} - u_i(x)^k \\
\bar{l}_i^{k+1} &= 2l_i^{k+1} - l_i^k \\
\bar{\beta}_{ij}^{k+1} &= 2\beta_{ij}^{k+1} - \beta_{ij}^k \\
\bar{\theta}_{ij}^{k+1} &= 2\theta_{ij}^{k+1} - \theta_{ij}^k \\
\bar{\eta}_{ij}^{k+1} &= 2\eta_{ij}^{k+1} - \eta_{ij}^k \\
\bar{\xi}_{ij}^{k+1} &= 2\xi_{ij}^{k+1} - \xi_{ij}^k
\end{aligned}
$$

Note that for the differential operators we use a non-local version of the gradient ∇_w and its respective adjoint operator $-\operatorname{div}_w$ which are defined on a neighbourhood of x denoted by $\mathcal{N}(x)$. For details see [8], and for a complete introduction to non-local operators and their applications in computer vision, we refer to [24].

References

1. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, 2001 (2001)
2. Chambolle, A., Cremers, D., Pock, T.: A convex approach for computing minimal partitions. Tech. rep. TR-2008-05, University of Bonn (2008)
3. Chambolle, A., Cremers, D., Pock, T.: A convex approach to minimal partitions. SIAM Journal on Imaging Sciences 5(4), 1113–1158 (2012)
4. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. JMIV 40(1), 120–145 (2011)
5. Delong, A., Gorelick, L., Veksler, O., Boykov, Y.: Minimizing energies with hierarchical costs. International Journal of Computer Vision 100(1), 38–58 (2012)
6. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: Pascal (VOC2010) Results
7. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. Int. J. Comput. Vision 59(2), 167–181 (2004)
8. Gilboa, G., Osher, S.: Nonlocal operators with applications to image processing. Multiscale Modeling & Simulation 7(3), 1005–1028 (2008)

9. Goldluecke, B., Cremers, D.: Convex relaxation for multilabel problems with product label spaces. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 225–238. Springer, Heidelberg (2010)
10. Goldluecke, B., Strekalovskiy, E., Cremers, D.: Tight convex relaxations for vector-valued labeling. SIAM Journal on Imaging Sciences (2013)
11. Klodt, M., Schoenemann, T., Kolev, K., Schikora, M., Cremers, D.: An experimental comparison of discrete and continuous shape optimization methods. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 332–345. Springer, Heidelberg (2008)
12. Kohli, P., Ladicky, L., Torr, P.H.S.: Robust higher order potentials for enforcing label consistency. International Journal of Computer Vision 82(3), 302–324 (2009)
13. Komodakis, N., Paragios, N., Tziritas, G.: Mrf optimization via dual decomposition: Message-passing revisited. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, pp. 1–8 (2007)
14. Komodakis, N., Tziritas, G.: Approximate labeling via graph cuts based on linear programming. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(8), 1436–1453 (2007)
15. Ladický, L., Russell, C., Kohli, P., Torr, P.: Inference methods for crfs with co-occurrence statistics. International Journal of Computer Vision 103(2), 213–225 (2013)
16. Leclerc, Y.G.: Region growing using the MDL principle. In: Proc. DARPA Image Underst. Workshop, April 6-8, pp. 720–726 (1990)
17. Lellmann, J., Kappes, J., Yuan, J., Becker, F., Schnörr, C.: Convex multi-class image labeling by simplex-constrained total variation. In: Tai, X.-C., Mørken, K., Lysaker, M., Lie, K.-A. (eds.) SSVM 2009. LNCS, vol. 5567, pp. 150–162. Springer, Heidelberg (2009)
18. Malik, J., Belongie, S., Leung, T., Shi, J.: Contour and texture analysis for image segmentation. Int. J. Comput. Vision 43(1), 7–27 (2001)
19. Nieuwenhuis, C., Toeppe, E., Cremers, D.: A survey and comparison of discrete and continuous multilabel segmentation approaches. International Journal of Computer Vision (2013)
20. Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In: ICCV (2011)
21. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An algorithm for minimizing the piecewise smooth mumford-shah functional. In: IEEE International Conference on Computer Vision (ICCV), Kyoto, Japan (2009)
22. Russell, B.C., Freeman, W.T., Efros, A.A., Sivic, J., Zisserman, A.: Using multiple segmentations to discover objects and their extent in image collections. In: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2006, vol. 2, pp. 1605–1614. IEEE Computer Society, Washington, DC (2006)
23. Strekalovskiy, E., Goldluecke, B., Cremers, D.: Tight convex relaxations for vector-valued labeling problems. In: ICCV (2011)
24. Werlberger, M., Unger, M., Pock, T., Bischof, H.: Efficient Minimization of the Non-Local Potts Model. In: ICSSVMCV (2011)
25. Yuan, J., Boykov, Y.: Tv-based multi-label image segmentation with label cost prior. In: BMVC, pp. 1–12 (2010)
26. Zach, C., Gallup, D., Frahm, J.-M., Niethammer, M.: Fast global labeling for real-time stereo using multiple plane sweeps. In: Vision, Modeling and Visualization (October 2008)
27. Zhu, S.C., Yuille, A.: Region competition: Unifying snakes, region growing, and bayes/mdl for multi-band image segmentation. PAMI 18, 884–900 (1996)

Convex Relaxations for a Generalized Chan-Vese Model

Egil Bae[1], Jan Lellmann[2], and Xue-Cheng Tai[3]

[1] Department of Mathematics, University of California, Los Angeles, USA
ebae@math.ucla.edu
[2] Department of Applied Mathematics and Theoretical Physics (DAMTP), University of Cambridge, UK
J.Lellmann@damtp.cam.ac.uk
[3] Department of Mathematics, University of Bergen, Norway
tai@math.uib.no

Abstract. We revisit the Chan-Vese model of image segmentation with a focus on the encoding with several integer-valued labeling functions. We relate several representations with varying amount of complexity and demonstrate the connection to recent relaxations for product sets and to dual maxflow-based formulations. For some special cases, it can be shown that it is possible to guarantee binary minimizers. While this is not true in general, we show how to derive a convex approximation of the combinatorial problem for more than 4 phases. We also provide a method to avoid overcounting of boundaries in the original Chan-Vese model without departing from the efficient product-set representation. Finally, we derive an algorithm to solve the associated discretized problem, and demonstrate that it allows to obtain good approximations for the segmentation problem with various number of regions.

1 Introduction

In this paper we focus image segmentation formulated as a variational problem. The general problem we are interested in, is to find a partition $\{\Omega_i\}_{i=1}^n$ of the image domain Ω, by minimizing an energy functional of the form

$$\min_{\{\Omega_i\}_{i=1}^n} \sum_{i=1}^n \int_{\Omega_i} f_i(I^0(x))\,dx + \alpha R(\{\partial\Omega_i\}_{i=1}^n)$$

$$\text{s.t.} \quad \bigcup_{i=1}^n \Omega_i = \Omega, \quad \bigcap_{i=1}^n \Omega_i = \emptyset. \tag{1}$$

Here $R(\{\partial\Omega_i\}_{i=1}^n)$ is a regularization term, and I^0 is a given input image. A popular choice for the regularizer is the Potts regularizer, which measures the total length of the region boundaries,

$$R(\{\partial\Omega_i\}_{i=1}^n) = \frac{1}{2}\sum_{i=1}^n |\partial\Omega_i|. \tag{2}$$

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 223–236, 2013.
© Springer-Verlag Berlin Heidelberg 2013

In order to build up algorithms for solving (1) numerically, one needs some representation of the regions in terms of functions instead of subsets. Over the last 30 years, several such representations have been proposed, including the level set method [16] and phase field method. Recently, there has been a particular interest in piecewise constant representations [12], where each regions is uniquely associated with a value of some binary or integer constrained functions. A reason for the popularity of this approach, is that very good convex relaxations often can be derived by relaxing the integrality constraints of the functions [6,18,11,24,17,1,3]. There are in particular three classical ways of representing multiple regions $\{\Omega_i\}_{i=1}^n$ in terms of piecewise constant functions

1. *Integer-valued* labeling function [12,13]: $\phi : \Omega \mapsto \{1, ..., n\}$ such that $\phi(x) = i$ if $x \in \Omega_i$, $i = 1, ..., n$.

2. *Simplex-constrained* vector function [11,24]: $v : \Omega \mapsto \Delta^n = \{v \in \mathbb{R}^n : \sum_{i=1}^n v^i = 1,\ v^i \in \{0, 1\},\ i = 1, ..., n\}$ such that

$$v^i(x) := \begin{cases} 1, x \in \Omega_i \\ 0, x \notin \Omega_i \end{cases}, \quad i = 1, \ldots, n.$$

   A related variant with a different parametrization of the unit simplex was proposed in [17].
3. $m = \log_2(n)$ overlapping *binary functions*: $\phi^1, ..., \phi^m : \Omega \mapsto \{0, 1\}$ such that $x \in \Omega_i$ iff $\phi^1(x)...\phi^m(x)$ is the binary representation of integer i. This representation was pioneered in a level set framework in [21] and the resulting optimization problem is often called the Chan-Vese model. The use of binary functions for the multiphase CV model in the continuous setting was done in [13,14,6]. It was observed in [6] that it is possible to convex relax these binary models.

For some special problems, convex relaxations exists that have proven to be exact, meaning that global minimizers of the original non-convex problems can be obtained from minimizers of the convex relaxations. This includes in particular problems with two regions [6], the labeling function representation in case of a regularization term which is convex in ϕ [18] and the Chan-Vese model with four regions under some conditions on the data term [3].

These relaxations have been motivated by the theory of discrete optimization, where it is known that the corresponding discrete optimization problems defined over a discrete image domain are submodular and can be solved efficiently by graph based optimization algorithm such as max-flow/min-cut. However, for the majority of variational segmentation problem of interest, including (1) with Potts regularizer (2), the corresponding discrete optimization problems are non-submodular (and actually NP-hard). Convex relaxations have been proposed for such problems that are not guaranteed to provide exact solutions in advance, but can yield good approximations in practice.

This paper aims to give an overview of different representations of the problem (1). In addition to focusing on the three approaches mentioned above, we propose

new representations as combinations of several labeling functions and several simplex constrained functions.

Secondly, we derive convex relaxations for the problems based on the convex envelope of the data fidelity term. As a special case, we obtain a convex relaxation of the Chan-Vese model with an arbitrary number of regions. Up until now, global optimization algorithms for this model have only been available in case of four regions [2,8,3]. In contrast to other relaxations [5] with more than four regions, ours is the tightest because it is based on the convex envelope of the non-convex data term. Furthermore, the number of unknowns grow as $O(log_2(n))$ instead of $O(n)$ in [5]. A simultaneous work [15] appearing in this conference also derives a convex relaxation of the Chan-Vese model by extending the max-flow model developed in [3].

The proposed relaxations are closely related to the recent work [20], which derived a convex relaxation for vector valued labeling problems. In contrast to [20], our original problems are not vector valued. Instead, we use a vector representation to significantly reduce the number of unknowns. We also derive the relaxation without on an initial simplex constrained conversion used in [20].

We derive a set of conditions which can be checked in advance to guarantee that a global minimizer is obtained from the relaxations. While in practice these seem to hold only in very rare cases, our approach can at least produce good approximations. the best approximations that are theoretically possible. A convex relaxation for Potts regularizer is a also derived by building on the vector relaxation and the work [17,3]. Efficient algorithms are proposed for all the problems based on Augmented Lagrangian methods.

2 Different Representations of the Partition as Vector-Valued Functions

2.1 2^m Regions with m Binary Functions

We start by focusing on the representation 3 given in the introduction, which is the binary version of the level set framework [21]. For each $i \in \{1, ..., n\}$, let $a_i^1 a_i^2 ... a_i^m$ denote the binary representation of i or any permutation of the digits in the binary representation. Define $w_0(s) := s$ and $w_1(s) := 1 - s$ and introduce m binary functions $\phi^1, ..., \phi^m : \Omega \mapsto \{0, 1\}$.

The general model in [21] could then be written in terms of polynomials in $\{\phi^i\}_{i=1}^m$ as

$$\min_{\{\phi^i\}_{i=1}^m} \int_\Omega \sum_{i=1}^n \prod_{k=1}^m w_{a_i^k}(\phi^k) f_i \, dx + \alpha \sum_{k=1}^m \int_\Omega |\nabla \phi^k| \tag{3}$$

subject to

$$\phi^i \in \mathbb{B} := \{\phi \in BV(\Omega) : \phi(x) \in \{0, 1\} \text{ for a.e. } x \in \Omega\}, \, i = 1, ..., m. \tag{4}$$

It is also possible to represent a number of n regions which is not a power of 2 by choosing m is the small integer such that $n < 2^m$ and setting $f_i = \infty$ for the $2^m - n$ number of excess indices i.

2.2 Product Space of Several Labeling Functions

A natural extension of the model in the previous section is to represent the image partition in terms of an *integer-valued* labeling function $\phi : \Omega \mapsto \{0, ..., n-1\}$ with the understanding that $\phi(x) = i$ if and only if $x \in \Omega_i$ for $i = 1, ..., n$. It has recently been established that such problems can be solved exactly if the regularizer is the total variation of the labeling function $R(\phi) = \int_\Omega |\nabla \phi|$ [9,18].

This can also be combined with the approach from the previous section, by representing the partition with several labeling functions $(\phi^1, ..., \phi^m)$ taking several integer values, as proposed in a level set framework in [7]. Denote by $\mathcal{L}_i = \{0, ..., N_i - 1\}$ the set of feasible values for ϕ^i, where N_i is the number of feasible integer values and define the vector function

$$\phi = (\phi^1, ..., \phi^m) : \Omega \to \mathcal{L}_1 \times ... \times \mathcal{L}_m \subset \mathbb{Z}^m \tag{5}$$

For each $x \in \Omega$, $\phi(x)$ can take $\prod_{i=1}^m N_i$ different values and thus be used to represent $n = \prod_{i=1}^m N_i$ regions. Let $\{a^i\}_{i=1}^n$ denote an enumeration of all feasible values for ϕ, i.e., for each $i = 1, ..., n$,

$$a^i = \begin{pmatrix} a_1^i \cdots a_m^i \end{pmatrix}^\top \tag{6}$$

such that $a_k^i \in \mathcal{L}_k$ for $k = 1, ..., m$. Region Ω_i can be encoded as

$$\Omega_i = \{x \in \Omega \text{ s.t. } \phi(x) = a^i\}, \quad i = 1, ..., n \tag{7}$$

However, this encoding is not unique, as the enumeration $\{a^i\}_{i=1}^n$ can be reordered in any way. There are $n!$ such reordeings and they can be formulated generally using a permutation matrix P as follows

$$[a^1 ... a^n] \leftarrow [a^1 ... a^n] \cdot P \tag{8}$$

The choice of permutation may have an effect on the quality of the relaxation. For instance [3] showed that a particular permutation of the four region model was crucial for producing exact global minimizers of the original problem. By introducing a function $f : \mathcal{L}_1 \times ... \times \mathcal{L}_m \times \Omega \mapsto \mathbb{R}$,

$$f(\phi(x), x) = \begin{cases} f_i(x), & \text{if } \phi(x) = a^i, \ \ i = 1, ..., n \\ +\infty, & \text{otherwise,} \end{cases} \tag{9}$$

we can define the regularized energy in terms of ϕ,

$$\min_\phi \int_\Omega f(\phi(x), x)\, dx + \alpha \sum_{i=1}^m \int_\Omega |\nabla \phi^i| \tag{10}$$

In case $N_1 = ... = N_m = 2$, the model (10) reduces to the Chan-Vese model (3). Note that due to the separable form of the regularizer, some boundaries will be counted more than once.

2.3 Product Space of Several Simplex Constrained Functions

A third way to represent the regions is in terms of several simplex constrained vector functions: let $v = (v^1, ..., v^m) : \Omega \mapsto \mathbb{R}^{N_1} \times ... \times \mathbb{R}^{N_m}$ be a set of unit vector functions which satisfy

$$\sum_{k=1}^{N_i} v_k^i(x) = 1, \quad v_k^i(x) \in \{0, 1\}, \quad k = 1, ..., N_i \; \forall x \in \Omega \tag{11}$$

The function v with the above constraint can represent $n = \prod_{i=1}^{m} N_i$ regions. For every $k_1 \in \{1, ..., N_1\}, ..., k_m \in \{1, ..., N_m\}$, let $i(k_1, ..., k_m) = k_1 + \sum_{j=2}^{m}(\prod_{i=1}^{j} N_i)k_j$ be the corresponding index, then each region can be described in terms of v by

$$\Omega_{i(k_1,...,k_m)} = \{x \in \Omega : v_{k_1}^1(x) = ... = v_{k_m}^m(x) = 1\} \tag{12}$$

In order to encode the data term we define

$$f(v(x), x) = \begin{cases} f_{i(k_1,...,k_m)}(x), & \text{if } v^1(x) = e_{k_1}, ..., v^m(x) = e_{k_m}, \\ +\infty, & \text{otherwise.} \end{cases} \tag{13}$$

The general segmentation model can then be formulated as

$$\min_{v \in B} \int_\Omega f(v(x), x) + \alpha \sum_{i=1}^{m} \sum_{k_i=1}^{N_i} \int_\Omega |\nabla v_{k_i}^i|. \tag{14}$$

An advantage of this representation compared to (10) is that the regularization term of (14) more closely resembles the Potts regularization term (2), and in fact exactly represents it for boundaries where only one of the v^i changes. For instance, if $m = 2$, the boundaries will be counted at most twice, with the majority being counted once.

3 Convex Relaxations Based on the Convex Envelope

In this section, we derive convex approximations for the models introduced in Sect. 2 based on their convex envelopes, which are defined as the largest convex function majorized by a given function and can under very general conditions be computed by computing the Legendre-Fenchel biconjugate [19].

This process ensures that the convexified energy is close to the original function. In some cases, minimizers of the original problem are also minimizers of the convex envelope. Conditions which guarantee this property in advance are derived in Sect. 4. In [20], a convex relaxation was proposed for vector valued labeling problems of the form (10) with arbitrary data terms. Our work is an adaptation of [20] with a few distinctions. The paper [20] focused on integer-constrained vector labeling, but first converted the problem to a simplex-constrained formulation and derived the convex relaxation based on this formulation. In contrast, we derive the convex envelope directly based on the integer labeling formulation, which leads to a new convex problem with fewer unknowns and a novel integer thresholding step.

3.1 Product Space of Several Labeling Functions

The energy functional (10) is composed of a sum of a non-convex data term and a convex regularizer. We ignore the regularization term in the following derivations, since it is already convex. Since deriving the full convex envelope is in general intractable, in the following we focus on deriving the convex envelope *pointwise* at each $x \in \Omega$, for f defined as in (9):

$$\begin{aligned} f^*(p(x),x) &= \sup_{u(x)\in\mathbb{R}^m} \big\{ \big(\sum_{i=1}^m p_i(x)u^i(x) \big) - f(u(x),x) \big\} \\ &= \max_{u(x)\in\{a_i\}_{i=1}^n} \big\{ \big(\sum_{i=1}^m p_i(x)u^i(x) \big) - f(u(x),x) \big\} \end{aligned}$$

The biconjugate is

$$\begin{aligned} f^{**}(\phi(x),x) &= \sup_{p(x)\in\mathbb{R}^m} \{ \sum_{i=1}^m \phi^i(x)p_i(x) - f^*(p(x),x) \} \\ &= \sup_{p(x)\in\mathbb{R}^m} \{ \sum_{i=1}^m \phi^i p_i(x) + \min_{u(x)\in\{a_i\}_{i=1}^n} \big\{ f(u(x),x) - \sum_{i=1}^m p_i(x)u^i(x) \big\} \} \\ &= \sup_{p(x)\in\mathbb{R}^m, p_0(x)\in\mathbb{R}} \sum_{i=1}^m \phi^i(x)p_i(x) + p_0(x) \qquad (15) \\ &\quad \text{s.t. } p_0(x) \le -\sum_{i=1}^n u^i(x)p_i(x) + f(u(x),x)\,, \ \ \forall u(x) \in \{a_i\}_{i=1}^n. \end{aligned}$$

Note that $f^{**}(\phi(x),x) = +\infty$ if ϕ is not in the convex hull of the points, $\mathrm{conv}\{a^1,...,a^n\}$: if not, the function $f^{**} + \delta_{\mathrm{conv}\{a^1,...,a^n\}}$ is strictly greater that f^{**} but still majorized by f, which contradicts the maximality of f^{**} as the convex hull of f. The overall problem with regularization we wish to solve is therefore

$$\min_{\phi\in BV(\Omega)} \sup_{p\in L^2(\Omega)^{m+1}} \int_\Omega p_0(x) + \sum_{i=1}^m \phi^i(x)p_i(x)\,dx + \alpha \sum_{i=1}^m \int_\Omega |\nabla \phi^i| \qquad (16)$$

$$\text{s.t. } p_0(x) + \sum_{i=1}^n u^i(x)p_i(x) \le f(u(x),x)\,, \ \ \forall u(x) \in \{a^i\}_{i=1}^n, \quad \forall x \in \Omega.$$

We want an integral solution $\phi^1,...,\phi^m$ to the minimization problem (16). However, it cannot in general be expected that the solution is integral at every point. Therefore, we apply a thresholding procedure with parameter $t \in (0,1]$ as follows

$$(\phi^i)^t(x) = \begin{cases} \lfloor \phi^i \rfloor, & \text{if } \phi^i(x) < \lfloor \phi^i(x) \rfloor + t \\ \lceil \phi^i \rceil, & \text{else,} \end{cases} \quad i = 1,...,m \qquad (17)$$

where $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ are the floor and ceiling functions respectively. If the constraint set is binary, i.e. $N_1 = ... = N_m = 2$, this corresponds to the standard thresholding procedure in [6].

3.2 Product Space of Several Simplex Constrained Functions

In the same way, one can derive the envelope relaxation of the model (14). We start by defining the vector $p = (p^1, ..., p^m)$ such that $p^k \in \mathbb{R}^{N_k}$. Computing the envelope pointwise for each $x \in \Omega$ and ignoring the regularization term, one ends up with

$$\sup_{\{\{p_i^k(x)\}_{i=1}^{N_k}\}_{k=1}^m} \sum_{k=1}^{m} \sum_{i=1}^{N_k} p_i^k(x) v_i^k(x) \tag{18}$$

subject to

$$\sum_{i=1}^{N_k} p_{a_i^k}^k(x) \leq f_i(x), \quad k = 1, .., m, \tag{19}$$

$$\sum_{i=1}^{N_k} v_i^k(x) = 1, \quad k = k, ..., m, \tag{20}$$

$$v_i^k(x) \in [0, 1], \quad i = 1, ..., N_k, \;\; k = 1, ..., m. \tag{21}$$

which is also the pointwise relaxation of the data term for vector labeling problems that was proposed in [20]. The objective function (18) with constraints (19)-(21) also arises if one computes the standard LP-relaxation (review: [22]) of the combinatorial data term (14). The model (14) can therefore be relaxed as

$$\sup_{\{\{p_i^k(x)\}_{i=1}^{N_k}\}_{k=1}^m} \sum_{k=1}^{m} \sum_{i=1}^{N_k} \int_\Omega p_i^k(x) v_i^k(x)\, dx + \alpha \sum_{i=1}^{m} \sum_{k_i=1}^{N_i} \int_\Omega |\nabla v_{k_i}^i|\, dx \tag{22}$$

subject to (19)–(21) for all $x \in \Omega$. Although we cannot expect a binary solution in general, an approximate partition can be obtained through the thresholding step

$$v_k^i(x) \leftarrow \begin{cases} 1, \text{ if } i = \arg\max_j v_k^j(x) \\ 0, \text{ otherwise.} \end{cases}, \quad i = 1, \ldots, n$$

Note that the Chan-Vese model can be obtained as a special case by substituting $v_1^k = 1 - \phi^k$ and $v_2^k = \phi^k$. The approach of Bae-Yuan-Tai [1] is also a special case of this general model where global minimization can be guaranteed under some moderate conditions.

4 Special Cases which Guarantee Global Minimizers

We show that under some conditions on the data term of (3), exact solutions will be produced by the relaxation (16). In particular, this is true if the data term is submodular. Observe that the energy in (3) pointwise consists of interactions between m binary variables. An energy function of two binary variables $E(\phi^1, \phi^2)$ is said to be *submodular* if [4,10]

$$E(1, 0) + E(0, 1) \leq E(1, 1) + E(0, 0). \tag{23}$$

A higher-order function $E(x^1, x^2, x^3)$ of 3 binary variables is said to be submodular if the projections onto all functions of two binary variables are submodular [4,10], i.e., if

$$E(b, x^2, x^3), \quad E(x^1, b, x^3), \quad E(x^1, x^2, b) \tag{24}$$

are submodular for every $b \in \{0, 1\}$.

In [3], a different relaxation was proposed for the model (3) with four regions, which was shown to be exact in case of a submodular data term. Specifically, it was shown (3) could be reformulated as

$$\min_{\phi^1, \phi^2 \in [0,1]} \alpha \int_\Omega |\nabla \phi^1| + \alpha \int_\Omega |\nabla \phi^2| \tag{25}$$

$$+ \int_\Omega (1 - \phi^1(x))C(x) + (1 - \phi^2(x))D(x) + \phi^1(x)A(x) + \phi^2(x)B(x)\, dx$$

$$+ \int_\Omega \max\{\phi^1(x) - \phi^2(x), 0\}E(x) - \min\{\phi^1(x) - \phi^2(x), 0\}F(x)\, dx$$

where $A, ..., F$ satisfy the linear system of equations

$$\begin{cases} A(x) + B(x) & = f_2(x) + \sigma(x) \\ C(x) + D(x) & = f_3(x) + \sigma(x) \\ A(x) + E(x) + D(x) & = f_1(x) + \sigma(x) \\ B(x) + F(x) + C(x) & = f_4(x) + \sigma(x) \end{cases}. \tag{26}$$

where σ is an arbitrary function. Consequently it was proved that an exact global minimizer could be obtained by thresholding any solution of the convex problem.

In the following, we show that the convex envelope relaxation (16) with $N_1 = N_2 = 2$ is equivalent to the formulation (25) in case the energy is submodular. In [3] it was observed that one possible solution of (26) is

$$\begin{aligned} &A = \max\{f_2 - f_4, 0\}, B = \max\{f_4 - f_3, 0\}, C = \max\{f_4 - f_2, 0\}, \\ &D = \max\{f_3 - f_4, 0\}, E = f_1 + f_4 - f_2 - f_3. \end{aligned} \tag{27}$$

Substituting this into the integrand of the data term of (25) yields

$$\phi^1(f_2 - f_4) + \phi^2(f_4 - f_3) + (f_1 + f_4 - f_2 - f_3)\max\{\phi^1 - \phi^2, 0\}. \tag{28}$$

for the data term. On the other hand, problem (15) is a linear program with in this case effectively 3 unknowns (note that here we use $a^0 = (1, 0), a^1 = (1, 1), a^2 = (0, 0), a^3 = (0, 1)$, so that the vector $(-1, -1, 1, 1)$ is in the null-space of the constraint matrix for p). Any solutions must be on corners of the constraint set, which are characterized by 3 of the constraints holding with equality, and can be computed as

$$p \in \{(f_1 - f_3, f_2 - f_1), (f_2 - f_4, f_4 - f_3), \tag{29}$$

$$(f_2 - f_4, f_2 - f_1), (f_1 - f_3, f_4 - f_3)\}. \tag{30}$$

Checking these possible solutions against the remaining constraint yields that the first two solutions are feasible and the last two solutions are infeasible iff $f_4 + f_1 - f_2 - f_3 \geqslant 0$, which is exactly the submodularity condition. Substituting these two solutions into (15) yields a compact expression for f^{**},

$$\max\{\phi^1(f_1 - f_3) + \phi^2(f_2 - f_1) + f_3, \phi^1(f_2 - f_4) + \phi^2(f_4 - f_3) + f_3\}, \quad (31)$$

which can be shown to be equivalent to (28), and consequently established the equivalence of (16) and (25) for submodular data.

It can also be shown that if the 8 region model is submodular, the relaxation satisfies the coarea formula and can therefore be thresholded while preserving the energy. More details will be provided in an extended version of this paper.

5 Potts Regularization Term

The model (1) does not correspond exactly to the Potts regularizer (2) because some of the region boundaries are counted multiple times. In the following, we present a way to derive a convex relaxation of the Potts regularization term by using the above representation as binary functions by introducing additional constraints on the dual variables.

This relaxation is inspired by the work of [17], which derived a convex relaxation of Potts model based on the labeling function representation of the partition. An important distinction is that the number of dual constraints in our relaxation grow as $O(\log_2(n)^2)$, whereas the number of constraints in [17] grow as $O(n^2)$. The relaxation was proposed for the case of 4 regions in [3], our contribution here is the generalization to 2^m regions.

The model (16) can be written with dual variables as

$$\min_{\phi \in BV(\Omega)} \sup_{p \in L^2(\Omega)^{m+1}, q^i \in C_\alpha} \int_\Omega p_0(x) + \sum_{i=1}^{m} \phi^i(x) p_i(x) + \phi^i(x) \operatorname{div} q^i(x)\, dx \quad (32)$$

$$\text{s.t. } p_0(x) + \sum_{i=1}^{n} u^i(x) p_i(x) \leq f(u(x), x)\,, \quad \forall u(x) \in \{a_i\}_{i=1}^n, \quad \forall x \in \Omega$$

where the constraint set C_α is defined as

$$C_\alpha = \{q \in C^\infty(\Omega)^{\dim(\Omega)} \,:\, |q|_\infty \leq \alpha\}. \quad (33)$$

where $|q|_\infty = \sup_{x \in \Omega} |q(x)|_2$. We are now interested in the binary case, i.e. $N_1 = ... = N_m = 2$. The convex relaxation for Potts model consists of the optimization problem (32) with the extra dual constraint set

$$(q^1, ..., q^m) \in C^P \quad (34)$$

$$= \left\{ \{q^i\}_{i=1}^m \in C_\alpha \;:\; |q^i - q^j|_\infty \leq \alpha,\ |q^i + q^j|_\infty \leq \alpha\,;\ \forall\, i < j \in \{1, ..., m\} \right\}.$$

If the functions $\phi^1, ..., \phi^m$ are binary, one can easily check that the last term of (32) corresponds to the Potts regularizer (2). The constraint set (34) contains $(\log_2(n))^2$ inequalities.

6 Algorithms

We derive algorithms for the problems (16) and (22). In this section, $(a^i)^k$ denotes component i of vector a at iteration k, and a^k denotes vector a at iteration k.

Define the set

$$C_p^1(x) = \{p(x) \in \mathbb{R}^m, p_0(x) \in \mathbb{R} : \\ p_0(x) + \sum_{j=1}^{m} u_j(x) p_j(x) \leq f(u(x), x)\,, \quad \forall u(x) \in \{a_i\}_{i=1}^n\}$$

Applying the dual formulation of total variation (32) and rearranging the terms, the model (16) can be reformulated as

$$\min_{\phi \in BV(\Omega)} \sup_{p \in L^2(\Omega)^{m+1}, q \in C_\alpha} \int_\Omega p_0(x) + \sum_{i=1}^{m} \phi^i(x)(p_i(x) + \operatorname{div} q^i(x))\, dx$$
$$\text{s.t. } p_0(x) + \sum_{i=1}^{n} u^i(x) p_i(x) \leq f(u(x), x)\,, \quad \forall u(x) \in \{a^i\}_{i=1}^n, \quad \forall x \in \Omega$$

Observe that ϕ^i can be interpreted as an unconstrained Lagrange multiplier for the constraint $p_i + \operatorname{div} q^i = 0$. Consequently, it possible to form the augmented Lagrangian functional

$$L(p, q, \phi) = \int_\Omega p_0 + \sum_{i=1}^{m} \phi^i (p_i + \operatorname{div} q^i)\, dx - \frac{c}{2} \sum_{i=1}^{m} ||p_i + \operatorname{div} q^i||^2 \qquad (35)$$

and solve the problem (16) by the augmented Lagrangian method as follows: initialize the starting points p^0, q^0, ϕ^0, and iterate, for $k = 0, 1, ...,$

$$\begin{aligned} (p)^{k+1} &= \arg\max_{(p) \in C_p^1} L(p, q^k, \phi^k), \\ (q^i)^{k+1} &= \arg\max_{q^i} L(p^{k+1}, q^i, \phi^k), && i = 1, ..., m \\ (\phi^i)^{k+1} &= (\phi^i)^k - c((p^i)^{k+1} + \operatorname{div}(q^i)^{k+1}), && i = 1, ..., m \end{aligned}$$

The first subproblem can be solved approximately as

$$(p)^{k+1} = \Pi_{C_p^1}(p^k + \delta_p \frac{\partial L}{\partial p}(p^k, q^k, \phi^k))$$

where $\frac{\partial L}{\partial p_0}(p^k, q^k, \phi^k) = 1$ and $\frac{\partial L}{\partial p_i}(p^k, q^k, \phi^k) = {\phi^i}^k - c(p_i^k - \operatorname{div} {q^i}^k)$, $i = 1, ..., m$. The projection $\Pi_{C_p^1}$ onto C_p cannot be computed in closed form, but can be computed iteratively by Dykstra's algorithm. The subproblem involving q can be solved iteratively by computing an ascent step followed by a simple projection onto C_α. In case of a Potts regularizer, we use Dykstra's algorithm to project q onto its constraint set as in [17,3].

Note that (35) resembles a max-flow problem where $p_i + \operatorname{div} q^i = 0$ is the flow conservation constraint. The algorithm above is in the same spirit as recently proposed continuous max-flow algoirthms [23,3] which have demonstrated to be very efficient in practice.

A similar Augmented Lagrangian algorithm can also be derived for (22). To save space, we skip the details here.

7 Experiments

In the experiments we have chosen the data term

$$f_i(x) = |I(x) - c_i|^\beta \quad \forall x \in \Omega, \quad i = 1, ..., n \tag{36}$$

with $\beta = 2$. The optimal parameters c_i, $i = 1, ..., n$ are calculated by iteratively minimizing for the regions and c_i until convergence. In Figure 2(f) they are chosen uniformly between 0 and 1 without subsequent updating. We use the relaxation (16) to find partition into various number of regions. Results with 8 regions, represented by 3 binary functions ($N_1 = N_2 = N_3 = 2$), are depicted in Figure 2 and 4. Results with 6 regions, which was represented by one binary function and one function taking 3 integer values ($N_1 = 2, N_2 = 3$), are shown in Figure 3 (a) and (b). Results with 16 regions in Figure 3 (c), represented by 4 binary functions ($N_1 = ... = N_4 = 2$) . Observe that the solutions $\phi^1, ..., \phi^m$ are binary/integer at most points. In order to produce a fully binary solution we threshold according to (17). To visualize the results, we have plotted $\phi^1, ..., \phi^3$ before thresholding . The results can also be depicted in a single image by the construction $I = c_i$ in Ω_i, $i = 1, ..., n$. We also depict I before thresholding, by using the polynomial in (3) to represent the regions in a soft manner before thresholding.

In the experiments with 8 regions, it is interesting to ask whether the submodularity conditions (24) can be satisfied at every point for a permutation of the labels, which would guarantee that a global minimizer can be obtained after

(a) (b)

Fig. 1. Test images

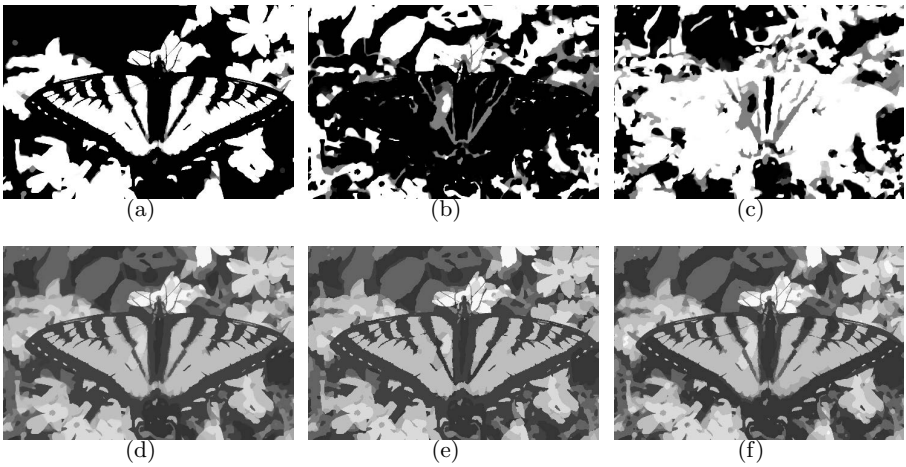


Fig. 2. 8 regions ($N_1 = N_2 = N_3 = 2$), $\alpha = 0, 1$: (a)-(c) $\phi^1, ..., \phi^3$ before threshold, (d) I before threshold, (e) I after threshold, (f) Potts relaxation after threshold

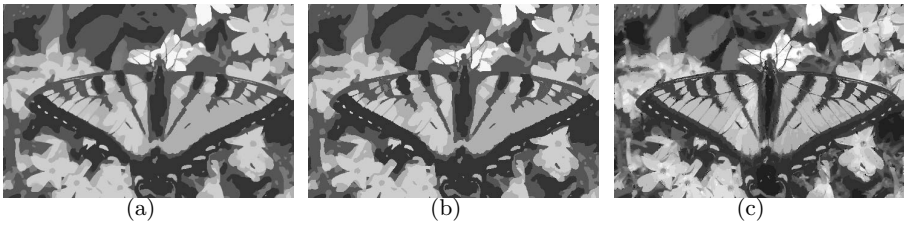


Fig. 3. (a) - (b) 6 regions ($N_1 = 2, N_2 = 3$), $\alpha = 0, 1$, (a) before threshold, (b) after threshold. (c) 16 regions ($N_1 = ... = N_4 = 2$)

Fig. 4. 8 regions, $\alpha = 0.1$. (a) Thresholded solution, (b)-(d) $\phi^1, ..., \phi^3$ before threshold

thresholding. Unfortunately, this was not the case for any of the possible permutations (8) of the representation, in contrast to the 4-region case, where this condition often holds [3].

This means that in most cases it is not possible to guarantee that a global minimizer will be obtained a priori. However the results seem to be good in practice and are in any case the best approximations that can be obtained using a local relaxation, i.e., a relaxation of the integrand.

8 Conclusions

We have summarized and generalized different representations of the regions in variational image segmentation models in terms of vector functions. Convex relaxations have been developed based on the convex envelope and connected to recent relaxations for product sets and to dual maxflow-based formulations. The relaxations contain a significantly lower number of unknowns than there are regions and are the tightest convex approximations that exist for the given set of problems. Efficient algorithms have been developed and experiments have demonstrated that good approximations for the segmentation problems can be obtained in practice.

Acknowledgement. This research has been supported by the Norwegian Research Council eVita project 214889, Award No. KUK-I1-007-43, made by King Abdullah University of Science and Technology (KAUST), EPSRC first grant No. EP/J009539/1, EPSRC/Isaac Newton Trust Small Grant, and Royal Society International Exchange Award No. IE110314.

References

1. Bae, E., Yuan, J., Tai, X.C.: Global minimization for continuous multiphase partitioning problems using a dual approach. International Journal of Computer Vision 92(1), 112–129 (2011)
2. Bae, E., Tai, X.-C.: Efficient global minimization for the multiphase Chan-Vese model of image segmentation. In: Cremers, D., Boykov, Y., Blake, A., Schmidt, F.R. (eds.) EMMCVPR 2009. LNCS, vol. 5681, pp. 28–41. Springer, Heidelberg (2009)
3. Bae, E., Tai, X.-C.: Efficient global minimization methods for image segmentation models with four regions. UCLA cam-report 11-82 (2011)
4. Billionnet, A., Minoux, M.: Maximizing a supermodular pseudo-boolean function: a polynomial algorithm for supermodular cubic functions. Discrete Appl. Math. (1985)
5. Brown, E.S., Chan, T.F., Bresson, X.: A convex relaxation method for a class of vector-valued minimization problems with applications to mumford-shah segmentation. UCLA, Applied Mathematics, CAM-report-10-43 (July 2010)
6. Chan, T.F., Esedoḡlu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM J. Appl. Math. 66(5), 1632–1648 (electronic) (2006)
7. Chung, G., Vese, L.A.: Image segmentation using a multilayer level-set approach. Computing and Visualization in Science 12(6), 267–285 (2009)
8. El-Zehiry, N., Grady, L.: Discrete optimization of the multiphase piecewise constant mumford-shah functional. In: Boykov, Y., Kahl, F., Lempitsky, V., Schmidt, F.R. (eds.) EMMCVPR 2011. LNCS, vol. 6819, pp. 233–246. Springer, Heidelberg (2011)

9. Ishikawa, H.: Exact optimization for Markov random fields with convex priors. IEEE Transactions on Pattern Analysis and Machine Intelligence 25, 1333–1336 (2003)
10. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 26, 65–81 (2004)
11. Lellmann, J., Kappes, J., Yuan, J., Becker, F., Schnörr, C.: Convex multi-class image labeling by simplex-constrained total variation. In: Tai, X.-C., Mørken, K., Lysaker, M., Lie, K.-A. (eds.) SSVM 2009. LNCS, vol. 5567, pp. 150–162. Springer, Heidelberg (2009)
12. Lie, J., Lysaker, M., Tai, X.-C.: A variant of the level set method and applications to image segmentation. Math. Comp. 75(255), 1155–1174 (2006)
13. Lie, J., Lysaker, M., Tai, X.C.: A piecewise constant level set framework. Int. J. Numer. Anal. Model 2(4), 422–438 (2005)
14. Lie, J., Lysaker, M., Tai, X.: A binary level set model and some applications to mumford-shah image segmentation. IEEE Transactions on Image Processing 15(5), 1171–1181 (2006)
15. Liu, J., Tai, X.-C., Leung, S.: A generic convexification and graph cut method for multiphase image segmentation. In: Heyden, A., Kahl, F., Olsson, C., Oskarsson, M., Tai, X.-C., Chairs, P. (eds.) EMMCVPR 2013. LNCS, vol. 8081, pp. 251–265. Springer, Heidelberg (2013)
16. Osher, S., Sethian, J.A.: Fronts propagating with curvature dependent speed: algorithms based on hamilton-jacobi formulations. J. Comput. Phys. 79(1), 12–49 (1988)
17. Pock, T., Chambolle, A., Bischof, H., Cremers, D.: A convex relaxation approach for computing minimal partitions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, Florida (2009)
18. Pock, T., Schoenemann, T., Graber, G., Bischof, H., Cremers, D.: A convex formulation of continuous multi-label problems. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 792–805. Springer, Heidelberg (2008)
19. Rockafellar, R.T., Wets, R.J.-B.: Variational Analysis, 2nd edn. Springer (2004)
20. Strekalovskiy, E., Goldluecke, B., Cremers, D.: Tight convex relaxations for vector-valued labeling problems. In: IEEE International Conference on Computer Vision (ICCV) (2011)
21. Vese, L.A., Chan, T.F.: A new multiphase level set framework for image segmentation via the Mumford and Shahs model. International Journal of Computer Vision 50, 271–293 (2002)
22. Werner, T.: A linear programming approach to max-sum problem: A review. IEEE Trans. Pattern Anal. Mach. Intell. 29(7), 1165–1179 (2007)
23. Yuan, J., Bae, E., Tai, X.C.: A study on continuous max-flow and min-cut approaches. In: IEEE Conference on Computer Vision and Pattern Recognition, USA, San Francisco, pp. 2217–2224 (2010)
24. Zach, C., Gallup, D., Frahm, J.-M., Niethammer, M.: Fast global labeling for real-time stereo using multiple plane sweeps. In: Vision, Modeling and Visualization Workshop (VMV) (2008)

Multiclass Segmentation by Iterated ROF Thresholding

Xiaohao Cai and Gabriele Steidl

Department of Mathematics,
University of Kaiserslautern, Kaiserslautern, Germany
{cai,steidl}@mathematik.uni-kl.de

Abstract. Variational models as the Mumford-Shah model and the active contour model have many applications in image segmentation. In this paper, we propose a new multiclass segmentation model by combining the Rudin-Osher-Fatemi model with an iterative thresholding procedure. We show that our new model for two classes is indeed equivalent to the Chan-Vese model but with an adapted regularization parameter which allows to segment classes with similar gray values. We propose an efficient algorithm and discuss its convergence under certain conditions. Experiments on cartoon, texture and medical images demonstrate that our algorithm is not only fast but provides very good segmentation results in comparison with other state-of-the-art segmentation models in particular for images containing classes of similar gray values.

1 Introduction

Throughout this paper, let $\Omega \subset \mathbb{R}^2$ be a bounded, open set and $f : \bar{\Omega} \to [0,1]$ a given image. In 1989, Mumford and Shah in [23] proposed to solve segmentation problems by minimizing over $(\Gamma, u) \in \Omega \times W_2^1(\Omega \setminus \Gamma)$ the energy

$$E_{\mathrm{MS}}(\Gamma, u) := \mathcal{H}^1(\Gamma) + \mu \int_{\Omega \setminus \Gamma} |\nabla u|^2 dx + \lambda \int_{\Omega} (u - f)^2 dx, \quad \lambda, \mu > 0,$$

where $\mathcal{H}^1$ denotes the $1D$ Hausdorff measure. The functional E_{MS} contains three terms: the regularity term on Γ in terms of its length, the regularity term imposing smoothness of u on areas $\Omega \setminus \Gamma$, and the data fidelity term. Related approaches in a spatially discrete setting were proposed in [8,18]. An early attempt to solve the challenging task of finding a minimizer of the non-convex, non-smooth Mumford-Shah functional was done by approximating it by a sequence of simpler elliptic problems in [3]. Many approaches to simplify the model were meanwhile proposed in the literature as, e.g., the convex relaxation of the piecewise smooth Mumford-Shah functional by functional lifting in [26]. A frequently applied strategy is to restrict the model to $\nabla u = 0$ on $\Omega \setminus \Gamma$ which results in the piecewise constant Mumford-Shah model

$$E_{\mathrm{PCMS}}(\Gamma, u) := \mathcal{H}^1(\Gamma) + \lambda \int_{\Omega} (u - f)^2 dx. \tag{1}$$

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 237–250, 2013.
© Springer-Verlag Berlin Heidelberg 2013

Assuming that $\Omega = \bigcup_{i=0}^{K-1} \Omega_i$ with pairwise disjoint sets Ω_i and $u(x) := m_i$ for $x \in \Omega_i$, $i = 0, \dots K-1$, the above functional can be rewritten as

$$E_{\mathrm{PCMS}}(\boldsymbol{\Omega}, \boldsymbol{m}) = \frac{1}{2} \sum_{i=0}^{K-1} \mathrm{Per}(\Omega_i; \Omega) + \lambda \sum_{i=0}^{K-1} \int_{\Omega_i} (m_i - f)^2 \, dx, \tag{2}$$

where $\mathrm{Per}(\Omega_i; \Omega)$ denotes the perimeter of Ω_i in Ω and $\boldsymbol{m} := (m_i)_{i=0}^{K-1}$, $\boldsymbol{\Omega} := (\Omega_i)_{i=0}^{K-1}$. For $K = 2$, the piecewise constant Mumford-Shah model is actually the model of the active contours without edges (Chan-Vese model) [16], i.e.

$$E_{\mathrm{CV}}(\Omega_1, m_0, m_1) = \mathrm{Per}(\Omega_1; \Omega) + \lambda \Big(\int_{\Omega_1} (m_1 - f)^2 \, dx + \int_{\Omega \setminus \Omega_1} (m_0 - f)^2 \, dx \Big). \tag{3}$$

One of the model's drawbacks is that it can easily get stuck in local minima. To overcome this drawback, a convex relaxation approach was proposed in [15]. More precisely, it was shown that the global minimizer of $E_{\mathrm{CV}}(\cdot, m_0, m_1)$ for fixed m_0, m_1 can be found by solving

$$\min_{0 \le u \le 1} \int_\Omega |\nabla u| + \lambda \int_\Omega \big((m_0 - f)^2 - (m_1 - f)^2 \big) u(x) dx, \tag{4}$$

and setting $\Omega_1 := \{x \in \Omega : u(x) > \rho\}$ for any $\rho \in (0, 1]$, see also [6,9]. In other words, (4) is a tight relaxation of the Chan-Vese model with fixed m_i, $i = 0, 1$. There are many other approaches for two-phase image segmentation based on the Chan-Vese model and its convex version, see, e.g., [31], [9] and [17].

In [28], Chan and Vese proposed a multiphase segmentation model using level sets. Convex (non-tight) relaxation approaches for the model with fixed $\boldsymbol{m}$ were proposed, e.g., in [21,22,25,29,30] and for the full model in [10]. For more details see also [5].

In [12] a two-stage image segmentation method which finds the solution of a convex variant of the Mumford-Shah model in the first stage followed by *one* thresholding step in the second stage was proposed. The applied functional was the Rudin-Osher-Fatemi (ROF) functional [27] supplemented by the *essential additional term* $\int |\nabla u|^2 \, dx$.

In this paper, we propose a new multiphase segmentation model based on *iteratively* thresholding the minimizer of the *original* ROF functional. In contrast to [12] we propose a strategy to update the thresholds and prove its convergence under certain conditions. There exists a clear relationship of our new model to the Chan-Vese model (3) which shows that a solution of (3) for a certain regularization parameter can actually be given by iteratively thresholding the ROF minimizer. Numerical examples demonstrate that our algorithm is not only fast but produces also very good results for images whose classes are close to each other. In particular it outperforms the algorithm in [12].

The paper is organized as follows: In Section 2, we introduce our segmentation model and discuss its the properties. We propose an efficient solution algorithm and provide a convergence analysis in Section 3. Finally, in Section 4, we test our algorithm on various synthetic and real-world images and compare it with other state-of-the-art segmentation algorithms.

2 Continuous Model

2.1 Notation

We briefly introduce the basic notation and relations which can be found, e.g., in [2,4]. In the following a 'set' is understood as a Lebesgue measurable set in $\mathbb{R}^2$, where we will consider equivalence classes of sets which are equal up to Lebesgue negligible sets. By $|A|$ we denote the Lebesgue measure of a set A. By $BV(\Omega)$ we denote the *space of functions of bounded variation*, i.e., the Banach space of functions $u : \Omega \to \mathbb{R}$ with finite norm $\|u\|_{BV} := \|u\|_{L_1(\Omega)} + TV(u)$, where

$$TV(u) := \sup \Big\{ \int_\Omega u(x) \mathrm{div}\varphi \, dx : \varphi \in \mathcal{C}_c^1(\Omega, \mathbb{R}^2), \|\varphi\|_\infty \leq 1 \Big\}.$$

The distributional first order derivative Du of u is a vector-valued Radon measure with total variation $|Du| = TV(u)$. In particular, we have for $u \in W_1^1(\Omega)$ that $Du = \nabla u \in L_1$ so that in this case $TV(u) = \int_\Omega |\nabla u| \, dx$. For a Lebesgue measurable set $A \subset \Omega$, the *perimeter* of A in Ω is defined by $\mathrm{Per}(A; \Omega) := TV(\chi_A)$, where χ_A denotes the *characteristic function* of A. Hence A is of finite perimeter, if its characteristic function has bounded variation. If A has a $\mathcal{C}^1$ boundary, then $\mathrm{Per}(A; \Omega)$ coincides with $\mathcal{H}^1(\partial A \cap \Omega)$. We define the mean of f on $A \subset \mathbb{R}^2$ by

$$\mathrm{mean}_f(A) := \begin{cases} \frac{1}{|A|} \int_A f \, dx & \text{if } |A| > 0, \\ 0 & \text{otherwise.} \end{cases}$$

We call (u^*, c^*) a *partial minimizer* of some objective function $E(u, c)$ if

$$\begin{aligned} E(u^*, c^*) &\leq E(u^*, c) \quad \text{for all feasible } c, \quad \text{and} \\ E(u^*, c^*) &\leq E(u, c^*) \quad \text{for all feasible } u. \end{aligned} \tag{5}$$

In case E is differentiable on its domain every partial minimizer contained in the interior of the domain is stationary, see, e.g., [19]. For example we see that a partial minimizer $(\boldsymbol{\Omega}^*, \boldsymbol{m}^*)$ of the piecewise constant Mumford-Shah model (2) with $\boldsymbol{\Omega}^* = (\Omega_i^*)_{i=0}^{K-1}$, $\boldsymbol{m}^* = (m_i^*)_{i=0}^{K-1}$ has to fulfill

$$m_i^* = \mathrm{mean}_f(\Omega_i^*), \quad i = 0, \ldots, K-1. \tag{6}$$

2.2 Model

We start by considering the segmentation into $K = 2$ classes. Let

$$\mathcal{E}(\Sigma, \tau) := \mathrm{Per}(\Sigma; \Omega) + \mu \int_\Sigma \tau - f \, dx, \quad \mu > 0. \tag{7}$$

Note that $\mathcal{E}(\emptyset, \tau) = 0$ and $\mathcal{E}(\Omega, \tau) = \mu \int_\Omega \tau - f \, dx$. Since f maps into $[0, 1]$, the global minimizer of $\mathcal{E}(\cdot, \tau)$ for fixed $\tau \leq 0$ is Ω and for $\tau \geq 1$ it is $\emptyset$. Therefore we restrict ourselves to $\tau \in (0, 1)$. We intend to find $(\Sigma^*, \tau^*) \in \Omega \times (0, 1)$ fulfilling

$$\begin{aligned} \mathcal{E}(\Sigma^*, \tau^*) &\leq \mathcal{E}(\Sigma, \tau^*) \quad \forall \Sigma \subset \Omega, \\ \tau^* &= \frac{1}{2}\big(\mathrm{mean}_f(\Sigma^*) + \mathrm{mean}_f(\Omega \backslash \Sigma^*)\big). \end{aligned} \tag{8}$$

Remark 1. Note that solving (8) is different from minimizing

$$\min_{\Sigma,\tau} \mathcal{E}(\Sigma,\tau) \quad \text{subject to} \quad \tau = \frac{1}{2}\big(\text{mean}_f(\Sigma) + \text{mean}_f(\Omega\backslash\Sigma)\big). \tag{9}$$

Consider the 1D example with the function $f(x) = x$ on $\Omega = (0,1)$ and restrict the attention to $\Sigma \in \{(0,b),(b,1)\}$. Then $\tau = \frac{1}{4}(1+2b)$ in (9) and we are searching for b. Now

$$\mathcal{E}(\Sigma,\tau) = 1 + \mu \int_I \tau - x\,dx = \begin{cases} 1 + \frac{1}{4}\mu b & \text{if } I = (0,b), \\ 1 - \frac{1}{4}\mu(1-b) & \text{if } I = (b,1) \end{cases}$$

which has no minimizer in $(0,1)$. On the other hand, let $\tau^* = \frac{1}{2}$. Then it can easily be checked that (Σ^*,τ^*) with $\Sigma^* = (\frac{1}{2},1)$ fulfills (8).

The following proposition ensures the existence of a global minimizer of $\mathcal{E}(\cdot,\tau)$ for fixed τ.

Proposition 1. *For any fixed $\tau \in (0,1)$, a global minimizer Σ of $\mathcal{E}(\cdot,\tau)$ in (7) can be found by solving the convex minimization problem*

$$\min_{u\in BV(\Omega),u\in[0,1]} TV(u) + \mu \int_\Omega (\tau - f)\,u\,dx \tag{10}$$

and then setting $\Sigma := \{x \in \Omega : u(x) > \rho\}$ for any $\rho \in [0,1)$.

For a proof we refer to Proposition 2.1 in the review paper [14]. This proof uses the same ideas as in [6,24] where the claim was shown for a.e. $\rho \in [0,1]$. Based on the next lemma, cf., [1, Lemma 4i)] and a smoothness argument, an explanation that the minimizing set Σ is unique for fixed τ was given in [14].

Lemma 1. *For fixed $0 < \tau_1 < \tau_2 < 1$, let Σ_i be minimizers of $\mathcal{E}(\cdot,\tau_i)$, $i = 1,2$. Then $|\Sigma_2\backslash\Sigma_1| = 0$ is fulfilled, i.e., $\Sigma_1 \supseteq \Sigma_2$ up to a negligible set.*

The relationship between our model (8) and the Chan-Vese model (3) is explained in the following proposition.

Proposition 2. (Relation between the Chan-Vese model and (8))
Assume that (Σ^*,τ^*), $\Sigma^* \notin \{\emptyset,\Omega\}$ is a solution of (8). Set $m_0^* := \text{mean}_f(\Sigma^*)$ and $m_1^* := \text{mean}_f(\Omega \setminus \Sigma^*)$. Let $\mathcal{O}$ be the set of partial minimizers of the Chan-Vese model (3) with parameter $\lambda := \frac{\mu}{2(m_1^* - m_0^*)}$. Then $\big(\Sigma^*, m_0^*, m_1^*\big) \in \mathcal{O}$.

Proof. Since $\Sigma^* \neq \emptyset$ is a minimizer of $\mathcal{E}(\cdot,\tau^*)$ we conclude $\int_{\Sigma^*} \tau^* - f\,dx < 0$ which implies $\tau^* < \text{mean}_f(\Sigma^*) = m_1^*$. Similarly, since $\Sigma^* \neq \Omega$, we see that

$$\begin{aligned} \text{Per}(\Sigma^*;\Omega) + \mu \int_{\Sigma^*} \tau^* - f\,dx &\le \mu \int_\Omega \tau^* - f\,dx, \\ 0 < \text{Per}(\Sigma^*;\Omega) &\le \mu \int_{\Omega\setminus\Sigma^*} \tau^* - f\,dx \end{aligned}$$

and consequently $m_0^* = \text{mean}_f(\Omega \setminus \Sigma^*) < \tau^*$. Therefore $m_0^* < m_1^*$. The set Σ^* is also a minimizer of $\mathcal{E}(\cdot, \tau^*) + C$ with the constant $C := \lambda \int_\Omega (m_0^* - f)^2 \, dx$. Regarding that $\tau^* = \frac{m_1^* + m_0^*}{2}$ we obtain

$$\begin{aligned}
\mathcal{E}(\Sigma, \tau^*) + C &= \text{Per}(\Sigma; \Omega) + \mu \int_\Sigma \tau^* - f \, dx + C \\
&= \text{Per}(\Sigma; \Omega) + \frac{\mu}{2(m_1^* - m_0^*)} \int_\Sigma (m_1^* - f)^2 - (m_0^* - f)^2 \, dx \\
&\quad + \lambda \int_\Omega (m_0^* - f)^2 \, dx \\
&= \text{Per}(\Sigma; \Omega) + \lambda \left(\int_\Sigma (m_1^* - f)^2 dx + \int_{\Omega \setminus \Sigma} (m_0^* - f)^2 \right).
\end{aligned}$$

By definition of m_i^*, $i = 0, 1$ and (6) we get the assertion. □

Since $0 < m_1^* - m_0^* \leq 1$ the parameter $\lambda = \frac{\mu}{2(m_1^* - m_0^*)}$ in the Chan-Vese model (3) is larger than μ and increases if $m_1^* - m_0^*$ becomes smaller. Hence, it is adapted to the difference between m_1^*, m_0^* and penalizes the data term more if this difference becomes smaller.

The following proposition has the important consequence that we can obtain a minimizer Σ of $\mathcal{E}(\cdot, \tau)$ by minimizing the ROF functional and subsequent thresholding of the minimizing function by τ.

Proposition 3. *The set* $\{x \in \Omega : u(x) > \tau\}$ *solves* (7) *if and only if the function* $u \in BV(\Omega)$ *solves the ROF model*

$$\min_{u \in BV(\Omega)} TV(u) + \frac{\mu}{2} \int_\Omega \big(u - f\big)^2 \, dx. \tag{11}$$

For the proof see [14, Proposition 2.6].

We generalize (7) and (8) to the multiclass case $K \geq 2$ by setting $\boldsymbol{\Sigma} := \{\Sigma_i\}_{i=1}^{K-1}$ and $\boldsymbol{\tau} := \{\tau_i\}_{i=1}^{K-1}$ with $0 < \tau_1 \leq \tau_2 \leq \ldots \leq \tau_{K-1} < 1$, and

$$\mathcal{E}(\boldsymbol{\Sigma}, \boldsymbol{\tau}) := \sum_{i=1}^{K-1} \big(\text{Per}(\Sigma_i; \Omega) + \mu \int_{\Sigma_i} \tau_i - f \, dx \big), \quad \mu > 0. \tag{12}$$

For fixed $\boldsymbol{\tau}$, we know by Lemma 1 that

$$\Omega \supseteq \Sigma_{\tau_1} \supseteq \Sigma_{\tau_2} \supseteq \cdots \supseteq \Sigma_{\tau_{K-1}} \supseteq \emptyset \tag{13}$$

and we see that the corresponding wanted segments

$$\Omega_i := \Sigma_i \setminus \Sigma_{i+1}, \; i = 0, \ldots, K-1, \; \Sigma_0 := \Omega, \; \Sigma_K := \emptyset \tag{14}$$

are pairwise disjoint and fulfill $\cup_{i=0}^{K-1} \Omega_i = \Omega$. We aim to find an ordered vector $\boldsymbol{\tau}^*$ and a corresponding nested set $\boldsymbol{\Sigma}^*$ with

$$\tau_i^* = \frac{1}{2}(m_{i-1}^* + m_i^*), \quad m_i^* := \text{mean}_f(\Omega_i^*), \quad i = 1, \ldots, K-1 \tag{15}$$

which minimizes $\mathcal{E}(\cdot, \boldsymbol{\tau}^*)$ among all sequences of nested sets.

3 Algorithmic Aspects

Our algorithm alternates the minimization of $\mathcal{E}(\cdot, \boldsymbol{\tau})$ in (12) for fixed $\boldsymbol{\tau}$ with $\tau_1 \leq \tau_2 \leq \ldots \leq \tau_{K-1}$ and the computation of $\boldsymbol{\tau}$ as in (15) for fixed sequences $\boldsymbol{\Sigma}$ of nested sets. By Proposition 3 the minimization of $\mathcal{E}(\cdot, \boldsymbol{\tau})$ in (12) for fixed $\boldsymbol{\tau}$ can be obtained by $K-1$ times thresholding the minimizer of the ROF functional. This is in particular efficient since the minimizer of the ROF functional remains the same during the whole thresholding process. We call the algorithm *thresholded ROF* (T-ROF).

Algorithm (T-ROF)

Initialization: $\tau^{(0)} = \left(\tau_i^{(0)}\right)_{i=1}^{K-1}$ with $0 < \tau_1^{(0)} < \cdots < \tau_{K-1}^{(0)} < 1$.

1. Compute the solution u of the ROF model (11).
2. For $k = 0, 1, \ldots,$ repeat
   2.1. Compute $\boldsymbol{\Sigma}^{(k)} = \left(\Sigma_i^{(k)}\right)_{i=1}^{K-1}$ by $\Sigma_i^{(k)} := \{x \in \Omega : u(x) > \tau_i^{(k)}\}$.
   2.2. Find $\Omega_i^{(k)} := \Sigma_i^{(k)} \backslash \Sigma_{i+1}^{(k)}$, $i = 0, \ldots, K-1$ with $\Sigma_0^{(k)} := \Omega$ and $\Sigma_K^{(k)} := \emptyset$.
   2.3. Compute $m_i^{(k)} := \mathrm{mean}_f(\Omega_i^{(k)})$, $i = 0, \ldots, K-1$.
   2.4. Update $\tau_i^{(k+1)} := \frac{1}{2}(m_{i-1}^{(k)} + m_i^{(k)})$, $i = 1, \ldots, K-1$.

We will prove the convergence of our algorithm under the following assumption:

(A) If $\Sigma_\tau, \Sigma_{\tilde{\tau}}$ are the minimizers of $\mathcal{E}(\cdot, \tau)$ and $\mathcal{E}(\cdot, \tilde{\tau})$ for any $0 < \tau < \tilde{\tau} < 1$ appearing in the algorithm, then

$$\tau \leq \mathrm{mean}_f(\Sigma_\tau \backslash \Sigma_{\tilde{\tau}}) \leq \tilde{\tau}. \tag{16}$$

The right-hand inequality in (16) is for example fulfilled if $\Sigma_{\tilde{\tau}}$ is also a minimizer of $\mathrm{Per}(\Sigma; \Sigma_\tau) + \mu \int_\Sigma \tilde{\tau} - f \, dx$. The left-hand inequality holds if $\Sigma_\tau \backslash \Sigma_{\tilde{\tau}}$ is also a minimizer of $\mathrm{Per}(\Sigma; \Omega \backslash \Sigma_{\tilde{\tau}}) + \mu \int_\Sigma \tau - f \, dx$. Using the above assumption we can prove the following lemma, see [11]:

Lemma 2. *Under the assumption* (A) *our T-ROF algorithm produces sequences* $(\boldsymbol{\tau}^{(k)})_k$ *and* $(\boldsymbol{m}^{(k)})_k$ *with the following properties:*

i) $0 \leq m_0^{(k)} \leq \tau_1^{(k)} \leq m_1^{(k)} \leq \cdots \leq m_{K-2}^{(k)} \leq \tau_{K-1}^{(k)} \leq m_{K-1}^{(k)}$
ii) *Set* $\tau_0^{(k)} := 0$ *and* $\tau_K^{(k)} := 1$. *If* $\tau_i^{(k)} \geq \tau_i^{(k-1)}$ *and* $\tau_{i+1}^{(k)} \geq \tau_{i+1}^{(k-1)}$, *then* $m_i^{(k)} \geq m_i^{(k-1)}$, $i = 0, \ldots, K-1$ *and this also holds true if* $\leq$ *is replaced everywhere by* $\geq$.

To prove the convergence of the sequence $(\boldsymbol{\tau}^{(k)})_k$, we define a sign sequence $\boldsymbol{\zeta}^{(k)} = (\zeta_i^{(k)})_{i=1}^{K-1}$ as follows: If $\tau_i^{(k)} \neq \tau_i^{(k-1)}$,

$$\zeta_i^{(k)} := \begin{cases} +1 & \text{if } \tau_i^{(k)} > \tau_i^{(k-1)}, \\ -1 & \text{if } \tau_i^{(k)} < \tau_i^{(k-1)}, \end{cases}$$

and otherwise

$$\zeta_i^{(k)} := \begin{cases} \zeta_j^{(k)} & \text{if } i = 1, \\ \zeta_{i-1}^{(k)} & \text{if } i \neq 1, \end{cases} \tag{17}$$

where $j = \min\{l \mid \tau_l^{(k)} \neq \tau_l^{(k-1)}\}$. By s_k we denote the number of sign changes in $\zeta^{(k)}$, for example, if $\boldsymbol{\zeta}^{(k)} = (\overbrace{+1,+1,+1}, \overbrace{-1,-1}, \overbrace{+1}, \overbrace{-1})$, then $s_k = 3$.

Lemma 3. i) *The number of sign changes s_k is monotone decreasing in k.*
ii) *If $\zeta_1^{(k+1)} \neq \zeta_1^{(k)}$, then we have the strict decrease $s_{k+1} < s_k$.*

Proof. i) Let $s_k = N$ and rewrite $(\boldsymbol{\tau}^{(k)})_k$ as

$$(\underbrace{\tau_0^{(k)}, \cdots, \tau_{l_1}^{(k)}}_{v_1^{(k)}}, \cdots, \underbrace{\tau_{i_j}^{(k)}, \cdots, \tau_{l_j}^{(k)}}_{v_j^{(k)}}, \cdots, \underbrace{\tau_{i_N}^{(k)}, \cdots, \tau_K^{(k)}}_{v_N^{(k)}}),$$

where $v_j^{(k)}$ contains those successive components of $(\boldsymbol{\tau}^{(k)})_k$ with the same sign.
1. If $\#v_j^{(k)} \geq 3$, we consider $\zeta_{i^*}^{(k+1)}$ with $i_j \leq i^* - 1 \leq i^* \leq i^* + 1 \leq l_j$, i.e., $\zeta_{i^*-1}^{(k)} = \zeta_{i^*}^{(k)} = \zeta_{i^*+1}^{(k)}$. WLOG let $\zeta_{i^*}^{(k)} = -1$. Then we obtain by Lemma 2 ii) that $m_{i^*-1}^{(k)} \leq m_{i^*-1}^{(k-1)}$ and $m_{i^*}^{(k)} \leq m_{i^*}^{(k-1)}$. Therefore

$$\tau_{i^*}^{(k+1)} = \frac{m_{i^*-1}^{(k)} + m_{i^*}^{(k)}}{2} \leq \frac{m_{i^*-1}^{(k-1)} + m_{i^*}^{(k-1)}}{2} = \tau_{i^*}^{(k)}$$

and consequently $\zeta_{i^*}^{(k+1)} = -1$ or $\zeta_{i_j}^{(k+1)} = \zeta_{i_j+1}^{(k+1)} = \cdots = \zeta_{i^*}^{(k+1)} = 1$. The case $\zeta_{i^*}^{(k)} = 1$ can be handled in the same way.
2. If there is no j such that $\#v_j^{(k)} = 1$, we consider $\zeta_{l_j}^{(k+1)}$ and $\zeta_{i_{j+1}}^{(k+1)}$ which are different by definition. WLOG let $\zeta_{l_j}^{(k)} = -1$ and $\zeta_{i_{j+1}}^{(k)} = +1$. Then, from Lemma 2 ii), we have

$$m_{l_j-1}^{(k)} \leq m_{l_j-1}^{(k-1)}, \quad m_{i_{j+1}}^{(k)} \geq m_{i_{j+1}}^{(k-1)}.$$

If $m_{l_j}^{(k)} \leq m_{l_j}^{(k-1)}$ (or $m_{l_j}^{(k)} \geq m_{l_j}^{(k-1)}$), then $\tau_{l_j}^{(k+1)} \leq \tau_{l_j}^{(k)}$ (or $\tau_{i_{j+1}}^{(k+1)} \geq \tau_{i_{j+1}}^{(k)}$). This means that $\zeta_{l_j}^{(k+1)} \neq \zeta_{l_j}^{(k)}$ and $\zeta_{i_{j+1}}^{(k+1)} \neq \zeta_{i_{j+1}}^{(k)}$ is not possible at the same time.
3. Finally, we consider the case $\#v_j^{(k)} = 1$ for all $j_1 \leq j \leq j_2$, where $\#v_{j_1-1}^{(k)} > 1$ and $\#v_{j_2+1}^{(k)} > 1$. We prove that from iteration $k \to k+1$ the signs of $\zeta_{i_{j_1}-1}^{(\cdot)}, \zeta_j^{(\cdot)}, \ldots, \zeta_{j_2}^{(\cdot)}, \zeta_{i_{j_2}+1}^{(\cdot)}$ can not change at the same time. WLOG assume that $\zeta_{i_{j_1}}^{(k)} = -1$ so that $\zeta_{i_j}^{(k)} = (-1)^{j-j_1+1}$ for $j_1 \leq j \leq j_2$, and $\zeta_{i_{j_1}-1}^{(k)} = \zeta_{i_{j_1}-2}^{(k)} = +1$ and $\zeta_{i_{j_2}+1}^{(k)} = \zeta_{i_{j_2}+2}^{(k)} = (-1)^{j_2-j_1}$. ¿From Lemma 2 ii), we know that

$$m_{i_{j_1}-2}^{(k)} \geq m_{i_{j_1}-2}^{(k-1)}, \quad \text{and} \quad \begin{cases} m_{i_{j_2}+1}^{(k)} \geq m_{i_{j_2}+1}^{(k-1)} & \text{if } j_2 - j_1 \text{ is even}, \\ m_{i_{j_2}+1}^{(k)} \leq m_{i_{j_2}+1}^{(k-1)} & \text{if } j_2 - j_1 \text{ is odd}. \end{cases} \tag{18}$$

If in $\zeta_{i_{j_1}-1}^{(\cdot)}$, $\zeta_{i_{j_1}}^{(\cdot)}$, $\cdots$, $\zeta_{i_{j_2}}^{(\cdot)}$, $\zeta_{i_{j_2}+1}^{(\cdot)}$ the signs change at the same time, we can deduce by Lemma 2 ii) that

$$\begin{cases} m_{i_{j_2}+1}^{(k)} < m_{i_{j_2}+1}^{(k-1)} & \text{if } j_2 - j_1 \text{ is even,} \\ m_{i_{j_2}+1}^{(k)} > m_{i_{j_2}+1}^{(k-1)} & \text{if } j_2 - j_1 \text{ is odd,} \end{cases}$$

which contradicts (18).

By parts 1–3, we see that $s_{k+1} \leq s_k$ for any $k \in \mathbb{N}$.

ii) If $\zeta_1^{(k+1)} \neq \zeta_1^{(k)}$, then $v_1^{(k)} = \tau_1^{(k)}$ and $\tau_2^{(k)} \in v_2^{(k)}$. By parts 1–3 of the proof we get $s_{k+1} < s_k$. This completes the proof. □

Now we can prove the convergence of our T-ROF algorithm with a slight modification. We divide the interval $[0, 1)$ into $n > 1$ disjoint subintervals $[\frac{i}{n}, \frac{i+1}{n})$, and define a projector $\mathcal{P}_n : [0, 1) \to \{\frac{i}{n} : i = 0, \ldots, n-1\}$ by $\mathcal{P}_n(x) := \frac{i}{n}$ if $x \in [\frac{i}{n}, \frac{i+1}{n})$. Clearly, $\mathcal{P}_n(x_1) \geq \mathcal{P}_n(x_2)$ if $x_1 \geq x_2$. We choose n large enough (say machine precision). Instead of $\boldsymbol{\tau}^{(k)}$ we compute in each step of the T-ROF algorithm the projection $\mathcal{P}_n(\boldsymbol{\tau}^{(k)})$ and continue the algorithm with these projected thresholds. Clearly, the statements of Lemma 2 and 3 remain true. For convenience we write again $\boldsymbol{\tau}^{(k)}$ for the output of the projected algorithm.

Theorem 1. *Under the assumption* (A)*, the sequence* $(\boldsymbol{\tau}^{(k)})_{k\in\mathbb{N}}$ *produced by the projected T-ROF algorithm converges to a vector* $\boldsymbol{\tau}^*$*.*

Proof. We prove the assertion by induction on the number of sign changes s_k in some iteration step k.

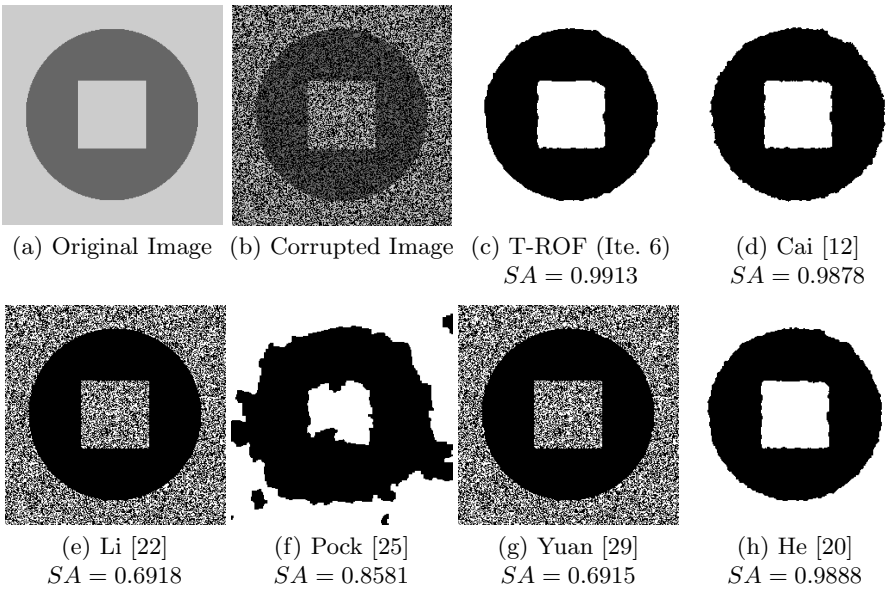
Assume that $s_k = 0$. WLOG let $\zeta_i^{(k)} = +1$, $i = 1, \ldots, K-1$, i.e., $\tau_i^{(k)} \geq \tau_i^{(k-1)}$. From Lemma 2 ii), we obtain $m_i^{(k)} \geq m_i^{(k-1)}$ and consequently $\tau_i^{(k+1)} \geq \tau_i^{(k)}$, $i = 1, \ldots, K-1$. Therefore $s_{k+1} = 0$ and $\zeta_i^{(k+1)} = +1$, $i = 1, \ldots, K-1$. This means that each sequence $(\tau_i^{(k)})_k$ is monotone increasing. Since the sequences are moreover bounded in $[0, 1]$, we conclude that $(\boldsymbol{\tau}^{(k)})_k$ converges.

Assume that $(\boldsymbol{\tau}^{(k)})_k$ converges if $s_k \leq N-1$ for some $k \in \mathbb{N}$.

We prove that in case $s_k = N$, there exists $\hat{k} > k$ such that $s_{\hat{k}} \leq N-1$. If there exists $\hat{k} > k$ such that $\zeta_1^{(\hat{k})} \neq \zeta_1^{(k)}$, we get $s_{\hat{k}} \leq N-1$ directly from Lemma 3 ii). If $\zeta_1^{(\hat{k})} = \zeta_1^{(k)}$ for all $\hat{k} > k$, then $(\tau_1^{(\hat{k})})_{\hat{k}>k}$ is monotone and bounded and converges consequently to some threshold τ_1^*. This threshold must be attained in the projected algorithm for some $k_1 > k$. Now we can repeat the same arguments with k_1 instead of k and $\zeta_2^{(\cdot)}$ instead of $\zeta_1^{(\cdot)}$ to see that $(\tau_2^{(k)})_k$ converges to a threshold τ_2^* which must be attained for some $k_2 > k_1$. Moreover, we have $\zeta_2^{(j)} = \zeta_1^{(j)}$ for all $j > k_1$. Repeating this procedure up to the final index $K-1$ we obtain the assertion. □

4 Numerical Results

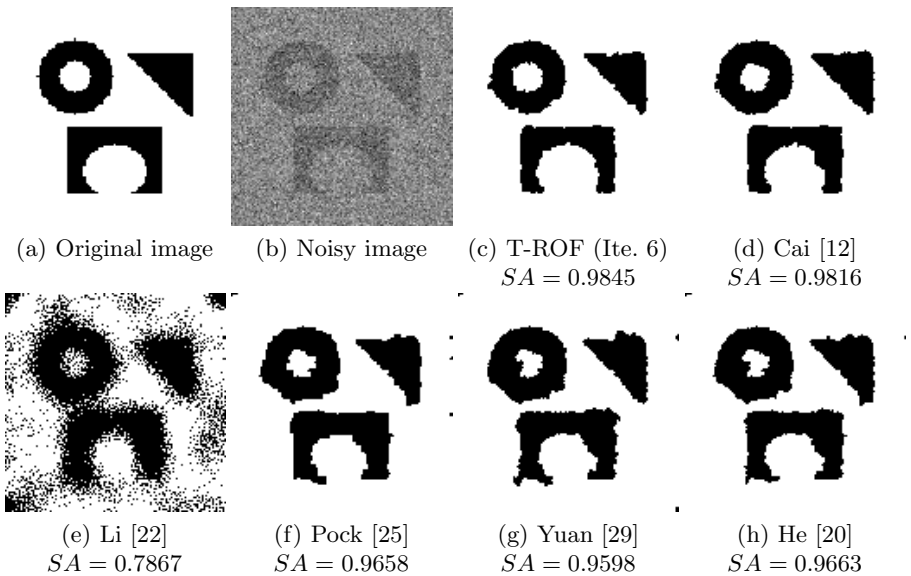
In this section, we test our method on various images. We actually use the T-ROF Algorithm with a discrete ROF model, see, e.g. [13], which minimizer is



(a) Original Image (b) Corrupted Image (c) T-ROF (Ite. 6) $SA = 0.9913$ (d) Cai [12] $SA = 0.9878$

(e) Li [22] $SA = 0.6918$ (f) Pock [25] $SA = 0.8581$ (g) Yuan [29] $SA = 0.6915$ (h) He [20] $SA = 0.9888$

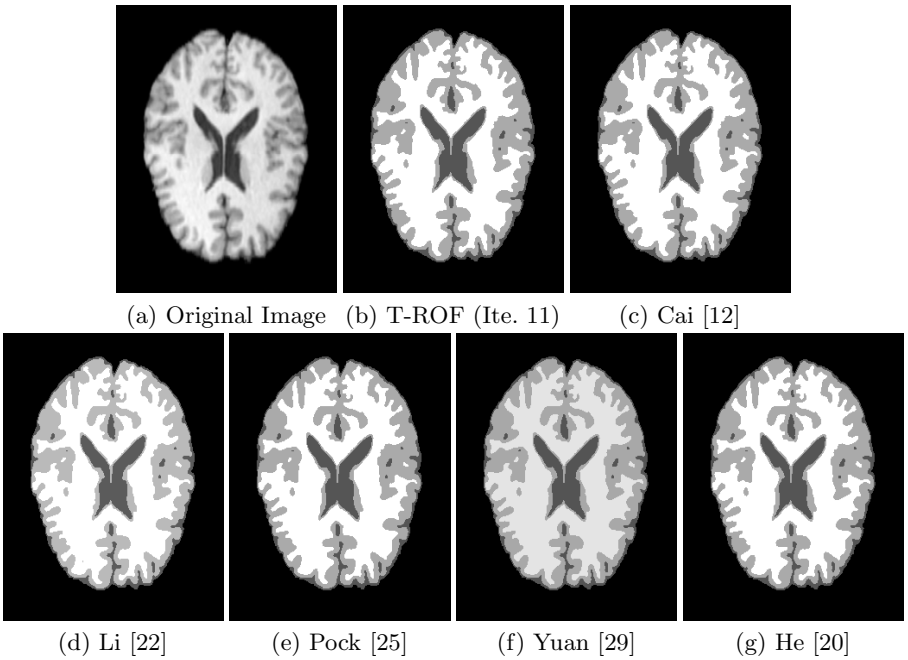
Fig. 1. Segmentation of two-class cartoon image (256×256) with some missing pixel values



(a) Original image (b) Noisy image (c) T-ROF (Ite. 6) $SA = 0.9845$ (d) Cai [12] $SA = 0.9816$

(e) Li [22] $SA = 0.7867$ (f) Pock [25] $SA = 0.9658$ (g) Yuan [29] $SA = 0.9598$ (h) He [20] $SA = 0.9663$

Fig. 2. Segmentation of two-class image (128×128) with close intensities



(a) Original Image (b) T-ROF (Ite. 11) (c) Cai [12]

(d) Li [22] (e) Pock [25] (f) Yuan [29] (g) He [20]

Fig. 3. Four-phase gray and white matter segmentation for a brain MRI image (319×256)

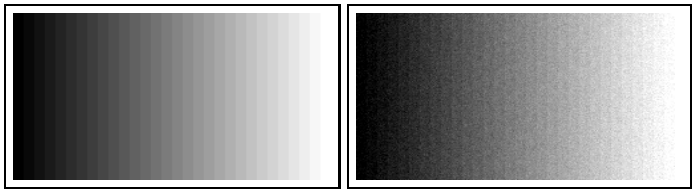


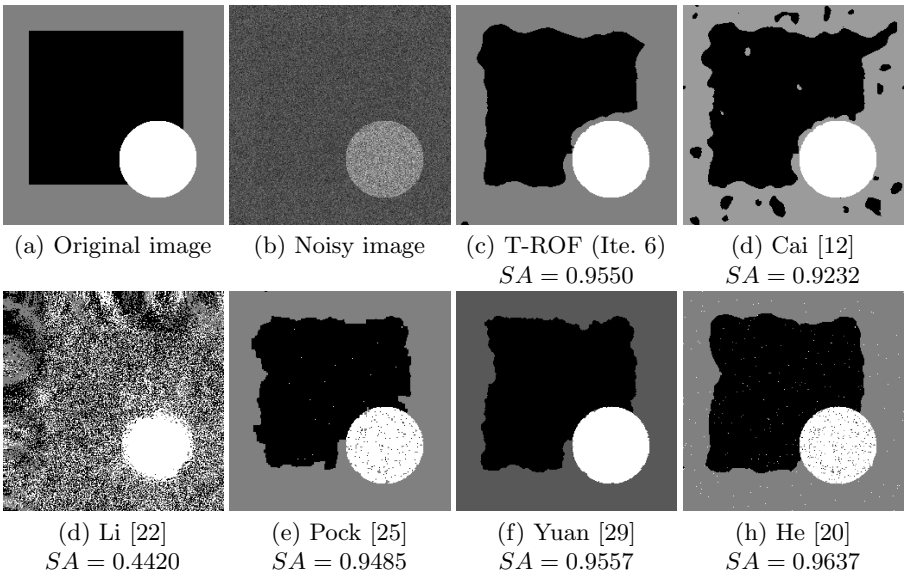
Fig. 4. Stripe image with 30 stripes (140×240) and its noisy version for the segmentation in Tab. 1

computed numerically by an ADMM algorithm with fixed inner parameter 2. Speedups by using more sophisticated methods will be considered in a future paper. The stopping criteria in our T-ROF algorithm for u and $\boldsymbol{\tau}$ are $\big(\|u^{(i)} - u^{(i-1)}\|_2\big)/\|u^{(i)}\|_2 \leq \epsilon_u$ and $\|\boldsymbol{\tau}^{(k)} - \boldsymbol{\tau}^{(k-1)}\|_2 \leq \epsilon_\tau$, where ϵ_u and ϵ_τ are fixed to 10^{-4} and 10^{-5}, respectively. The initialization of $(\tau_i^{(0)})_{i=1}^{K-1}$ was computed by the fuzzy C-means method [7] with 100 iteration steps.

We will compare our method with the recently proposed multiclass segmentation methods [12,20,22,25,29]. Note that the methods [25,29] work with the fixed fuzzy C-means codebook and we do not update the codebook. Such update is however involved in [20]. The default stopping criterion used in [20,22,25] is the maximum iteration steps; the default stopping criterion used in [29] is 10^{-4} and

Table 1. Parameter μ, iteration steps, CPU time in seconds, and SA for example 4

		Li [22]	Pock [25]	Yuan [29]	He [20]	Cai [12]	T-ROF
Five phases	μ	80	100	10	50	10	8
	Ite.	100	100	87	100	41	84 (4)
	Time	3.87	6.25	4.33	16.75	1.33	1.39
	SA	0.9946	0.9965	0.9867	0.9968	0.9770	0.9986
Ten phases	μ	80	100	10	50	10	8
	Ite.	100	100	102	100	41	84 (5)
	Time	7.71	15.41	9.79	38.52	2.11	2.33
	SA	0.8545	0.9984	0.9715	0.9848	0.8900	0.9967
Fifteen phases	μ	80	100	10	50	10	8
	Ite.	100	100	208	100	41	84 (5)
	Time	11.56	28.21	33.21	63.67	3.06	3.74
	SA	0.7715	0.9993	0.9730	0.9904	0.5280	0.9933



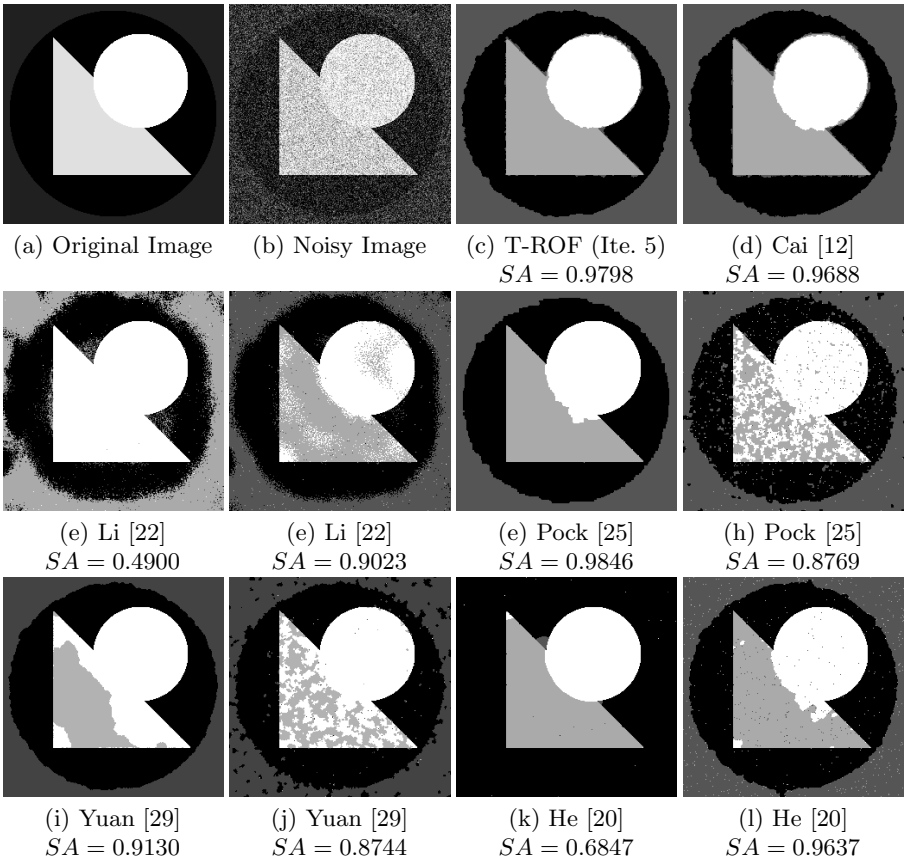
(a) Original image (b) Noisy image (c) T-ROF (Ite. 6) $SA = 0.9550$ (d) Cai [12] $SA = 0.9232$

(d) Li [22] $SA = 0.4420$ (e) Pock [25] $SA = 0.9485$ (f) Yuan [29] $SA = 0.9557$ (h) He [20] $SA = 0.9637$

Fig. 5. Segmentation of three-class image (256×256) containing phases with close intensities

maximum iteration steps 300; the default stopping criterion used in [12] is 10^{-4}. We choose the regularization parameter μ of the fidelity term in all the methods by judging the *segmentation accuracy* (SA) defined as

$$\text{SA} = \frac{\#\text{correctly classified pixels}}{\#\text{all pixels}}.$$



(a) Original Image (b) Noisy Image (c) T-ROF (Ite. 5) $SA = 0.9798$ (d) Cai [12] $SA = 0.9688$

(e) Li [22] $SA = 0.4900$ (e) Li [22] $SA = 0.9023$ (e) Pock [25] $SA = 0.9846$ (h) Pock [25] $SA = 0.8769$

(i) Yuan [29] $SA = 0.9130$ (j) Yuan [29] $SA = 0.8744$ (k) He [20] $SA = 0.6847$ (l) He [20] $SA = 0.9637$

Fig. 6. Segmentation of four-class image (256×256) with close gray values

We show the results for two two-class and four multiclass images, where all computations were done on a MacBook with 2.4 GHz processor and 4GB RAM. For further examples we refer to [11].

Example 1 is a two-class cartoon image with some missing pixel values. The segmentation results are shown in Fig. 1. We see that only methods [12,20] and our method gives good results. Indeed a codebook update is required here.

Example 2 is a two-class image with close intensities generated as follows: We have added Gaussian noise with mean 0 and variance 10^{-8} to constant image with gray value 0.5. The noisy image is obtained by keeping the pixel values belonging to the white parts of the original image and reducing the pixel values belonging to the black parts by a factor of 2×10^{-4}. Fig. 2 shows the results of the various algorithms. Except the method [22], all models produce reasonable results.

Example 3 is a four-class gray and white matter segmentation for a brain MRI image from [25]. Fig. 3 gives the results. We can see that, all the methods work well for this kind of image. However our method with 11 τ-value updates is faster

than the other methods, e.g., three times faster than the algorithm of Pock et al. [25] with the assigned parameters.

Example 4 segments the noisy stripe image in Fig. 4 (b), which is generated by imposing Gaussion noise with mean 0 and variance 10^{-3} on the clean image Fig. 4 (a) with 30 stripes. The results for a 5, 10 and 15 class segmentation are listed in Table 1.

Example 5 is a three-class image with close intensities. The test in image Fig. 5 is generated using the same way as those in Example 2 with Gaussian noise of mean 0 and variance 10^{-2}, the scalars used in the black and white parts are 0.1 and 0.6, respectively. For the results of the different methods see Fig. 5.

Example 6 shows a four-class image with close intensities. Fig. 6 (a) and (b) are the original image and the noisy image generated by using Gaussian noise with mean 0 and variance 3×10^{-2}. Figs. 6 (c)–(l) provide two results for each method using different representative parameters μ, where one parameter is optimal with respect to the SA.

References

1. Alter, F., Caselles, V., Chambolle, A.: A characterization of convex calibrable sets in R^N. Mathematische Annalen 332, 329–366 (2005)
2. Ambrosio, L., Fusco, N., Pallara, D.: Functions of Bounded Variation and Free Discontinuity Problems. Oxford University Press, Oxford (2000)
3. Ambrosio, L., Tortorelli, V.: Approximation of functions depending on jumps by elliptic functionals via t-convergence. Communications in Pure and Applied Mathematics 43, 999–1036 (1990)
4. Attouch, H., Buttazzo, G., Michaille, G.: Variational Analysis in Sobolev and BV Spaces. SIAM, Philadelphia (2006)
5. Bar, L., Chan, T., Chung, G., Jung, M., Kiryati, N., Mohieddine, R., Sochen, N., Vese, L.: Mumford and Shah model and its applications to image segmentation and image restoration. In: Handbook of Mathematical Imaging, pp. 1095–1157. Springer (2011)
6. Bellettini, G., Paolini, M., Verdi, C.: Convex approximations of functionals with curvature. Mat. Appl. 2, 297–306 (1991)
7. Bezdek, J.C., Ehrlich, R., Full, W.: FCM: The fuzzy c-means clustering algorithm. Computers & Geosciences 10, 191–203 (1984)
8. Blake, A., Zisserman: Visual Reconstruction. MIT Press (1987)
9. Bresson, X., Esedoglu, S., Vandergheynst, P., Thiran, J., Osher, S.: Fast global minimization of the active contour/snake model. Journal of Mathematical Imaging and Vision 28, 151–167 (2007)
10. Brown, E., Chan, T., Bresson, X.: Completely convex formulation of the Chan-Vese image segmentation model. International Journal of Computer Vision 98, 103–121 (2012)
11. Cai, X.: A new multiclass image segmentation model by ROF thresholding. Preprint University of Kaiserslautern (2013)
12. Cai, X., Chan, R., Zeng, T.: A two-stage image segmentation method using a convex variant of the Mumford-Shah model and thresholding. SIAM Journal on Imaging Sciences 6, 368–390 (2013)

13. Chambolle, A.: An algorithm for total variation minimization and applications. Journal of Mathematical Imaging and Vision 20, 89–97 (2004)
14. Chambolle, A., Caselles, V., Novaga, M., Cremers, D., Pock, T.: An introduction to total variation for image analysis. HAL Preprint, hal-00437581 (2009)
15. Chan, T.F., Esedoglu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM Journal on Applied Mathematics 66, 1632–1648 (2006)
16. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Transactions on Image Processing 10, 266–277 (2001)
17. Dong, B., Chien, A., Shen, Z.: Frame based segmentation for medical images. Commun. Math. Sci. 32, 1724–1739 (2010)
18. Geman, S., Geman, D.: Stochastic relation, Gibbs distributions, and the Baysian resoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence 6(6), 721–741 (1984)
19. Gorski, J., Pfeiffer, F., Klamroth, K.: Biconvex sets and optimization with biconvex functions - a survey and extensions. Mathematical Methods of Optimization Research 66, 373–407 (2007)
20. He, Y., Hussaini, M.Y., Ma, J., Shafei, B., Steidl, G.: A new fuzzy c-means method with total variation regularization for image segmentation of images with noisy and incomplete data. Pattern Recognition 45, 3463–3471 (2012)
21. Lellmann, J., Schnörr, C.: Continuous multiclass labeling approaches and algorithms. SIAM Journal on Imaging Science (to appear)
22. Li, F., Ng, M., Zeng, T., Shen, C.: A multiphase image segmentation method based on fuzzy region competition. SIAM Journal on Imaging Sciences 3, 277–299 (2010)
23. Mumford, D., Shah, J.: Optimal approximation by piecewise smooth functions and associated variational problems. Communications on Pure and Applied Mathematics, 577–685 (1989)
24. Nikolova, M., Esedoglu, S., Chan, T.F.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM Journal on Applied Mathematics 66, 1632–1648 (2006)
25. Pock, T., Chambolle, A., Cremers, D., Bischof, H.: A convex relaxation approach for computing minimal partitions. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 810–817 (2009)
26. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An algorithm for minimizing the piecewise smooth Mumford-Shah functional. In: ICCV 2009 (2009)
27. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D 60, 259–268 (1992)
28. Vese, L., Chan, T.: A multiphase level set framework for image segmentation using the Mumford and Shah model. International Journal of Computer Vision 50, 271–293 (2002)
29. Yuan, J., Bae, E., Tai, X.-C., Boykov, Y.: A continuous max-flow approach to Potts model. In: European Conference on Computer Vision, pp. 379–392 (2010)
30. Zach, C., Gallup, D., Frahm, J., Niethammer, M.: Fast global labeling for real-time stereo using multiple plane sweeps. In: Vision, Modeling, and Visualization Workshop (2008)
31. Zhang, Y., Matuszewski, B., Shark, L., Moore, C.: Medical image segmentation using new hybrid level-set method. In: Fifth International Conference BioMedical Visualization: Information Visualization in Medical and Biomedical Informatics, 71–76 (2008)

A Generic Convexification and Graph Cut Method for Multiphase Image Segmentation

Jun Liu[1], Xue-Cheng Tai[2], and Shingyu Leung[3]

[1] School of Mathematical Sciences
Laboratory of Mathematics and Complex Systems,
Beijing Normal University, Beijing 100875, P.R. China
[2] Department of Mathematics, University of Bergen, Norway
[3] Department of Mathematics, Hong Kong University of Science and Technology,
Clear Water Bay, Hong Kong
jliu@bnu.edu.cn, tai@mi.uib.no, masyleung@ust.hk

Abstract. We propose a unified graph cut based global minimization method for multiphase image segmentation by convexifying the non-convex image segmentation cost functionals. As examples, we shall apply this method to the non-convex multiphase Chan-Vese (CV) model and piecewise constant level set method (PCLSM). Both continuous and discretized formulations will be treated. For the discrete models, we propose a unified graph cut algorithm to implement the CV and PCLSM models, which extends the result of Bae and Tai [1] to any phases CV model. Moreover, in the continuous case, we further improve the model to be convex without any conditions using a number of techniques that are unique to the continuous segmentation models. With the convex relaxation and the dual method, the related continuous dual model is convex and we can mathematically show that the global minimization can be achieved. The corresponding continuous max-flow algorithm is easy and stable. Experimental results show that our model is very efficient.

1 Introduction

Many multiphase image segmentation models are non-convex and thus the corresponding numerical algorithms may sometimes get stuck at a local minimum close to the initial condition and produce undesirable segmentation results. For example, the multiphase Chan-Vese (CV) model [2] to partition an image into n parts by using $\log_2 n$ level set functions is non-convex. Its global minimization can not be guaranteed. Another multiphase segmentation method is to use a piecewise constant level set method PCLSM [3, 4] to represent different classes, the constraint of imposing the label function to be a piecewise constant function is non-convex, and thus the global minimization for such a model also can not be guaranteed.

Some efforts on the global minimization have been done in recent years. For the discrete methods, it is well-known that the global minimization can be attained by the graph cut approach. However, the graph cut method can only

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 251–265, 2013.
© Springer-Verlag Berlin Heidelberg 2013

minimize some particular energies [5]. For a modified PCLSM [6], the global minimization can be obtained by Ishikawa graph cut method [7]. As to the multiphase CV model, generally speaking, the global minimization can not be achieved by the graph cut method since its associated discrete energy does not satisfy the graph representation condition in [5]. When a convex condition holds for the data term, Bae-Tai [8] have showed that the 4-phase CV model can be globally optimized by graph cut. If the convex condition fails, the authors also propose a truncation method to approximately minimize the energy. In [8], the weights assignment for the graph is derived from a nonnegative solution for some coupled linear equations, and the related convex condition makes sure that there is at least one nonnegative solution. To find such a nonnegative solution is not easy for any arbitrary phases segmentation, and thus this method is not convenient to be extended to any phases CV partition problems.

In the continuous case, convex relaxation method (e.g. [9–15]) is very popular in recent years. The main idea of the convex relaxation is to relax the binary characteristic function into a continuous interval $[0, 1]$ such that the non-convex original problem becomes convex. Solving such a relaxed convex problem can enable one to find a global minimizer, and then the global binary solution of the original problem can be obtained by a threshold process. The functional lifting method [10] can be regarded as a convex relaxation of PCLSM, while the multi-dimensional generalization of the functional lifting [16] ensure that one can get a convex formulation of multiphase CV model. The continuous max-flow [15] approach shows that finding a max-flow on a discrete graph, namely graph cut method, corresponding to solving a continuous primal-dual problem. It gives the connections between the discrete approach and continuous method. More interestingly, it was found that the "cut" is just the Lagrangian multiplier for the flow conservation constraint when maximizing the total flow. Recently, some experimental comparisons between discrete and continuous segmentation methods has been given in [17] for some selected continuous multi-labelling approaches. It would be interesting to see a systematical comparison including these continuous max-flow with Augmented Lagrangian approaches.

The non-convex property of both PCLSM and CV models comes from the existence of some non-convex multiplication function terms. In this paper, we shall show that both of these two models can be written as some similar linear term plus multiplication function terms. Based on the graph cut minimization theory [5] and the property of *max* function which is associated to the convex envelope of the multiplication function, we propose a unified graph cut method to globally solve these two models. Since each term of the cost functional is convex after using the envelope functions, and thus the related relaxation continuous problem is also convex. Our method can easily handle any phases segmentation problems. For the discrete PCLSM, finding max-flows on the proposed graph is faster than Ishikawa's [7]. For the discrete 4-phase CV model, the proposed graph would be the same as [8], but it is easier to handle any phases segmentation. Moreover, to drop the convex condition in K-phase CV model in graph cut method, we propose a continuous relaxation max-flow method for multiphase

CV. Compared the earlier graph cut method, the proposed model is convex without any condition. Experimental results have shown that this can improve the quality of the segmentation results. A simultaneous work [18] appearing in this conference also derives a convex relaxation for the Chan-Vese model with any number of phases. However, their approach is based on directly computing the convex envelope of the data term.

The rest of the paper is organized as follows: section 2 is a brief introduction of the fundamental energy minimization theory with graph cut; in section 3, we propose the graph construction for the PCLSM and CV models; section 4 contains a continuous relaxation max-flow model for CV with super-level set representation; we show some experimental results and comparisons with other methods in section 5; finally, some conclusions and discussions are presented in section 6.

2 Energy Minimization with Graph Cut

In [5], the authors have given the condition of what energies can be minimized by graph cut. In this section, we shall briefly review the main results of [5].

Let $v_1, v_2, \cdots, v_n \in \{0, 1\}$, $\mathcal{E}$ be a function of some binary variables. Then the following theorems hold:

Theorem 1 ([5]). *All the functions $\mathcal{E}(v_i)$ of one binary variable can be minimized by graph cut.*

Theorem 2 ([5]). *A function $\mathcal{E}(v_1, v_2)$ of 2 binary variables can be minimized by graph cut if and only if $\mathcal{E}$ is submodular, i.e. $\mathcal{E}(0,0)+\mathcal{E}(1,1) \leqslant \mathcal{E}(0,1)+\mathcal{E}(1,0)$. More generally, $\mathcal{E}(v_1, \cdots, v_n)$ of n binary variables can be minimized by graph cut if and only if $\mathcal{E}$ is submodular.*

Theorem 3 (additivity, [5]). *The sum of finite number of submodular functions is submodular.*

From the additivity theorem 3, one can conclude that if there are n functions which can be minimized by n different graphs, then the sum of the n functions also can be minimized by graph cut. This can be done by simply putting the vertices together and adding the n graphs' edge weights together (if any graphs have no edge between two vertices, one can add an edge with weight 0). For the proofs of these theorems, please refer to [5].

Suppose a s-t graph $\mathbb{G} = (\mathbb{V}, \mathbb{E})$ is constituted by a set of vertices $\mathbb{V}$ and a set of directed edges $\mathbb{E}$. Here there are two special distinguished vertices in $\mathbb{V}$, the source s and the sink t. A cut on the graph $\mathbb{G}$ is denoted by $(\mathbb{V}_t, \mathbb{V}_s)$, which is to partition the vertices $\mathbb{V}$ into two disjoint connected set $\mathbb{V}_s$ and $\mathbb{V}_t$ such that $s \in \mathbb{V}_s$ and $t \in \mathbb{V}_t$. For all binary variable $v_i \in \{0, 1\}$, let $v_i = 0$ if the associated vertex belongs to $\mathbb{V}_s$ and $v_j = 1$ for vertex belongs to $\mathbb{V}_t$. Based on these theorems, we have the following conclusions:

Proposition 1 (linear function). *The minimization problem* $v_i^* = \arg\min_{v_i \in \{0,1\}} \{\mathcal{E}(v_i) = a_i v_i\}$, *where* a_i *is a known coefficient, corresponds to finding the min-cut on graph displayed in Fig.1(a).*

Proposition 2 (piecewise linear function). *The minimization problem* $(v_i^*, v_j^*) = \arg\min_{v_i, v_j \in \{0,1\}} \{\mathcal{E}(v_i, v_j) = b_{ij} \max\{v_j - v_i, 0\}\}$, *where* $b_{ij} \geqslant 0$ *is a known coefficient, corresponds to find the min-cut on graph displayed in Fig.1(b).*

These two properties are very important to construct a graph to minimize the energies of PCLSM and CV models. We shall show that both of these two models can be minimized by solving min cuts on graphs which constituted by sum of these two graphs. Since there are some multiplication functions terms in PCLSM and CV, for convenience, we firstly construct a graph to minimize such a term according to the propositions 1 and 2.

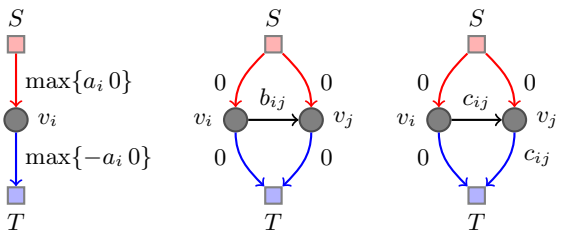
The multiplication function $\mathcal{E}(v_1, \cdots, v_n) = -c \prod_{i=1}^n v_i$, where $0 \leqslant v_i \leqslant 1$ and c is a known coefficient, is non-convex. From convex analysis theory (e.g. [19]), one can get its convex envelope

$$\mathcal{E}^{**}(v_1, \cdots, v_n) = \begin{cases} c \max\{-v_1, -v_2, \cdots, -v_n\}, & c \geqslant 0, \\ -c \max\{\sum_{i=1}^n v_i - n, 0\}, & c < 0, \end{cases}$$

which is the tightest convex function below $\mathcal{E}$. Moreover, in binary case, i.e. $v_i \in \{0, 1\}$, we have $\mathcal{E} = \mathcal{E}^{**}$. Thus, to minimize $\mathcal{E}$ with the binary constraint can be replaced by finding the minimizer of the convex function $\mathcal{E}^{**}$ with the same constraint. This is also the main idea of the convex relaxation method for product label spaces in [20]. When $c \geqslant 0$, in the two variables case, one can get

$$\mathcal{E}^{**}(v_1, v_2) = c \max\{v_2 - v_1, 0\} - c v_2.$$

Thus, based on the theorems 2, 3, and propositions 1, 2, we have the following conclusion:



(a) $a_i v_i$ (b) $b_{ij} \max\{v_j - v_i, 0\}$ (c) $-c_{ij} v_i v_j$

Fig. 1.

Corollary 1 (multiplication function). *The multiplication* $\mathcal{E}(\boldsymbol{v}) = -c_n \prod_{i=1}^{n} v_i, v_i \in \{0,1\}$ *is submodular if and only if the coefficient* $c_n \geqslant 0$. *In particular, the 2 variables multiplication* $\mathcal{E}(v_i, v_j) = -c_{ij} v_i v_j, v_i, v_j \in \{0,1\}$ *is submodular if and only if the coefficient* $c_{ij} \geqslant 0$. *Moreover, if* $c_{ij} \geqslant 0$, *the related binary minimization problem corresponds to find the min-cut on the graph displayed in Fig.1(c).*

3 Graph Construction for PCLSM and CV Models

In this section, we shall use the results in section 2 to construct graphs to solve PCLSM and CV models.

3.1 PCLSM

Now, we use the former method to minimize PCLSM with graph cut.

With the label function representation, the energy of multi-phase segmentation problem can be written as the following modified PCLSM [3]

$$\mathcal{E}^{PCLSM}(l) = \sum_{k=0}^{K-1} \int_{\Omega} \delta_{l,k} d^k \mathrm{d}x + \mu \int_{\Omega} |\nabla l| \mathrm{d}x, \tag{1}$$

Here $l : \Omega \to \{0, 1, \cdots, K-1\}$ is an unknown integer function whose K different values are used to represent the K number of phases, and

$$\delta_{l,k} = \begin{cases} 1, l(x) = k, \\ 0, l(x) \neq k. \end{cases}$$

The first term is the data term which gives the classification criterion, and each d^k should depend on the input image I. For example, $d^k(x) = |I(x) - c^k|^\lambda$, $\lambda = 1, 2$ represents that the pixels are classified in terms of the intensity means $\{c^k\}_{k=1}^K$. In this paper, we suppose $d^k \geqslant 0$ are known. While the second term is the regularization term and μ is a parameter which controls the balance of these two terms. This functional is non-convex because of the existence of composite function of delta function and l. However, with the help of the γ-super-level set function

$$\phi^\gamma(x) = \begin{cases} 1, \text{ when } \ l(x) \geqslant \gamma, \\ 0, \text{ when } \ l(x) < \gamma, \end{cases} \tag{2}$$

we have

$$\phi^k(x) - \phi^{k+1}(x) = \delta_{l(x),k}, \tag{3}$$

for $k = 0, 1, \cdots, K-1$. Together with the generalized co-area formula [21]

$$\int_{\Omega} |\nabla l| \mathrm{d}x = \int_{\Omega} \left(\int_{-\infty}^{+\infty} |\nabla \phi^\gamma(x)| \mathrm{d}\gamma \right) \mathrm{d}x,$$

the functional can be formulated as

$$\mathcal{E}^{PCLSM}(\boldsymbol{\phi}) = \sum_{k=0}^{K-1} \int_{\Omega} (\phi^k - \phi^{k+1}) d^k \mathrm{d}x + \mu \sum_{k=1}^{K-1} \int_{\Omega} |\nabla \phi^k| \mathrm{d}x, \tag{4}$$

where $\boldsymbol{\phi} = (\phi^1, \phi^2, \cdots, \phi^{K-1}), \phi^1 \geqslant \phi^2 \geqslant \cdots \geqslant \phi^{K-1}$, and $\phi^0 = 1, \phi^K = 0$.

The functional (4) is now convex, compared with the original formulation (1). The above process is the functional lifting method (FLM) that has been developed in [10], but the data term is slightly different from [10]. The data term here is linear but L^1 in [10].

In discrete case, (4) can be globally minimized by graph cut. In the next, we shall construct such a graph.

Let $\mathbb{P}$ be the set of mesh grid points in Ω, and $\mathbb{N}_p^4$ be the set of 4 nearest neighbors of $p \in \mathbb{P}$. For $\Omega \subset \mathbb{R}^2$, $\mathbb{P} = \{(i,j) \subset \mathbb{Z}^2\}$ and for each $p = (i,j) \in \mathbb{P}$

$$\mathbb{N}_p^4 = \{(i \pm 1, j), (i, j \pm 1)\},$$

Let ϕ_p^k, d_p^k be the function values of ϕ^k and d^k at $p \in \mathbb{P}$. If we choose the anisotropic TV

$$TV_1(\phi^k) = \int_{\Omega} |\nabla \phi^k|_1 \mathrm{d}x = \int_{\Omega} (|\partial_{x_1} \phi^k| + |\partial_{x_2} \phi^k|) \mathrm{d}x_1 \mathrm{d}x_2$$

for the regularization term in (4), and employ the difference schemes

$$|\partial_{x_1} \phi^k| = \frac{|\partial_{x_1} \phi^k|}{2} + \frac{|\partial_{x_1} \phi^k|}{2} = \frac{|\phi_{i+1,j}^k - \phi_{i,j}^k|}{2} + \frac{|\phi_{i-1,j}^k - \phi_{i,j}^k|}{2},$$
$$|\partial_{x_2} \phi^k| = \frac{|\partial_{x_2} \phi^k|}{2} + \frac{|\partial_{x_2} \phi^k|}{2} = \frac{|\phi_{i,j+1}^k - \phi_{i,j}^k|}{2} + \frac{|\phi_{i,j-1}^k - \phi_{i,j}^k|}{2},$$

then the discrete formulation of (4) should be

$$\mathcal{E}^{PCLSM-D}(\boldsymbol{\phi}) = \sum_{k=0}^{K-1} \sum_{p \in \mathbb{P}} \left(\phi_p^k - \phi_p^{k+1}\right) d_p^k + \frac{\mu}{2} \sum_{k=1}^{K-1} \sum_{p \in \mathbb{P}} \sum_{q \in \mathbb{N}_p^4} |\phi_q^k - \phi_p^k|.$$

From propositon 1, $\phi_p^k d_p^k, -\phi_p^{k+1} d_p^k$ can be minimized by solving the min-cuts on graphs displayed in Fig.2(a) and 2(b), respectively. According to the additivity theorem 3 and combining the constraint condition $\phi^1 \geqslant \cdots \geqslant \phi^{K-1}$, the data term at each pixel $p \in \mathbb{P}$ in the above energy can be minimized through searching the min-cut on graph defined in Fig.2(c). For a cut $(\mathbb{V}_s, \mathbb{V}_t)$, we say $(\mathbb{V}_s, \mathbb{V}_t)$ is a feasible cut when the cost of the cut $C(\mathbb{V}_s, \mathbb{V}_t) < +\infty$. Besides, let us denote

$$\mathbb{B} = \{\boldsymbol{\phi} = (\phi^1, \phi^2, \cdots, \phi^{K-1}) : \phi^k \in \{0,1\}, 1 = \phi^0 \geqslant \phi^1 \geqslant \phi^2 \geqslant \cdots \geqslant \phi^{K-1} \geqslant \phi^K = 0\}. \tag{5}$$

Then we have the following result:

Proposition 3. *There is a one-to-one correspondence between the feasible cuts of graph $\mathbb{G} = (\mathbb{V}, \mathbb{E})$ defined on Fig.2(c) and the binary super-level set function $\boldsymbol{\phi}$, and*

$$\min_{(\mathbb{V}_s, \mathbb{V}_t)} C(\mathbb{V}_s, \mathbb{V}_t) = \min_{\boldsymbol{\phi} \in \mathbb{B}} \sum_{k=0}^{K-1} \sum_{p \in \mathbb{P}} \left(\phi_p^k - \phi_p^{k+1}\right) d_p^k.$$

As to the regularization term, since ϕ_p^k, ϕ_q^k are binary and we have

$$\frac{\mu}{2}\sum_{k=1}^{K-1}\sum_{p\in\mathbb{P}}\sum_{q\in\mathbb{N}_p^4}|\phi_q^k-\phi_p^k| = \frac{\mu}{2}\sum_{k=1}^{K-1}\sum_{p\in\mathbb{P}}\sum_{q\in\mathbb{N}_p^4}|\phi_q^k-\phi_p^k|^2 = \frac{\mu}{2}\sum_{k=1}^{K-1}\sum_{p\in\mathbb{P}}\sum_{q\in\mathbb{N}_p^4}(\phi_q^k+\phi_p^k-2\phi_q^k\phi_p^k).$$

By applying the additivity theorem 3 and corollary 1, the regularization term can be minimized by solving the min-cut on the graph displayed in Fig.3.

Finally, we can get the graph for minimizing the energy $\mathcal{E}^{PCLSM-D}$ by simply adding the edge weights of graphs defined in Fig.2(c) and Fig.3(b) together according to the additivity of the graph.

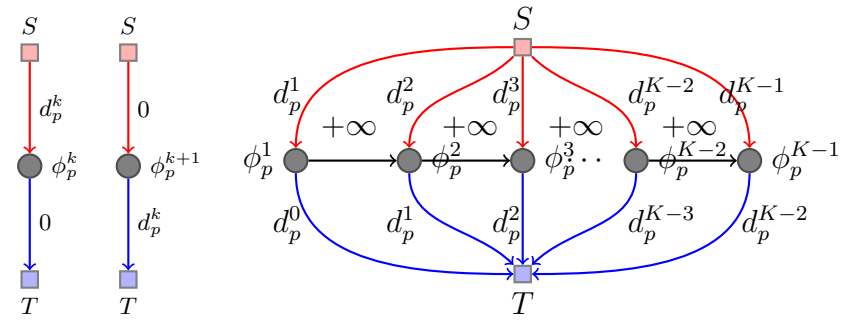


Fig. 2. Graph construction for data term at each pixel $p \in \mathbb{P}$ in PCLSM

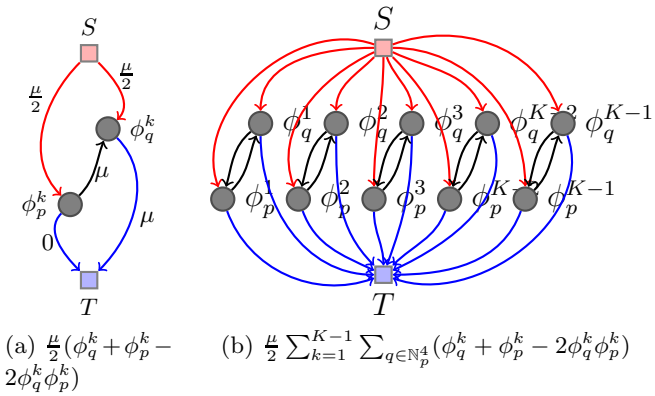


Fig. 3. Graph construction for regularization term at each pixel $p \in \mathbb{P}$ in PCLSM. In Fig.3(b), all the weights of red and blue edges (t-link) are 4μ, and all the weights of black edges (n-link) are μ.

3.2 CV Model

The Chan-Vese model [22] is a popular image segmentation model, a generalization of Chan-Vese model has been proposed in [2] to partition an image into K parts by using $\log_2 K$ level set functions. In [4], vector binary functions were used to represent the different phases and this has a very close relationship to the things we are going to discuss here. For the general CV model with any number of phases, we firstly derive a general formulation for multi-phase CV model from the binary representation of integers. It is well known that for any integer $k \in \{0, 1, \cdots, K-1\}$, there is a mapping $\Lambda : \mathbb{Z}^+ \cup \{0\} \to \mathbb{B}$ such that $\Lambda(k) = b_k^{M-1} \cdots b_k^1 b_k^0$, where $\mathbb{B}$ is the set of the binary representation of integers. Here $b_k^m \in \{0, 1\}, m = 0, \cdots, M-1$, $M = \lceil \log_2 K \rceil$, and $\lceil \cdot \rceil$ is a ceiling operator. For example, the binary representation of integer $\Lambda(5) = 101$. In essence, the CV model is closely related to this binary representation. For simplification, we first consider the $K = 2^M, M = 1, 2 \cdots$ case. Let us write the binary representation of the label function $l : \Omega \to \{0, 1, \cdots, 2^M - 1\}$ as $\Lambda(l) = \psi^{M-1} \cdots \psi^0$, and in this case it is easy to check Λ is a 1-1 mapping, then

$$\delta_{l,k} = \delta_{\Lambda(l),\Lambda(k)} = \delta_{\psi^{M-1}\cdots\psi^0, b_k^{M-1}\cdots b_k^0} = \prod_{m=0}^{M-1} \delta_{\psi^m, b_k^m} = \begin{cases} 1, \psi^m = b_k^m, \\ 0, else. \end{cases}$$

Using the γ-super-level set function representation for ψ^m and let us denote

$$\phi^{m,b_k^m}(x) = \begin{cases} 1, \text{when } \ \psi^m(x) \geqslant b_k^m, \\ 0, \text{when } \ \psi^m(x) < b_k^m, \end{cases}$$

for all $m = 0, 1, \cdots, M-1$. Now, we have

$$\phi^{m,b_k^m} - \phi^{m,b_k^m+1} = \delta_{\psi^m, b_k^m},$$

and thus

$$\delta_{l,k} = \prod_{m=0}^{M-1} (\phi^{m,b_k^m} - \phi^{m,b_k^m+1}).$$

In the above equation,

$$\phi^{m,0} = 1, \phi^{m,1} = \psi^m, \phi^{m,2} = 0,$$

since $b_k^m, \psi^m \in \{0, 1\}$, so the unknown variables are only $\boldsymbol{\phi}^{\cdot,1} = (\phi^{0,1}, \phi^{1,1}, \cdots, \phi^{M-1,1})$.

If the segmentation phase $K \neq 2^M$, i.e. $2^{M-1} < K < 2^M$, then Λ may not be a 1-1 mapping, which means that we may use serval labels to indicate one class. This can be achieved by adding several d^{K-1} in the energy.

Replacing the δ function in (4) with the above expression and modifying the regularization term, we get the multi-phase CV model for any K phases:

$$\mathcal{E}^{CV-K}(\boldsymbol{\phi}^{\cdot,1}) = \sum_{k=0}^{2^M-1} \int_\Omega d^k \prod_{m=0}^{M-1} (\phi^{m,b_k^m} - \phi^{m,b_k^m+1}) \mathrm{d}x + \mu \sum_{m=0}^{M-1} \int_\Omega |\nabla \phi^{m,1}| \mathrm{d}x, \tag{6}$$

where $d^k = d^{K-1}$ when $K-1 \leqslant k \leqslant 2^M-1$. Let point out that the choice of d^k when $K-1 \leqslant k \leqslant 2^M-1$ is not unique, for example, one can use $d^k = \frac{\sum_{k=0}^{K-1} d^k}{K}$ for $k > K-1$.

The discrete formulation of the regularization term in CV model (6) is

$$\mathcal{R}^{CV-K}(\boldsymbol{\phi}) = \frac{1}{2}\sum_{m=0}^{M-1}\sum_{p\in\mathbb{P}}\sum_{q\in\mathbb{N}_p^4}|\phi_q^m - \phi_p^m|,$$

which can be minimized by graph cut in terms of our earlier discussion.

Let $i_0 < i_1 < \cdots < i_{m-1}$ be any m-combination of the set $\{0,1,\cdots,M-1\}$ and $\mathbb{S}^m$ be a set which contains the C_M^m different m-combinations. For simplicity, we rewrite $\phi^{m,1}$ as ϕ^m. In the discrete case, the data term in CV model (6) can be expressed as

$$\mathcal{D}^{CV-K}(\boldsymbol{\phi}) = \sum_{p\in\mathbb{P}}\left(d_p^0 + \sum_{m=1}^{M}\sum_{i_0,i_1,\cdots,i_{m-1}\in\mathbb{S}^m}\sum_{t_0=0}^{1}\sum_{t_1=0}^{1}\cdots\sum_{t_{m-1}=0}^{1} c_{t_{m-1},\cdots,t_0}^{i_{m-1},\cdots,i_0}\phi_p^{i_0}\phi_p^{i_1}\cdots\phi_p^{i_{m-1}}\right),$$

where the coefficients

$$c_{t_{m-1},\cdots,t_0}^{i_{m-1},\cdots,i_0} = (-1)^{\left(m-\sum_{j=0}^{m-1}t_j\right)} d_p^{\sum_{j=0}^{m-1}t_j 2^{i_j}}.$$

Now, the data term $\mathcal{D}^{CV-K}$ only contains some multiplying functions, we can use the former theorems. According to proposition 1, corollary 1 and theorem 3, we can get the following result

Proposition 4. *The K-phases discrete CV model can be exactly minimized by graph cut if the coefficients $\sum_{t_0=0}^{1}\cdots\sum_{t_{m-1}=0}^{1} c_{t_{m-1},\cdots,t_0}^{i_{m-1},\cdots,i_0} \leqslant 0$ for all $2 \leqslant m \leqslant M$.*

In particular, when $K=3$, $M = \lceil \log_2 3 \rceil = 2$, then

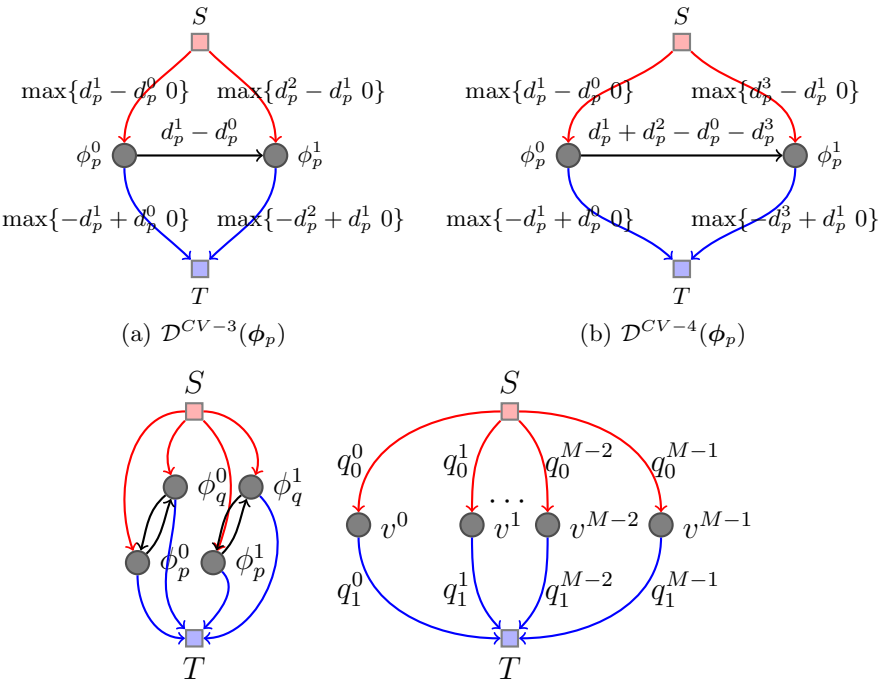
$$\begin{aligned}\mathcal{D}^{CV-3}(\boldsymbol{\phi}) &= \textstyle\sum_{p\in\mathbb{P}}\left(d_p^0 + (c_0^0+c_1^0)\phi_p^0 + (c_0^1+c_1^1)\phi_p^1 + (c_{0,0}^{1,0}+c_{0,1}^{1,0}+c_{1,0}^{1,0}+c_{1,1}^{1,0})\phi_p^0\phi_p^1\right)\\ &= \textstyle\sum_{p\in\mathbb{P}}\left(d_p^0 + (-d_p^0+d_p^1)\phi_p^0 + (-d_p^0+d_p^2)\phi_p^1 + (d_p^0-d_p^1)\phi_p^0\phi_p^1\right).\end{aligned}$$

Similarly, for $K=4$,

$$\begin{aligned}\mathcal{D}^{CV-4}(\boldsymbol{\phi}) &= \textstyle\sum_{p\in\mathbb{P}}\left(d_p^0 + (c_0^0+c_1^0)\phi_p^0 + (c_0^1+c_1^1)\phi_p^1 + (c_{0,0}^{1,0}+c_{0,1}^{1,0}+c_{1,0}^{1,0}+c_{1,1}^{1,0})\phi_p^0\phi_p^1\right)\\ &= \textstyle\sum_{p\in\mathbb{P}}\left(d_p^0 + (-d_p^0+d_p^1)\phi_p^0 + (-d_p^0+d_p^2)\phi_p^1 + (d_p^0-d_p^1-d_p^2+d_p^3)\phi_p^0\phi_p^1\right).\end{aligned}$$

Thus we have the following conclusion:

Corollary 2. *The 3-phase discrete CV model can be exactly minimized by graph cut if and only if $d^0 - d^1 \leqslant 0$. Similarly, the condition of the 4-phase discrete CV model is $d^0 - d^1 - d^2 + d^3 \leqslant 0$. When these condition holds, for each pixel p, $\mathcal{D}^{CV-3}(\boldsymbol{\phi}_p), \mathcal{D}^{CV-4}(\boldsymbol{\phi}_p), \mu\mathcal{R}^{CV-3\ or\ 4}(\boldsymbol{\phi}_p)$ can be minimized by finding the min cuts on graphs defined on Fig.4(a), Fig.4(b) and Fig.4(c), respectively.*



(a) $\mathcal{D}^{CV-3}(\boldsymbol{\phi}_p)$ (b) $\mathcal{D}^{CV-4}(\boldsymbol{\phi}_p)$

(c) $\mu\mathcal{R}^{CV-3\ or\ 4}(\boldsymbol{\phi}_p)$ (d) The continuous relaxation max-flow for CV (data term).

Fig. 4. Graph construction for 3,4-phase CV model. In Fig.4(c), all the weights of red and blue edges (t-link) are 4μ, and all the weights of black edges (n-link) are μ.

Therefore, to minimize the 3 or 4-phase discrete model can be implemented by solving the min cut of the graph which is constituted by putting the graphs defined by data term and regularization term together.

In 4-phase case, the corollary 2 coincides to result in [1, 8]. Here, we extend the results to any phases CV model, which is not easy to handle with the method of [1, 8].

In fact, when the condition $\sum_{t_0=0}^{1} \cdots \sum_{t_{m-1}=0}^{1} c_{t_{m-1},\cdots,t_0}^{i_{m-1},\cdots,i_0} \leqslant 0$ in proposition 4 holds, the non-convex term $-\phi_p^{i_0}\phi_p^{i_1}\cdots\phi_p^{i_{m-1}}$ can be replaced by its convex envelope $\max\{-\phi^{i_0}, -\phi^{i_1}, \cdots, -\phi^{i_{m-1}}\}$ since these two functionals have the same minimizer in the binary case. As in [1], we call this condition as convex condition. To get a convex model, one can relax the non-convex constraint $\phi^i \in \{0, 1\}$ to an interval $[0, 1]$. In the continuous case, we can get a convex model for K-phase CV model.

As for the convex condition in proposition 4, it depends on the segmentation data. In the 4-phase case, it have been theoretically analyzed in [8]. In many real image segmentation problems, it may hold that 4 cluster are sufficient. However, we note that this condition would become stricter when the number of the phases

increases, thus in the discrete case, the K-phase CV model may not be exactly optimized by graph cut if the segmentation data is very bad (fails to satisfy the convex condition). To overcome this flaw, one may use an approximation graph cut method by cutting off the coefficient $\sum_{t_0=0}^{1}\cdots\sum_{t_{m-1}=0}^{1} c_{t_{m-1},\cdots,t_0}^{i_{m-1},\cdots,i_0}$ as 0 when the condition fails. However, such an approximation method may not be the original CV model. Here, we shall propose another continuous relaxation max-flow method, which is also a convex model but without any convex conditions. Refer to [18] in this proceedings for another method to derive a convex relaxation for the multiphase CV model. The relaxation of [18] is tight for any phases, but need conditions to guarantee convexity.

4 Continuous Convex Relaxation Max-Flow for CV

The continuous max-flow method for 2-phase CV model and Potts model has been proposed by Yuan *etc.* in [15, 23]. It has been extended to 4-phase CV model by Bae-Tai in [1] with the earlier mentioned convex condition. Here, we propose a convex continuous max-flow method for any phases CV model but without any convex conditions.

For multi-phase CV model, the regularization term is convex and to convert it to continuous max-flow would be the same as [1,15,23,24]. Here we only discuss the data term.

Firstly, we construct a graph displayed in Fig.4(d): for K phases CV model, we copy $M = \lceil \log_2 K \rceil$ vertices $v^m, m = 0, \cdots, M-1$ at each pixel x; let us denote the edges between source s and vertex v^m, sink t and vertex v^m as q_0^m and q_1^m, respectively. For $k = 0, 1, \cdots, K-1$, let the binary representation of k be $\Lambda(k) = b_k^{M-1}\cdots b_k^0$. We impose the capabilities of these edges satisfy the following condition

$$\sum_{m=0}^{M-1} q_{b_k^m}^m \leqslant d^k, k = 0, 1\cdots, 2^M - 1, \tag{7}$$

where $d^k = d^{K-1}$ when $K - 1 \leqslant k \leqslant 2^M - 1$.

The max-flow problem is to find the maximum capabilities of the flows which stream from source s to sink t under a flow preserving condition and the maximum capabilities condition (7). Together with the graph defined in Fig.4(d) and the regularization term, the proposed max-flow problem for any K-phase CV model can be written as

$$\begin{cases} \max\limits_{\boldsymbol{q}_0,\boldsymbol{q}_1,\boldsymbol{p}\in\mathbb{C}} \int_\Omega \sum\limits_{m=0}^{M-1} q_1^m(x)\mathrm{d}x, \\ q_0^m(x) - q_1^m(x) - \nabla\cdot\boldsymbol{p}^m(x) = 0, m = 0, 1, \cdots, M-1, \\ \sum_{m=0}^{M-1} q_{b_k^m}^m(x) \leqslant d^k(x), k = 0, 1\cdots, 2^M - 1. \end{cases} \tag{8}$$

Here, $\boldsymbol{q}_0 = (q_0^0, \cdots, q_0^{M-1}), \boldsymbol{q}_1 = (q_1^0, \cdots, q_1^{M-1})$, the second equation is the flow preserving condition of vertex v^m, in which $\boldsymbol{p} = (\boldsymbol{p}^0, \cdots, \boldsymbol{p}^{M-1}) \in \mathbb{C} = \{\boldsymbol{p} : ||\boldsymbol{p}^m||_\infty \leqslant \mu, m = 0, 1, \cdots, M-1.\}$ is the associated flow function for the TV term in CV model, and the third inequality is just the maximum capabilities condition (7).

Applying the Lagrange multiplier method and let $\boldsymbol{\phi} = (\phi^0, \cdots, \phi^{M-1})$ be the M Lagrange multipliers, then the problem (8) can be formulated as the following saddle point problem:

$$\begin{cases} \min\limits_{\boldsymbol{\phi}} \max\limits_{\boldsymbol{q}_0, \boldsymbol{q}_1, \boldsymbol{p} \in \mathbb{C}} \int_\Omega \sum\limits_{m=0}^{M-1} (\phi^m(x) q_0^m(x) + (1 - \phi^m(x)) q_1^m(x) - \phi^m(x) \nabla \cdot \boldsymbol{p}^m(x)) \, \mathrm{d}x, \\ \sum_{m=0}^{M-1} q_{b_k^m}^m(x) \leqslant d^k(x), k = 0, 1 \cdots, 2^M - 1. \end{cases} \tag{9}$$

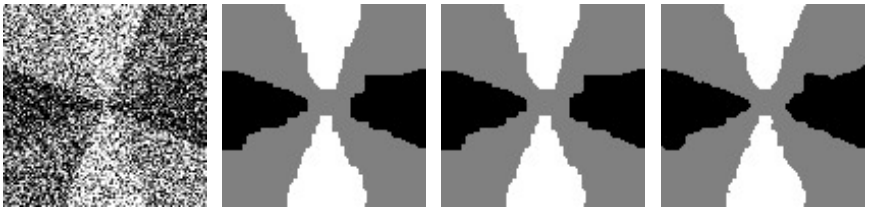
In the above saddle point problem, the term $\mathcal{J}(\boldsymbol{\phi}) = \max\limits_{\boldsymbol{q}_0, \boldsymbol{q}_1, \boldsymbol{p} \in \mathbb{C}} \int_\Omega \sum\limits_{m=0}^{M-1} \phi^m(x) q_0^m(x) + (1 - \phi^m(x)) q_1^m(x) \mathrm{d}x$ with the constraint condition (7) is associated to the data term $\mathcal{D}^{CV-K}(\boldsymbol{\phi})$ in CV model (6). For this two terms, we can prove that they have the same global minimization.

Proposition 5. *If $\boldsymbol{\phi}^*$ is a binary minimizer of $\mathcal{D}^{CV-K}(\boldsymbol{\phi})$, then $\boldsymbol{\phi}^*$ is also a minimizer of $\mathcal{J}(\boldsymbol{\phi})$ under constraint (7). Moreover, $\mathcal{D}^{CV-K}(\boldsymbol{\phi}^*) = \mathcal{J}(\boldsymbol{\phi}^*) = \int_\Omega \min\{d^0(x), \cdots, d^{2^M-1}(x)\} \mathrm{d}x$.*

As to find the saddle point of (9), it can be done by projection gradient method since the functional is linear and the constraint sets are convex for each variable. We do not plan to list the details of the algorithm here.

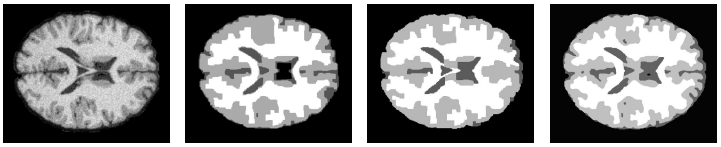
5 Experimental Results

In this section, we shall give some numerical examples with the proposed method. The first one is to partition a synthetic image in Fig. 5(a) into 3 parts. The PCLSM, CV model and continuous convex relaxation max-flow (CCRM) of CV are applied to segment this image. For the PCLSM and CV, we use the constructed graph cut to solve them. The final segmentation results are shown in Fig.5(b)-Fig.5(d), respectively. As can seen from the figure, the result produced by CCRM is smoother than the discrete methods. This is due to the fact that we may employ an isotropic TV in the continuous case but not with graph cuts. At the current CPU program implementation, the discrete graph cut method is faster than CCRM. However, the discrete graph cut algorithm is not easy to work with the parallel program, thus the continuous method CCRM with GPU implementation would be faster than the discrete ones. The second example displayed in Fig. 6 shows the segmentation results of applying the proposed method to a real brain MRI. In this figure, the brain image is partition into 4 parts. As can be found again, the boundary in result of CCRM is smoother than others.



(a) Size: 100×100 (b) PCLSM (c) CV (d) CCRM

Fig. 5. A comparison between discrete graph cut algorithm and continuous convex relaxation max-flow (CCRM). The cost of the CPU time for PCLSM, CV and CCRM are 0.0352s,0.3504s,0.4282s, respectively.



(a) Size591×709 (b) PCLSM (c) CV (d) CCRM

Fig. 6. Different algorithms for real images. The regularization parameters for each method are manually chosen. The clusters is 4.

6 Conclusion and Discussion

We have proposed a unified method to minimize the multiphase image segmentation PCLSM and CV models with discrete graph cut and continuous max-flow. This method is developed by considering the convexification of the multiplying functions with graph cut method. For PCLSM, we construct a graph which is different from the earlier Ishikawa method [7]. For CV models, we extend the result in [1, 6] to any phases and propose a saddle point problem for any phases CV model. Compared the original CV method, the proposed convex relaxation max-flow method is convex and thus it can get the global minimization. Moreover, we show that the key idea of CV model is to use the binary expression to represent a integer, thus one can extend it to any n-decimal numeral system .

Acknowledgements. Liu was supported in part by National Natural Science Foundation of China (No. 11201032), and Leung was supported in part by the HKUST grant RPC11SC06.

References

1. Bae, E., Tai, X.C.: Efficient global minimization methods for image segmentation models with four regions. Technical Report 11-82, UCLA CAM Report (2011)
2. Vese, L.A., Chan, T.F.: A multiphase level set framework for image segmentation using the mumford and shah model. International Journal of Computer Vision 50(3), 271–293 (2002)

3. Lie, J., Lysaker, M., Tai, X.C.: A variant of the level set method and applications to image segmentation. Mathematics of Computation 75(255), 1155–1174 (2006)
4. Lie, J., Lysaker, M., Tai, X.C.: A binary level set model and some applications to mumford-shah image segmentation. IEEE Transactions on Image Processing 15(5), 1171–1181 (2006)
5. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via grah cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(2), 147–159 (2004)
6. Bae, E., Tai, X.C.: Graph cut optimization for the piecewise constant level set method applied to multiphase image segmentation. In: Scale Space and Variational Mehotds in Computer Vision, pp. 1–13 (2009)
7. Ishikawa, H.: Exact optimization for markov random fields with convex priors. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(10), 1333–1336 (2003)
8. Bae, E., Tai, X.-C.: Efficient global minimization for the multiphase chan-vese model of image segmentation. In: Cremers, D., Boykov, Y., Blake, A., Schmidt, F.R. (eds.) EMMCVPR 2009. LNCS, vol. 5681, pp. 28–41. Springer, Heidelberg (2009)
9. Chan, T.F., Esedoglu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM Journal on Applied Mathematics 66(5), 1632–1648 (2006)
10. Pock, T., Schoenemann, T., Graber, G., Bischof, H., Cremers, D.: A convex formulation of continuous multi-label problems. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 792–805. Springer, Heidelberg (2008)
11. Zach, C., Gallup, D., Frahm, J.M., Niethammer, M.: Fast global labeling for real-time stereo using multiple plane sweeps. In: Vision, Modeling and Visualization Workshop, VMV (2008)
12. Pock, T., Chambolle, A., Cremers, D., Bischof, H.: A convex relaxation approach for computing minimal partitions. In: IEEE CVPR 2009 (2009)
13. Lellmann, J., Kappes, J., Yuan, J., Becker, F., Schnörr, C.: Convex multi-class image labeling by simplex-constrained total variation. In: Tai, X.-C., Mørken, K., Lysaker, M., Lie, K.-A. (eds.) SSVM 2009. LNCS, vol. 5567, pp. 150–162. Springer, Heidelberg (2009)
14. Bae, E., Yuan, J., Tai, X.C.: Global minimization for continuous multiphase partitioning problems using a dual approach. International Journal of Computer Vision 92(1), 112–129 (2011)
15. Yuan, J., Bae, E., Tai, X.-C., Boykov, Y.: A continuous max-flow approach to potts model. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part VI. LNCS, vol. 6316, pp. 379–392. Springer, Heidelberg (2010)
16. Brown, E.S., Chan, T.F., Bresson, X.: Completely convex formulation of the chan-vese image segmentation model. International Journal of Computer Vision 98, 103–121 (2012)
17. Claudia, N., Eno, T., Daniel, C.: A survey and comparison of discrete and continuous multi-label optimization approaches for the potts model. International Journal of Computer Vision, 1–18 (2013)
18. Bae, E., Lellmann, J., Tai, X.-C.: Convex relaxations for a generalized chan-vese model. In: Heyden, A., Kahl, F., Olsson, C., Oskarsson, M., Tai, X.-C., Chairs, P. (eds.) EMMCVPR 2013. LNCS, vol. 8081, pp. 223–236. Springer, Heidelberg (2013)

19. Ekeland, I., Temam, R.: Convex Analysis and Variational Problems. SIAM Classics in Applied Mathematics. SIAM, Philadelphia (1999)
20. Goldluecke, B., Cremers, D.: Convex relaxation for multilabel problems with product label spaces. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 225–238. Springer, Heidelberg (2010)
21. Fleming, W., Rishel, R.: An integral formula for total gradient variation. Archiv der Mathematik 11, 218–222 (1960)
22. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Transactions on Image Processing 10(2), 266–277 (2001)
23. Yuan, J., Bae, E., Tai, X.C.: A study on continuous max-flow and min-cut approaches. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, USA, pp. 2217–2224 (2010)
24. Liu, J., Tai, X.C., Leung, S., Huang, H.: A new continuous max-flow algorithm for multiphase image segmentation using super-level set functions. Technical report, UCLA report 12-81 (2012)

Segmenting Planar Superpixel Adjacency Graphs w.r.t. Non-planar Superpixel Affinity Graphs

Bjoern Andres[1,⋆], Julian Yarkony[2,⋆], B.S. Manjunath[2], Steffen Kirchhoff[1], Engin Turetken[3], Charless C. Fowlkes[4], and Hanspeter Pfister[1]

[1] Harvard University
[2] UC Santa Barbara
[3] EPFL
[4] UC Irvine

Abstract. We address the problem of segmenting an image into a previously unknown number of segments from the perspective of graph partitioning. Specifically, we consider minimum multicuts of superpixel affinity graphs in which all affinities between non-adjacent superpixels are negative. We propose a relaxation by Lagrangian decomposition and a constrained set of re-parameterizations for which we can optimize exactly and efficiently. Our contribution is to show how the planarity of the adjacency graph can be exploited if the affinity graph is non-planar. We demonstrate the effectiveness of this approach in user-assisted image segmentation and show that the solution of the relaxed problem is fast and the relaxation is tight in practice.

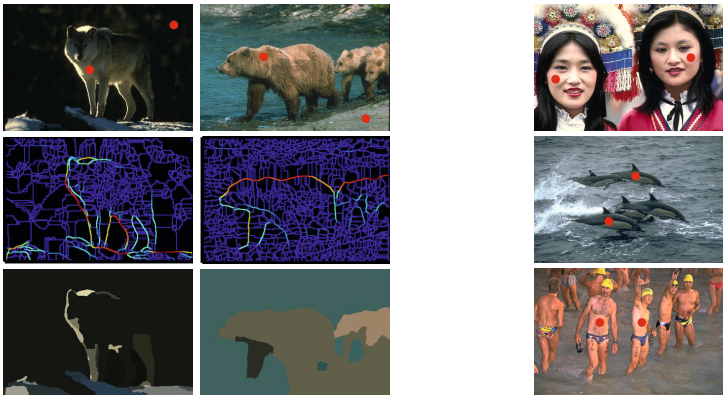
1 Introduction

The formalization of the image segmentation problem as a multicut problem has recently attracted considerable attention [2,3,6,9,10,15,17]. This problem consists in finding a partition of a weighted superpixel adjacency graph into connected components (segments) such that the set of edges that straddle different segments (the multicut) has minimum total weight. A positive edge weight penalizes and a negative edge weight rewards all segmentations in which the connected superpixels are in different segments. The weights depend on the image in a way this is typically learned from data. They are called an *affinity*.

With the notable exception of [10], recent work has focused on the problem described so far in which affinities are only defined for pairs of *adjacent* superpixels. This has been motivated by three reasons. First and primary is that affinities between adjacent superpixels can be estimated reasonably well with methods such as the global probability of boundary [4]. Second is that the number of affinities grows linearly in the number of superpixels which is beneficial for modeling and optimization. Third is that superpixel adjacency graphs are *planar* which affords a relaxation of the multicut problem that can be solved efficiently and is tight in practice [17], although the multicut problem for planar graphs remains NP-hard [5].

⋆ Authors contributed equally.

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 266–279, 2013.
© Springer-Verlag Berlin Heidelberg 2013



(a) (b)

Fig. 1. a) In natural images, we often observe pixels at different locations that clearly belong to different segments but no clear position where the segment boundary should be located. This occurs whenever there is a smooth transition from the color of one region to another color of another region. In this case, affinities between adjacent superpixels do not provide strong cues which leads to errors in the segmentation. We show in Section 5 that negative affinities between non-adjacenct superpixels can overcome this problem. **b)** At the same time, *positive* affinities between non-adjacent superpixels do not convey strong cues because separate segments can have similar texture.

In this paper, we study the generalization of the problem that allows for negative affinities between non-adjacent superpixels. The objective is to find a segmentation of the superpixel adjacency graph such that the sum of affinities between superpixels in different segments in minimal. We propose a relaxation by Lagrangian decomposition and a constrained set of re-parameterizations for which we can optimize exactly and efficiently. We demonstrate the effectiveness of this approach in user-assisted image segmentation and show that the solution of the relaxed problem is fast and the relaxation is tight in practice.

Our concentration on *negative* non-local affinities is motivated by the following observations: In natural images, we often observe pixels at different locations that clearly belong to different segments but no clear position where the segment boundary should be located. This occurs whenever there is a smooth transition from the color of one region to another color of another region (Fig. 1a). In this case, affinities between adjacent superpixels do not provide strong cues which leads to errors in the segmentation. We show in Section 5 that negative affinities between non-adjacenct superpixels can overcome this problem. At the same time, *positive* affinities between non-adjacent superpixels do not convey strong cues because separate segments can have similar texture (Fig. 1b). A conceptual difficulty of non-local positive affinities is discussed in Appendix A.1.

2 Related Work

The multicut problem is known to be NP-complete, [7], even for planar graphs [5]. In computer vision, this combinatorial optimization problem has been used

to formalize image segmentation. To date, multicuts of superpixel adjacency graphs [2,17] are among the closest, in terms of partition metrics, to the man-made segmentations in the Berkeley Segmentation Benchmark [4].

Recent work on the multicut problem for computer vision application has afforded an exact cutting plane algorithm for general graph that is applicable if edge weights are strong [3], efficient greedy algorithms suitable for large problems and problems where edge weights are weak [6], as well as diverse linear programming (LP) relaxations [9,10,12,15,17]. The Lagrangian decomposition we propose in this paper is built on [17] which considers optimal multicuts of planar graphs. Our decomposition extends [17] by allowing for the introduction of negative affinities between non-adjacent superpixels.

3 Image Segmentation by Multicuts

We now introduce the mathematical framework for our discussion. We consider a graph (V, E) in which vertices indicate superpixels and weighted edges quantify an affinity between superpixels. A partition of the node set is encoded as a labeling $x \in \{0,1\}^E$ of edges. We use $x_e = 0$ to indicate that the pair of vertices connected by edge e are in the same segment and use $x_e = 1$ to indicate that the pair are in separate segments. We call edge e *uncut* if $x_e = 0$ and *cut* if $x_e = 1$. The multicut of a partition is the set of all edges that are cut.

Edge weights $\theta \in \mathbb{R}^E$ define an objective function over partitions. Here, θ_e is called the affinity of the superpixels connected by the edge e. Negative values indicate a reward for superpixels to be in separate segments and positive values indicate a cost for superpixels to be in separate segments. A partition that minimizes the sum of the affinities of cut edges is called a minimum partition. The corresponding multicut is called a minimum multicut. We write the objective value of a multicut (indicated by) x as the inner product $\theta^T x$.

Not all binary labelings x correspond to partitions. Consider a graph of three superpixels a, b, c which are connected by edges ab, ac, bc. Consider also the labeling $x_{ab} = 1, x_{ac} = 0, x_{bc} = 0$. This labeling states that superpixels a and b are in separate segments while saying that both a and b are in the same segment as c, which is contradiction.

A necessary and sufficient condition for a labeling to define a partition is that there are no edges e such that $x_e = 1$ within a connected component [8]. This condition can be written as a set of inequalities as follows: For every cycle of edges c and every edge f in c, if $x_f = 1$, then at least one other edge in the cycle is cut. These inequalities are called the *cycle inequalities*. The multicut problem can be written as an integer program with cycle inequalities.

$$\min_{x \in \mathbb{R}^E} \quad \theta^T x \tag{1}$$

$$\text{subject to} \sum_{e \in c \setminus f} x_e \geq x_f \qquad \forall c \in \text{cycles}(V, E)\ \forall f \in c \tag{2}$$

$$x_e \in \{0,1\} \qquad \forall e \in E \ .$$

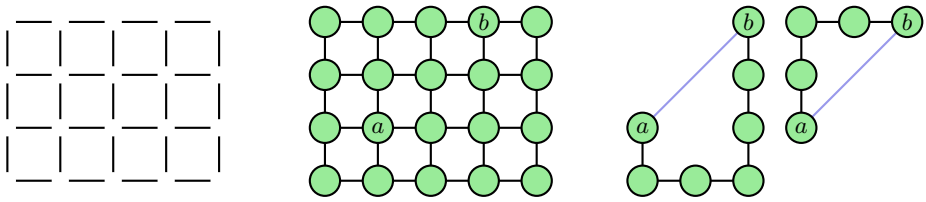


Fig. 2. We propose a Lagrangian decomposition of the multicut problem that consists of the following sub-problems: 1. the independent edge sub-problem, 2. the planar multicut sub-problem, 3. a set of cycle sub-problems.

4 Lagrangian Decomposition

We now propose a Lagrangian decomposition of the multicut problem that consists of the sub-problems depicted in Fig. 2. In the independent edge sub-problem, edges can be cut or uncut independently. In the planar multicut sub-problem, any solution must be a partition of the vertices and thus obey the cycle inequalities. Affinities between non-adjacent superpixels are not included in the planar multicut sub-problem. In a cycle sub-problem an edge between non-adjacent superpixels is associated with a path between the corresponding vertices of that edge. A cycle sub-problem enforces that the number of edges cut in that cycle is not equal to one, thus enforcing the cycle inequalities for that cycle.

We use θ^{ie}, θ^{pmc}, and θ^{c} to denote the affinities of the independent edge, planar multicut and the c-th cycle sub-problem, respectively. The corresponding solutions are x^{ie}, x^{pmc}, x^{c}. For short hand, we refer to edges between adjacent superpixels as planar edges and edges between non-adjacent superpixels as non-planar edges. We use θ^{NP} to denote the affinities of the non-planar edges and θ^{P} to denote the affinities of planar edges. We index non-planar edges with $\hat{e}$ and planar edges with e. Given the sub-problems, we write the Lagrangian decomposition below.

$$\max_{[\theta^{ie},\theta^{pmc},\theta^{c}\ \forall c]} \min_{x^{ie}} (\theta^{ie})^T x^{ie} + \min_{x^{pmc}} (\theta^{pmc})^T x^{pmc} + \sum_{c\in C} \min_{x^{c}} (\theta^{c})^T x^{c} \quad (3)$$

subject to

$$\theta^{ie} + \theta^{pmc} + \sum_{c\in C} \theta^{c} = \theta \quad (4)$$

4.1 The Planar Multicut Sub-problem

Each of the sub-problems is tractable, except for the planar multicut sub-problem which is NP-hard [5]. To make this sub-problem tractable, we constrain the re-parameterization. Specifically, we enforce that the minimal objective value of

the planar multicut sub-problem is zero[1]. Consequently, the empty multicut is optimal and thus, optimization is tractable. In order for the objective value of the planar multicut sub-problem to be zero, it is sufficient to enforce that every 2-colorable segmentation[2] has a non-negative value [17].

We introduce a matrix Z in which each row corresponds to a 2-colorable segmentation. We index 2-colorable segmentations by r. For every index r and every edge e, $Z_{re} = 1$ indicates that e is cut in the r-th 2-colorable segmentation. The constraint that every 2-colorable partition has a non-negative value is formalized below.

$$Z\theta^{pmc} \geq 0 \tag{5}$$

4.2 The Independent Edge Sub-problem

The affinities θ^{ie} in the independent edge sub-problem have important properties. First, we establish $\theta^{ie} \leq 0$ for every $e \in E$. Suppose there exists an edge e such that $\theta^{ie}_e > 0$. Then, $x^{ie}_e = 0$ in any optimal solution of the independent edge sub-problem. Notice that setting $\theta^{pmc}_e := \theta^{pmc}_e + \theta^{ie}_e$ and $\theta^{ie}_e := 0$ does not loosen the bound of Lagrangian decomposition and does not decrease the optimal value of the planar multicut sub-problem. Thus, we can restrict the re-parameterization to allocate non-positive affinity to every edge in the independent edge sub-problem. Furthermore, this means we can take the value of the independent edge sub-problem to be the sum of the terms in θ^{ie} as cutting every edge in the independent edge sub-problem is an optimal solution.

Another important property is that the affinity in the independent edge sub-problem for a given edge e is lower bounded by $\min(\theta_e, 0)$. This property is established in Appendix A.2

4.3 The Cycle Sub-problems

The affinities θ^c in the cycle sub-problem have two importrant properties at the optimal re-parameterization. These properties are established in Appendix A.3.

The first property is that the values of the affinities of the planar edges in a given cycle sub-problem are exactly opposite of the value of the affinity of the corresponding non-planar edge. Thus, the affinities in each cycle sub-problem are defined by one parameter. We denote the parameter associated with the c-th cycle as ψ^c. The affinity of the non-planar edge is $-\psi^c$ and the affinity of each planar edges is ψ^c. We use ψ to denote the concatenation of all ψ^c. Second that the affinity of the non-planar edges is strictly non-positive $\psi^c \geq 0$. Notice also that the optimal value of each cycle sub-problem is zero so the term corresponding to them in (3) can be removed.

[1] Notice that we impose a requirement on θ^{pmc} not a requirement on x^{pmc}.

[2] A 2-colorable segmentation is a partition of the set of superpixels into connected subsets (segments) such that every segment can be given a different color from all adjacent segments, using only two colors overall.

The second property is that the sum of the affinities of the copies of a given non-planar edge $\hat{e}$ across cycle sub-problems can be no less than $\theta^{\mathrm{NP}}_{\hat{e}}$. We use a matrix W to define this constraint. We index W by $\hat{e}, c$ where $W_{\hat{e}c} = 1$ indicates that non-planar edge $\hat{e}$ is present in the c-th cycle sub-problem. The constraint is written formally below.

$$-W\psi \geq \theta^{\mathrm{NP}} \tag{6}$$

4.4 Value of a Re-parameterization

We now discuss the value of a given re-parameterization. Since the optimal values of the cycle sub-problems and the planar multicut sub-problem are zero, the value of a given re-parameterization is the optimal value of the independent edge sub-problem. If we re-write the re-parameterization condition in (4), we notice that $\theta^{ie} = \theta - \theta^{pmc} - \sum_c \theta^c$. We now discuss the mapping of ψ values to $\sum_c \theta^c$. This is done via a matrix operation $Y\psi$. Matrix Y is indexed by e, c where $Y_{ec} = 1$ indicates that planar edge e is present in cycle c. We write the value of a re-parameterization that obeys all of the constraints discussed in this section below.

$$\min_{X^{ie}} (\theta^{ie})^t X^{ie} \;=\; 1^T\theta^{ie} \;=\; 1^T(\theta^{\mathrm{P}} - \theta^{pmc} - Y\psi) + 1^T(\psi + \theta^{\mathrm{NP}}) \;. \tag{7}$$

The term $1^T(\theta^{\mathrm{P}} - \theta^{pmc} - Y\psi)$ corresponds to the value of the contribution to the objective from planar edges. The term $1^T(\psi + \theta^{\mathrm{NP}})$ corresponds to the value of the contribution to the objective from non-planar edges. We now write the constraint that each planar edge have non-positive affinity in the independent edge sub-problem.

$$\theta^{\mathrm{P}} - \theta^{pmc} - Y\psi \leq 0 \tag{8}$$

We use (6) to ensure that each non-planar edge has non-positive affinity in the independent edge sub-problem. We write the constraint that the affinity in the single edge sub-problem for a given edge e is lower bounded by $\min(\theta_e, 0)$ as follows.

$$\min([0, \theta^{\mathrm{P}}]) \leq \theta^{\mathrm{P}} - \theta^{pmc} - Y\psi \tag{9}$$

In summary, the lower bound of Lagrangian decomposition is the optimal value of the linear program below.

$$\max_{\theta^{pmc}, \psi} \quad 1^T(\theta^{\mathrm{P}} - \theta^{pmc} - Y\psi) + 1^T(\psi + \theta^{\mathrm{NP}}) \tag{10}$$

$$\text{subject to} \quad Z\theta^{pmc} \geq 0 \tag{11}$$

$$-W\psi \geq \theta^{\mathrm{NP}} \tag{12}$$

$$\theta^{\mathrm{P}} - \theta^{pmc} - Y\psi \leq 0 \tag{13}$$

$$\min([0, -\theta^{\mathrm{P}}]) \leq -\theta^{pmc} - Y\psi \tag{14}$$

$$\psi \geq 0 \tag{15}$$

4.5 Optimization of the Lower Bound

Since we can not enumerate all of the constraints of the form $Z\theta^{pmc} \geq 0$ or enumerate all the cycle sub-problems, we adopt a cutting plane approach in the dual. Given a set of cycle sub-problems and a set of 2-colorable partitions Z, we solve the LP in (10)–(15), then compute the most violated constraint corresponding to a 2-colorable partition. This is done by minimizing the following objective over the set of 2-colorable partitions which we denote as P_2 .

$$\min_{\bar{X} \in P_2} \theta^{pmc} \bar{X} \tag{16}$$

We can compute the minimum value 2-colorable segmentation $\bar{X}$ in time $O(N^{3/2} \log(N))$ where N is the number of planar edges [14]. This computation is fast in practice (less than a second in all our applications). For every segment in $\bar{X}$, we add the constraint corresponding to the cut that separates this segment from the other segments. We repeat this until the minimum value 2-colorable partition has nearly 0 value. At any point in optimization, we can lower bound the optimal value of the planar multicut sub-problem by 3/2 times the value of the optimal 2-colorable partition as described in [17].

Next, we solve the dual of the Lagrangian decomposition, which provides a non-integral partition. We then find violated cycle inequalities in the non-integral partition and add the corresponding sub-problems to Lagrangian decomposition. The addition of cycle sub-problems corresponds to adding columns to Y and W. The dual of Lagrangian decomposition is written below. It is derived in the supplement. We use ϕ to simplify notation where $\phi = \min(0, -\theta^{\mathrm{P}})$.

$$\begin{aligned} \min_{\gamma \geq 0, \omega \geq 0, \delta \geq 0, \beta \geq 0} \quad & 1^T(\theta^{\mathrm{P}} + \phi + \theta^{\mathrm{NP}}) - \phi^T Z^T \gamma \\ & - (\theta^{\mathrm{NP}})^T \omega + ((-\theta^{\mathrm{P}})^T - \phi^T)\beta && (17) \\ \text{subject to} \quad & (Z^T \gamma + \beta) = 1 + \delta && (18) \\ & W^T \omega \geq Y^T \beta - Y^T 1 + 1 - Y^T \delta && (19) \end{aligned}$$

The vector $Z^T\gamma$ defines a non-integral partition. The vector ω determines which non-planar edges are cut. Here $\omega_{\hat{e}} = 1$ indicates that edge $\hat{e}$ is uncut and $\omega_{\hat{e}} = 0$ indicates that edge $\hat{e}$ is cut (not a typo). Violated cycle inequalities are then found in this partition and the corresponding cycle sub-problems are added to Lagrangian decomposition. Violated cycle inequalities only correspond to cycles involving a non-planar edge. Violated cycle inequalities can be found using Dijkstra's algorithm to find a path between the superpixels connected by any non-planar edge $\hat{e}$ for which the fractional number of cut edges is minimal. If the fractional number of cut edges on this path is less than $1 - \omega_{\hat{e}}$, we add the cycle sub-problem containing this path to the Lagrangian decomposition. The iteration terminates when no violated cycle inequalities are found in $Z^t\gamma$.

4.6 Constructing a Segmentation

We now consider two approaches for converting of $Z^T\gamma$ to an integral partition. The first approach, which is called simple rounding, is based on rounding $Z^T\gamma$ to a partition. We cut every edge e such that $(Z^T\gamma)_e \geq T$ where T is a parameter. We uncut all cut edges within connected components. To select the optimal T we can try all unique values of $Z^T\gamma$ or a fixed number of values uniformly spaced over the interval $[0, 1]$. Notice that $Z^T\gamma \geq T$ defines which non-planar edges are cut.

For a more principled approach, we solve an integer program that constructs a partition using a weighted sum of the 2-colorable segmentations with non-zero γ. In practice, the number of 2-colorable segmentations with non-zero γ is small (tens in our applications). We now discuss the integer program explicitly.

We use $\ddot{X}$ to describe the integral partition that we are constructing. We use a vector $\ddot{\gamma}$ to describe a fractional multicut. We index $\ddot{\gamma}$ with r where $\ddot{\gamma}_r$ defines how much of the r-th 2-colorable partition is included in X. Vector $\ddot{\gamma}$ has only indices corresponding to non-zero indices of γ. We use $\ddot{\omega}$ to define which of the non-planar edges are uncut. We use $\ddot{\omega}$ to penalize non-planar edges being uncut.

Our ILP is built on the following constraints. We enforce that a planar edge with negative affinity e can only be cut if the multicut defined by $\ddot{\gamma}$ cuts this edge at least once. We enforce that a planar edge with positive affinity e can only be uncut if, for every partition r, $Z_{r\hat{e}}\ddot{\gamma}_r = 0$. We enforce that a non-planar edge $\hat{e}$ can only be cut in X if the cycle inequalities from the primal LP are satisfied. We write this optimization as an ILP below. The solution of this ILP does not make up a substantial portion of the computation time in practice.

$$\min_{\ddot{X},\ddot{\omega},\ddot{\gamma}} \quad (\theta_e^{\mathrm{P}})^T \ddot{X}_e - (\theta^{\mathrm{NP}})^T \ddot{\omega} + 1^T\theta^{\mathrm{NP}} \tag{20}$$

$$\text{s.t.} \quad \ddot{X}_e \leq \sum_{r\ s.t.\ Z(r,e)=1} \ddot{\gamma}_r \qquad \forall e\ s.t.\ \theta_e < 0 \tag{21}$$

$$\ddot{\gamma}_c \leq \ddot{X}_e \quad \forall r\ s.t.\ Z(r,e) = 1;\ \ \forall e\ s.t.\ \theta_e \geq 0 \tag{22}$$

$$\ddot{\omega}_{\hat{e}} \geq 1 - \sum_{r\ s.t.\ [\exists e \in c \quad Z(r,e)=1]} \ddot{\gamma}_e; \qquad \forall c\ s.t.\ W(\hat{e}, c) = 1 \tag{23}$$

$$\ddot{X}_e \in \{0, 1\} \quad \forall e \tag{24}$$

$$\ddot{\omega}_{\hat{e}} \in \{0, 1\} \quad \forall \hat{e} \tag{25}$$

$$\ddot{\gamma}_r \in [0, 1] \quad \forall r \tag{26}$$

Once $\ddot{X}$ is constructed, cut edges within connected components are uncut and the partition is returned. In practice, the objective value of the segmentation obtained by solving the ILP is nearly the same as the lower bound (cf. Section 5.2). The entire Lagrangian decomposition algorithm is detailed in Algorithm 1.

Algorithm 1. Semi-Planar Multicuts

Initialize Z=[], W=[], Y=[];
while *true* **do**
  while *true* **do**
    Solve LP in (10)
    $\bar{x} \leftarrow \arg\min_{x^{2c}} \sum_e -\lambda_e x_e$
    $Z \leftarrow [Z \cup \bar{x}]$
    if $\sum_e -\lambda_e \bar{x}_e \sim 0$ **then**
      break
    end
  end
  $\gamma \leftarrow$ solve LP in (17)
  for $\hat{e}$ **do**
    Use Dijkstra's algorithm to find a path between the superpixels connected by edge $\hat{e}$ for which the fractional number of cut edges is minimal. If this number is less than $1 - \omega_{\hat{e}}$, add a cycle sub-problem corresponding to this path to the Lagrangian decomposition.
  end
end
$\gamma \leftarrow$ solve LP in (17)
round $Z^T\gamma$ using ILP in (20) or apply simple rounding on $Z^T\gamma$ to produce a final segmentation X.

5 Application

5.1 Effectiveness

To demonstrate the effectiveness of Algorithm 1, we implement a workflow for inter-active image segmentation that requires only trivial user input for both the joining and cutting of segments. In the beginning, the user is presented with an optimal multicut of a superpixel segmentation. This multicut is computed using only affinities between adjacent superpixels. Subsequently, any combination of two types of input is accepted to correct errors in the segmentation: 1. Scribbles on the image to indicate pixels that belong to the same segment, 2. Pairs of points in the image to indicate pixels that belong to different segments.

The evidence provided by scribbles is incorporated into the multicut problem by constraining variables to 0. The evidence provided by pairs of points is incorporated as a non-local affinity with sufficiently large negative weight. Alternatively, these affinities could also be incorporated as hard constraints. Hard constraints are processed similarly to non-local affinities with finite negative weight, except that the corresponding variable is fixed to 1 and the constant contribution to the objective function, $-\infty$, is ignored. Either way, every pair of points excludes from the solution set all segmentations in which the points are in different segments.

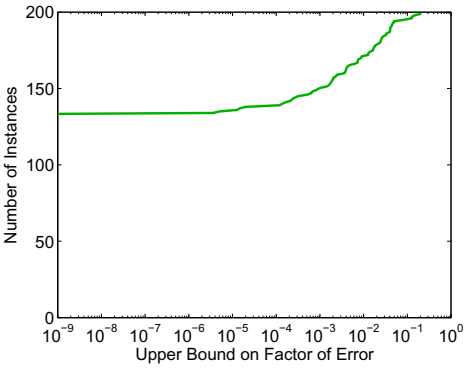


Fig. 3. The fraction by which the value of a segmentation output by Alg. 1 deviates from the global optimum (first axis) is small for a large number of instances (second axis). 132 of 200 instances are solved to optimality. For 170 of the 200 instances, the error is less than one percent. Depicted is the upper bound on the error provided by the gap between the value of a segmentation and the LP lower bound.

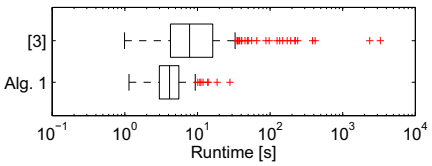


Fig. 4. Alg. 1 terminates after 4.73 s in the mean, 4.16 s in the median and 28.05 s in the worst case of the 200 instances. This is significantly faster than the an exact algorithm for general graphs [3] that takes 51.32 s in the mean, 7.79 s in the median and 3307.42 s (55 minutes) in the worst case.

At any point in time, the user can request that the augmented problem with additional constraints be solved to obtain an updated segmentation. There is no requirement for the user to focus on one or the other type of errors, nor to prefer one over the other type of input. Results that confirm the effectiveness of the algorithm and the consistency of our implementation are depicted in Fig. 5.

5.2 Quality of Bounds and Runtime

To examine the bounds provided by Alg. 1 and to compare its runtime to that of the exact algorithm for general graphs [3], we set up segmentation problems with negative non-local affinities for the 200 test images of the BSD500 benchmark. To ensure that experiments are unbiased, i.e. not subjective to specific user input, we construct these problems automatically as follows.

Starting with the same superpixels and affinities between adjacent superpixels as in [17], we search the proximity of every superpixel for the superpixel whose mean color differs maximally from that of the first superpixel. For every image, we add the 50 strongest non-local affinities. Both, our implementation of Alg. 1 and the optimized C++ implementation [1] of [3], use IBM ILOG Cplex as an LP and ILP solver, respectively. We use Blossom V [11] for computing the optimal 2-colorable partitions.

The fraction by which the value of a segmentation output by Alg. 1 deviates from the global optimum is depicted on the first axis in Fig. 3. More precisely, it is the upper bound on this deviation provided by the gap between the value

of a segmentation and the LP lower bound. This gap is zero for 132 of the 200 benchmark images, indicating that these problems are solved to optimality by Alg. 1. For 170 of the 200 instances, the error is less than one percent.

The distribution of runtimes for the 200 problems is depicted in Fig. 4. Unlike the exact algorithm for general graphs whose worst-case runtime is prohibitive, Alg. 1 is fast enough for inter-active applications.

6 Conclusions

We motivated the use of negative affinities between non-adjacent superpixels in the multicut formulation of the image segmentation problem. For the resulting NP-hard combinatorial optimization problem, we proposed a relaxation by Lagrangian decomposition and a constrained set of re-parameterizations for which we can optimize exactly and efficiently by solving an LP. We implemented a cutting plane approach in the dual, along with an algorithm to construct segmentation from fractional solutions. This algorithms finds optimal to near optimal solutions of segmentation problems with non-local negative affinities for natural images. In our applications, it is more than 10 times faster on average than an exact algorithm based on integer programming.

A Appendix

A.1 Difficulty of Positive Non-local Terms

We now consider why positive non-local terms are difficult. Let v and w be superpixels connected by a positive non-local term whose corresponding edge is x_k. Let $S(v, w)$ the set of all rings such that if all edges in a particular ring are cut then the positive edge x_k must also be cut. The corresponding addition to the LP relaxation is written below.

$$\forall r \in S(p_1, p_2) \quad x_k + \sum_{e \in r} x_e \leq |r| \ . \tag{27}$$

These constraints that appear elegantly in the form of separator inequalities in [13] are remarkably fragile and almost always result in fractional solutions to even trivial multicut problems. Moreover, these fractional solutions are uninformative with regards to what the optimal solution is. As an example, consider five superpixels s_a, s_b, s_c, s_d, s_e in the plane. Let s_a be surrounded on all sides by s_b, s_c, s_d, let s_e be a neighbor of s_b, s_c, s_d and disconnected from s_a. Also, let s_a, and s_e be paired by an infinitely strong positive term. The remaining terms are described as follows:

$$\theta_{s_a,s_b} = \theta_{s_a,s_c} = \theta_{s_a,s_d} = \infty \tag{28}$$

$$\theta_{s_e,s_b} = \theta_{s_e,s_c} = \theta_{s_e,s_d} = -1 \tag{29}$$

$$\theta_{s_b,s_c} = \theta_{s_c,s_d} = \theta_{s_b,s_d} = \infty \tag{30}$$

Clearly, the optimal solution is to put each superpixel in the same connected component, and the cut cost is zero. However, the optimal fractional solution is to cut each repulsive edge with value $-2/3$, and objective value is -2.

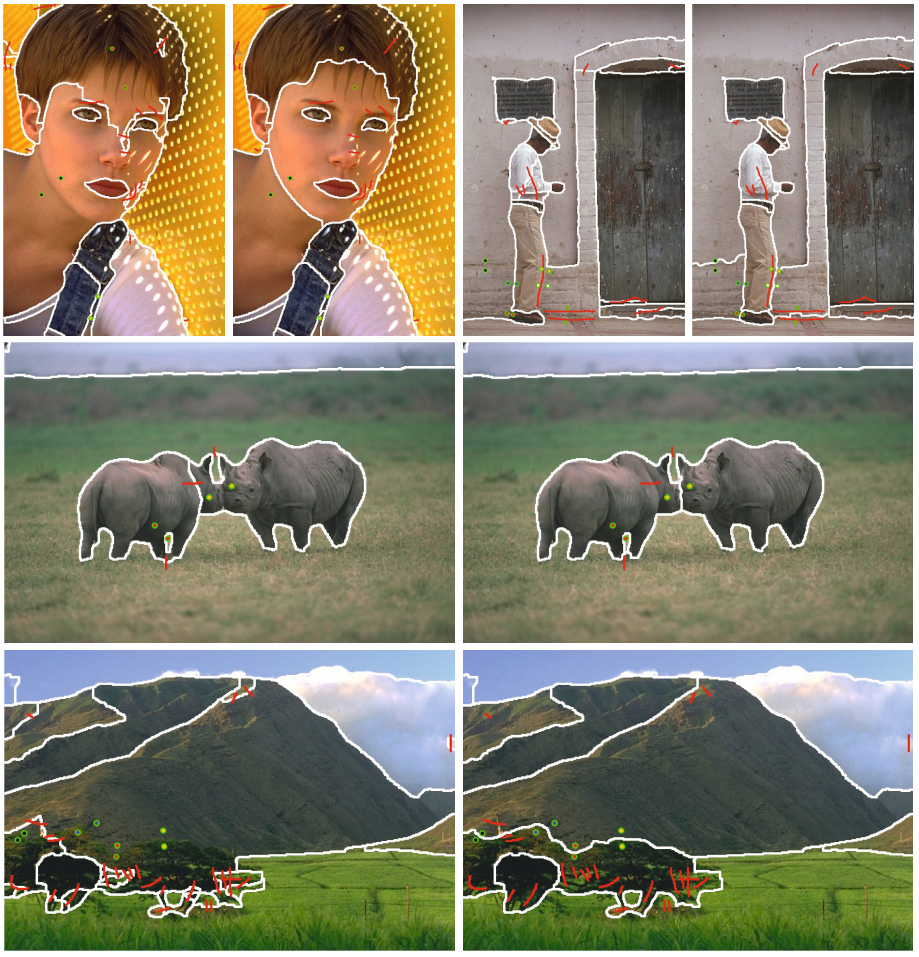


Fig. 5. To demonstrate the effectiveness of Algorithm 1, we implement a workflow for inter-active image segmentation that requires only trivial user input for both the joining and cutting of segments. Presented with an initial and typically imperfect segmentation, the user can scribble on the image (red) to indicate pixels that belong to the same segment and click on pairs of points (green) that belong to different segments. The latter are incorporated as non-local negative affinities.

A.2 Lower Bound on Affinities in the Independent Edge Sub-problem

We now establish that $\min([0, -\theta^{\mathrm{P}}]) \leq -\theta^{pmc} - Y\psi$. To do this, we rely on a theorem established in [16] which states that given affinities θ for an instance of the planar multicut problem, the affinities in the planar multicut sub-problem $-\theta^{pmc}$ are lower bounded by $\min([-\theta, 0])$. In [16], no non-planar edges are considered.

Given optimal parameters ψ, we have a description for a planar multicut problem instance. Thus, the parameters of the independent edge sub-problem and the planar multicut sub-problem satisfy the following inequality.

$$\min([-\theta^{\mathrm{P}} + Y\psi, 0]) \leq -\theta^{pmc} \ . \tag{31}$$

Suppose that, at the optimal re-parameterization, the constraint

$$\min([0, -\theta^{\mathrm{P}}]) \leq -\theta^{pmc} - Y\psi \tag{32}$$

is not satisfied. Thus, there must be a planar edge e such that the constraint is unsatisfied for. In this case, there must exist a cycle sub-problem c containing e such that $\psi_e^c \geq 0$. We now define α and σ as follows.

$$\alpha = -\min([-\theta_e, 0]) - \theta_e^{pmc} - Y\psi \tag{33}$$

$$\sigma = \max([0, \psi^c + \alpha]) - \psi^c \ . \tag{34}$$

Now set $\psi^c = \max([0, \psi^c + \alpha])$. For each edge $f \neq e$ in c set $\theta_f^{pmc} = \theta_f^{pmc} - \sigma$. Notice that the bound remains constant, no additional edges are unsatisfied and the constraint of the form $\min([0, -\theta_e^{\mathrm{P}}]) \leq -\theta_e^{pmc} - (Y\psi)_e$ is $-\sigma$ closer to being satisfied. This process can be repeated until the constraint $\min([0, -\theta^{\mathrm{P}}]) \leq -\theta^{pmc} - Y\psi$ is satisfied for all edges.

A.3 Properties of Cycle Sub-problems

We now consider the constraints (6) on the parameters of the cycle sub-problems discussed in Section 4.3. We establish that they do not loosen the lower bound of Lagrangian decomposition as follows. Assume that we have maximized the lower bound of Lagrangian decomposition and that these constraints are violated. We then alter the re-parameterization without loosening the bound so as to enforce the relevant constraints for a single cycle. This process can be repeated for every cycle in which the constraints are violated.

Consider a given cycle sub-problem c and no structure in the affinities. Suppose that there are zero or two or more edges with non-positive affinity. If this is the case, move all negative affinity to the independent edge sub-problem and all positive affinity to the planar multicut sub-problem. If the non-planar edge has positive affinity, move the positive affinity to other copies of that edge in cycle sub-problems and the independent edge sub-problem. Preserve that the affinity in the independent edge sub-problem for each edge is non-positive. Finally, remove cycle sub-problem c from the decomposition. Notice that these alterations to the re-parameterization do not loosen the lower bound of the decomposition.

Now consider the case where there is one negative affinity edge in cycle sub-problem c. Set the affinity of the edge with negative affinity to the opposite of the affinity of the edge with the smallest non-negative affinity. We denote the value of the smallest non-negative affinity as ϕ^c. Now, set all affinities on edges with non-negative affinity to ϕ^c. This last step is done because no optimal configuration of cycle sub-problem c cuts a non-negative affinity edge e unless

that edge has affinity equal to ϕ^c. We send the "excess" positive affinity from cycle sub-problem c to the planar multicut sub-problem, which does not loosen the lower bound of the decomposition.

Since the addition of cycle sub-problems induces non-planar edges to but uncut or induces planar edges along a path connecting the ends of a non-planar edge to be cut, it is clear that the affinity associated with the non-planar edge in each given cycle sub-problem is non-positive.

References

1. Andres, B.: `https://github.com/bjoern-andres/graph`
2. Andres, B., Kappes, J.H., Beier, T., Köthe, U., Hamprecht, F.A.: Probabilistic image segmentation with closedness constraints. In: ICCV (2011)
3. Andres, B., Kroeger, T., Briggman, K.L., Denk, W., Korogod, N., Knott, G., Koethe, U., Hamprecht, F.A.: Globally optimal closed-surface segmentation for connectomics. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part III. LNCS, vol. 7574, pp. 778–791. Springer, Heidelberg (2012)
4. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. TPAMI 33(5), 898–916 (2011)
5. Bachrach, Y., Kohli, P., Kolmogorov, V., Zadimoghaddam, M.: Optimal coalition structures in graph games. arXiv ePrint (2011)
6. Bagon, S., Galun, M.: Large scale correlation clustering optimization. arXiv ePrint, abs/1112.2903 (2011)
7. Bansal, N., Blum, A., Chawla, S.: Correlation clustering. Machine Learning 56, 89–113 (2004)
8. Chopra, S., Rao, M.R.: The partition problem. Math. Program. 59, 87–115 (1993)
9. Kappes, J.H., Speth, M., Andres, B., Reinelt, G., Schnörr, C.: Globally optimal image partitioning by multicuts. In: Boykov, Y., Kahl, F., Lempitsky, V., Schmidt, F.R. (eds.) EMMCVPR 2011. LNCS, vol. 6819, pp. 31–44. Springer, Heidelberg (2011)
10. Kim, S., Nowozin, S., Kohli, P., Yoo, C.D.: Higher-order correlation clustering for image segmentation. In: NIPS (2011)
11. Kolmogorov, V.: Blossom V: a new implementation of a minimum cost perfect matching algorithm. Mathematical Programming Computation 1(1), 43–67 (2009)
12. Nowozin, S., Jegelka, S.: Solution stability in linear programming relaxations: graph partitioning and unsupervised learning. In: ICML, pp. 769–776 (2009)
13. Nowozin, S., Lampert, C.H.: Global interactions in random field models: A potential function ensuring connectedness. SIAM J. Img. Sci. 3(4), 1048–1074 (2010)
14. Shih, W.-K., Wu, S., Kuo, Y.S.: Unifying maximum cut and minimum cut of a planar graph. IEEE Trans. Comput. 39(5), 694–697 (1990)
15. Vitaladevuni, S.N.P., Basri, R.: Co-clustering of image segments using convex optimization applied to EM neuronal reconstruction. In: CVPR (2010)
16. Yarkony, J.: MAP inference in Planar Markov Random Fields with Applications to Computer Vision. PhD thesis, University of California, Irvine (2012)
17. Yarkony, J., Ihler, A., Fowlkes, C.C.: Fast planar correlation clustering for image segmentation. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VI. LNCS, vol. 7577, pp. 568–581. Springer, Heidelberg (2012)

Contour-Relaxed Superpixels

Christian Conrad[1], Matthias Mertz[1], and Rudolf Mester[1,2]

[1] VSI Lab, CS Dept., Goethe University Frankfurt, Germany
[2] Computer Vision Laboratory, ISY, Linköping University, Sweden

Abstract. We propose and evaluate a versatile scheme for image presegmentation that generates a partition of the image into a selectable number of patches (*'superpixels'*), under the constraint of obtaining maximum homogeneity of the 'texture' inside of each patch, and maximum accordance of the contours with both the image content as well as a Gibbs-Markov random field model. In contrast to current state-of-the art approaches to superpixel segmentation, 'homogeneity' does not limit itself to smooth region-internal signals and high feature value similarity between neighboring pixels, but is applicable also to highly textured scenes. The energy functional that is to be maximized for this purpose has only a very small number of design parameters, depending on the particular statistical model used for the images.

The capability of the resulting partitions to deform according to the image content can be controlled by a single parameter. We show by means of an extensive comparative experimental evaluation that the compactness-controlled *contour-relaxed superpixels* method outperforms the state-of-the art superpixel algorithms with respect to boundary recall and undersegmentation error while being faster or on a par with respect to runtime.

1 Introduction

The history of image segmentation research, when regarded on the scale of decades, exhibits clearly discernible phases during which certain method paradigms have been dominant. There has been a period of statistical models during the 1980/90ies, largely induced by the requirements and operating conditions of remote sensing and (somewhat later) image communication. Currently, in the early 2010ies, segmentation research is clearly dominated by graph-based methods, partially also variational methods. However, it is important not to mix and possibly confuse the image model (comprising the 'energy function' that should be minimal for a 'good' segmentation), vs. the optimization scheme.

We propose and extensively evaluate in the following a method for computing superpixels, that is: relatively small regions that are expected to be homogenous with respect to their internal texture. In contrast to most other currently discussed superpixel approaches, this method can be deduced from a statistical model, and homogeneity refers to 'real' texture, not the constricted interpretation of having locally smooth, or even constant, image values.

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 280–293, 2013.
© Springer-Verlag Berlin Heidelberg 2013

Furthermore, the features used for the segmentation process can be almost arbitrarily combined from choices such as a) gray values, b) color vectors, c) texture features (= outputs of texture operators applied onto a spatial neighborhood centered on the regarded pixel), d) depth values (e.g. from an active depth camera) e) motion vectors, and many more. Thus we speak of a whole family of segmentation modules that can be constructed from the basic approach. The only constraint is that the individual feature channels can be regarded as uncorrelated – for reasons that become clear later (section 4.1). The two main contributions of the paper are a statistically sound approach to obtain a superpixel energy function, allowing explicit control of compactness, and an extensive comparative evaluation with state-of-the art approaches for a) the base line gray value version of *contour-relaxed superpixels* and b) the color version.

2 Related Work

The term 'superpixels' dates way back in the 1980ies and has been popularized by Ren and Malik [1]. It can be defined as a connected group of pixels similar with respect to certain features, i.e. their color value. A superpixel segmentation of an image is an oversegmentation into typically many subregions, all of them expected to be a proper subset of exactly one 'semantic' region. Usually, superpixel segmentations serve as precursor to higher level tasks such as object segmentation, motion estimation or tracking [1–4] where computations are more efficiently done on some few hundred groups of such pixels [1, 3, 5] instead of $10^5 - 10^6$ pixels.

Superpixels can be obtained using standard segmentation algorithms, e.g. the well known mean-shift [6] and watershed [7] algorithms. However, it has been shown that these 'general purpose' algorithms typically produce superpixels of highly irregular shape and size and do not allow to directly control the number of superpixels [8, 9]. It is a common view in the literature [8–10] that superpixel algorithms should include a compactness constraint and should allow to directly control the number of superpixels (to avoid a costly parameter search) such that superpixels are similar in size. While non-compact superpixels can adapt to quite complicated object shapes, they risk a larger extent of *undersegmentation error*, a measure of overlap between a single superpixel and multiple objects. In our approach, the degree of compactness can be controlled by a single parameter. Variation of this parameter allows a system designer to chose between highly compact superpixels and an almost 'fluid' behavior that aligns superpixels well even with complicated boundaries.

The various approaches proposed since [1] can roughly be grouped into *graph-based* and *gradient-based* methods. Among the most popular graph-based methods one finds the superpixel algorithms based on normalized cuts [1, 5, 11] and Felzenszwalb & Huttenlocher's approach [12] who compute a superpixel segmentation by solving a shortest spanning tree problem. A drawback of [12] is that they do not encode a compactness constraint. A problem of the normalized cut approach [1] is its computational effort, since (depending on the image size) a

single run of the algorithm may take minutes on hardware as of 2012. In the 'superpixel lattices' proposal [13], superpixels are forced to conform to a grid by introducing a topology constraint leading to superpixels with a rectangular shape. SP-Lattice is among the fastest superpixel methods up to now [13, 14], but produces results of lower quality compared to more recent work [8, 9, 14].

More recently, Veksler et al. [9] and [14] computed superpixel segmentations within the well known *graph cuts* and *expansion moves* framework. Veksler et al. cover an image with overlapping square patches of fixed size which are subsequently stitched during optimization using $\alpha-$expansion. They define a second order energy function containing a data term modeling the likelihood for a specific label for a single pixel and a prior term resembling the Potts model. Two different flavors of the approach are proposed, namely 'compact' and 'constant' superpixels, respectively. The 'compact' superpixels have the inherent problem that the likelihood term does not differentiate between pixels having different color, but assigns low energy to all labels (patches) that overlap the regarded pixel. Therefore neighboring pixels are likely to be assigned the same label regardless of their color, leading to superpixels possibly containing strong discontinuities. In 'constant superpixels', a different data term is used which is based on the distance in color space between the regarded pixel and the color of the center pixel of the current label thus encouraging superpixels of constant or similar intensity. For more than 400 superpixels, the constant version of [9] outperforms TurboPixels, the normalized cut approach as well as the method by Felzenszwalb & Huttenlocher. However, while Veksler et al. encode a compactness constraint, their method does not allow to control the number of superpixels directly but would need a parameter search to do so. The work of Zhang et al. [14] relies on pseudo-boolean optimization and is inspired by the one by Veksler et al., trying to overcome the aforementioned difficulties by design. They report runtime figures much lower than the ones by Veksler et al. which are independent of the number of superpixels. However the quality of their superpixels with respect to boundary recall R_B and undersegmentation error E_{us} is worse than Veksler's.

TurboPixels [8], and more recently SLIC superpixels [10] are prominent gradient based methods. Such methods do not formulate segmentation as some graph based related problem such as *mincut* but rather optimize the energy function in a gradient ascent/descent sense. TurboPixels are based on the geometric flow implemented via level sets. The method allows to directly control the number of superpixels and integrates a compactness constraint. In the SLIC ('simple linear iterative clustering') method [10], the computation of superpixels is cast as a clustering problem in a five dimensional feature space consisting of the three `Lab` color channels and the pixel 2d coordinates. The scheme starts with regularly sampling N cluster centers in image space, subsequently perturbing the center locations such that they lie at the lowest image gradient position within a small sub window. Then pixels are assigned to the best fitting cluster center within a small neighborhood and cluster centers are recomputed based on the ℓ_1 norm between old and new center positions. The method can directly control the number of superpixels and encodes a compactness constraint. It is shown [10] that SLIC

outperforms TurboPixels, the method of Felzenszwalb & Huttenlocher, and the normalized cut approach of [1, 5].

3 Outline of the New Approach

In this work we propose a superpixel algorithm that has the following advantages: (a) direct control of the number of superpixels, (b) control of the compactness of the superpixels by setting a (single) compactness parameter κ, thus superpixels can be allowed to adapt to complicated shapes if needed, and (c) most importantly: an explicit statistical modeling of superpixel shape and content allows to perform the segmentation on an arbitrary number of feature channels, e.g. intensity-only, color, depth, or 'real' texture feature vectors. The energy function to be optimized is derived from this statistical image model, and it turns out that already a local optimization is sufficient for yielding results competitive or superior to the state of the art. The image model used for *contour-relaxed superpixels* is based on homogeneously textured regions; this includes the plain model of smooth (=quasi-constant) gray or color values, but goes significantly beyond this.

4 Theory of *Contour-Relaxed Superpixels*

In this section, we summarize the theoretical basis on which the *contour-relaxed superpixels* approach is built. Attempts to formulate the segmentation problem as an estimation task can already be found in the very early literature. Still, the potentials and advantages of a well-founded statistical model for low-level segmentation are not reflected in the current literature. We build on the fundamental model used by Mester et al. [15, 16] and transform it into a competitive superpixel approach by introducing a compactness term.

Let the total set of measurements on the image array (= a 2D-array of vectors), be summarized in a single huge vector $\boldsymbol{z}$. Let $\mathcal{Q} = \{R_1, R_2, \ldots R_n\}$ be a partition of the image array. The measurements *inside* of each region are the outcomes of region-specific stochastic processes, and each such process is associated with an individual parameter vector $\boldsymbol{\theta}_i = \boldsymbol{\theta}(R_i)$. The combination of a partition $\mathcal{Q}$ and the parameter ensemble $\{\boldsymbol{\theta}\} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots \boldsymbol{\theta}_n\}$ is denoted as the 'array state' $\mathcal{S}$. Each completely specified array state $\mathcal{S}$ induces a joint probability density $p(\boldsymbol{z}, \mathcal{Q}, \{\boldsymbol{\theta}\}) = p(\boldsymbol{z}, \mathcal{S})$ for the ensemble of random variables $\boldsymbol{z}$. Given an image measurement vector $\boldsymbol{z}$, the notion 'segmentation' means to find an array state $\mathcal{S}$ which has a high likelihood to have generated the observed image vector $\boldsymbol{z}$.

4.1 Deriving the Segmentation 'Energy Function' from Maximum-A-Posteriori (MAP) Principles

The particular combination of a partition $\mathcal{Q}$ and the corresponding model parameters $\{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots \boldsymbol{\theta}_n\}$ that maximizes the probability density function

$$p(\mathcal{S}|\boldsymbol{z}) = p(\mathcal{Q}\ ,\ \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots \boldsymbol{\theta}_n\}|\boldsymbol{z}) \tag{1}$$

is considered as the maximum-a-posteriori (MAP) estimate of the array state $\mathcal{S}$. From Bayes' theorem we obtain

$$p(\mathcal{S}|\boldsymbol{z}) = p(\boldsymbol{z}|\mathcal{S}) \cdot p(\mathcal{S})/p(\boldsymbol{z}). \tag{2}$$

With the observed image vector $\boldsymbol{z}$ being fixed, $p(\boldsymbol{z})$ is merely a normalizer. Thus we search the particular array state $\mathcal{S}$ which maximizes the target function J

$$J := \quad p(\boldsymbol{z}|\mathcal{S}) \cdot p(\mathcal{S}) = p(\boldsymbol{z},\ \mathcal{S}) = p(\boldsymbol{z}|\mathcal{Q}\,, \boldsymbol{\theta}) \cdot p(\boldsymbol{\theta}|\mathcal{Q}) \cdot p(\mathcal{Q}). \tag{3}$$

The joint density in Eq. 3 consist of the prior probability $p(\mathcal{Q})$ for the regarded partition $\mathcal{Q}$, and the conditional densities for $\boldsymbol{z}$ given the individual region model processes. Using a Gibbs random field (GRF) with discrete two-element cliques, $p(\mathcal{Q})$ is expressed as

$$p(\mathcal{Q}) = \frac{1}{Z} \cdot \exp\left(-\sum_{c_i} V_c(c_i) \right). \tag{4}$$

Here, c_i denote all maximal cliques of size 2. The potentials $V_c(c_i)$ depend on whether the label values in such a clique are identical or not (Potts model). While computing the partition function Z is intractable in general, it is not needed here, as we are only interested in the MAP estimate of Eq. 3.

The region-specific model parameter vectors $\boldsymbol{\theta}$ are considered as unknown deterministic parameters, (i.e. we assume a 'flat', uninformative prior for them) such that only the distribution for the partition $\mathcal{Q}$ appears in $p(\mathcal{S})$

$$p(\mathcal{S}) = p(\boldsymbol{\theta}, \mathcal{Q}) = \alpha \cdot p(\mathcal{Q}). \tag{5}$$

The model parameters $\boldsymbol{\theta}$ are obtained by maximizing the term $p(\boldsymbol{z}|\mathcal{S})$ with respect to the parameter $\boldsymbol{\theta}$ while the partition $\mathcal{Q}$ is fixed, that means by a region-specific maximum likelihood estimation ('EM style'):

$$\begin{aligned} p(\boldsymbol{z}, \mathcal{S}) &= p(\boldsymbol{z}|\mathcal{S}) \cdot p(\mathcal{S}) \\ &= p(\boldsymbol{z}|\mathcal{Q}\,, \{\hat{\boldsymbol{\theta}}^{ML}(\mathcal{Q})\}) \cdot p(\mathcal{Q}) \cdot \alpha \end{aligned}$$

The texture processes of the individual regions are considered to be pairwise statistically independent between regions. This means that knowing the complete texture signal inside region R_i does not yield *any* information on the texture signal inside region $R_j, j \neq i$. Thus, the joint probability density of observing *all* the texture signals (= the complete vector-valued image $\boldsymbol{z}$) can be written as a product, bearing in mind that this is an approximation (due to the ML parameter estimate):

$$p(\boldsymbol{z}|\mathcal{Q}) = \prod_{R_i} \prod_{k} p(\boldsymbol{z}_{ik}|\boldsymbol{\theta}_i^{ML}) \tag{6}$$

Here i denotes the region, and index k varies over the feature channels. This plain double product is of course the result of the independence between regions, on one hand, and the (assumed) independence between feature channels, on the other hand, which may be a coarse but effective approximation in some cases.

4.2 The Optimization: 'Contour Relaxation'

Given an initial partition Q_0, J in Eq. 3 is maximised by variation of pixel labels. Each grid point x_0 which is located on the contour of a region is regarded and it is checked whether a change of its initial region label into a another label occuring in its neighborhood leads to an increase of target function J (Eq. 3). If this is the case, the change of the label is carried out. The focusing on *contour pixels* and a *local subset* of labels is essential for the speed and explains how the method is related to the general ICM framework [17].

Due to the conditional independence structure induced by a pairwise GRF, only the 8 cliques including site x_0 need to be taken into account. Hence the expression $p(\mathcal{Q})$ (Eq. 4) can be factorized into

$$p(\mathcal{Q}) = k_1 \cdot \exp(-n'_B B - n'_C C), \tag{7}$$

where only the second factor depends on the label of x_0. Here, n'_B and n'_C denote the numbers of inhomogeneous horizontal/vertical and diagonal cliques where B and C are the associated costs, respectively.

Given the partition $\mathcal{Q}$, the conditional likelihood of the image data $\boldsymbol{z}$ can be factorized into a constant and a variable term:

$$p(\boldsymbol{z} \mid \mathcal{Q}) = k_2 \cdot \prod_{\{\mathcal{R}_j\}} p\big(\boldsymbol{z}(\mathcal{R}_j) \mid \boldsymbol{\theta}(\mathcal{R}_j)\big), \tag{8}$$

with the variable product here comprising only those regions R_j that may include pixel x_0. From the set of legal choices of $q(x_0)$, the label maximizing

$$p(\boldsymbol{z}, \mathcal{Q}) = k_1 \cdot k_2 \cdot \exp(-n'_B B - n'_C C) \cdot \prod_{\{\mathcal{R}_j\}} p\big(\boldsymbol{z}(\mathcal{R}_j) \mid \boldsymbol{\theta}(\mathcal{R}_j)\big) \tag{9}$$

is then assigned to point x_0. Due to the extreme value range of (9), it is computationally mandatory to minimize the negative log of (9) instead of directly maximizing it. This yields the main part of the *energy function* L to be minimized in the *contour-relaxed superpixels* framework (see sec. 5.1).

This scheme denoted as 'contour relaxation' is performed by scanning the whole image using the 'coding scheme' proposed by Besag [18] to avoid directional preferences. The computational expense is rather low, as only pixels on the region boundaries are considered during optimization. The parameter values B and C of the Gibbs model are far from being critical. See section 6 for the values used in our experiments, and the number of passes over the image array.

5 Specific Design Features of the Proposed Approach

The combination of a probabilistic target function to be maximized (or minimized, if the negative logarithm is regarded) and a greedy iterative optimization scheme, as described in the preceding section, leaves of course ample space for

selecting the features, the shape of the assumed distributions (Gaussian, Laplacian, etc.) in the feature channels. The number of channels, and the kind of information assigned to the channels (gray values, color components, depth values, texture features, ...) allows many variants of the fundamental scheme, yielding a whole *family* of segmentation methods. What is *not* needed are weight factors that weigh the relative importance of the individual features; since the moments of the distributions are estimated in the course of the process, the correct balance between the features is obtained automatically — this is certainly one of the main advantages of the approach.

What remains to be explained is how the functional form of the feature pdfs, the likelihood terms appearing in the energy function, and the actual data in the regions are tied together. This is done by simply computing *unnormalized moments* of the data $f_k(\boldsymbol{x}_i)$ in each region R_j, that is:

$$N := \sum_{i \in R_j} 1 \qquad S := \sum_{i \in R_j} f_k(\boldsymbol{x}_i) \qquad Q := \sum_{i \in R_j} f_k^2(\boldsymbol{x}_i) \tag{10}$$

From these *sufficient statistics* N, S, and Q, the ML estimates of the pdf parameters in channel k can be computed, and thus the likelihood for a given ensemble of data values can be determined.

5.1 Adding a Compactness Term κ

Experiments with the original model from [15, 16] show that using this universal segmentation scheme for the particular purpose of computing superpixels suffers from the compactness of the regions not being directly controllable. We introduce an additional compactness term that ensures that the (fundamentally fluid) regions do not create too wriggled, too elongated regions, as this could be the case for a pure Gibbs-Markov random field model with realistic parameters.

Carrying over the cost functional L from the MAP criterion in section 4, we can now enforce spatial compactness for the regions to be formed, and define a new cost functional $\tilde{L}$ by adding an extra 'regularization' term which penalizes the squared deviation between the spatial location $\boldsymbol{x}$ of the pixels in the region R_j and the center $\boldsymbol{m}$ of the region R_j, as follows

$$\tilde{L} = L + \kappa \cdot \sum_{\boldsymbol{x} \in R_j} (\boldsymbol{x} - \boldsymbol{m}(R_j))^T (\boldsymbol{x} - \boldsymbol{m}(R_j)), \tag{11}$$

where κ is a parameter which controls the compactness of the regions. It is straightforward to show that $\tilde{L}$ can be fully expressed in terms of these 'spatial moments' introduced in Eq. 11, the ensemble of clique potentials, and the sufficient statistics N, S, and Q previously defined, if the *functional form* of the feature distributions is given (e.g. a Gaussian model). In case that a Laplacian pdf is chosen, a corresponding set of sufficient statistics can likewise be defined, by incorporating the sum of the absolute values of the scalar features instead of squares. Likewise, other distributions such as χ^2 or Rayleigh pdfs can be used.

6 Evaluation and Experimental Results

The *Contour-relaxed superpixels*[1] approach has been evaluated for many of the numerous variations comprised by the approach; we present in the following the most relevant test results as permitted space allows. The quality of the method is quantitatively expressed by the two established measures *'boundary recall'* (R_B) and *'undersegmentation error'* (E_{us}) evaluated on the 300 images from the *Berkeley Segmentation Database* (BSDS300) [19] which contains ground truth segmentations provided by human subjects. The boundary recall R_B is a measure of how well the superpixel boundaries align with ground truth segments, while the undersegmentation error E_{us} measures the degree of bleeding caused by superpixels overlapping more than one ground truth segment. We compute the undersegmentation error E_{us} using the definition of [8] and the boundary recall using the MATLAB code provided by the BSD benchmark.

The experiments have been carried out based on a straightforward single threaded and unoptimized C/C++ implementation. For images available in the Berkeley database with a resolution of 481×321 (or 321×481) pixels, one iteration of the contour relaxation on a single channel takes about 40 ms on an Intel Xenon 2.8GHz processor.

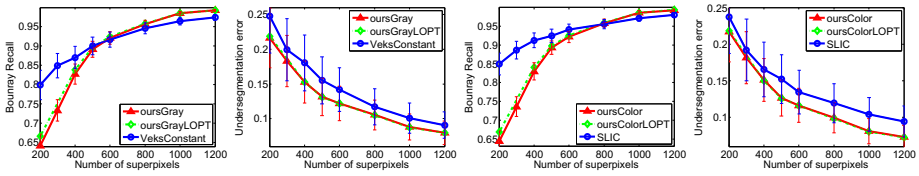


Fig. 1. Column 1 and 2: Benchmark results using intensity and compactness feature: Average boundary recall, and undersegmentation error including error bars ($\pm\sigma/2$) on the BSD300 dataset (300 images). **Column 3 and 4:** Benchmark results using color and compactness feature. Best viewed in color.

As the BSD benchmark differentiates between gray and color image segmentation, we divide our evaluation in two parts, superpixels computed on gray value images and color images, respectively. We compare our results with the state-of-the-art superpixel methods for both settings, namely the approach by Veksler et al. [9] and SLIC superpixels [10], where both of them also include a compactness term. Note that there are many other algorithms available, e.g. Turbo-Pixels [8], NCuts [1], Lattice Superpixels [13] and recently QPBO superpixels [14]. However, it has been shown [10, 14] that SLIC and the approach by Veksler et al. outperform the aforementioned algorithms with respect to boundary recall and undersegmentation error but are coequal or better with respect to runtime performance. Therefore we will exclude these methods from the qualitative evaluation but will reference them within the discussion of the runtime

[1] Code will soon be available at `http://www.vsi.cs.uni-frankfurt.de`

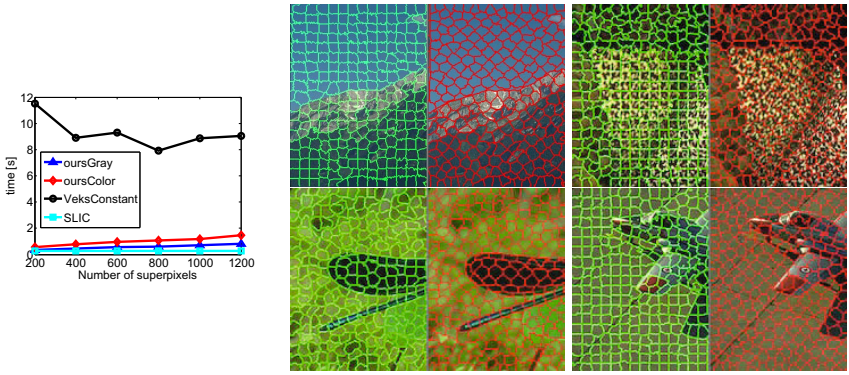


Fig. 2. (left) Runtime performance: Average runtime in seconds averaged over the BSD300 dataset (300 images). **(right)** Typical results obtained with our method when initialized with axis aligned square blocks (green boundaries), and when initialized with a diamond pattern (red boundaries).

performance. Note that we compare our superpixel approach to its most serious competitors but do not include full (object based) segmentation algorithms, e.g. gPb-owt-ucm and its variants [20], as this would not be a fair comparison. However, our approach could be used as an integral part of such a full segmentation.

Single Channel Superpixels. Within the following experiments, we configure the algorithm to use a single gray value feature channel and the compactness term. We control the number of generated superpixels by initializing the label array with a 'blind' segmentation consisting of equally-sized square blocks. Following this initialization, we perform twelve passes of the contour relaxation. The clique costs for inhomogenous horizontal or vertical and diagonal cliques were set to $B = 0.3$ and $C = \frac{\sqrt{0.3}}{\sqrt{2}}$, respectively, The value of the compactness parameter κ was 0.015. These parameters have been kept constant for all the experiments. We compute the boundary recall and undersegmentation error for 200, 300, 400, 500, 600, 800, 1000 and 1200 superpixels by averaging over these measures for the 300 images from the BSD. We compare our results with the approach by Veklser et al. [9] using their publicly available implementation[2] while setting the parameters to the ones reported in their paper. Note that it is not possible to directly control the number of superpixels generated by their algorithm but indirectly with a patch size parameter. We set the patch size parameter such that on average the method computes, say, 500 superpixels while the exact number of superpixels produced can vary for each individual image. Figure 1 shows the boundary recall and undersegmentation error for the proposed method (`oursGray`) and the one by Veksler et al., specifically the 'constant' version (`vekslerConst`) (cf. Sec. 2). Note that Veksler et al. use a global optimization method (graph cuts), while we achieve our results merely with local greedy optimization. In order to allow an evaluation of the influence

[2] `http://www.csd.uwo.ca/faculty/olga/Code/`

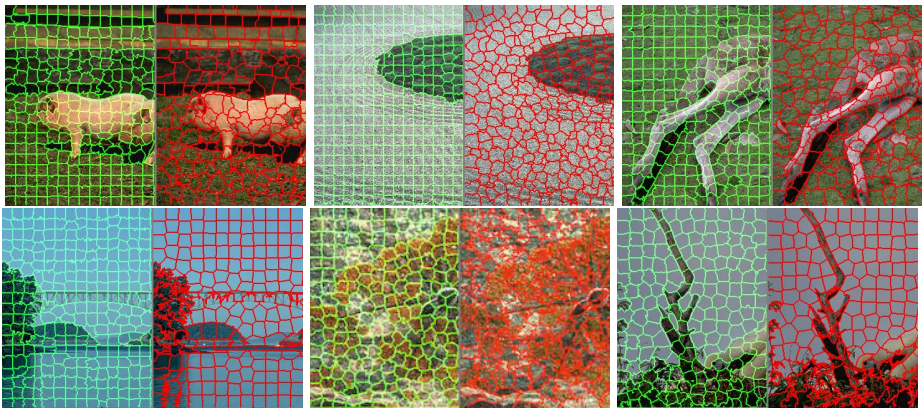


Fig. 3. *Contour-relaxed superpixels* results on a set of test images from the Berkeley database. The green boundaries show the superpixel results produced by our method compared to (top row) Veksler et al. in red when initialized with 500 (column 1 and 3) and 600 (column 2) superpixels and (bottom row) to SLIC in red when initialized with 400 (column 1 and 2) and 500 (column 3) superpixels. Best viewed in color.

of the number of iterations during optimization, we additionally compute the R_B and E_{us} measures while letting our method perform as many runs of the contour relaxation until no label changes occur, that is until a local optimum is reached (`oursGrayLOPT` in Fig. 1). From Fig. 1 it can be seen that while `vekslerConst` achieves a better boundary recall when the desired number of superpixels is relatively small (200 to 400), our *contour-relaxed superpixels* method performs better in terms of boundary recall for 550 or more superpixels. Furthermore, our method consistently outperforms its competitor `vekslerConst` with respect to undersegmentation error. Figure 1 also shows that the actual number of iterations of the contour relaxation influences the results only marginally. While more iterations slightly improve the boundary recall for 200 to 400 superpixels, they nearly have no influence on the undersegmentation error and are non-critical in practice. Figure 2 (left) shows graphs of the average running time over 200 to 1200 superpixels. While `vekslerConst` roughly takes 7 to 12 seconds, our method only takes 0.3 to 0.7 seconds. Note that the runtime of our method slightly increases with the number of superpixels: as more superpixels lead to more contour pixels in the label array, more pixels have to be visited and evaluated for a possible label change. Our method also compares favorably to Lattice- and QPBO superpixels where the authors of [14] report an average runtime for QPBO superpixels (independent of the number of superpixels) of 0.5 seconds, but are outperformed by the Veksler et al. approach with respect to boundary recall and undersegmentation error. Furthermore, [14] report that their method is usually 10% slower than Lattice superpixels. Our method has a comparable runtime while outperforming both methods in terms of quality, as expressed by R_B and E_{us}.

Multi-channel Superpixels. Within the following experiments we set up our algorithm using three independent feature channels one for each of the three YUV color channels and the compactness term. We use the same label array initialization as for the single feature case and only rescale the compactness parameter κ to 0.045 to account for the increased number of feature channels. Again, these parameters have been kept fixed while computing the boundary recall and undersegmentation error for 200, 300, 400, 500, 600, 800, 1000 and 1200 superpixels by averaging over these measures for the 300 images from the BSD. We compare our results with SLIC superpixels using their publicly available implementation[3] where the exact number of superpixels can be controlled. As for the approach by Veksler et al., we set the parameters of the SLIC implementation to the ones reported by the authors in the corresponding paper. Figure 1 shows the boundary recall and undersegmentation error for the proposed method (`oursColor` and `oursColorLOPT`) and SLIC (`SLIC`). It can be seen that we consistently perform better than `SLIC` with respect to undersegmentation error even for few superpixels. Note that the margin by which we outperform `SLIC` increases with the number of superpixels. For 200 to 750 superpixels `SLIC` achieves a better boundary recall, however, our method is coequal or better than `SLIC` for 750 or more superpixels. As for the single feature case, the actual number of iterations of contour relaxation only has a marginal effect on the results as can be seen from the boundary recall and undersegmentation error for `oursColorLOPT` compared to `oursColor`. Figure 2 (left) shows graphs of the average running time over 200 to 1200 superpixels. Here, `SLIC` takes 0.25 seconds independent of the number of superpixels while our method needs 0.6 to 1.7 seconds depending on the number of superpixels. Note that our method can easily be parallelized, that is by computing the region statistics for each channel in parallel thus making it possible to achieve a similar runtime as for the single channel version with 0.3 to 0.7 seconds. Furthermore our runtime performance also compares favorably over TurboPixels and NCuts which take several seconds and more than 30 seconds (both depending on the number of superpixels, see [8]), respectively. For a qualitative evaluation, Fig. 3 shows example superpixel segmentations using the proposed *contour-relaxed superpixels* approach, the method of Veksler et al. and SLIC. One can see that the obtained boundary maps respect the prominent boundaries of the image, and furthermore, the effect of the compactness term is clearly visible, as the resulting regions are compact and regularly shaped. As can be seen in Fig. 3, the *contour-relaxed superpixels* approach considers the strongly textured areas as homogeneous and textured, and does not spend 'energy' on complicated region contours, in contrast to Veksler et al. and SLIC shown besides. Furthermore, Fig. 2 (right) shows that the initial segmentation only has a minor influence on the results and that our approach does not have an intrinsic bias towards a specific boundary layout (horizontal/vertical). While in areas of homogeneous texture the segmentation stays close to the initial partition, superpixels adapt to shapes as necessary.

[3] `http://ivrg.epfl.ch/supplementary_material/RK_SLICSuperpixels/index.html`

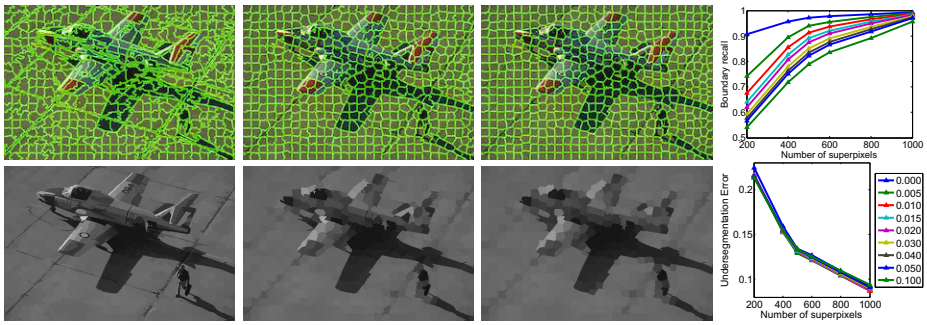


Fig. 4. Influence of varying the compactness weight: Top row, columns 1-3: *contour-relaxed superpixels* results for the 'plane' image when setting the compactness weight to 0, 0.015 and 0.03, respectively. Bottom row, columns 1-3: Region mean images corresponding to the superpixel segmentations in the top row. Column 4 shows the boundary recall and undersegmentation error over the number of superpixels averaged over all images from the BSDS300 while varying the compactness weight between 0 and 0.1. Best viewed in color.

Increasing Boundary Recall by Lowering the Compactness Weight. The precision with which the boundaries are recovered can additionally be improved, leading to a higher boundary recall by decreasing the weight of the compactness parameter κ whereas the number of relaxation passes (here: 12) only has a minor influence as shown previously. Figure 4 shows several superpixel segmentations and corresponding region mean images for the 'plane' image from the BSD visualizing the effect of lowering the compactness weight on the resulting superpixels. When the compactness term is set to 0 we can achieve a boundary recall of around 0.9 for only 200 superpixels as shown in the recall graph in Fig. 4. The spread of a superpixel is then only governed by the inhomogeneous clique potentials. In contrast to the common view in the literature [8–10] that superpixel algorithms should include a compactness term in order to avoid bleeding effects, the non-compact version of our method is only marginally influenced with respect to the undersegmentation error as shown in the graph in Fig. 4, as due to our model the superpixels still respect object boundaries. However, in this setting superpixels have a highly irregular shape which might be suboptimal for different applications. Furthermore Fig. 4 shows that when increasing the compactness weight superpixels tend to be more regular in shape and have similar spatial extent. This also comes at a cost, namely that the region mean images do not represent the input image as well as the one for the non-compact version, but are merely a subsampled version.

Segmenting a 'Texture-Only' Image. Our final (extreme) experiment shows how our approach compares to the method of Veksler et al. and SLIC when applied to a synthetic image where all regions have the same mean gray value but differ in variance. Figure 5 (top row) depicts that SLIC fails to extract any useful superpixels, while the method of Veksler et al. has problems especially in regions

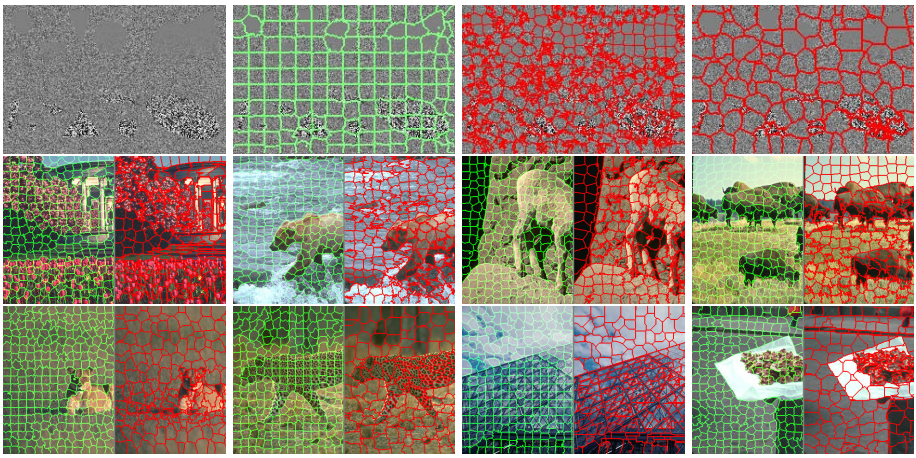


Fig. 5. Top row: Texture (or noise) sensitivity: Test using an image with regions differing only in variance, exploring the case of highly textured regions. Left to right: input image, results from proposed method, `SLIC` and `vekslerConst`. **Middle and bottom row** show typical results for highly textured scenes obtained with the proposed method (green boundaries) compared to `SLIC` (red boundaries, second row), and `vekslerConst` (red boundaries, third row), respectively. Note that all methods were initialised with the same number of superpixels, but `SLIC` and `vekslerConst` may reduce the number of superpixels in the final result. Best viewed in color.

of high variance. While our method also does not give fully satisfactory results in this extreme case, it can be seen that, due to the emphasis on statistically homogeneity instead of smoothness, the proposed method does not spend 'energy' on random region contours and regards textured areas as such. Furthermore, Fig. 5 (row 2 and 3) shows that this is not a contrived experiment, but that this effect is also present in natural images.

7 Conclusions

We have enhanced the *contour-relaxed superpixels* approach and performed an extensive qualitative and quantitative evaluation of this method which is based on a statistical image model and a simple, but efficient optimization scheme. Due to the new compactness term, our approach performs, in terms of standard superpixel benchmarks, comparable, mostly even better than state-of-the-art approaches such as SLIC or the Veksler et al. method. This suggests that the choice of the energy function (derived from a statistical model in our case) might be at least equally important as the optimization method. Computationally, the proposed approach compares similar or favorably against leading state-of-the-art methods. The design parameters of the method allow to tune its behavior in a goal-directed way between high precision of boundaries, strong homogeneity of the region-internal texture, and smoothness/compactness of boundaries.

Acknowledgements. This work was supported by the German Federal Ministry of Education and Research (BMBF) in the project Bernstein Fokus Neurotechnologie – Frankfurt Vision Initiative 01GQ0841, and in parts supported by the ELLIIT programme funded by the Swedish Government.

References

1. Ren, X., Malik, J.: Learning a classification model for segmentation. In: ICCV (2003)
2. Ochs, P., Brox, T.: Object segmentation in video: A hierarchical variational approach for turning point trajectories into dense regions. In: ICCV (2011)
3. Wang, S., Lu, H., Yang, F., Yang, M.: Superpixel tracking. In: ICCV (2011)
4. Gorelick, L., Delong, A., Veksler, O., Boykov, Y.: Recursive MDL via graph cuts: Application to segmentation. In: ICCV (2011)
5. Mori, G., Ren, X., Efros, A.A., Malik, J.: Recovering human body configurations: combining segmentation and recognition. In: CVPR, pp. 326–333 (2004)
6. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. PAMI 24, 603–619 (2002)
7. Vincent, L., Soille, P.: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. PAMI 13, 583–598 (1991)
8. Levinshtein, A., Stere, A., Kutulakos, K., Fleet, D., Dickinson, S., Siddiqi, K.: Turbopixels: Fast superpixels using geometric flows. PAMI (2009)
9. Veksler, O., Boykov, Y., Mehrani, P.: Superpixels and supervoxels in an energy optimization framework. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 211–224. Springer, Heidelberg (2010)
10. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Suesstrunk, S.: SLIC Superpixels. Technical Report Nr. 149300, EPFL, Lausanne (CH) (2010)
11. Shi, J., Malik, J.: Normalized cuts and image segmentation. TPAMI 22 (2000)
12. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. IJCV (2004)
13. Moore, A., Prince, S., Warrell, J., Mohammed, U., Jones, G.: Superpixel lattices. In: CVPR (2008)
14. Zhang, Y., Hartley, R., Mashford, J., Burn, S.: Superpixels via pseudo-boolean optimization. In: ICCV (2011)
15. Mester, R., Conrad, C., Guevara, A.: Multichannel segmentation using contour relaxation: Fast super-pixels and temporal propagation. In: Heyden, A., Kahl, F. (eds.) SCIA 2011. LNCS, vol. 6688, pp. 250–261. Springer, Heidelberg (2011)
16. Guevara, A., Conrad, C., Mester, R.: Boosting segmentation results by contour relaxation. In: ICIP (2011)
17. Besag, J.: On the statistical analysis of dirty pictures. Journal of the Royal Statistical Society, Series B 48, 259–302 (1986)
18. Besag, J.: Spatial interaction and the statistical analysis of lattice systems. Journal of the RSS, Series B 36, 192–236 (1974)
19. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: ICCV, vol. 2, pp. 416–423. IEEE (2001)
20. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. TPAMI (2010)

Sparse-MIML: A Sparsity-Based Multi-Instance Multi-Learning Algorithm

Chenyang Shen[1], Liping Jing[2], and Michael K. Ng[1]

[1] Department of Mathematics, Hong Kong Baptist University
[2] School of Computer and Information Technology Beijing Jiaotong University

Abstract. Multi-Instance Multi-Label (MIML) learning is one of challenging research problems in machine learning. The main aim of this paper is to propose and develop a novel sparsity-based MIML learning algorithm. Our idea is to formulate and construct a transductive objective function for labels indicator to be learned by using the method of random walk with restart that exploits the relationships among instances and labels of objects, and computes the affinities among the objects. Then sparsity can be introduced in the labels indicator of the objective function such that relevant and irrelevant objects with respect to a given class can be distinguished. The resulting sparsity-based MIML model can be given as a constrained convex optimization problem, and it can be solved very efficiently by using the augmented Lagrangian method. Experimental results on benchmark data have shown that the proposed sparse-MIML algorithm is computationally efficient, and effective in label prediction for MIML data. We demonstrate that the performance of the proposed method is better than the other testing MIML learning algorithms.

Keywords: Sparsity, multi-instance multi-label data, label ranking, Markov chain, iterative methods.

1 Introduction

In this paper, we study Multi-Instance Multi-Label (MIML) learning problems [13]. Let $\mathcal{X}$ be a set of objects and $\mathcal{Y}$ be a set of labels or classes. We denote the size of $\mathcal{Y}$ by $c = |\mathcal{Y}|$, and the size of $\mathcal{X}$ by $m = m' + m" = |\mathcal{X}|$, where m' and $m"$ are the sizes of training data and testing data, respectively. In the single-instance single-label learning setting, each object (i.e., only one instance) $X \in \mathcal{X}$ is assigned a single class $Y \in \mathcal{Y}$. In the multi-instance multi-label setting, the training data set is $\{(X_1, Y_1), (X_2, Y_2), ..., (X_m, Y_m)\}$, where the i-th object $X_i = \{x_1^{(i)}, ..., x_{n_i}^{(i)}\}$ contains a bag of n_i instances, and $Y_i = \{y_1^{(i)}, ..., y_{l_i}^{(i)}\} \subset \{1, 2, ..., c\}$ is a set of labels assigned to X_i. The testing data set is $X_{m'+1}, ..., X_{m'+m"}$ without labels information. It is clear that the single-instance single-label scenario is a special case of MIML setting where $n_i = 1$ and $l_i = 1$ for all X_i and Y_i. The MIML learning task is to learn a classifier $\Phi : \mathcal{X} \to \mathcal{Y}$ which minimizes the probability that $\hat{Y} \neq \Phi(\hat{X})$ on a testing object $\hat{X}$ with its set of labels $\hat{Y}$. Compared

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 294–306, 2013.
© Springer-Verlag Berlin Heidelberg 2013

to single-instance single-label learning problems, the MIML learning problems are more practical to represent complicated objects which have multiple semantic meanings. For example, an image can contain multiple patches where each can be represented by an instance, and the image can belong to several classes simultaneously; a document can contain multiple sections where each of which can be represented as an instance, and the document can belong to different categories.

In this paper, we are interested in classifiers which can generate a ranking of labels for a given object such that its correct labels receive higher ranking than the other irrelevant labels. Our aim is to compute labels indicator F for a testing $\hat{X}$ with its set of labels $\hat{Y}$ such that the labels of $\hat{X}$ in $\mathcal{Y}$ should be ordered according to F, i.e., if $F(\hat{X}, y) > F(\hat{X}, y')$, then the label y is ranked higher than the label y'. The classifier is evaluated in terms of its ability to predict a good approximation for $\hat{Y}$ based on the ranking by F. In particular, we would like to obtain the ranking of labels in $\hat{Y}$ is higher than the ranking of those not in $\hat{Y}$.

The main contribution of this paper is to propose and develop a novel sparsity-based MIML learning algorithm to compute a ranking of labels associated with objects of multiple instances. We exploit the relationships among instances and labels of objects so that the ranking of a class label to an object depends on the affinity metric between the bag of instances of this object and the bag of instances of the other objects, and the ranking of a class label of similar objects. By employing instance-to-object relation matrix and computing the affinity metric among the objects, a transductive objective function for labels indicator to be learned can be constructed. Moreover, sparsity can be introduced in the labels indicator of the objective function such that relevant and irrelevant objects with respect to a given class can be distinguished. For example, we show in Figure 1 the sparsity of the association of class labels and testing objects from the two benchmark MIML data sets [13]. We see from the figure that each object associates a few classes. The resulting sparsity-based MIML model can be formulated as a constrained convex optimization problem, and it can be solved very efficiently by using the augmented Lagrangian method based on separable structure of the objective function. Experimental results on benchmark data will show that the proposed sparse-MIML algorithm is computationally efficient, and effective in label prediction for MIML data, and its performance is better than the other MIML learning algorithms.

The rest of the paper is organized as follows. In Section 2, we formulate the objective function based on affinity metric and sparsity. In Section 3, we present the proposed algorithm. In Section 4, we review the related works. In Section 5, we present and discuss the experimental results on two benchmark MIML data sets. Finally, we give some concluding remarks in Section 6.

2 The Proposed Model

The main idea of the method of random walk with restart is to set up the affinities among objects on MIML data, and initialize label information from

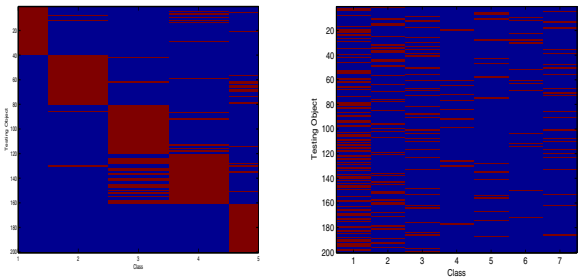


Fig. 1. Sparsity of MIML image data (left) and MIML text data (right). A red (blue) pixel refers that an object belongs (does not belong) to a class.

labeled objects [7]. As there are multiple instances among objects in MIML data, we evaluate the affinity between the two objects which may have different set of instances. An affinity matrix is constructed to represent affinity among instances, and then employ instance-to-object-relation matrix to transfer the affinity information among instances to the label information among objects. We consider that the instances are ordered as follows:

$$\underbrace{x_1^{(1)}, ..., x_{n_1}^{(1)}}_{\text{1st object}}, \underbrace{x_1^{(2)}, ..., x_{n_2}^{(2)}}_{\text{2nd object}}, \cdots, \underbrace{x_1^{(m)}, ..., x_{n_m}^{(m)}}_{\text{mth object}}$$

For simplicity, we set n to be the total number of instances in the MIML data, i.e., $n = \sum_{i=1}^{m} n_i$. Let $a_{i,j,s,t}$ be the affinity between the s-th instance of the i-th object and the t-th instance of the j-th object. Here we employ the Gaussian kernel as the affinity function similar to that used in other MIML learning algorithms [11–13]:

$$a_{i,j,s,t} = \exp\left[\frac{-||x_s^{(i)} - x_t^{(j)}||_2^2}{2\sigma^2}\right],$$

where $||\cdot||_2$ is the Euclidean distance. An n-by-n block matrix $\mathbf{A} = [\mathbf{A}_{i,j}]$ where the (i,j)-th block is an n_i-by-n_j matrix $\mathbf{A}_{i,j} = [a_{i,j,s,t}]$ with $s = 1, \cdots, n_i$ and $t = 1, \cdots, n_j$ is obtained. Note that $\mathbf{A}$ is a symmetric matrix. Moreover, a block diagonal matrix $\mathbf{B} = [\mathbf{B}_{i,j}]$ where the (i,j)-th block is a zero matrix except $i = j$. For the (i,i)-th block, $\mathbf{B}_{i,i}$ is a 1-by-n_i matrix where all its entries are equal to 1. This block indicates the relation between the i-th object and its association instances. The size of $\mathbf{B}$ is m-by-n, and it refers to be an object-to-instance relation matrix that can be used to transfer from the affinity information at the instance level to the object level. The resulting m-by-m matrix

$$\mathbf{S} = \mathbf{BAB}^T \tag{1}$$

represents the affinities among objects.

For label prediction problem with partial labeled data, the method of random walk with restart has probability α to return to the labeled objects. It can be

interpreted that during each time step each object receives the label information from its neighbors via the Markov chain, and also retains its initial label information. The parameter α in between 0 and 1 specifies the relative amount of the information from its neighbors and its initial label information. In this approach, the steady state probabilities give ranking of labels to indicate the importance of a set of labels to a unlabeled object [7]. The steady state probabilities can be computed via the following recursive procedure:

$$\mathbf{F}_{(t+1)} = (1-\alpha)\mathbf{S}\mathbf{D}^{-1}\mathbf{F}_{(t)} + \alpha\mathbf{P}, \tag{2}$$

where $\mathbf{D}$ is a diagonal matrix with its main diagonal entry given by $[\mathbf{D}]_{i,i} = \sum_{k=1}^{m}[\mathbf{S}]_{k,i}$. Here $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_c]$ is the labels indicator to be learned, where $\mathbf{f}_l$ is a vector of size m corresponding to the l-th class label, and $\mathbf{F}_{(t)}$ refers to the iterate of $\mathbf{F}$ at the t-th iteration. Also $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_c]$ is the given labels of training objects, where $\mathbf{p}_l$ is a vector of size m corresponding to the l-th class label. One way to construct $\mathbf{p}_l$ is by using an uniform distribution on the objects with the label class l. More precisely,

$$[\mathbf{p}_l]_i = \begin{cases} 1/e_l, & \text{if } l \in Y_i \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

where e_l is the number of objects with the label class l in the training data. Note that the summation of entries of $\mathbf{p}_l$ is equal to 1, and the summation of entries of $\mathbf{f}_l$ is also equal to 1 via the iteration in (2).

By using the iteration in (2), we have

$$\mathbf{F}_{(t)} = (1-\alpha\mathbf{S}\mathbf{D}^{-1})^{t-1}\mathbf{F}_{(0)} + \alpha\sum_{i=0}^{t-1}((1-\alpha)\mathbf{S}\mathbf{D}^{-1})^{i}\mathbf{D},$$

where $\mathbf{F}_{(0)}$ is the initial estimate of the labels indicator. Since α is in between 0 and 1 and the spectral radius of $\mathbf{S}\mathbf{D}^{-1}$ is equal to 1[1], we have $\lim_{t\to\infty}((1-\alpha)\mathbf{S}\mathbf{D}^{-1})^{t-1}\mathbf{F}_{(0)} = \mathbf{0}$, and $\lim_{t\to\infty}\sum_{i=0}^{t-1}((1-\alpha)\mathbf{S}\mathbf{D}^{-1})^{i}\mathbf{P} = (\mathbf{I}-(1-\alpha)\mathbf{S}\mathbf{D}^{-1})^{-1}\mathbf{P}$, where $\mathbf{0}$ is a zero matrix, and $\mathbf{I}$ is an m-by-m matrix. Hence, the sequence $\{\mathbf{F}_{(t)}\}$ converges to $\mathbf{F}^* = \alpha(\mathbf{I}-(1-\alpha)\mathbf{S}\mathbf{D}^{-1})^{-1}\mathbf{P}$ or

$$\hat{\mathbf{F}}^* = \alpha(\mathbf{I}-(1-\alpha)\mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2})^{-1}\hat{\mathbf{P}}, \tag{4}$$

where $\hat{\mathbf{F}}^* = \mathbf{D}^{-1/2}\mathbf{F}^*$ and $\hat{\mathbf{P}} = \mathbf{D}^{-1/2}\mathbf{P}$. Indeed, this solution can be obtained by minimizing the following objective function:

$$J_1(\hat{\mathbf{F}}) = tr(\hat{\mathbf{F}}^T(\mathbf{I}-\mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2})\hat{\mathbf{F}}) + \mu\|\hat{\mathbf{F}}-\hat{\mathbf{P}}\|_F^2. \tag{5}$$

where $tr(\cdot)$ is the trace of a matrix, $\|\cdot\|_F$ denotes the Frobenius norm of a matrix, and $\mu = \frac{\alpha}{1-\alpha}$. Note that the matrix $\mathbf{I}-\mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2}$ can be interpreted as the

[1] Each column sum of $\mathbf{S}\mathbf{D}^{-1}$ is equal to 1 and the results follow by using the Perron Frobenius Theorem.

graph Laplacian, see [2]. As $\mathbf{I} - \mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2}$ is similar to $\mathbf{I} - \mathbf{S}\mathbf{D}^{-1}$ and the spectral radius of $\mathbf{S}\mathbf{D}^{-1}$ is also equal to 1. Therefore, $\mathbf{I} - \mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2}$ is positive semi-definite, and (5) is a convex optimization problem. The optimization problem in (5) makes use of relationships among instances and labels of objects. An object, which contains a bag of instances that are highly similar to bags of instances of the other objects with a particular label, receives a high ranking of this label.

2.1 The Sparse Model

We note from (2) that the calculation of each column of $\mathbf{F}$ is independent, and the solution $\mathbf{f}_l^*$ is dependent on the iteration matrix $(1-\alpha)\mathbf{S}\mathbf{D}^{-1}$ and the l-th label vector $\mathbf{p}_l$. The disadvantage of this approach is that we do not make use of given label information from different classes to compute labels indicator. The main contribution of this paper is to explore the property of $\mathbf{F}$ (or $\hat{\mathbf{F}} = \mathbf{D}^{-1/2}\mathbf{F}$) to enhance classification accuracy by using given label information from different classes. Our idea is to use the sparsity of $\mathbf{F}$ in the objective function. H The aim is to distinguish the relevant and irrelevant objects with respect to a given class, i.e., each column vector of $\mathbf{F}$: $[[\mathbf{F}]_{1,l}, [\mathbf{F}]_{2,l}, \cdots, [\mathbf{F}]_{n,l}]$ should be sparse. By combining these two sparse constraints in the objective function, we can reformulate J_1 in (5) as follows:

$$\min_{\hat{\mathbf{F}}} J(\hat{\mathbf{F}}) = tr(\hat{\mathbf{F}}^T\mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2}\hat{\mathbf{F}}) + \mu\|\hat{\mathbf{F}} - \hat{\mathbf{P}}\|_F^2 + \lambda\sum_{l=1}^{c}\sqrt{\sum_{i=1}^{n}[\hat{\mathbf{F}}]_{i,l}^2} \quad (6)$$

Since the element $[\mathbf{F}]_{i,l}$ is in between 0 and 1, $[\hat{\mathbf{F}}]_{i,l}$ is in between 0 and $b = \sqrt{\sum_{i=1}^{n}[\mathbf{S}]_{k,i}}$.

The first term of the right-hand side in the above cost function is the smoothness constraint, which means that a proper labels indicator $\hat{\mathbf{F}}$ should not change too much between neighbor objects. To minimize the first term, we expect that if two objects X_i and X_j are close (i.e., $[\mathbf{S}]_{i,j}$ is large or $(\mathbf{I} - \mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2})$ is small), $[\hat{\mathbf{F}}]_{i,\cdot}$ and $[\hat{\mathbf{F}}]_{j,\cdot}$ are also close to each other. The second term is the fitting constraint, which means that a good labels indicator should not change too much from the initial label assignment. The third term is the sparsity requirement, which means that each object labels indicator should be corresponding to a few categories. Because of the sparsity constraint, both intra-class (column vector of $\hat{\mathbf{F}}$ and inter-class (row vector of $\hat{\mathbf{F}}$) label rankings are considered together in the model. The trade-off between these three terms is captured by two positive parameters μ and λ.

After solving $\hat{\mathbf{F}}$ (i.e, $\mathbf{F}$) by using (6), we generate a ranking of the possible labels for a testing object X_i by ordering the values of each row of $\mathbf{F}$: $[\mathbf{F}]_{i,1}, [\mathbf{F}]_{i,2}, \cdots, [\mathbf{F}]_{i,c}$ where c refers to the number of classes, such that its correct labels receive higher ranking than the other irrelevant labels.

3 The Proposed Algorithm

The objective function in (6) is a constrained nonlinear optimization problem with respect to $\hat{\mathbf{F}}$. Note that $J(\hat{\mathbf{F}})$ is a quadratic and continuously differentiable function and the constraints are bounded and closed, and therefore there is one and only one minimizer of J. To handle the constrains of $\hat{\mathbf{F}}$ efficiently, we introduce two new variables $\hat{\mathbf{G}}$ and $\hat{\mathbf{H}}$ and set $\hat{\mathbf{F}} = \hat{\mathbf{G}}$ and $\hat{\mathbf{G}} = \hat{\mathbf{H}}$. The corresponding augmented Lagrangian function J_a of (6) is given by

$$\begin{aligned} & J_a(\hat{\mathbf{F}}, \hat{\mathbf{G}}, \hat{\mathbf{H}}, \Gamma^{(1)}, \Gamma^{(2)}) \\ &= tr(\hat{\mathbf{F}}^T(\mathbf{I} - \mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2})\hat{\mathbf{F}}) + \mu\|\hat{\mathbf{G}} - \hat{\mathbf{P}}\|_F^2 + \lambda \sum_{l=1}^{c} \sqrt{\sum_{i=1}^{n} [\hat{\mathbf{F}}]_{i,l}^2} + \\ & \mathbf{\Gamma}^{(1)} \circ (\hat{\mathbf{F}} - \hat{\mathbf{G}}) + \mathbf{\Gamma}^{(2)} \circ (\hat{\mathbf{G}} - \hat{\mathbf{H}}) + \beta_1\|\hat{\mathbf{F}} - \hat{\mathbf{G}}\|_F^2 + \beta_2\|\hat{\mathbf{G}} - \hat{\mathbf{H}}\|_F^2, \end{aligned} \quad (7)$$

with $0 \leq [\hat{\mathbf{H}}]_{i,l} \leq b$, where $\mathbf{\Gamma}^{(1)}$ and $\mathbf{\Gamma}^{(2)}$ are the Lagrangian multipliers of same size of $\hat{\mathbf{F}}$, $\circ$ denotes the entrywise multiplication of two matrices, and β_1 and β_2 are positive penalty parameters which are used to control the convergence of the augmented Lagrangian method (ALM)[2]. Now $(\hat{\mathbf{F}}, \hat{\mathbf{H}})$ and $\hat{\mathbf{G}}$ in (7) can be minimized separately, and each minimization subproblem has a closed-form solution. More specifically, with a given initial $(\hat{\mathbf{G}}_{(0)}, \mathbf{\Gamma}_{(0)}^{(1)}, \mathbf{\Gamma}_{(0)}^{(2)})$, the ALM approaches the solutions of (7) via the following iterative scheme (t refers to the iteration index):

$$\begin{aligned} \hat{\mathbf{F}}_{t+1} &= \arg\min_{\hat{\mathbf{F}}} J_a(\hat{\mathbf{F}}, \hat{\mathbf{G}}_{(t)}, \hat{\mathbf{H}}_{(t)}, \mathbf{\Gamma}_{(t)}^{(1)}, \mathbf{\Gamma}_{(t)}^{(2)}) \\ &= ((\mathbf{I} - \mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2}) + \beta_1\mathbf{I} + \mu\mathbf{I})^{-1}(\mu\hat{\mathbf{P}} + \beta_1\hat{\mathbf{G}}_t - \mathbf{\Gamma^{(1)}}_t); \end{aligned} \quad (8)$$

for $1 \leq l \leq c$ (for each column),

$$\begin{aligned} \hat{\mathbf{H}}_{:,l}^{t+1} &= \arg\min_{\hat{\mathbf{H}}} J_a(\hat{\mathbf{F}}_{(t+1)}, \hat{\mathbf{G}}_{(t)}, \hat{\mathbf{H}}, \mathbf{\Gamma}_{(t)}^{(1)}, \mathbf{\Gamma}_{(t)}^{(2)}) \quad \text{subject to} \quad 0 \leq [\hat{\mathbf{H}}]_{i,l} \leq b \\ &= \frac{\hat{\mathbf{G}}_{:,l}^{t+1} + \frac{1}{\beta_2}(\mathbf{\Gamma^{(2)}}_t)_{:,l}}{\|\hat{\mathbf{G}}_{:,l}^{t+1} + \frac{1}{\beta_2}(\mathbf{\Gamma^{(2)}}_t)_{:,l}\|_2} \max\left\{\left\|\hat{\mathbf{G}}_{:,l}^{t+1} + \frac{1}{\beta_2}(\mathbf{\Gamma^{(2)}}_t)_{:,l}\right\|_2 - \frac{\lambda}{\beta_2}, 0\right\}; \end{aligned} \quad (9)$$

$$\hat{\mathbf{G}}_{t+1} = \max\left\{\frac{\mathbf{\Gamma^{(1)}}_t - \mathbf{\Gamma^{(2)}}_t + \beta_1\hat{\mathbf{F}}_{t+1} + \beta_2\hat{\mathbf{H}}_t}{\beta_1 + \beta_2}, 0 \cdot \mathbf{E}\right\} \quad (10)$$

where $\mathbf{E}$ is a matrix of all entries being equal to one, and the maximum operation is entrywise based comparison;

$$\mathbf{\Gamma^{(1)}}_{t+1} = \mathbf{\Gamma^{(1)}}_t + \frac{\beta_1}{2}(\hat{\mathbf{F}}^{t+1} - \hat{\mathbf{G}}^{t+1}); \quad (11)$$

$$\mathbf{\Gamma^{(2)}}_{t+1} = \mathbf{\Gamma^{(2)}}_t + \frac{\beta_2}{2}(\hat{\mathbf{G}}^{t+1} - \hat{\mathbf{H}}^{t+1}). \quad (12)$$

[2] Theoretically, the ALM method is convergent for any constant $\beta_1, \beta_2 > 0$. If necessary, we can adjust their values of β_1 and β_2 dynamically in order to achieve better numerical performance, see [5] for a detailed discussion.

Theoretically, the convergence of the iteration steps in (8), (9), (10), (11) and (12) can be guaranteed and the the iterate converges to the the unique minimizer of J in (6).

We remark that all the computations of $\hat{\mathbf{G}}_{(t+1)}$, $\hat{\mathbf{H}}_{(t+1)}$, $\mathbf{\Gamma}^{(1)}_{(t+1)}$, $\mathbf{\Gamma}^{(2)}_{(t+1)}$ can be carried out independently at each entry. These computation tasks can be done very efficiently. As for calculation of $\hat{\mathbf{F}}_{(t+1)}$, a m-by-m linear system is required to solved. The main computational cost and storage is to build and store the whole affinity matrix $\mathbf{S}$ or $\mathbf{A}$ in (1). In the implementation, we figure out that there are many entries of the affinity matrix $\mathbf{A}$ being close to zero. Therefore it may not be necessary to store these entries such that the computational cost and the storage can be reduced. For example, we can generate a sparse and symmetric affinity matrix $\mathbf{A}_\kappa$ as follows [8]:

$$a_{i,j,s,t} = \begin{cases} \exp[-||x_s^{(i)} - x_t^{(j)}||_2 \Big/ 2\sigma^2], \text{ if } x_t^{(j)} \in \mathcal{N}_\kappa, \\ 0, \qquad\qquad\qquad \text{otherwise.} \end{cases}$$

where $\mathcal{N}_\kappa$ is the set of κ nearest neighbors of instance $x_s^{(i)}$. The κ nearest neighbors can be efficient searched by using kd-tree implementation. When $\mathbf{A}_\kappa$ is sparse, the resulting affinity matrix $\mathbf{S}_\kappa = \mathbf{B}\mathbf{A}_\kappa\mathbf{B}^T$ is also sparse and the corresponding sparse linear system can be solved very efficiently.

4 Related Works

Multi-Instance (MI) learning problems [6] and Multi-Label (ML) learning problems [3] can be regarded as degenerated problems of Multi-Instance Multi-Label learning problems. Therefore, it is natural to solve a MIML problem by decomposing it into ML or MI learning problems. For example, a MIML algorithm based on support vector machine (MIML-SVM) has been developed [12] by transforming a MIML problem into several single-instance multi-label problems. Similarly, by using a lazy learning approach to multi-label learning to replace the support vector machine used in MIML-SVM, a MIML-kNN algorithm is derived [13]. On the other hand, a MIML algorithm based on boosting (MIML-Boost) [12] has been developed by transforming a MIML problem into several multi-instance single-label problems. Remark that it is very time-consuming to use MIML-Boost, see [10]. Also for these two approaches, there may be some useful information to be lost in the process and therefore the classification performance may be degraded. Recently, Zhang and Zhou [9] designed a M^{3}MIML algorithm based on regularization to explicitly exploit the relationships between instances and labels. Experimental results have shown that this algorithm achieves superior performance than MIML-SVM and MIML-Boost, but the cost of this learning algorithm is quite high.

In [7], the method of random walk with restart (MIML-RWR) is developed for learning MIML data. The key iteration is given in (2). Experimental results on

benchmark data have shown that the classification performance of MIML-RWR is competitive with those of MIML-SVM, MIML-Boost and M^{3}MIML, but the computational time required is less than those by the other three algorithms. Their main idea of the MIML-RWR is basically a kind of nearest neighbor approach which makes use of neighbors' information to learn the correct labels. The drawback of the algorithm is that each column of the solution is independent, see (4), and the algorithm does not make use of each class information to learn MIML data. In the proposed sparse-MIML algorithm, we also make use of neighbors' information to learn the correct labels, but we also employ the sparsity to learn intra-class and inter-class label information among the objects.

5 Experimental Results

To evaluate the performance of the proposed sparse-MIML algorithm, we conduct experiments on two benchmark MIML data sets. The first one is the image classification task given in [12] to study the MIML framework[3]. In summary, this data set contains 2,000 scene images taken from five possible class labels (desert, mountains, sea, sunset and trees) and each image is represented as a bag of 9 instances in 15-dimensional using the SBN image bag generator [4]. Another data set is the widely used Reuters-21578 text collection in text categorization[4]. In summary, this data set contains 2,000 documents with multiple classes and represented as a bag of instances based on the techniques of sliding windows, where each instance corresponds to a text segment enclosed in a sliding window [1]. The description of these two data sets is listed in Table 1, see [11] for the detailed information. The text and image data are both preprocessed in our experiments. Each instance is normalized such that its Euclidean norm is equal to 1. Also we follow the setting in [11], we normalize the image data on each dimension in the range between $[0, 5]$ as shown in the M^{3}MIML algorithm. In the tests, we compare sparse-MIML with MIML-RWR[5]. All the comparisons are performed in a computer running a server environment with 2.66GHz CPU and 3.5GB memory.

The performance of multi-label prediction is evaluated by four multi-label ranking metrics: one-error, ranking loss, coverage and average precision. One-error computes how many times the top-ranked label is not relevant. Ranking loss computes the average fraction of label pairs that are not correctly ordered. Coverage determines how far one needs to go in the list of labels to cover all the relevant labels of an instance. This measure is loosely related to the precision at the level of perfect recall. Average precision determines the percentage of relevant labels among all labels that are ranked above. For their detailed definitions, we refer to [6].

[3] available at `http://lamda.nju.edu.cn/datacode/miml-image-data.htm`

[4] available at `http://lamda.nju.edu.cn/datacode/miml-text-data.htm`

[5] MIML-Boost, MIML-SVM, M^{3}MIML and MIML-kNN are not tested as its classification performance is not better than MIML-RWR as reported in [8].

Table 1. The description of image and text data for MIML learning

					Instance per bag			Labels per object (c)		
Data set	Objects	Instances	Features	classes	min	max	mean+std.	c=1	c=2	c≥3
Text	2,000	7,119	243	7	2	26	3.56+2.71	1701	290	9
Image	2,000	18,000	15	5	9	9	9.00+0.00	1543	442	15

Table 2. Percentage improvement of sparse-MIML over MIML-RWR on the text data

sparse-MIML	one-error	coverage	rankLoss	1-avgPrec
50	16.33%	4.27%	12.90%	15.00%
100	12.63%	3.45%	10.00%	12.07%
200	11.58%	3.16%	10.00%	10.37%
500	7.37%	0.87%	6.67%	6.90%
1000	1.06%	-1.74%	0.00%	0.00%
1500	6.38%	1.45%	6.67%	5.26%
all	2.15%	-3.20%	0.00%	1.75%

Table 3. Percentage improvement of sparse-MIML over MIML-RWR on the image data

sparse-MIML	one-error	coverage	rankLoss	1-avgPrec
50	9.78%	2.15%	4.88%	5.00%
100	8.05%	2.94%	5.36%	6.70%
200	7.51%	3.80%	5.78%	6.51%
500	9.88%	5.02%	7.30%	8.56%
1000	9.66%	6.15%	9.19%	4.80%
1500	9.64%	5.76%	8.47%	8.94%
all	10.81%	5.48%	8.76%	9.17%

5.1 Classification Performance

We compare the average performance sparse-MIML with MIML-RWR for the two MIML data sets. In particular, we show how many the percentage of performance improvement of sparse-MIML over MIML-RWR by varying the values of κ respectively in the construction of sparse affinity matrix $\mathbf{S}_\kappa$ in Section 3. When $\kappa =$ all, it means that all the entries are used. For the sparse-MIML, we fix $\alpha = 0.99$ and $\lambda = 0.11$ for the text data and $\alpha = 0.64$ and $\lambda = 0.17$ for the image data. For the the MIML-RWR algorithm, we choose its corresponding optimal parameters. We see from Tables 2 and 3 that sparse-MIML consistently performs better than MIML-RWR across all evaluation metrics and data sets in most cases. We also note that the computational times of sparse-MIML and MIML-RWR algorithms are about the same (sparse-MIML: about 200 seconds and MIML-RWR: about 184 seconds), and both increase when κ increases. We would like to report that the proposed algorithm takes 100 iterations in average in order to satisfy the convergence criteria $\|\hat{\mathbf{F}}_{t+1} - \hat{\mathbf{F}}_t\|_F^2 / \|\hat{\mathbf{F}}_t\|_F^2 \leq 1 \times 10^{-15}$.

Table 4. Percentage improvement of sparse-MIML over MIML-RWR for different sizes of training text data

sparse-MIML	one-error	coverage	rankLoss	1-avgPrec
10%	1.90%	1.02%	0.00%	2.02%
20%	10.45%	1.15%	0.00%	4.82%
30%	5.83%	-1.71%	0.00%	4.00%
40%	10.71%	0.00%	-2.78%	1.43%
50%	2.88%	-2.70%	-6.06%	0.00%
60%	4.90%	-2.79%	-3.13%	0.00%
70%	12.24%	-1.14%	0.00%	6.67%
80%	7.45%	-3.49%	3.33%	3.45%
90%	10.75%	0.59%	3.57%	7.27%

Table 5. Percentage improvement of sparse-MIML over MIML-RWR for different sizes of training image data

sparse-MIML	one-error	coverage	rankLoss	1-avgPrec
10%	1.72%	0.94%	1.33%	1.50%
20%	0.81%	1.58%	1.50%	1.65%
30%	1.12%	2.36%	-0.52%	0.43%
40%	2.03%	0.5%	0.00%	1.79%
50%	2.67%	1.01%	1.12%	2.27%
60%	3.90%	2.23%	3.41%	3.69%
70%	5.45%	3.47%	5.11%	5.09%
80%	4.97%	2.22%	4.17%	4.31%
90%	8.05%	2.94%	5.36%	6.70%

In the next experiment, we test the performance of sparse-MIML algorithm with respect to the number of training examples. We randomly pick up 10, 20, 30, 40, 50, 60, 70, 80, 90 percentages of the data set as training data. The remaining data set is used for testing data. The performance is measured by averaging 10 trials by randomly selected data using this procedure. Tables 4 and 5 show the improvement of classification performance of sparse-MIML over MIML-RWR on the text and image data sets respectively when $\kappa = 100$. We note that the classification performance when $\kappa = 100$ is higher than those in the other values of κ in Tables 2 and 3. For the text data, here it is optimal to use the same set of parameters in Table 2. For the image data, we use the same set of parameters in Table 3 when the percentages of the data are in between 30 and 90. However, when the percentages of the image data is 10 and 20, we need to tune the parameters to be $\alpha = 0.99$ and $\lambda = 0.1$. We see from Tables 4 and 5 that the performance in one-error of sparse-MIML is usually slightly better than that of MIML-RWR except some cases in coverage and rankLoss. Note that when we tune the parameters for sparse-MIML method, the optimal result is obtained by finding the smallest one-error. This optimal result may not correspond to the optimal result in the other evaluation metrics like coverage and

rankLoss. Therefore, there are some negative values appearing in the columns of coverage and rankLoss in Tables 4 and 5.

5.2 Effect of Parameters

In this subsection, we study the effect of parameters α and λ to the classification performance of the sparse-MIML algorithm. In Figures 2 and 3, we show the performance of the proposed method with 10% and 90% training image data for different values of α and λ. When 10% training data is used, the optimal parameters are $\alpha = 0.99$ and $\lambda = 0.1$. We need to make use of affinity information from unlabeled data (α is close to 1) to determine the ranking labels. When 90% training data is used, the optimal parameters are $\alpha = 0.64$ and $\lambda = 0.17$. Because there is more labeled data, the usage of affinity information from unlabeled data can be reduced and the algorithm make use of the sparsity to enhance classification performance. It is interesting to note that the optimal values of parameters are quite consistent for all four evaluation metrics. In Figure 4, we also show the performance of the proposed method with 90% training text data for different values of α and λ. Indeed, the graphs are the about the same as those in Figure 4 when 10% data is used, and the optimal parameters are the same for 10% and 90% training data to be used. It seems that this MIML problem for image example may be more difficult than that for text example.

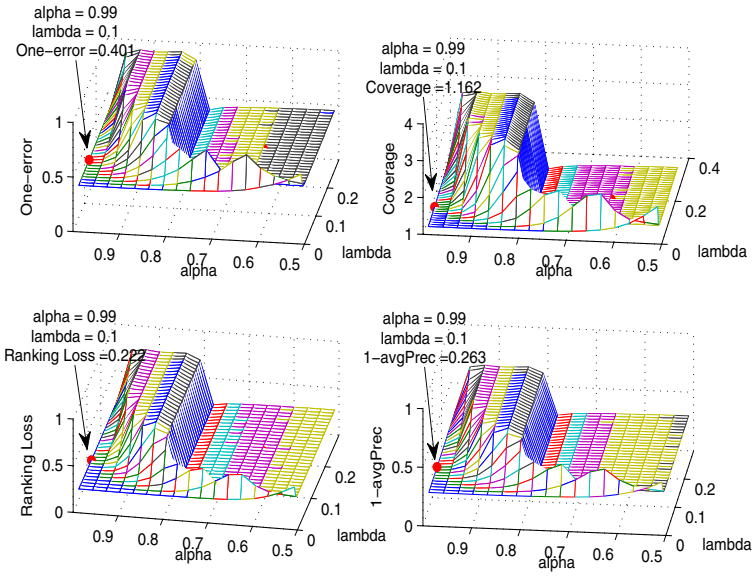


Fig. 2. The performance of sparse-MIML with 10% training image data for different values α and λ

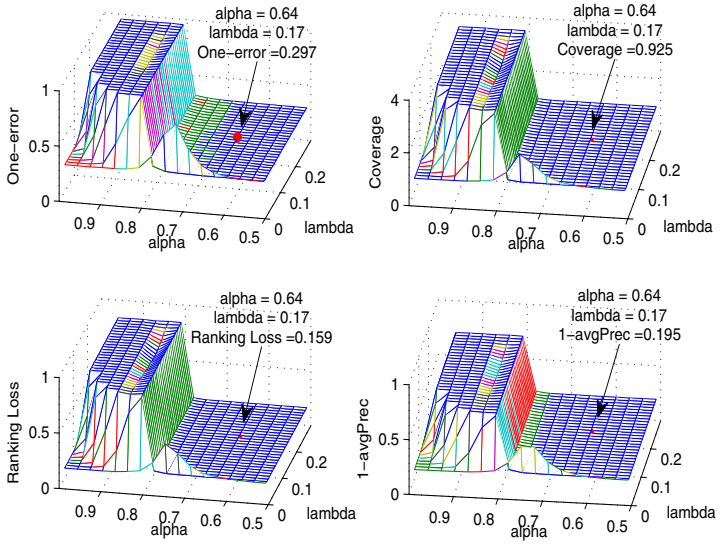


Fig. 3. The performance of sparse-MIML with 90% training image data for different values α and λ

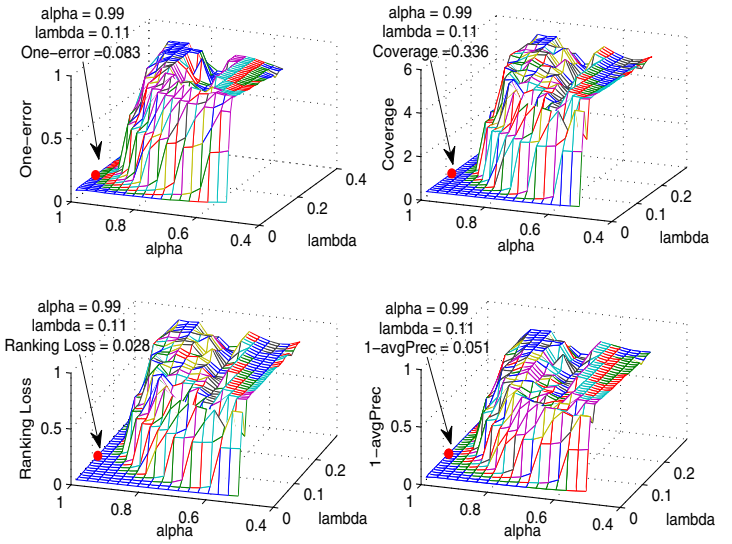


Fig. 4. The performance of sparse-MIML with 90% training text data for different values α and λ

6 Concluding Remarks

In this paper, we have developed a sparse-MIML algorithm to determine the rank of class labels for objects in multi-instance multi-label data. The experimental results have demonstrated that the proposed algorithm is efficient and effective. In the future, we would like to explore how to determine the optimal parameters in the proposed algorithm, and apply the algorithm to the other large scale MIML problems.

References

1. Andrews, S., Tsochantaridis, I., Hofmann, T.: Support vector machines for multiple-instance learning. Advances in Neural Information Processing Systems, 577–584 (2003)
2. Chung, F.: Spectral graph theory. Amer Mathematical Society (1997)
3. Dietterich, T., Lathrop, R., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artificial Intelligence 89(1-2), 31–71 (1997)
4. Maron, O., Ratan, A.: Multiple-instance learning for natural scene classification. In: Proceedings of the 15th IEEE ICML International Conference on Machine Learning (1998)
5. Ng, M., Weiss, P., Yuan, X.: Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods. SIAM Journal on Scientific Computing 32(5), 2710–2736 (2010)
6. Schapire, R., Singer, Y.: Boostexter: A boosting-based system for text categorization. Machine Learning 39(2), 135–168 (2000)
7. Tong, H., Faloutsos, C., Pan, J.: Random walk with restart: fast solutions and applications. Knowledge and Information Systems 14(3), 327–346 (2008)
8. Wu, Q., Ng, M., Ye, Y.: Markov-Miml: A Markov chain-based multi-instance multi-label learning algorithm. Knowledge and Information Systems (2013)
9. Zha, Z., Hua, X., Mei, T., Wang, J., Qi, G., Wang, Z.: Joint multi-label multi-instance learning for image classification. In: Proceedings of the IEEE CVPR Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
10. Zhang, M.: A k-nearest neighbor based multi-instance multi-label learning algorithm. In: 2010 22nd IEEE International Conference on Tools with Artificial Intelligence (ICTAI), vol. 2, pp. 207–212 (2010)
11. Zhang, M., Zhou, Z.: M3miml: A maximum margin method for multi-instance multi-label learning. In: Proceedings of the 8th IEEE ICDM International Conference on Data Mining, pp. 688–697 (2008)
12. Zhou, Z., Zhang, M.: Multi-instance multi-label learning with application to scene classification. Advances in Neural Information Processing Systems 19, 1609 (2007)
13. Zhou, Z., Zhang, M., Huang, S., Li, Y.: Multi-instance multi-label learning. Artificial Intelligence 176(1), 2291–2320 (2012)

Consensus Clustering with Robust Evidence Accumulation

André Lourenço[1,2], Samuel Rota Bulò[3],
Ana Fred[2], and Marcello Pelillo[3]

[1] Instituto Superior de Engenharia de Lisboa, Lisbon, Portugal
[2] Instituto de Telecomunicações, Instituto Superior Técnico, Lisbon, Portugal
[3] DAIS, Universitá Ca' Foscari Venezia, Venice, Italy

Abstract. Consensus clustering methodologies combine a set of partitions on the clustering ensemble providing a consensus partition. One of the drawbacks of the standard combination algorithms is that all the partitions of the ensemble have the same weight on the aggregation process. By making a differentiation among the partitions the quality of the consensus could be improved. In this paper we propose a novel formulation that tries to find a median-partition for the clustering ensemble process based on the evidence accumulation framework, but including a weighting mechanism that allows to differentiate the importance of the partitions of the ensemble in order to become more robust to noisy ensembles. Experiments on both synthetic and real benchmark data show the effectiveness of the proposed approach.

Keywords: Clustering Algorithm, Clustering Ensembles, Median-Partition, Evidence Accumulation Clustering, Clustering Selection, Clustering Weighting.

1 Introduction

The combination of multiple sources of information either in the supervised or unsupervised learning setting allows to obtain improvements on the classification performance. In the unsupervised paradigm, this task is difficult due to the label correspondence problem, i.e., the lack of explicit correspondences between the cluster labels produced by the different clustering algorithms. This problem is made more serious if additionally clusterings with different numbers of clusters are allowed in the ensemble. Clustering ensemble methods, also known as consensus clustering methods, propose a formalism to tackle this problem, allowing the combination of a set of base clustering algorithms into a single consensus partition.

Recent surveys present an overview on this research topic [1, 2]. One of the main approaches is known as *median-partition* [2], where the consensus solution is obtained as the partition having lowest divergence from all the partitions in the clustering ensemble. Another significant approach, known as Evidence Accumulation Clustering (EAC) [3], is based on object *co-occurences*, where the

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 307–320, 2013.
© Springer-Verlag Berlin Heidelberg 2013

consensus is obtained through a voting process among the objects. Specifically, the consensus clustering problem is addressed by summarizing the information of the ensemble into a pairwise *co-association matrix*, where each entry holds the fraction of clusterings in the ensemble in which a given pair of objects is placed in the same cluster. By doing so, the label correspondence problem is implicitly solved. This matrix, which is regarded to as a similarity matrix, is used to feed a pairwise similarity clustering algorithm to deliver the final consensus clustering. In [3] agglomerative hierarchical algorithms are used to extract the consensus partition and in [4] a graph partitioning algorithm (METIS [5]) are used.

In [6] a more principled way of using the information in the co-association matrix has been proposed. Specifically, the problem of extracting a consensus partition is posed as a matrix factorization problem involving the co-association matrix, where the factor matrix is left-stochastic, *i.e.* nonnegative with columns summing up to one. Each column of the factor matrix can be interpreted as a multinomial distribution expressing the probabilities of each object of being assigned to each cluster. In [7], a probabilistic model for the co-association matrix has been proposed, entitled Probabilistic Evidence Accumulation for Clustering Ensembles. In this model, the entries of the co-association matrix are regarded as independent observations of binomial random variables counting the number of times two objects occur in a same cluster. These random variables are parametrized by the unknown assignments of objects to clusters, which are in turn estimated by adopting a maximum-likelihood approach. In [8] a new formulation is proposed that constitute a generalization of [6] which is solved in way which is close in spirit to [7]. This method, entitled PEACE, creates sparse co-association matrices by a simple uniform sampling criterion and exploits this sparsity to achieve sub-linear iterations in the consensus clustering algorithm.

One of the drawbacks of previous combination methodologies is that all input partitions of the ensemble have the same weight in the aggregation process, when in fact some of them are less important than other [9]. The partitions of the ensemble may come from different algorithms, or from the same algorithm with different initializations. It was shown that the diversity on the clustering ensemble leads to an enhancement on the performance [10], but extreme cases introduce to much variability leading to a significant drop on the performance. Moreover the clustering ensemble can be composed by a subset of partitions highly correlated that can decrease significantly the variability biasing the consensus solution to one of the one of the input partitions.

The problem of weighting differently each of the base clustering solutions was already studied in the literature [9, 11–16]. In Duarte et al. [11] the weighting of the partitions is obtained using internal and relative clustering validation indexes, and the combination is performed using the Evidence Accumulation Clustering algorithm. Fern and Lin [12] define two important quantities that should be considered on the selection process, namely quality calculated between each ensemble and a consensus solution, and diversity of the ensemble. They propose three different heuristics that jointly consider these criteria. Hong et al. [15] also ground their algorithm on these criteria, and use a re-sampling-based method to estimate

them. Jia et al. [16] only use a quality criterion on their selection mechanism. In Li and Ding [9], the weighted consensus clustering is based on nonnegative matrix factorization framework but focus only on the quality criterion. Vega-Pons et al. [13] follow the idea of finding the median partition but weighting differently each partition, finding their relevance through an intermediate step.

In this work, we propose a consensus clustering approach based on the evidence accumulation framework, which includes a weighting mechanism that allows to differentiate the importance of the partitions of the ensemble in order to become more robust to noisy ensembles. Our approach tries to find a median-partition, *i.e.* minimizing its divergence from the other partitions in the ensemble, in a way that takes into account the co-occurrences of objects in clusters. Additionally, we jointly optimize the importance of each partition in the ensemble by means of weight variables representing a discrete probability distribution over the set of partitions in the ensemble. To overcome the occurrence of trivial weightings, *i.e.* putting full mass on a single partition, we introduce two regularization mechanisms which lead to two different formulations. In the first formulation, regularization is achieved by restricting the set of feasible probability distributions determining the partitions' importance weights. In the second formulation, a classical ℓ_2 regularization is adopted. We determine the median-partition and the weights vector by means of an alternating optimization procedure, which guarantees to find a local solution. Finally, we perform experiments on ensembles derived from synthetic and real datasets to assess the validity of our model.

The paper is organized as follows: in Section 2 we present the notation used throughout the paper and we introduce our robust consensus clustering model and the two related formulations deriving from the use of different regularization techniques. In Section 3, we present the optimization procedure adopted to find a consensus clustering solution, together with the weights associated to each partition in the ensemble, according to the two proposed formulations. Section 4 is devoted to assessing the effectiveness of the proposed approach. Finally, we draw conclusions in Section 5.

2 Formulation

We start introducing the basic notation and definitions adopted throughout the paper. We denote *sets* with upper-case calligraphic-style letters (*e.g.*, $\mathcal{X}$), column *vectors* with bold lower-case letters (*e.g.*, $\boldsymbol{x}$), *matrices* with upper-case typewriter-style letters (*e.g.*, $\mathtt{X}$), *indices* with lower-case letters (*e.g.*, i) and *constants* with lower-case typewriter-style letters (*e.g.*, n). We denote by $\mathbb{1}_P$ the indicator function yielding 1 if proposition P holds true, 0 otherwise. We indicate with $\mathtt{A}_{ij}$ the ijth component of matrix $\mathtt{A}$ and with $\boldsymbol{v}_i$ the ith component of vector $\boldsymbol{v}$. The vector of all 1s of size k is denoted by $\mathbf{1}_\mathsf{k}$, the subscript being dropped where size is unambiguous. Let $\mathtt{A}$ be a $\mathsf{k} \times \mathsf{n}$ matrix. The *transposition* of $\mathtt{A}$ is denoted by $\mathtt{A}^\top$. The *Frobenious norm* of $\mathtt{A}$ is denoted by $\|\mathtt{A}\| = \sqrt{\sum_{ij} \mathtt{A}_{ij}^2}$. The sets of real and nonnegative real numbers are denoted by $\mathbb{R}$ and $\mathbb{R}_+$ as usual. We compactly write $[\mathsf{n}]$ for the set $\{1, \ldots, \mathsf{n}\}$. We denote by $\lfloor r \rfloor$ and $\lceil r \rceil$

the *floor* and *ceil* operators giving the largest integer value upper bounded by $r \in \mathbb{R}$ and the smallest integer value lower bounded by r, respectively. The set of *left-stochastic* $\mathsf{k} \times \mathsf{n}$ matrices is denoted by $\mathcal{S} = \left\{ \mathtt{X} \in \mathbb{R}_{+}^{\mathsf{k} \times \mathsf{n}} : \mathtt{X}^{\top} \mathbf{1}_{\mathsf{k}} = \mathbf{1}_{\mathsf{n}} \right\}$ and we denote by $\mathcal{S}_{01} = \mathcal{S} \cap \{0,1\}^{\mathsf{k} \times \mathsf{n}}$ the set of *binary* left-stochastic matrices.

Let $\mathcal{X} = \{\boldsymbol{x}_i\}_{i \in [\mathsf{n}]}$ be a set of n data points. An *ensemble of clusterings* of $\mathcal{X}$ is a collection $\mathcal{E} = \{\mathtt{X}^{(u)}\}_{u \in [\mathsf{m}]}$ of m partitions, obtained by running different algorithms (*e.g.*, different parametrizations and/or initializations) on the data set $\mathcal{X}$. Each partition $\mathtt{X}^{(u)} \in \mathcal{S}_{01}$ can be regarded as a binary left-stochastic matrix where $\mathtt{X}_{ki}^{(u)}$ indicates whether the ith data point belongs to cluster k in the uth partition.

A *consensus clustering* (or consensus partition) for an ensemble $\mathcal{E}$ is typically defined as a partition minimizing its divergence from the other partitions in the ensemble. This definition, however, implicitly assumes that the ensemble is noiseless and, thus, that all clusterings should be given equal importance during the establishment of a consensus partition. In order to be more robust to noisy elements, we introduce a probability distribution over the set of partitions in the ensemble that allows to automatically tune the importance of each partition in the ensemble. Formally, a consensus, median partition under this setting can be found as the solution of the following optimization problem:

$$\begin{aligned} \min \textstyle\sum_{u \in \mathcal{U}} \alpha_u d(\mathtt{Z}, \mathtt{X}^{(u)}) \\ \text{s.t. } \mathtt{Z} \in \mathcal{S}_{01} \\ \boldsymbol{\alpha} \in \Delta \,, \end{aligned} \tag{1}$$

where $d(\cdot, \cdot)$ is a function providing a distance between the partitions given as arguments, and Δ is the set of m-dimensional vectors representing a discrete probability distribution.

We select the distance function $d(\cdot, \cdot)$ by following the EAC principles. In specific, we implicitly sidestep the problem of cluster correspondences in the computation of the distance between partitions, by counting the errors in the pairwise cluster co-occurrences as follows

$$d(\mathtt{Z}, \mathtt{X}) = \left\| \mathtt{Z}^{\top} \mathtt{Z} - \mathtt{X}^{\top} \mathtt{X} \right\|^2 . \tag{2}$$

Indeed, this distance counts the number of times two data points are assigned the same cluster in $\mathtt{X}$ but different ones in $\mathtt{Z}$ and vice versa. Moreover, for convenience, instead of attacking (1) directly, we relax the troublesome integer constraints by replacing the left-stochastic binary matrix variable $\mathtt{Z}$ with a left-stochastic *real* matrix variable $\mathtt{Y} \in \mathcal{S}$ yielding the following relaxed continuous optimization problem:

$$\begin{aligned} \min \textstyle\sum_{u \in \mathcal{U}} \alpha_u \left\| \mathtt{Y}^{\top} \mathtt{Y} - {\mathtt{X}^{(u)}}^{\top} \mathtt{X}^{(u)} \right\|^2 \\ \text{s.t. } \mathtt{Y} \in \mathcal{S} \\ \boldsymbol{\alpha} \in \Delta \,. \end{aligned} \tag{3}$$

We can finally project $\mathtt{Y}$ back on $\mathcal{S}_{01}$ by performing a maximum a posteriori choice over each column of $\mathtt{Y}$.

Unfortunately, the optimization problem in (3) is ill posed, because trivial distributions, putting full mass on a single partition $u \in [\mathsf{m}]$, lead to an optimal solution by setting $\mathtt{Y} = \mathtt{X}^{(u)}$. The same issue clearly afflicts also the original formulation in (1). To overcome this problem, we need a form of regularization on the probability distribution $\boldsymbol{\alpha}$. In this paper, we propose two different regularization solutions that will be described in the next subsections, one acting on the feasible domain of (3), the other acting on its objective function.

2.1 Regularization of $\boldsymbol{\alpha}$ Using a Restricted Simplex

The first formulation forces the consensus clustering to agree with at least a share $0 < \rho \leq 1$ of the partitions in the ensemble. By doing so, noisy partitions might be excluded from the objective and thus their importance can be nullified and, at the same time, trivial weighting solutions can be excluded as a minimum number of partitions should actively be involved in the minimization. This is achieved by constraining $\boldsymbol{\alpha}$ to lie in a *ρ-restricted simplex* Δ_ρ which is defined as

$$\Delta_\rho = \left\{ \boldsymbol{\alpha} \in \mathbb{R}_+^{\mathsf{m}} \,:\, \boldsymbol{\alpha}^\top \mathbf{1} = 1 \,,\; \boldsymbol{\alpha} \leq \rho \mathbf{1} \right\} ,$$

where $\rho \geq \mathsf{m}^{-1}$, since otherwise the set Δ_ρ would be empty. Since ρ can be regarded as the largest probability that can be taken by an element of $\boldsymbol{\alpha}$, automatically at least $\lfloor \rho^{-1} \rfloor$ entries of $\boldsymbol{\alpha}$ have to be strictly positive. Clearly, if $\rho \geq 1$ we fall back to the standard simplex, *i.e.* $\Delta_\rho = \Delta$ for all $\rho \geq 1$.

By changing the domain of $\boldsymbol{\alpha}$ in (3) to a ρ-restricted simplex, with $\mathsf{m}^{-1} \leq \rho < 1$ we get the following regularized formulation:

$$\begin{aligned} \min & \textstyle\sum_{u \in \mathcal{U}} \alpha_u \left\| \mathtt{Y}^\top \mathtt{Y} - {\mathtt{X}^{(u)}}^\top \mathtt{X}^{(u)} \right\|^2 \\ \text{s.t. } & \mathtt{Y} \in \mathcal{S} \\ & \boldsymbol{\alpha} \in \Delta_\rho \,. \end{aligned} \tag{4}$$

This formulation falls back to the unweighted case when $\rho = \mathsf{m}^{-1}$ and to the unregularized case (3) when $\rho \geq 1$.

2.2 Regularization of $\boldsymbol{\alpha}$ Using $\boldsymbol{\ell_2}$-Norm

Our second formulation, considers the following, classical ℓ_2 regularization parametrized by $\lambda \geq 0$:

$$\begin{aligned} \min & \textstyle\sum_{u \in \mathcal{U}} \alpha_u \left\| \mathtt{Y}^\top \mathtt{Y} - {\mathtt{X}^{(u)}}^\top \mathtt{X}^{(u)} \right\|^2 + \frac{\lambda}{2} \|\boldsymbol{\alpha}\|^2 \\ \text{s.t. } & \mathtt{Y} \in \mathcal{S} \\ & \boldsymbol{\alpha} \in \Delta \,. \end{aligned} \tag{5}$$

This formulation falls back to (3) by taking $\lambda = 0$ and to the unweighted case when $\lambda \to \infty$. Indeed, the probability distribution $\boldsymbol{\alpha}$ is pushed towards the uniform distribution as the regularization constant λ increases.

It is interesting to notice that (5) presents a special case of elastic net regularization [17] in which the parameter related to the ℓ_1 regularization is taken to infinity and becomes a constraint in Δ.

3 Algorithm

Solving (4) or (5) is in general a hard problem. We propose here for both an alternating, local optimization procedure which interleaves updates of the cluster assignments $\mathtt{Y}$ and updates of the weights vector $\boldsymbol{\alpha}$. The update procedure for the cluster assignments $\mathtt{Y}$ is the same for both the regularized formulations and will be addressed in the next subsection, while the update procedure of $\boldsymbol{\alpha}$ is in general different for (4) or (5) and it will be addressed in Subsection 3.2 and 3.3, respectively.

3.1 Optimization of $\mathtt{Y}$ in (4) and (5)

Assume $\boldsymbol{\alpha}$ to be fixed and consider the problem of optimizing (4) or (5) with respect to $\mathtt{Y}$ only. This yields a non-convex continuous optimization problem which can be conveniently rewritten into the following one, which shares with (4) and (5) the same local minimizers:

$$\begin{array}{l} \min \left\| \mathtt{Y}^\top \mathtt{Y} - \sum_{u \in \mathcal{U}} \alpha_u {\mathtt{X}^{(u)}}^\top \mathtt{X}^{(u)} \right\|^2 \\ \text{s.t. } \mathtt{Y} \in \mathcal{S} \,. \end{array} \tag{6}$$

The equivalence between (4)/(5) and (6) can be grasped by noting that the terms depending on $\mathtt{Y}$ are the same in both the optimization problems and the objectives differ simply by an additive, constant term. The matrix $\sum_{u \in \mathcal{U}} \alpha_u {\mathtt{X}^{(u)}}^\top \mathtt{X}^{(u)}$ can be regarded as the *weighted co-association matrix*. Indeed, when $\boldsymbol{\alpha}$ is the uniform distribution, we fall back to the classic notion of co-association matrix as originally defined in [3].

A local solution of the optimization problem in (6) can be efficiently computed using the approach proposed in [8], which is a primal, line search approach that iteratively improves the objective by optimizing one column of $\mathtt{Y}$ at time. Each update has a linear complexity both in time and space. Another advantage of this optimization approach is that it can handle sparsified versions of the optimization problem in (6), where the Frobenious norm runs over a sparse subset of the entries of the matrix given as argument, which is useful in the case of large datasets. In the sparisified scenario, the time complexity of every line search can be reduced to sub-linear.

3.2 Optimization of $\boldsymbol{\alpha}$ in (4)

Assume $\mathtt{Y}$ to be fixed in (4) and let us focus on the optimization of the vector-valued variable $\boldsymbol{\alpha}$ only. By letting $\boldsymbol{d} \in \mathbb{R}_+^{\mathsf{m}}$ be a vector with entries

$$d_u = \left\| \mathtt{Y}^\top \mathtt{Y} - {\mathtt{X}^{(u)}}^\top \mathtt{X}^{(u)} \right\|^2 , \tag{7}$$

we can rewrite the optimization problem restricted to $\boldsymbol{\alpha}$ as follows:

$$\begin{aligned} \min\ & \boldsymbol{\alpha}^\top \boldsymbol{d} \\ \text{s.t. } & \boldsymbol{\alpha}^\top \mathbf{1} = 1 \\ & \boldsymbol{\alpha} \leq \rho \mathbf{1} \\ & \boldsymbol{\alpha} \in \mathbb{R}_+^{\mathsf{m}} . \end{aligned} \tag{8}$$

This is a linear programming problem, whose solution can be readily computed as stated by the following proposition.

Proposition 1. *Assume without loss of generality that* $\boldsymbol{d}$ *satisfies the relation* $d_u \leq d_v$ *for all* $0 \leq u \leq v \leq \mathsf{m}$. *Let* $r = 1 - \rho\lfloor \rho^{-1} \rfloor$. *A solution of* (8) *is given by* $\alpha_u = \rho \mathbb{1}_{u \leq \lfloor \rho^{-1} \rfloor} + r \mathbb{1}_{u = \lceil \rho^{-1} \rceil}$, *for all* $u \in [\mathsf{m}]$.

Proof. We proceed with a proof by contradiction. Assume that $\boldsymbol{\alpha}$ is not a solution of (8) and let $\boldsymbol{\beta} \in \Delta_\rho$ be a solution of (8), which exists since the feasible set is compact. Clearly, $\boldsymbol{\beta} \neq \boldsymbol{\alpha}$. The solution $\boldsymbol{\beta}$ must satisfy the property that $\beta_u \geq \beta_v$ for all $0 \leq u \leq v \leq \mathsf{m}$. Otherwise, by swapping the elements in $\boldsymbol{\beta}$ indexed by a pair of indices violating the condition, we would yield a better solution, contradicting the optimality of $\boldsymbol{\beta}$.

Now, let $p \in [\mathsf{m}]$ be the smallest index satisfying $\beta_p > \alpha_p$. Hence, $\alpha_u \geq \beta_u$ holds for all $u < p$. Necessarily, $p > \lfloor \rho^{-1} \rfloor$ because by construction $\alpha_u = \rho$ for all $0 \leq u \leq \lfloor \rho^{-1} \rfloor$ and therefore it cannot be exceeded by β_u. Moreover, by construction, $\alpha_u = 0$ for all $u > \lceil \rho^{-1} \rceil$, which implies that $\alpha_u \leq \beta_u$ for all $u \geq p$. By exploiting these relations and by the non-increasing ordering on $\boldsymbol{d}$, we derive that

$$\begin{aligned} \boldsymbol{d}^\top (\boldsymbol{\alpha} - \boldsymbol{\beta}) &= \left[\sum_{i=1}^{p-1} d_i (\alpha_i - \beta_i) \right] - \left[\sum_{i=p}^{\mathsf{m}} d_i (\beta_i - \alpha_i) \right] \\ &\leq d_p \left[\sum_{i=1}^{p-1} \alpha_i - \beta_i \right] - d_p \left[\sum_{i=p}^{\mathsf{m}} \beta_i - \alpha_i \right] = d_p \left[\sum_{i \in [\mathsf{m}]} \alpha_i - \beta_i \right] = 0 , \end{aligned}$$

which contradicts the non-optimality of $\boldsymbol{\alpha}$. □

The condition required on $\boldsymbol{d}$ can be met by sorting the vector $\boldsymbol{d}$ in ascending order and by keeping track of the induced permutation. The latter can be used at the end to reorder the solution $\boldsymbol{\alpha}$ using the inverse mapping.

3.3 Optimization of $\boldsymbol{\alpha}$ in (5)

Assume $\mathtt{Y}$ to be fixed in (5) and let us focus on the optimization of the vector-valued variable $\boldsymbol{\alpha}$ only. By taking $\boldsymbol{d} \in \mathbb{R}_+^{\mathsf{m}}$ as defined in (7), we reduce (5) to following convex, quadratic optimization problem:

$$\begin{aligned} \min\ & \boldsymbol{\alpha}^\top \boldsymbol{d} + \tfrac{\lambda}{2} \boldsymbol{\alpha}^\top \boldsymbol{\alpha} \\ \text{s.t. } & \boldsymbol{\alpha}^\top \mathbf{1} = 1 \\ & \boldsymbol{\alpha} \in \mathbb{R}_+^{\mathsf{m}} . \end{aligned} \tag{9}$$

Fortunately, also this optimization problem can be readily computed in linear time. The solution procedure is detailed in the following proposition.

Proposition 2. *Assume without loss of generality that* $\boldsymbol{d}$ *satisfies the relation* $d_u \leq d_v$ *for all* $0 \leq u \leq v \leq \mathsf{m}$. *Let* $\boldsymbol{y} \in \mathbb{R}^{\mathsf{m}}$ *be defined as*

$$y_u = \frac{1}{u}\left(1 + \sum_{v=1}^{u} \frac{d_v}{\lambda}\right)$$

and let w *be the largest element in* $[\mathsf{m}]$ *satisfying* $y_u > d_u/\lambda$. *A solution of* (5) *is given by* $\alpha_u = \mathbb{1}_{u \leq w}(y_w - d_u/\lambda)$, *for all* $u \in [\mathsf{m}]$.

Proof. We start showing that $y_u \leq y_{u+1}$ for all $u > w$. To this end, note that the relation $y_u \leq d_u/\lambda$ holds for all $u > w$ by definition of w. Then

$$y_u = \frac{u+1}{u}\left(y_{u+1} - \frac{1}{u+1}\frac{d_{u+1}}{\lambda}\right) \leq \frac{u+1}{u}\left(y_{u+1} - \frac{1}{u+1}y_{u+1}\right) = y_{u+1}\,. \quad (10)$$

By repeated application of this relation, we have that $y_u \leq y_v$ for all $v \geq u$ and $u > w$.

Since the optimization problem in (9) is convex, the Karush-Kuhn-Tucker (KKT) necessary conditions for optimality are also sufficient. Hence, a solution $\boldsymbol{\alpha}$ satisfying the following KKT conditions, for some value of the Lagrangian multipliers $\gamma \in \mathbb{R}$ and $\boldsymbol{\mu} \in \mathbb{R}^{\mathsf{m}}_{+}$, is a solution of (9):

$$d_u + \lambda\alpha_u - \gamma - \mu_u = 0\,, \qquad \forall u \in [\mathsf{m}] \quad (11)$$

$$\boldsymbol{\alpha}^{\top}\mathbf{1} = 1\,, \quad (12)$$

$$\boldsymbol{\alpha}^{\top}\boldsymbol{\mu} = 0\,. \quad (13)$$

We proceed by showing that the solution computed as detailed in the proposition satisfies the KKT conditions, *i.e.* we have to show that Equations (11)-(13) are satisfied for a specific choice of the Lagrangian multipliers and that our choice of $\boldsymbol{\mu}$ also satisfies the non-negativity constraint. We start noting that $\alpha_u > 0$ for all $u \leq w$, for we have that $y_w > d_w/\lambda \geq d_u/\lambda$ for all $u \leq w$, and $\alpha_u = 0$ for all $u > w$ by construction. Set $\gamma = \lambda y_w$. For all $u \leq w$, Equation (11) is satisfied by taking $\mu_u = 0$, while for all $u > w$ it is satisfied by taking $\mu_u = d_u - \lambda y_w$. This choice of the elements of $\boldsymbol{\mu}$ clearly satisfies Equation (13). Moreover, $\mu_u \geq 0$ is clearly satisfied for all $u \leq w$ with equality and it is also satisfied for all $u > w$ because $y_w \leq y_u \leq d_u/\lambda$ holds by the relation proven at the beginning of this proof and by definition of w. We conclude by showing that also Equation (12) holds. Indeed,

$$\sum_{u=1}^{\mathsf{m}} \alpha_u = \sum_{u=1}^{w} \alpha_u = w y_w - \sum_{u=1}^{w} \frac{d_u}{\lambda} = 1\,.$$

□

As in the case of the previous formulation, the condition required for $\boldsymbol{d}$ can be met by sorting the vector in ascending order and by keeping track of the induced permutation, which will be used to recover the original ordering on the solution vector $\boldsymbol{\alpha}$.

3.4 Summary of the Algorithm

As anticipated at the beginning of this section, the algorithm used to optimize (4) and (5) alternates between the optimization of the cluster assignment probabilities $\mathtt{Y}$ and the optimization of the weights vector $\boldsymbol{\alpha}$.

The pseudo-code of the algorithm is shown in Algorithm 1. The input consists of the ensemble of clusterings $\mathcal{E} = \{\mathtt{X}^{(u)}\}_{u\in[\mathsf{m}]}$ the maximum number of desired clusters k and the regularization parameter, namely ρ for the formulation in (4) and λ for the formulation in (5). At line 1, we initialize the matrix of probabilistic cluster assignments by randomly sampling an element of $\mathtt{S}$, or by considering a uniform distribution for all objects. At line 2, we initialize the weights to the uniform distribution. Lines 3-6 represent the alternating optimization loop, which is iterated until a stopping criterion is met, *e.g.* maximum number of iterations has been reached, or the difference of either $\mathtt{Y}$ or $\boldsymbol{\alpha}$ between two consecutive iterations is below a given threshold. At line 4, we optimize (6), which is equivalent to optimizing either regularized formulation with respect to $\mathtt{Y}$. The solution is obtained by using the algorithm described in [8]. At the next line, we focus on optimizing the weights vector-valued variable $\boldsymbol{\alpha}$. Based on the chosen formulation, we optimize (8) or (9) following the procedure described in Proposition 1 or Proposition 2, respectively. Both solutions can be obtained efficiently in linear time. Finally, once we exit the optimization loop, we project $\mathtt{Y}$ on the set of left-stochastic binary matrices obtaining matrix $\mathtt{X} \in \mathcal{S}_{01}$, and we return $\mathtt{X}$ and $\boldsymbol{\alpha}$.

Algorithm 1. Algorithm description

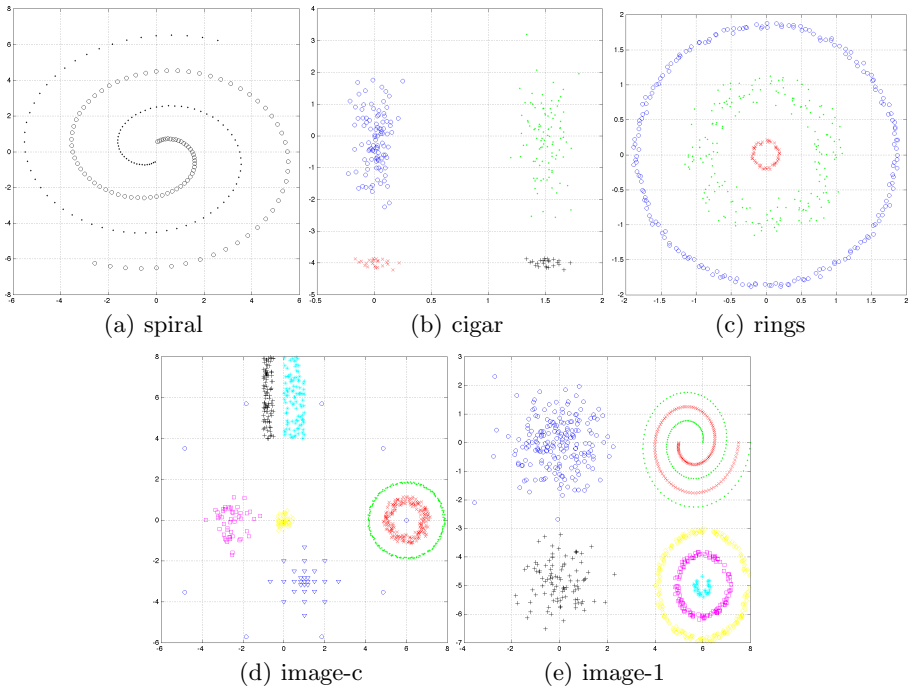
Require: $\mathcal{E} = \{\mathtt{X}^{(u)}\}_{u\in[\mathsf{m}]}$: ensemble of clusterings
Require: k: maximum number of desired clusters
Require: $\mathsf{m}^{-1} \leq \rho \leq 1$, regularization parameter for formulation (4), or $\lambda > 0$, regularization parameter for formulation (5)
1: Initialize $\mathtt{Y} \in \mathcal{S}$
2: $\boldsymbol{\alpha} \leftarrow \mathsf{m}^{-1}\mathbf{1}$
3: **repeat**
4: $\quad\mathtt{Y} \leftarrow$ solve (6) using the approach in [8]
5: $\quad\boldsymbol{\alpha} \leftarrow$ solve (4) or (5) based on the desired formulation
6: **until** termination condition is met
7: $\mathtt{X} \leftarrow$ project $\mathtt{Y}$ on $\mathcal{S}_{01}$
8: **return** $\mathtt{X}$, $\boldsymbol{\alpha}$

4 Experimental Evaluation

We evaluate our formulation using synthetic and real-world datasets from the UCI Machine Learning Repository. We compare the performance of our algorithm against the algorithm in [8], which we refer to as Un-weighted. We call our two algorithms Weighted+Δ_ρ and Weighted+ℓ_2 corresponding to the variants described in Section 3.2 and 3.3, respectively. For all the experiments,

Table 1. Benchmark datasets - synthetic: (s-1) spiral, (s-2) cigar, (s-3) rings, (s-4) image-c, (s-5) image-1; real: (r-1) iris, (r-2) wine, (r-3) house-votes, (r-4) ionsphere, (r-5) std-yeast-cell, (r-6) breast-cancer, (r-7) optdigits.

Data-Sets	s-1	s-2	s-3	s-4	s-5	r-1	r-2	r-3	r-4	r-5	r-6	r-7
k	2	4	3	7	8	3	3	2	2	5	2	10
n	200	250	450	739	1000	150	178	232	351	384	683	1000
$\mathsf{k}_{\min} - \mathsf{k}_{\max}$	2-10	2-10	2-10	7-40	8-40	3-20	4-20	2-20	4-20	5-20	2-20	10-50



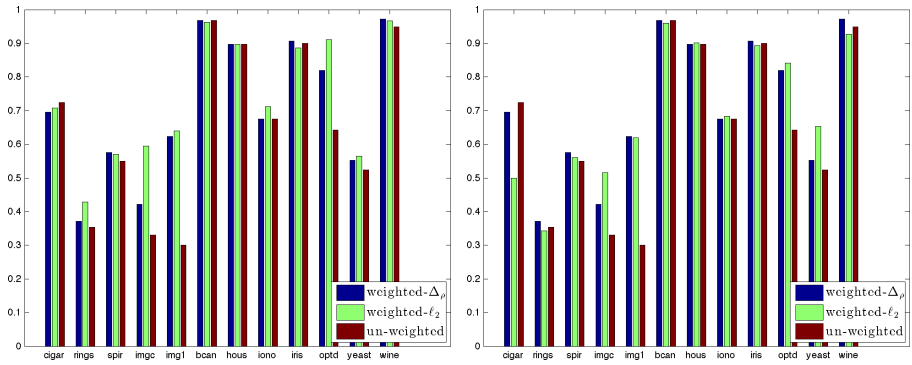
(a) spiral (b) cigar (c) rings

(d) image-c (e) image-1

Fig. 1. Sketch of the Synthetic Datasets

we used the following setting for the regularization parameters of our algorithms: $\rho = (0.8\,\mathsf{m})^{-1}$ and $\lambda = 0.5\,\mathsf{n}^2$.

We performed different series of experiments to compare the performance of our approaches on several types of ensembles: i) *k-means ensemble* - consisting of $\mathsf{m} = 150$ partitions generated running the classical k-means algorithm [18] with different number of clusters, and different initializations; ii) *mixed ensemble* - consisting of $\mathsf{m} = 56$ partitions generated by multiple algorithms (agglomerative hierarchical algorithms: single, average, ward, centroid link; k-means[18]; spectral clustering [19]) with different number of clusters iii) *noisy ensemble* - an ensemble with noisy partitions obtained from previous ensembles, changing a percentage of the partitions of the ensemble to random partitions.



(a) k-means ensemble (b) mixed ensemble

Fig. 2. Performance Evaluation for k-means and mixed ensembles in terms of Accuracy. The weighted approaches WEIGHTED+Δ_ρ and WEIGHTED+ℓ_2 are compared against the un-weighted one.

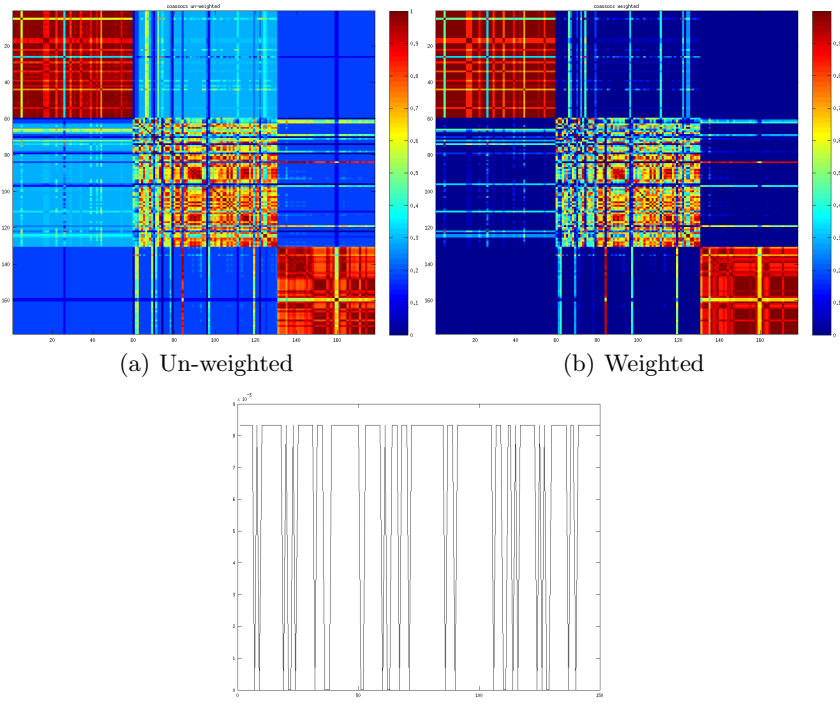
We assess the quality of a consensus partition by comparing it against the ground truth partition. In order to compare two hard clusterings we adopt the $\mathcal{H}$ criterion introduced in [20], which computes the fraction of correct cluster assignments considering the best cluster matching between the consensus partition and the ground-truth partition.

Table 1 summarizes the main characteristics of the UCI and synthetic datasets used in the evaluation (number of ground truth clusters k and number of samples n) and reports also the range of number of clusters used during the ensemble generation $\{\mathsf{k}_{min} - \mathsf{k}_{max}\}$. Figure 1 illustrates the synthetic datasets used in the evaluation: (a) spiral; (b) cigar; (c) rings; (d) image-c (e) image-1 .

4.1 Non-Noisy Ensembles

Figure 2 presents the performance results in terms of $\mathcal{H}$ criterion for the k-means and mixed ensembles. In both types of ensembles we can see that the proposed weighted consensus clustering approaches perform on average better than the unweighted one, as expected, even though we have a fixed parameterization for the regularization on all datasets. If we compare the two types of regularization on the weighted algorithms, there is no clear winner, the ℓ_2-regularized being slightly better on average. Overall, we have that in the k-means ensemble the weighted algorithms obtain better result in 8 out of 12 datasets, while in the mixed ensemble they obtain better result in 5 out of 12 datasets. On the remaining datasets the performance of weighted and unweighted formulations perform comparably well.

In Figure 3 we present the co-association matrices of the weighted and unweighted situation and the weights that were obtained by the weighted version (in this case obtained by the WEIGHTED+Δ_ρ approach). The colour scheme on the co-association matrices goes from blue (zero similarity), to red (highest similarity). The un-weighted co-association was transformed into the weighted



(a) Un-weighted (b) Weighted

(c) Weights

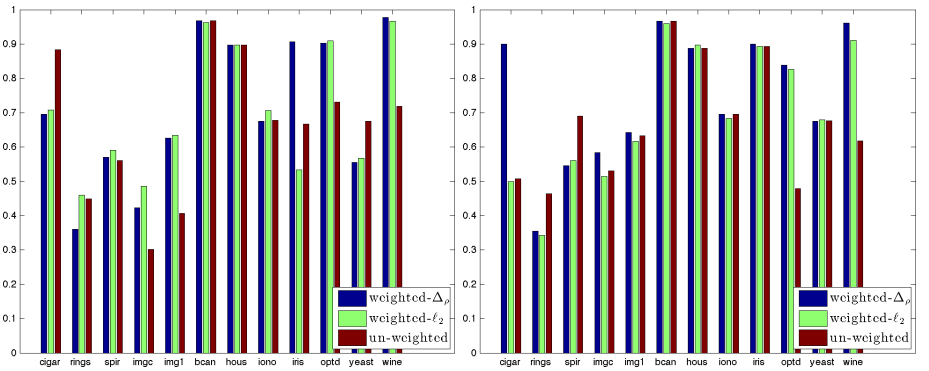
Fig. 3. Example of Co-association matrices. On the left the unweighted version of the co-association, and on the right the weighted co-association, obtained after weighting the partitions.

co-association using the weights presented below, which selectively tune the importance of each partition, turning the matrix into a more structured one.

4.2 Noisy Ensembles

Figure 4 presents the performance obtained on the noisy ensembles, which have been obtained from the k-means and mixed ensembles of the previous section by substituting 20% of the partitions with randomly generated ones. Our purpose is to assess the robustness of the approach to outliers in the ensembles. The results are evaluated in terms of the $\mathcal{H}$ criterion.

As we can see, the performance of weighted approaches tends to be more stable than the un-weighted version. There are isolated dataset where the un-weighted version improved the performance (when compared to non-noisy ensembles), but this situation is not generalizable to the other datasets. The opposed situation was also observed, with the weighted approaches improving the performance (when compared with non-noisy ensembles), but the general trend was to conserve the previous result. if we compare the two types of regularization, we see that WEIGHTED+Δ_ρ apparently was more stable, preserving in more situations the previous result.



(a) k-means ensemble (b) mixed ensemble

Fig. 4. Performance Evaluation for the noisy k-means and mixed ensembles in terms of Accuracy. The weighted approaches WEIGHTED+Δ_ρ and WEIGHTED+ℓ_2 are compared against the unweighted one.

5 Conclusions and Future Work

One of the drawbacks of the classical clustering combination methodologies is that all the partitions of the ensemble have the same weight in the combination process. In this paper we propose a consensus clustering approach with a weighting mechanism that allows to select a subset of the ensemble becoming more robust to noisy ensembles. Our approach tries to find a median-partition based on co-occurences of objects in clusters. We follow an alternating optimization procedure, which allows the determination of the median-partition and the weights vector. Experiments on syntethic and real-world datasets show that the proposed approach outperforms state-of-the-art approaches delivering more robust results. Future work will focus on the application of this framework to large-scale problems.

Acknowledgments. This work was partially funded by FCT under grants SFRH/PROTEC/49512/2009, PTDC/EEI-SII/2312/2012 (LearningS project), and by the ADEETC from Instituto Superior de Engenharia de Lisboa, whose support the authors gratefully acknowledge.

References

1. Ghosh, J., Acharya, A.: Cluster ensembles. WIREs Data Mining and Knowledge Discovery 1(4), 305–315 (2011)
2. Vega-Pons, S., Ruiz-Shulcloper, J.: A survey of clustering ensemble algorithms. IJPRAI 25(3), 337–372 (2011)
3. Fred, A., Jain, A.: Combining multiple clustering using evidence accumulation. IEEE Trans Pattern Analysis and Machine Intelligence 27(6), 835–850 (2005)
4. Strehl, A., Ghosh, J.: Cluster ensembles - a knowledge reuse framework for combining multiple partitions. J. of Machine Learning Research 3 (2002)

5. Karypis, G., Kumar, V.: A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM J. Sci. Comput. 20(1), 359–392 (1998)
6. Rota Bulò, S., Lourenço, A., Fred, A., Pelillo, M.: Pairwise probabilistic clustering using evidence accumulation. In: Hancock, E.R., Wilson, R.C., Windeatt, T., Ulusoy, I., Escolano, F. (eds.) SSPR&SPR 2010. LNCS, vol. 6218, pp. 395–404. Springer, Heidelberg (2010)
7. Lourenço, A., Rota Bulò, S., Rebagliati, N., Fred, A., Figueiredo, M., Pelillo, M.: Probabilistic evidence accumulation for clustering ensembles. In: 2nd Int. Conf. on Pattern Recognition Applications and Methods, ICPRAM 2013 (2013)
8. Lourenço, A., Rota Bulò, S., Rebagliati, N., Fred, A., Figueiredo, M., Pelillo, M.: Consensus clustering using partial evidence accumulation. In: Sanches, J.M., Micó, L., Cardoso, J.S. (eds.) IbPRIA 2013. LNCS, vol. 7887, pp. 69–78. Springer, Heidelberg (2013)
9. Li, T., Ding, C.: Weighted Consensus Clustering. In: Proceedings of 2008 SIAM International Conference on Data Mining (SDM 2008) (2008)
10. Hadjitodorov, S.T., Kuncheva, L.I., Todorova, L.P.: Moderate diversity for better cluster ensembles. Inf. Fusion 7(3), 264–275 (2006)
11. Duarte, F.J.F., Fred, A.L.N., Rodrigues, F., Duarte, J.M.M., Lourenço, A.: Weighted evidence accumulation clustering using subsampling. In: Proceedings of the 6th International Workshop on Pattern Recognition in Information Systems, PRIS 2006, In conjunction with ICEIS, pp. 104–116 (2006)
12. Fern, X.Z., Lin, W.: Cluster ensemble selection. Stat. Anal. Data Min. 1(3), 128–141 (2008)
13. Vega-Pons, S., Correa-Morris, J., Ruiz-Shulcloper, J.: Weighted cluster ensemble using a kernel consensus function. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) CIARP 2008. LNCS, vol. 5197, pp. 195–202. Springer, Heidelberg (2008)
14. Azimi, J., Fern, X.: Adaptive cluster ensemble selection. In: Proceedings of the 21st International Jont Conference on Artifical Intelligence, IJCAI 2009, pp. 992–997. Morgan Kaufmann Publishers Inc., San Francisco (2009)
15. Hong, Y., Kwong, S., Wang, H., Ren, Q.: Resampling-based selective clustering ensembles. Pattern Recognition Letters 30(3), 298–305 (2009)
16. Jia, J., Xiao, X., Liu, B., Jiao, L.: Bagging-based spectral clustering ensemble selection. Pattern Recognition Letters 32(10), 1456–1467 (2011)
17. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. J. of the Royal Stat. Society, Series B, 301–320 (2005)
18. Jain, A.K., Dubes, R.: Algorithms for Clustering Data. Prentice Hall (1988)
19. Ng, A.Y., Jordan, M.I., Weiss, Y.: On spectral clustering: Analysis and an algorithm. In: NIPS, pp. 849–856. MIT Press (2001)
20. Meilă, M.: Comparing clusterings by the variation of information. In: Schölkopf, B., Warmuth, M.K. (eds.) COLT/Kernel 2003. LNCS (LNAI), vol. 2777, pp. 173–187. Springer, Heidelberg (2003)

Variational Image Segmentation and Cosegmentation with the Wasserstein Distance

Paul Swoboda and Christoph Schnörr

Image and Pattern Analysis Group & HCI
Dept. of Mathematics and Computer Science, University of Heidelberg

Abstract. We present novel variational approaches for segmenting and cosegmenting images. Our supervised segmentation approach extends the classical Continuous Cut approach by a global appearance-based data term enforcing closeness of aggregated appearance statistics to a given prior model. This novel data term considers non-spatial, deformation-invariant statistics with the help of the Wasserstein distance in a single global model. The unsupervised cosegmentation model also employs the Wasserstein distance for finding the common object in two images. We introduce tight convex relaxations for both presented models together with efficient algorithmic schemes for computing global minimizers. Numerical experiments demonstrate the effectiveness of our models and the convex relaxations.

Keywords: Wasserstein distance, (co)segmentation, convex relaxation.

1 Introduction

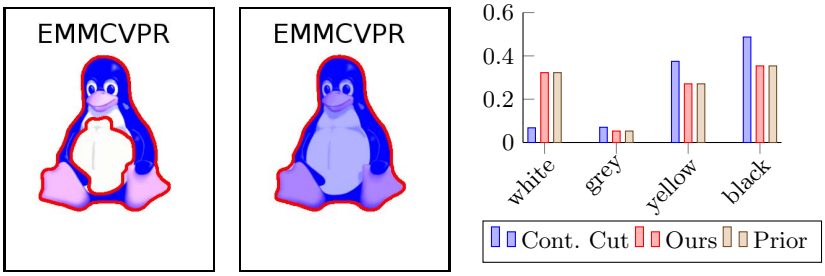
The segmentation problem for k classes consists of finding a partition $(\Omega_1, \dots, \Omega_k)$ of a domain Ω, which means $\Omega_1, \dots, \Omega_k \subset \Omega$, $\Omega_i \cap \Omega_j = \varnothing$ for $i \neq j$ and $\bigcup_{i=1}^k \Omega_i = \Omega$, such that an energy $E(\Omega_1, \dots, \Omega_k)$ is minimized. A commonly used energy functional comes from the minimal partition problem:

$$E(\Omega_1, \dots, \Omega_k) = \frac{1}{2} \sum_{i=1}^{k} \mathrm{Per}(\Omega_i; \Omega) + \sum_{i=1}^{k} \int_{\Omega_i} d^i(x) dx \,, \tag{1}$$

where $\mathrm{Per}(\Omega_i, \Omega)$ is the perimeter of the set Ω_i in Ω and $d^i \in L^1(\Omega), i \in \{1, \dots, k\}$. By minimizing the above functional, k sets are found such that their boundaries are short and the areas they cover are dictated by which potential function d^i has the lowest value. See [5, 12, 16] for treatments of this problem, including relaxations, discretizations and extensions of the minimization problem (1). In the case of two classes this is the well-known Continuous Cut segmentation model, see [8]. This model can be exactly solved by variational methods, see [9].

Often the potential functions $d^i(x) = -\log(p^i(I(x)))$ are chosen as the negative log-likelihood of some probability density p^i modelling the data. Using such potentials d^i poses in general the following problems:

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 321–334, 2013.
© Springer-Verlag Berlin Heidelberg 2013



(a) Cont. Cut (b) Ours (c) Foreground histograms

Fig. 1. Inadequacy of local costs for segmentation. Figure (a) shows the result of the Continuous Cut segmentation, Figure (b) the result of our approach and Figure (c) the resulting and prior foreground color histograms. The blue areas in Figures (a) and (b) denote the areas determined to be foreground by the respective algorithms. The ground truth foreground is the penguin, while the background is the white area behind it as well as the "EMMCVPR" inscription. We set $d^i(x) = -\log(p^i(I(x)))$ in the Continuous Cut model with accurate distributions p^i for the two classes. White and black color can be found in fore- and background, hence local potentials d^i for both classes are not discriminative or may lead to wrong segmentations. Although the local potentials d^i used in the Continuous Cut model indicate that the "EMMCVPR" inscription should be foreground, it is labelled correctly as background, because the regularization strength is set high. However the white belly of the penguin is labelled wrong, because white is more probable to be background and the regularizer is not able to fill in the correct information. In contrast, our approach correctly determines fore- and background, because it works on the appearance histograms of the *whole* segmentation and enforces them to be close to the prespecified ones as can be seen in Figure (c).

1. For some probability densities p^i the resulting potential functions d^i may not be discriminative or even misleading for some $x \in \Omega$. See Figure 1 for an illustration.
2. For individual components of the resulting partition, the corresponding appearance measures may not match well the model distributions p^i.
3. In unsupervised settings like cosegmentation, which is the task of finding the same object in two different images, we have no knowledge of the probability distribution coming from the object we wish to cosegment. Consequently, no probability models p^i or potential functions d^i are available and must be inferred as part of the optimization problem.

These problems more or less persist, even if we use more elaborate potential functions. We resolve this issue by making our data term *dependent* on the whole segmentation.

We propose to solve the first and second of the stated problems by introducing a global term which directly works on global appearance measures. By using such

a term, we force each of the subsets Ω_i of the partition $(\Omega_1, \ldots, \Omega_k)$ to have an appearance measure which is near a prespecified one. To approach the third problem, we introduce a closely related global term, which depends on both appearance measures of the common object in the two images and ensures that they are similar.

1.1 Related Work

Segmentation. Foreground/background segmentation with the Wasserstein distance was already proposed in the two papers [15] and [7].

Peyré et al. introduce in [15] a data term based on the Wasserstein distance and an approximation thereof for reasons of efficiency. The model proposed there is not convex, so it may get stuck in local minima. By contrast, we derive a fully convex model and work directly with the Wasserstein distance.

The work of Chan et al. in [7] boils down to the Continuous Cut model. The novelty is the computation of the local costs d^1 and d^2 from (1). They are computed by comparing patches around pixels to a foreground and a background histogram with the Wasserstein distance. The model remains convex, as it amounts to solving a Continuous Cut, so global minimizers can be computed very efficiently with existing methods. Our approach differs in that we use the Wasserstein distance (i) on arbitrary images opposed to grayvalue images and (ii) as a truly global data term that *depends* on the segmentation. We point out however that the limitation to grayvalue images in [7] is only made for computational reasons as the one dimensional Wasserstein distance is very fast to compute and is not an inherent limitation of the algorithm in [7].

Cosegmentation. Rother et al. introduce in [18] the cosegmentation task into the literature. To solve the problem, they propose to find a MAP configuration of an MRF with pairwise potentials for spatial coherency and a global constraint to actually cosegment two images. The resulting MRF is not easy to optimize however, and the authors employ a trust region algorithm, which they call trust region graph cut. The algorithm they employ is not guaranteed to find a global optimum, may get stuck in local optima and is dependent upon initialization. In comparison, we solve a convex relaxation that is not dependent upon initialization and gives a reasonably tight global optimum of the relaxed problem.

Vicente et al. give in [19] an overview over several models for cosegmentation. They all have in common that they seek the object to be cosegmented to have similar appearance histograms. The approaches considered in [19] fall into two categories: (i) the histogram matching term may not be very general or (ii) may be difficult to optimize. Approaches falling into category (ii) are solved with EM-type algorithms which alternatingly compute appearance models and then match according to them. Our approach can match appearance measures very flexibly and leads to a *single convex model*, hence solving both of the problems of the approaches encountered in the paper [19].

Another approach to cosegmentation is presented in [20], where object proposals for the objects to be cosegmented are computed and taken as labels in a graphical model. This approach is different from ours, as it relies heavily on object proposals, which are computed with sophisticated but mathematically less explicit methods from the realm of computer vision. For these proposals a big array of complex features is computed. These features are used to compare objects in different images and find the matching ones. Our model does not need object proposals to be computed but finds the cosegmented objects in a mathematically more explicit variational manner by minimizing one single convex energy function. Still, sophisticated features can be introduced in our model as well, however this is not the focus of this paper.

1.2 Contribution

We present

- A new variational model for supervised segmentation with global appearance-based data-terms, see Section 2,
- a new variational model for unsupervised cosegmention of two images based on the similarity of the appearance measures of the respective cosegmentations, see Section 3,
- convex relaxations for both models together with efficient numerical schemes to minimize them, see Section 4,
- experimental validation of the proposed approach, see Section 5.

1.3 Notation

For vectors or vector valued functions $u = \left(u^1, \dots, u^k\right)^\top$ we will denote its i-th entry by u^i. Throughout the paper let $\Omega \subset \mathbb{R}^l$ be the image domain, typically $\Omega = [0,1]^2$. We will denote images by $I, I_1, I_2 : \Omega \to \mathcal{M}$. Images will take values in a measurable space $(\mathcal{M}, \Sigma)$. $\mathcal{M}$ denotes the values an image can take, while $\Sigma \subset 2^{\mathcal{M}}$ is a σ-algebra over $\mathcal{M}$. We also assume we are given a measurable similarity function $c : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$. An example is the k-dimensional euclidean space with the Borel σ-algebra: $(\mathcal{M}, \Sigma) = (\mathbb{R}^l, \mathcal{B}(\mathbb{R}^l))$, $c(v_1, v_2) = \|v_1 - v_2\|^p$. For gray-value images we have $l = 1$ and for color images $l = 3$.

For $v \in \mathcal{M}$ consider the dirac measure $\delta_v(A) = \begin{cases} 0, v \notin A \in \Sigma \\ 1, v \in A \in \Sigma \end{cases}$.

Given a measurable subset $\Theta \subset \Omega$ of the image domain and an image $I : \Omega \to \mathcal{M}$, consider the measure $\mu^I_\Theta : \Sigma \to \mathbb{R}_+$ which records the values which I takes on the subset Θ:

$$\mu^I_\Theta = \int_\Theta \delta_{I(x)} dx \,. \tag{2}$$

Please note that the right hand side of (2) is a measure-valued integral, hence again a measure. It follows that for a measurable set $A \in \Sigma$, we have $\mu^I_\Theta(A) = \int_\Theta \mathbb{1}_{\{I(x) \in A\}} dx$, which is the area in $\Theta \subset \Omega$ where I takes values in $A \subset \mathcal{M}$.

Therefore, the measure μ^I_Θ captures the appearance of the image region $\Theta \subset I$. See Figure 2 for an illustration.

In the discrete case, i.e. $\Omega = \{1, \dots, n\}$, $\mathcal{M} = \{1, \dots, m\}$, the appearance measure μ^I_Θ is the histogram of the image values on the subset Θ: $\mu^I_\Theta(A) = \#\{x \in \Theta : I(x) \in A\}$. The general setup however allows to state the model in a continuous setting and makes notation easier.

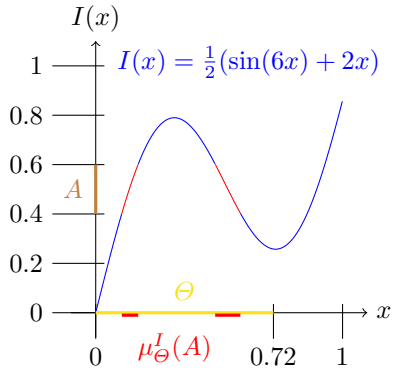


Fig. 2. Illustration of the construction of the appearance measure $\mu^I_\Theta(A)$ for a subregion $\Theta = [0, 0.72] \subset \Omega = [0, 1]$ and a subset of values $A = [0.4, 0.6] \subset \mathcal{M} = [0, 1]$. The blue parts of the curve $I(x) = \frac{1}{2}(\sin(6x) + 2x)$ do not contribute to $\mu^I_\Theta(A)$ while the red ones do. Note that Definition (2) applies also to vector-valued and more generally to $\mathcal{M}$-valued images.

For a convex formulation of our models we introduce the space of functions

$$\mathcal{E}_k = \left\{ u \in BV(\Omega)^k \, : u(x) \in \{e_1, \dots, e_k\} \text{ a.e. } x \in \Omega \right\}, \tag{3}$$

where e_i are the unit vectors in $\mathbb{R}^k$ and $BV(\Omega)$ is the space of functions of bounded variations, see [2] for an introduction to this topic.

$\mathcal{E}_k$ is not a convex set and therefore it is not amenable for use in minimization problems in practice. Hence we consider the convex hull of $\mathcal{E}_k$:

$$\Delta_k = \left\{ u \in BV(\Omega)^k \, : u(x) \in \operatorname{conv}\{e_1, \dots, e_k\} \text{ a.e. } x \in \Omega \right\}, \tag{4}$$

which is the space of functions having values in the k-dimensional unit simplex.

1.4 Wasserstein Distance

Given two measures $\mu_1, \mu_2 : \Sigma \to \mathbb{R}_+$ with $\mu_1(\mathcal{M}) = \mu_2(\mathcal{M})$, the Wasserstein distance $W(\mu_1, \mu_2) \in \mathbb{R}$ of these two measures is computed by evaluating the cost of an optimal rearrangement of μ_1 onto μ_2 with regard to a similarity function c on $\mathcal{M}$. Specifically, consider the space of all *rearrangements* of μ_1 onto μ_2, that is all measures on $\mathcal{M} \times \mathcal{M}$ with marginals μ_1 and μ_2:

$$\Pi(\mu_1, \mu_2) = \{\pi \text{ a measure on } \mathcal{M} \times \mathcal{M} \, : \begin{array}{l} \pi(A \times \mathcal{M}) = \mu_1(A) \\ \pi(\mathcal{M} \times B) = \mu_2(B) \end{array} \quad \forall A, B \in \Sigma\}. \tag{5}$$

Measures in Π are also known as *coupling measures* or *transport plans* in the literature. We will stick to the name *coupling measures*. The Wasserstein distance

is defined as the infimum over all possible rearrangements with regard to the cost c:

$$W(\mu_1, \mu_2) = \inf_{\pi \in \Pi(\mu_1, \mu_2)} \int_{\mathcal{M} \times \mathcal{M}} c \, d\pi \,, \tag{6}$$

It can be shown that under mild assumptions on c the infimum is attained and the distance is finite, see [21] for an in-depth treatise of the Wasserstein distance. The Wasserstein distance is a metric on the space of probability measures for c a metric on $\mathcal{M}$, hence it gives a reasonable distance for measures for c properly chosen.

The minimization problem (6) has linear objective and constraints and is therefore a linear optimization problem, which means it is globally solvable. Moreover it is jointly convex in both of its arguments under mild conditions as well, so it is naturally usable in a convex variational setting, see Theorem 4.8 in [21]

Finally, the Wasserstein distance offers much flexibility in modelling similarity and dissimilarity of measures by choosing an appropriate cost function c in (6).

2 Variational Model for Supervised Segmentation

We will combine into a single variational problem the spatial regularization from the minimal partition problem (1), appearance measures from subsets of the image domain constructed by (2) and the Wasserstein distance (6) for comparing the resulting measures to obtain a new model for segmenting images.

We assume in this setting that one image $I : \Omega \to \mathcal{M}$ and k probability measures μ^i over $\mathcal{M}$ are given. For a partition $(\Omega_1, \ldots, \Omega_k)$ of Ω we enforce the measures $\mu^I_{\Omega_i}$ to be similar to the prespecified measures μ^i by using the Wasserstein distance (6).

Replacing the data term with the potential functions d^i in the minimal partition problem (1) by the Wasserstein distance yields

$$E_{seg}(\Omega_1, \ldots, \Omega_k) = \frac{1}{2} \sum_{i=1}^{k} \mathrm{Per}(\Omega_i, \Omega) + \sum_{i=1}^{k} W\left(\mu^I_{\Omega_i}, |\Omega_i| \cdot \mu^i\right). \tag{7}$$

The additional multiplicative factor $|\Omega_i|$ in the second argument of the Wasserstein distance above is needed to ensure that measures of equal mass are compared, as otherwise the Wasserstein distance is ∞. This is due to the fact that the space (5) of coupling measures Π is empty for measures of differing masses.

Minimizing (7) over all partitions $(\Omega_1, \ldots, \Omega_k)$ of Ω results in partitions, which have regular boundaries due to the perimeter term, and the appearance measures of the partition $\mu^I_{\Omega_i}$ being similar to the given appearance measures μ^i. Note that the measures $\mu^I_{\Omega_i}$ depend on the partition through Ω_i.

As for the minimal partition problem in [5, 9, 12, 16], we replace the sets Ω_i by indicator functions $u^i = \mathbb{1}_{\Omega_i}$ and minimize over them.

Proposition 1. *Let* $u^i = \mathbb{1}_{\Omega_i}$. *Then* (7) *is equal to*

$$J_{seg}(u) = \tfrac{1}{2}\sum_{i=1}^k \int_\Omega |Du^i|\,dx + \sum_{i=1}^k W\Big(\int_\Omega u^i(x)\delta_{I(x)}dx, \int_\Omega u^i(x)dx \cdot \mu^i\Big)\,, \tag{8}$$

where the Total Variation $\int_\Omega |Du^i|\,dx$ *is to be understood as*

$$\int_\Omega |Du^i|\,dx \coloneqq \sup\left\{\int_\Omega u^i \cdot \operatorname{div}(g)dx \;:\; g \in C_c^1(\Omega), \|g\|_\infty \le 1\right\}. \tag{9}$$

Minimizing (7) *over all partitions* $(\Omega_1, \dots, \Omega_k)$ *such that each* Ω_i *has a finite perimeter is equivalent to minimizing* (8) *over* $u \in \mathcal{E}_k$ *given by* (3).

Proof. A partition $(\Omega_1, \dots, \Omega_k)$ corresponds to a vector-valued function $u \in \mathcal{E}_k$ bijectively by $\Omega_i \Leftrightarrow u^i = \mathbb{1}_{\Omega_i}$. By the Coarea formula $\operatorname{Per}(\Omega_i, \Omega) = \int_\Omega |Du^i|\,dx$ holds, see [2]. The Wasserstein term is equal, since

$$\mu^I_{\Omega_i} = \int_{\Omega_i} \delta_{I(x)}dx = \int_\Omega u^i(x)\delta_{I(x)}dx \quad \text{and} \quad |\Omega_i| = \int_\Omega u^i(x)dx\,. \tag{10}$$

Thus, $J_{seg}(u) = E_{seg}(\Omega_1, \dots, \Omega_k)$, which proves the first claim.

The equivalence of both minimization problems stems from the fact, that sets of finite perimeter correspond bijectively to indicator functions of finite variation, see again [2], and partitions correspond bijectively to vector-valued functions such that $\sum_{i=1}^k u^i = \mathbb{1}$ and $u \in \{0,1\}^k$, hence

$$\inf_{u \in \mathcal{E}_k} J_{seg}(u) = \inf_{(\Omega_1,\dots,\Omega_k) \text{ is a partition}} E_{seg}(\Omega_1, \dots, \Omega_k)\,, \tag{11}$$

which proves the second claim.

The functional $J_{seg}(\cdot)$ from (8) is convex, as the Total Variation term is convex and the Wasserstein term is so as well by Theorem 4.8 in [21]. However $\mathcal{E}_k$ is a nonconvex set, so taken together minimizing $\min_{u\in\mathcal{E}_k} J_{seg}(u)$ is not a convex problem. Thus, for practically finding a minimizer of (8), we have to relax the domain over which we optimize. The following problem is convex, as Δ_k is the convex hull of $\mathcal{E}_k$:

$$\inf_{u \in \Delta_k} J_{seg}(u)\,. \tag{12}$$

Remark 1. It is possible to introduce additional local costs $d^i : \Omega \to \mathbb{R}$ without compromising convexity of (8), i.e. to minimize

$$\inf_{u \in \Delta_k} J_{seg}(u) + \sum_{i=1}^k \int_\Omega d^i(x)u^i(x)dx\,. \tag{13}$$

Numerically it comes at a marginal cost to do so. However we chose not to use local costs to demonstrate most directly the power of the global Wasserstein cost.

Remark 2. (8) is the Continuous Cut model when we choose $k = 2$, two points $v_1, v_2 \in \mathcal{M}$ and $\mu^1 = \delta_{v_1}$ and $\mu^2 = \delta_{v_2}$, as then we can replace the Wasserstein distance by multiplication with a local data term. The resulting model is the minimal partition problem (1) for two classes. [9] shows that a global minimizer of the non-relaxed problem can be obtained by thresholding.

3 Variational Model for Unsupervised Cosegmentation

Let two images $I_1, I_2 : \Omega \to \mathcal{M}$ be given and let $\mathcal{M}$ and c be as above. Suppose an object is present in both images, but we have no information about the appearance, location or size of it, Thus, we consider the fully unsupervised setting. The task is to search for two sets $\Omega_1, \Omega_2 \subset \Omega$ such that Ω_1 and Ω_2 are the areas occupied in I_1 resp. I_2 by the common object. Let $\mu^{I_1}_{\Omega_1}$ and $\mu^{I_2}_{\Omega_2}$ be the appearance measures of the common object in images I_1 and I_2 respectively. We know that both appearance measures should be very similar. Therefore we will use the Wasserstein distance $W\left(\mu^{I_1}_{\Omega_1}, \mu^{I_2}_{\Omega_2}\right)$ as a penalization term for enforcing similarity of the appearance measures $\mu^{I_1}_{\Omega_1}$ and $\mu^{I_2}_{\Omega_2}$.

Consider the energy

$$E_{coseg}(\Omega_1, \Omega_2) = \sum_{i=1}^{2} \mathrm{Per}(\Omega_i, \Omega) + W\left(\mu^{I_1}_{\Omega_1}, \mu^{I_2}_{\Omega_2}\right) + \sum_{i=1}^{2} P \cdot |\Omega \backslash \Omega_i| \tag{14}$$

where $P > 0$ and $P \cdot |\Omega \backslash \Omega_i|$ penalizes not selecting an area as the common object. This latter term is called the ballooning term in [19] and is needed to avoid the empty cosegmentation. Minimizing (14) results in two sets Ω_1 and Ω_2 which have a short boundary due to the perimeter term and such that the appearance measures $\mu^{I_1}_{\Omega_1}$ and $\mu^{I_2}_{\Omega_2}$ are similar. Note that neither $\mu^{I_1}_{\Omega_1}$ nor $\mu^{I_2}_{\Omega_2}$ are known but completely depend on the segmentation.

The main difference between the segmentation model (7) and the cosegmentation model (14) is that in the segmentation model the second argument in the Wasserstein distance is fixed while we allow it to vary in the cosegmentation model.

By the same arguments as in Section 2 and Proposition 1, we can establish a similar correspondence between (14) and a suitable convex formulation in the space of indicator functions.

Proposition 2. *Let* $u^i = \mathbb{1}_{\Omega_i}$. *Then* (14) *is equal to*

$$J_{coseg}(u^1, u^2) = \begin{array}{l} \sum_{i=1}^{2} \int_\Omega |Du^i| \, dx + W\left(\int_\Omega u^1(x)\delta_{I_1(x)} dx, \int_\Omega u^2(x)\delta_{I_2(x)} dx\right) \\ + \sum_{i=1}^{2} P \cdot \int_\Omega (1 - u^i(x)) \, dx \end{array}. \tag{15}$$

Minimizing $E_{coseg}(\Omega_1, \Omega_2)$ (14) *over all sets* $\Omega_1, \Omega_2 \subset \Omega$ *with finite perimeter is equivalent to minimizing* $J_{coseg}(u^1, u^2)$ *over all* $\{0, 1\}$*-valued functions of finite variation.*

As in Section 2, J_{coseg} is convex, whereas the space of $\{0,1\}$-valued functions is not. Relaxing to functions $u^i : \Omega \to [0,1]$ yields a convex relaxation.

Note that due to aggregating the appearance in the two measures $\mu^{I_1}_{\Omega_1}$ and $\mu^{I_2}_{\Omega_2}$ in a *translation-, rotation- and deformation-invariant* way, the resulting cosegmentation energy also exhibits these properties.

Remark 3. (14) implicitly defines the size constraint $|\Omega_1| = |\Omega_2|$, since the Wasserstein distance requires both measures to have equal mass. Weakening this constraint is beyond the scope of this paper.

4 Numerical Implementation

It is common to solve convex large-scale non-smooth problems with first order algorithms like [3, 6, 10]. To efficiently solve our models with such schemes, it is necessary to split our energies into suitable convex funtions, such that the proximity operators for each function can be computed efficiently. Our splitting results in $2+k$ convex non-smooth functions for the segmentation functional (8) and 3 such functions with an additional linear term for the cosegmentation functional (15). We use the Generalized Forward-Backward Splitting Algorithm [17], which can handle an arbitrary number of convex functions in a flexible way.

In practice our image domain is discrete. Here we assume $\Omega = \{1, \dots, n\}^2$. The gradient operator will be approximated by forward differences.

We can rewrite the energy function (8) for the segmentation problem as follows by splitting variables for the gradient operator:

$$J_{seg}(u,g) = \chi_{\{\nabla u = g\}} + \chi_{\{u \in \Delta_k\}} + \|g\| + \sum_{i=1}^{k} W^i_{seg}(u^i)\,, \tag{16}$$

where $W^i_{seg}(u) = W\left(\sum_{x\in\Omega} u(x)\delta_{I(x)}, \left(\sum_{x\in\Omega} u(x)\right)\mu^i\right)$ are the Wasserstein terms in (8) and $\chi_{\text{True}} = 0$, $\chi_{\text{False}} = +\infty$ stands for the indicator function. The energy (15) for the cosegmentation problem can be split as follows:

$$J_{coseg}(u,g) = \sum_{i=1}^{2} \left\{\chi_{\{\nabla u^i = g^i\}} + \|g^i\|\right\} + \langle d, u\rangle + \chi_{\{u\in[0,1]^{|\Omega|}\}} + W_{coseg}(u^1, u^2)\,, \tag{17}$$

where $W_{coseg}(u_1,u_2) = W\left(\sum_{x\in\Omega} u^1(x)\delta_{I_1(x)}, \sum_{x\in\Omega} u^2(x)\delta_{I_2(x)}\right)$ is the Wasserstein term in (15) and $\langle d, u\rangle$ takes care of the balloning term.

Solving (8) and (15) with the Generalized Forward-Backward Splitting algorithm from [17] requires solving efficiently the proximity operators for each convex function in (16) and (17). The proximity operator for a function G at point u^0 is defined by

$$\text{prox}_G(u^0) = \text{argmin}_u \frac{1}{2}\|u - u^0\|^2 + G(u)\,. \tag{18}$$

Proximity operators for all the convex functions in (16) and (17) except for the Wasserstein term can be computed very efficiently by standard methods:

- $\text{prox}_{\delta_{\{\nabla u = g\}}}(u^0, g^0)$ is the projection onto the set $\{\nabla u = g\}$ and can be computed with Fourier transforms.
- $\text{prox}_{\Delta_k}(u^0)$ is the projection onto the simplex and can be computed in a small finite number of steps with the algorithm from [14].
- $\text{prox}_{\|g\|}(g^0)$ amounts to computing the shrinkage operator.

See again [17] concerning how these proximity operators are combined.

The Wasserstein proximity operator can be computed efficiently with the technique detailed below.

4.1 Dimensionality Reduction for the Proximity Operator of the Wasserstein Distance

In general, computing the proximity operator of the Wasserstein distance can be expensive and requires solving a quadratic program with $|\Omega| + |\mathcal{M}|^2$ variables. However due to symmetry we can significantly reduce the size of the quadratic program to $|\mathcal{M}|^2$ variables, such that the Wasserstein proximation step is *independent of the size of the image*.

In practice we will solve the problem on an image grid $\Omega = \{1, \dots, n\}^2$ and the number of values a pixel can take is usually significantly smaller than the number of pixels (e.g. 256 values for gray-value images and for color pictures we may cluster the colors to reduce the number of distinct values as well, while the number of pixels $|\Omega| = n^2$ can be huge). Hence, we may assume $|\Omega| \gg |\mathcal{M}|$.

In the following we only discuss the segmentation case due to space constraints.

Due to the representation of the Wasserstein distance (6), the proximity operator $\text{prox}_{W^i_{seg}}(u^0) = \text{argmin}_u \|u - u^0\|^2 + W^i_{seg}(u)$ of the Wasserstein distance in the segmentation problem (16) can be written equivalently as

$$\begin{array}{rl} \text{argmin}_{\{u,\pi\}} & \sum_{x\in\Omega}(u(x) - u^0(x))^2 + \int_{\mathcal{M}\times\mathcal{M}} c(v_1, v_2)\, d\pi(v_1, v_2) \\ \text{s.t.} & \pi(\mathcal{M} \times A) = \sum_{\{x \in I^{-1}(A)\}} u(x) \quad \forall A \subset \mathcal{M} \\ & \pi(B \times \mathcal{M}) = \left(\sum_{x\in\Omega} u(x)\right) \mu^i(B) \quad \forall B \subset \mathcal{M} \\ & \pi \geq 0 \end{array} \tag{19}$$

Note that the Wasserstein distance term above is invariant to permutations of values inside each set $\{I^{-1}(v)\}\ \forall v \in \mathcal{M}$. The quadratic term $\sum_{x\in\Omega}(u(x) - u^0(x))^2\, dx$ also possesses similar symmetries. This enables us to reduce the number of variables as follows:

Let $n_v = \#\{I^{-1}(v)\}$ be the number of pixels which take the value $v \in \mathcal{M}$ and let $\mu^0 = \sum_{x\in\Omega} u^0(x)\delta_{I(x)}$. Consider the problem

$$\begin{array}{rl} \text{argmin}_{\pi\in\mathcal{P}(\mathcal{M}\times\mathcal{M})} & \int_{\mathcal{M}} n_v \cdot \left(\pi(\mathcal{M} \times \{v\}) - \mu^0(\{v\})\right)^2 dv + \int_{\mathcal{M}\times\mathcal{M}} c(v_1, v_2)\, d\pi(v_1, v_2) \\ \text{s.t.} & \pi(B \times \mathcal{M}) = \pi(\mathcal{M} \times \mathcal{M}) \cdot \mu^1(B) \quad \forall B \subset \mathcal{M} \\ & \pi \geq 0 \end{array} \tag{20}$$

The relation between the two minimization problems (19) and (20) is:

Lemma 1. *The minimization problems* (19) *and* (20) *are equivalent in the following sense: For* $I(x) = v \in \mathcal{M}$ *the optimal solutions* $\hat{u}$ *of* (19) *and* $\hat{\pi}$ *of* (20) *correspond to each other via the relation*

$$\hat{u}(x) = u^0(x) + \frac{\hat{\pi}(\mathcal{M} \times \{v\}) - \mu^0(\{v\})}{n_v}. \tag{21}$$

Lemma 1 allows for efficiently solving (19) via (20) and (21).

5 Experiments

To show the performance of our method we have restricted ourselves to only consider colors as features. Hence the features alone are not very distinctive, but the whole energy function makes our approach work. Our label space $\mathcal{M}$ is the CIE 1931 color space and our cost function c will be derived from the euclidean distance on the above color space. *More sophisticated features can be used in our variational models with no additional computational cost in the minimization procedure. Choosing such features however goes beyond the scope of this paper, that is purely devoted to the novel variational approach, rather than to specific application scenarios.* Also, more sophisticated regularizers can be employed as well, e.g. one could vary weights in the total variation term or use nonlocal versions of it, see [11] for the latter.

5.1 Segmentation

In our experimental setting we assume that we have probability measures μ^1, μ^2 at hand for the foreground and background classes, which we employ in the global Wasserstein data-term. We could in addition determine potential functions to enhance segmentation results and solve model (13), e.g. by $d^i(x) = -\log(p^i(I(x)))$, where p^i is the density of μ^i. We chose to not use the latter to show the strength of the global Wasserstein term alone and the tightness of our relaxation. See [5, 9, 12, 16] for numerical examples of segmentation results with potential functions alone.

For the foreground and background appearance measures we chose a part of the foreground and background of the image respectively and constructed prior appearance measures μ^1, μ^2 from them. In a preprocessing step, we clustered the color values of the image by the k-means method [13]. The number of prototypes was set to 50. The quadratic problem in the prox-step (20) of the Wasserstein distance is thus a 50×50 convex quadratic problem and efficiently solvable. We conducted four experiments with textured objects, for which it is not always easy to find discriminative prototypical vectors, but where the color histogram catches much information about the objects' appearance, see figure 3. Note for example that the cheetah's fur has the same color as the sand in the image, but the distribution of the black dots and the color of the rest of the fur is still distinctive. The fish has black regions, exactly as in the background, but the white and black pattern is distinctive again, so a reasonable segmentation can be obtained.

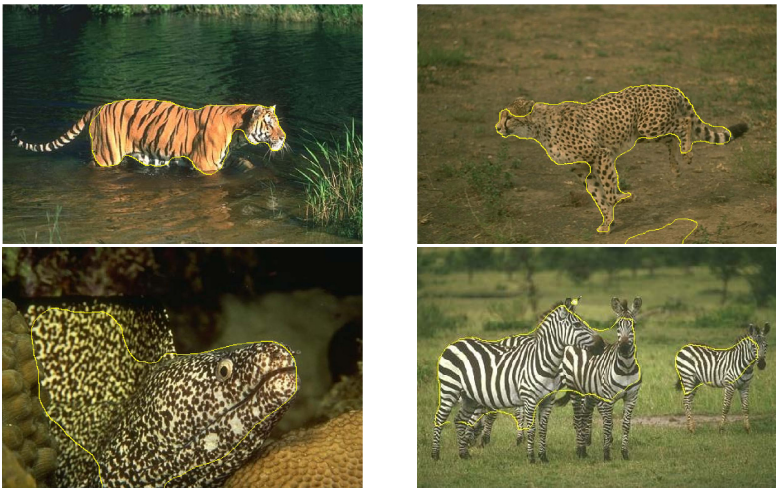


Fig. 3. Supervised segmentation experiments with global segmentation-dependent data term using the Wasserstein distance. Note that because the results correspond to *global* optima of a single convex functional, undesired parts of the partition are solely due to the – in our case: simple color – features and the corresponding prior appearance measures.

5.2 Cosegmentation

For cosegmentation we first subdivide the image into superpixels with SLIC [1]. Then we modify the cost function c as follows: For each superpixel in image 1 we consider k nearest superpixels in image 2 and vice versa. For these pairs we let c be the euclidean distance. For all other pairs of superpixels we set c to ∞. Obviously, the optimal transport plan will be zero where the distance c is ∞, hence we may disregard such variables. By this procedure we reduce the problem size and computational complexity substantially while not reducing the quality of the solution. The prox-step $\text{prox}_{W_{coseg}}(u^1, u^2)$ can be further reduced with a technique similar to the one presented in Section 4.1.

Four experiments can be seen in figure 4. The foreground objects were taken from the dataset [4]. We rotated these objects, translated them and added different backgrounds. As the Wasserstein term does not depend upon location and spatial arrangement of the pixels contributing to the cosegmentation, we could find the common objects independently of where and in which orientation they were located in the images without explicitly enumerating over all different possible such configurations, but by *solving a single convex optimization problem to its global optimum*. Note that in this unsupervised setting, no prior knowledge about the objects is used.

In both experimental settings our method produced functions u^i which were nearly indicator functions except on some parts of the boundaries. Empirically, our relaxation seems to be quite tight.

Fig. 4. Unsupervised cosegmentation: foreground regions in two images are separated at arbitrary locations where the Wasserstein distance between the corresponding histograms is small. This distance depends on the unknown segmentation, and both are consistently determined by a single convex variational problem. No prior knowwledge at all was used in these unsupervised experiments.

6 Conclusion

We presented new variational models for segmentation and cosegmentation. Both utilize the Wasserstein distance as a global term for enforcing closeness between suitable appearance measures. We also derived convex relaxations of the models and presented efficient numerical methods for minimizing them. Both models can be easily augmented by using different regularizers or additional data terms and any features known from the literature.

Acknowledgements. The authors would like to thank Marco Esquinazi for helpful discussions.

References

1. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Susstrunk, S.: SLIC Superpixels. Technical report, EPFL (June 2010)
2. Ambrosio, L., Fusco, N., Pallara, D.: Functions of Bounded Variation and Free Discontinuity Problems (Oxford Mathematical Monographs). Oxford University Press, USA (2000)
3. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. Found. Trends Mach. Learning 3(1), 1–122 (2010)
4. Wang, J., Gelautz, M., Kohli, P., Rott, P., Rhemann, C., Rother, C.: Alpha matting evaluation website

5. Chambolle, A., Cremers, D., Pock, T.: A Convex Approach to Minimal Partitions. SIAM J. Imag. Sci. 5(4), 1113–1158 (2012)
6. Chambolle, A., Pock, T.: A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. Journal of Mathematical Imaging and Vision 40(1), 120–145 (2011)
7. Chan, T., Esedoglu, S., Ni, K.: Histogram Based Segmentation Using Wasserstein Distances. In: Sgallari, F., Murli, A., Paragios, N. (eds.) SSVM 2007. LNCS, vol. 4485, pp. 697–708. Springer, Heidelberg (2007)
8. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Trans. Imag. Proc. 10(2), 266–277 (2001)
9. Chan, T.F., Esedoglu, S., Nikolova, M.: Algorithms for Finding Global Minimizers of Image Segmentation and Denoising Models. SIAM J. Appl. Math. 66(5), 1632–1648 (2006)
10. Eckstein, J., Bertsekas, D.P.: On the Douglas—Rachford splitting method and the proximal point algorithm for maximal monotone operators. Mathematical Programming 55, 293–318 (1992)
11. Gilboa, G., Osher, S.: Nonlocal Operators with Applications to Image Processing. Multiscale Modeling & Simulation 7(3), 1005–1028 (2008)
12. Lellmann, J., Schnörr, C.: Continuous Multiclass Labeling Approaches and Algorithms. SIAM J. Imag. Sci. 4(4), 1049–1096 (2011)
13. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proc. 5th Berkeley Symp. Math. Stat. Probab., Univ. Calif. 1965/1966, vol. 1, pp. 281–297 (1967)
14. Michelot, C.: A finite algorithm for finding the projection of a point onto the canonical simplex of rn. J. Optim. Theory Appl. 50(1), 195–200 (1986)
15. Peyré, G., Fadili, J., Rabin, J.: Wasserstein Active Contours. Technical report, Preprint Hal-00593424 (2011)
16. Pock, T., Schoenemann, T., Graber, G., Bischof, H., Cremers, D.: A Convex Formulation of Continuous Multi-label Problems. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 792–805. Springer, Heidelberg (2008)
17. Raguet, H., Fadili, J., Peyré, G.: Generalized Forward-Backward Splitting. Technical report, Preprint Hal-00613637 (2011)
18. Rother, C., Minka, T., Blake, A., Kolmogorov, V.: Cosegmentation of image pairs by histogram matching - incorporating a global constraint into mrfs. In: CVPR, pp. 993–1000. IEEE, Washington, DC (2006)
19. Vicente, S., Kolmogorov, V., Rother, C.: Cosegmentation revisited: Models and optimization. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 465–479. Springer, Heidelberg (2010)
20. Vicente, S., Rother, C., Kolmogorov, V.: Object cosegmentation. In: CVPR, pp. 2217–2224. IEEE (2011)
21. Villani, C.: Optimal Transport: Old and New, 1st edn. Grundlehren der mathematischen Wissenschaften. Springer (November 2008)

A Convex Formulation for Global Histogram Based Binary Segmentation

Romain Yıldızoğlu, Jean-François Aujol, and Nicolas Papadakis

Univ. Bordeaux, Institut de Mathématiques de Bordeaux (IMB, UMR 5251),
351 Cours de la Libération F-33400 Talence, France
{romain.yildizoglu,jean-francois.aujol,
nicolas.papadakis}@math.u-bordeaux1.fr
http://www.math.u-bordeaux1.fr/imb

Abstract. In this paper, we present a general convex formulation for global histogram-based binary segmentation. The model relies on a data term measuring the histograms of the regions to segment w.r.t. reference histograms as well as TV regularization allowing the penalization of the length of the interface between the two regions. The framework is based on some l^1 data term, and the obtained functional is minimized with an algorithm adapted to non smooth optimization. We present the functional and the related numerical algorithm and we then discuss the incorporation of color histograms, cumulative histograms or structure tensor histograms. Experiments show the interest of the method for a large range of data including both gray-scale and color images. Comparisons with a local approach based on the Potts model or with a recent one based on Wasserstein distance also show the interest of our method.

Keywords: segmentation, global histogram, convex, structure tensor, non smooth optimization.

1 Introduction

Image segmentation has been the subject of active research for more than 20 years (see e.g. [2,11] and references therein). For instance, we can refer to the seminal work of Mumford and Shah [16], or to its very popular approximation with level sets developed by Chan and Vese in [9]. This last work provides a very flexible algorithm to segment an image into two homogeneous parts, each one being characterized by its mean gray level value.

In the case of textured images, a lot of extensions of [9] have been proposed to enhance the mean value image segmentation model. For instance, local histograms are used in [31,17], Gabor filters in [27] and [20], wavelet packets in [3] and textures are characterized thanks to the structure tensor in [6,26].

When considering the global histograms of the regions to segment, there also exist a large body of literature [1,7,15,14] also based on [9]. Recent works make use of the Bhattacharyya distance [30] or the Wasserstein distance [21] to compare globally the histograms. It is important to notice that this class of approaches involves complex shape gradient computations [12] for the level set

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 335–349, 2013.
© Springer-Verlag Berlin Heidelberg 2013

evolution equation. Moreover, as these methods all rely on the evolution of a level set function [19], it leads to non convex methods that are sensible to the initialization choice and only a local minimizer of the associated energy is computed. Other models as in [25,28,4,13] that use graph-based methods and also [24] obtain good results without level-sets, but these algorithms are not bound to give a global minimum of the original energy.

Recently, convexification methods have been proposed to tackle this problem, as in [18,23,8,5,29]. The original Chan-Vese model [9] can indeed be convexified, and a global solution can be efficiently computed, for instance with a primal-dual algorithm. A simple thresholding of this global solution provides then a global minimizer of the original non convex problem. Up to our knowledge, such approaches have not been developed yet for global histogram segmentation with length boundary regularization.

The contribution of this paper is thus to introduce a convex model to segment an image into two parts, each region being characterized by its global histogram. This convex model is minimized efficiently by using a non smooth convex optimization algorithm [8]. The model is first developed for the case of grayscale images, and then extended to the case of features based on the structure tensor, and finally to the case of color images.

With respect to the previously mentioned global histogram based segmentation methods relying on the evolution of a level set function, the approach here is much simpler. On the one hand, the proposed algorithm is faster than the ones based on Wasserstein distance [17,21], and on the other hand despite the simplicity of the approach the obtained segmentation results are very good while being independent of the initialization.

The plan of the paper is the following. We introduce our model for global histogram based binary segmentation in Section 2. Our functional relies on some l^1 norm for the data term, and we propose an efficient numerical scheme based on a non smooth convex optimization algorithm. We present in Section 3 numerous numerical examples to show the strength of the method, as well as its limitations. In Section 4, we illustrate the importance of global histogram comparisons to get a global image segmentation and we give some comparisons with the approach of [21]. We conclude in Section 5 and we present some future venues of research.

2 A Convex Variational Formulation

The problem tackled in this paper concerns the segmentation of an image into two parts through global histogram constraints. Our approach is motivated by two issues. The first one concerns the dependence to the initialization, the second is the problem of locality in the histogram comparison.

Notations

Let Ω be the image domain: we assume Ω to be a non empty open bounded subset of $\mathbb{R}^2$ with Lipschitz boundary. We note $|\Omega|$ the size of Ω and $\langle ., . \rangle$ the

standard inner product on $L^2(\Omega)$, that is $\langle u, v\rangle = \sum_{x\in\Omega} u(x)v(x)$ in the discrete setting. Let $I : \Omega \mapsto \Lambda \subset \mathbb{R}^k$ be the image, Λ finite, h^0 and h^1 two given reference histograms: $\Lambda \to [0;1]$, with $\sum_{\lambda\in\Lambda} h^i(\lambda) = 1$, $i = 0, 1$. We aim at estimating a binary segmentation represented by $u : \Omega \to \{0, 1\}$, where the histogram computed on the region $\Omega_0 := \{x \in \Omega, u(x) = 0\}$ is close to h^0 (resp. the histogram on $\Omega_1 := \{x \in \Omega, u(x) = 1\}$ should be close to h^1). The histogram on the region Ω_1 is then computed as:

$$h_u(\lambda) = \frac{1}{|\Omega_1|}\sum_{x\in\Omega} u(x)\mathbb{1}_{I=\lambda}(x) = \frac{1}{\sum_{x\in\Omega} u(x)}\sum_{x\in\Omega} u(x)\mathbb{1}_{I=\lambda}(x) \tag{1}$$

To realize the segmentation, we first require a metric between histograms, and we will thus consider a norm $\| \, . \, \|$ on $\mathbb{R}^\Lambda$. In order to control the length of the interface between the two parts of the partition, a total variation regularization is also considered. The segmentation can therefore be obtained by minimizing the following non-convex energy :

$$J(u) = TV(u) + \left|\left|(h_u - h^1)_{\lambda\in\Lambda}\right|\right| + \left|\left|(h_{1-u} - h^0)_{\lambda\in\Lambda}\right|\right| \tag{2}$$

Convexification

In the aim of defining a convex model, we propose to reformulate the data term comparing histograms. To that end, we can first observe that

$$\| \, h_u - h^1 \, \| = \left|\left|\left(\frac{1}{\sum_\Omega u(x)}\sum_\Omega u(x)\mathbb{1}_{I=\lambda}(x) - h^1(\lambda)\right)_{\lambda\in\Lambda}\right|\right| \tag{3}$$

$$= \left|\left|\frac{1}{\sum_\Omega u(x)}\left(\sum_\Omega u(x)\mathbb{1}_{I=\lambda}(x) - \left(\sum_\Omega u(x)\right)h^1(\lambda)\right)_{\lambda\in\Lambda}\right|\right| \tag{4}$$

Assuming that the size of the area Ω_1 defined by $|\Omega_1| = \sum_\Omega u(x)$ is known, we can only keep the distance

$$\left|\left|\left(\sum_\Omega u(x)\left(\mathbb{1}_{I=\lambda}(x) - h^1(\lambda)\right)\right)_{\lambda\in\Lambda}\right|\right| \tag{5}$$

which is convex in u. Note that a similar problem is tackled in [10] in the slightly different framework of cosegmentation with no total variation, and a solution is found without modifying this term. The very same convex reformulation can be done for the second term involving $1 - u$ and the histogram h^0, thus removing a factor $\frac{1}{\sum_\Omega(1-u(x))} = \frac{1}{|\Omega_0|}$.

With such normalizations, the data terms of the two partitions are no more balanced. Nevertheless, a weighting factor $\beta \in [0, 1]$ can be introduced. This factor represents the ratio $\beta = \sum_\Omega u(x)/|\Omega| = |\Omega_1|/|\Omega|$. As $|\Omega| = |\Omega_0| + |\Omega_1|$, one can see that $\sum_\Omega(1 - u(x))/|\Omega| = |\Omega_0|/|\Omega| = (|\Omega| - |\Omega_1|)/|\Omega| = 1 - \beta$, which gives the normalization factor of the second term.

Defining $g^1_\lambda(x) := \mathbb{1}_{I=\lambda}(x) - h^1(\lambda)$, $g^0_\lambda(x) := \mathbb{1}_{I=\lambda}(x) - h^0(\lambda)$, the final convex model reads:

$$J(u) = TV(u) + \frac{1}{\beta} \parallel (\langle u, g^1_\lambda \rangle_\Omega)_{\lambda\in\Lambda} \parallel + \frac{1}{1-\beta} \parallel (\langle 1-u, g^0_\lambda \rangle_\Omega)_{\lambda\in\Lambda} \parallel . \quad (6)$$

It now remains to choose the distance to compare histograms.

l^1 Data Term

Up to now, the general model has been designed for any data norm. In this section, we choose the l^1 norm: $\parallel h \parallel_{l^1} = \sum_{\lambda\in\Lambda} |h(\lambda)|$. We therefore have the following energy:

$$\begin{aligned} J(u) &= TV(u) + \mu\frac{1}{\beta} \sum_{\lambda\in\Lambda} \left|\langle u, g^1_\lambda \rangle_\Omega\right| + \mu\frac{1}{1-\beta} \sum_{\lambda\in\Lambda} \left|\langle 1-u, g^0_\lambda \rangle_\Omega\right| \quad (7)\\ &= TV(u) + \mu\frac{1}{\beta} \sum_{\lambda\in\Lambda} \left|\sum_{x\in\Omega} u(x) g^1_\lambda(x)\right| + \mu\frac{1}{1-\beta} \sum_{\lambda\in\Lambda} \left|\sum_{x\in\Omega} (1-u(x)) g^0_\lambda(x)\right| . \end{aligned}$$

One can observe that our global data term depends on the whole state $\{u(x), x \in \Omega\}$, for each value $\lambda \in \Lambda$. We want to minimize $J(u)$ on the set $BV(\Omega, \{0,1\})$. As this binary set is not convex, we relax our problem by minimizing $J(u)$ on $BV(\Omega, [0,1])$. We will get back to the original domain by thresholding the solution of the relaxed problem. However, there is no guarantee that this strategy should lead to minimizers of the original problem, and different choices of thresholds may lead to different solutions.

Finally notice that if the original problem has many global minimizers, the result given by the convex minimization can be such that we do not know how to use it in order to get back to the original problem. Even if specific scenarios could be built to make such situation happen, this issue never occurred in our experiments with non-synthetic data.

Since our functional can be written as the sum of non-smooth convex terms, some of them being the composition of a l^1 norm and a linear operator, we will write the problem with dual variables. Let us write A and B the linear operators $A : u \mapsto (\langle u, g^1_\lambda \rangle_\Omega)_{\lambda\in\Lambda}$, $B : u \mapsto (\langle u, g^0_\lambda \rangle_\Omega)_{\lambda\in\Lambda}$. We can write the problem as a saddle point one

$$\arg \max_{q_1,q_2,q_3} \arg\min_u E(q_1, q_2, q_3, u) \quad (8)$$

where

$$\begin{aligned} E(q_1, q_2, q_3, u) &= \langle q_1, \nabla u \rangle_\Omega + \langle q_2, Au \rangle_\Lambda + \langle q_3, B(1-u) \rangle_\Lambda \quad (9)\\ &\quad -\chi_{B(0,1)}(q_1) - \chi_{[-\mu\frac{1}{\beta}, \mu\frac{1}{\beta}]}(q_2) - \chi_{[-\mu\frac{1}{1-\beta}, \mu\frac{1}{1-\beta}]}(q_3) + \chi_{[0,1]}(u)\\ &= \langle q, Ku \rangle_{\Omega\times\Lambda^2} - F^*(q) + G(u) \quad (10) \end{aligned}$$

with $q = (q_1, q_2, q_3) \in (\mathbb{R}^\Omega \times \mathbb{R}^\Omega) \times \mathbb{R}^\Lambda \times \mathbb{R}^\Lambda$ of dimension $2|\Omega| + 2|\Lambda|$, $K : u \mapsto (\nabla u, Au, -Bu)$, $G(u) = \chi_{[0,1]}(u)$, $F^*(q) = \chi_{B(0,1)}(q_1) + \chi_{[-\mu\frac{1}{\beta}, \mu\frac{1}{\beta}]}(q_2) +$

$\chi_{[-\mu\frac{1}{1-\beta},\mu\frac{1}{1-\beta}]}(q_3) - \langle q_3, B1\rangle$, $B(0,1)$ the closed unit disc of $\mathbb{R}^2$. In the previous expressions, χ_C denotes the characteristic function of the convex set C, i.e. $\chi_C(x) = 0$ if $x \in C$, $\chi_C(x) = \infty$ if $x \notin C$. We optimize this criterion with a primal-dual scheme as explained here-after.

Optimization with a Preconditioned Primal-Dual Algorithm (PPD)

We now give the general formulation of the PPD algorithm, and then write the algorithm applied to our functional.

Let X, Y be two finite-dimensional vector spaces. We write $\langle ., . \rangle$ the standard inner products, $K : X \to Y$ a linear operator and $G : X \to \mathbb{R} \cup \{\infty\}$ and $F^* : Y \to \mathbb{R} \cup \{\infty\}$ some convex functions (F^* being the Legendre Fenchel transform of F [22]). We want to solve

$$\min_{x\in X} \max_{y\in Y} \langle Kx, y\rangle + G(x) - F^*(y). \tag{11}$$

In [22], a preconditioned primal-dual algorithm has been proposed to solve such convex problem. The algorithm reads:

Algorithm 1. Preconditioned primal-dual algorithm ([22])

$u^{k+1} = (I + T\partial G)^{-1}\left(u^k - TK^t z^k\right)$
$z^{k+1} = (I + \Sigma\partial F^*)^{-1}\left(z^k - \Sigma K(2u^{k+1} - u^k)\right)$

with T and Σ symmetric positive definite matrices such that $\| \Sigma^{\frac{1}{2}} K T^{\frac{1}{2}} \|^2 < 1$ and $(I + T\partial G)^{-1}(\hat{u}) := \arg\min_{u\in U} G(u) + \frac{1}{2\tau}\left\langle T^{-1}(u - \hat{u}), u - \hat{u}\right\rangle$. Such process converges to a saddle point in $O(\frac{1}{k})$ (with k the number of iterations). As proposed in [22], one can take $T = \mathrm{diag}(\tau_1, ..., \tau_n)$ and $\Sigma = \mathrm{diag}(\sigma_1, ..., \sigma_m)$ with $\tau_j < \frac{1}{\sum_{i=1}^m |K_{i,j}|}$, $\sigma_i < \frac{1}{\sum_{j=1}^n |K_{i,j}|}$.

Application of PPD to Functional (8)

Introducing the following relations:

$$\begin{cases} Tu(x) = \tau_x u(x), & \tau_x < \frac{1}{2+\sum_{\lambda\in\Lambda}(|g^1_\lambda(x)|+|g^0_\lambda(x)|)}, \\ A^t q_2 = \sum_{\lambda\in\Lambda} q_2(\lambda) g^1_\lambda, \\ B^t q_3 = \sum_{\lambda\in\Lambda} q_3(\lambda) g^0_\lambda, \\ \sigma_1 < \frac{1}{4}, \\ \Sigma_2 q_2(\lambda) = \sigma_{2,\lambda} q_2(\lambda), & \sigma_{2,\lambda} < \frac{1}{\sum_{x\in\Omega} g^1_\lambda(x)}, \\ \Sigma_3 q_3(\lambda) = \sigma_{3,\lambda} q_3(\lambda), & \sigma_{3,\lambda} < \frac{1}{\sum_{x\in\Omega} g^0_\lambda(x)}, \\ b = B1, \end{cases} \tag{12}$$

the PPD Algorithm 1 applied to our functional (7) gives the Algorithm 2, where P_C is the orthogonal projection on the convex C:

Algorithm 2.

$$u^{k+1} = P_{[0,1]}\left(u_k - T\left(A^t q_2^k - B^t q_3^k - \mathrm{div}(q_1^k)\right)\right)$$
$$q_1^{k+1} = P_{B(0,1)}\left(q_1^k + \sigma_1 \nabla(2u^{k+1} - u^k)\right)$$
$$q_2^{k+1} = P_{[-\mu\frac{1}{\beta}, \mu\frac{1}{\beta}]}\left(q_2^k + \Sigma_2 A(2u^{k+1} - u^k)\right)$$
$$q_3^{k+1} = P_{[-\mu\frac{1}{1-\beta}, \mu\frac{1}{1-\beta}]}\left(q_3^k - \Sigma_3\left(B(2u^{k+1} - u^k) - b\right)\right).$$

3 Applications

We now present some experiments on synthetic and natural images, assuming that the histograms of the two areas to segment are known. They are in fact given by manual scribbles selected by the user in our practical applications (they are displayed on the images with red rectangles for the first region and green rectangles for the second one).

The balance between the data term and the regularization term is tuned by the user.

The parameter β is always fixed at 0.5, we show in Figure 4 that changing it can improve the result, but its value did not seem critical in our experiments. As our functional is convex, the process is simply initialized with $u = 0.5$ in all our experiments, which is a main difference with existing approaches based on global histograms [17,21,30]. The final solution is currently obtained by thresholding the estimated solution u^* with $u^* \geq 0.5$, the choice of the threshold leading to slightly different results.

3.1 Grayscale Images

In practical applications, images are always corrupted by some noise. In order to get a more robust algorithm, we can work with cumulative histograms instead of histograms. Situations may happen where a pixel value is represented in none of the reference distributions h^0 and h^1. For instance, let us consider that the reference histograms have been learned from a clean image. If this histogram is finely quantified and sparse, adding some noise (or an offset) to the image may lead to such issue. In this case, the values of a lot of noised pixels may belong to empty bins of both reference histograms and the segmentation will fail.

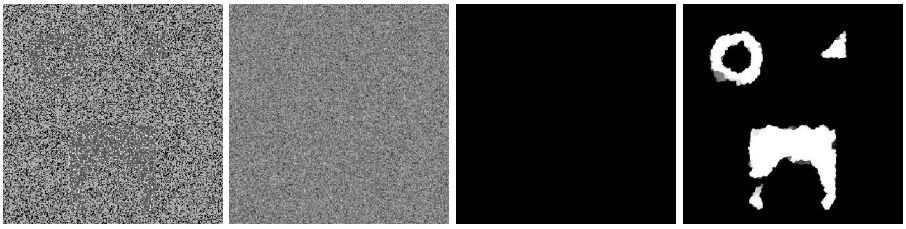
Cumulative histograms are one way to tackle this problem.

All the derivations we have made so far in the paper for histogram based segmentation hold true for cumulative histograms and we get the following functional:

$$J(u) = TV(u) + \mu \sum_{\Lambda} \left(\frac{1}{\beta} \left| \left\langle u, g^1_{C,\lambda} \right\rangle_\Omega \right| + \frac{1}{1-\beta} \left| \left\langle 1-u, g^0_{C,\lambda} \right\rangle_\Omega \right| \right) \tag{13}$$

with $g_{C,\lambda} = \sum_{\gamma \leq \lambda} g^0_\lambda = \mathbb{1}_{I \leq \lambda} - h^0_C(\lambda)$, and the cumulative histogram is obtained as $h^0_C(\lambda) = \sum_{\gamma \leq \lambda} h^0(\gamma)$. The associated numerical schemes are therefore similar.

In order to show the robustness of cumulative histograms to noise, we segment in Figure 1 a synthetic image (a) with two areas that have the same color mean but different histograms with two non zero values, and we add a Gaussian noise to the image (b). We first estimate the reference histograms on the clean image. The objective is here to see if it is possible to retrieve the objects in the noisy image, knowing the histograms on the clean image. We show the results obtained with the data terms based on (c) histograms and (d) cumulative histograms. As expected, the simple histogram comparison fails, whereas the cumulative version of the model is able to give good segmentations of the objects.



(a) Initial image (b) Noisy image (c) Histograms (d) Cumulative histograms

Fig. 1. Segmentation of a noisy image with two zones that have the same color mean but different histograms, with $\beta = 0.5$, $\lambda = 0.07$ and 100 bins. The cumulative histograms version produces a good estimation, whereas the simple histogram comparison fails. The results are here not thresholded in order to show that the failure of the histogram model is not due to the choice of the threshold.

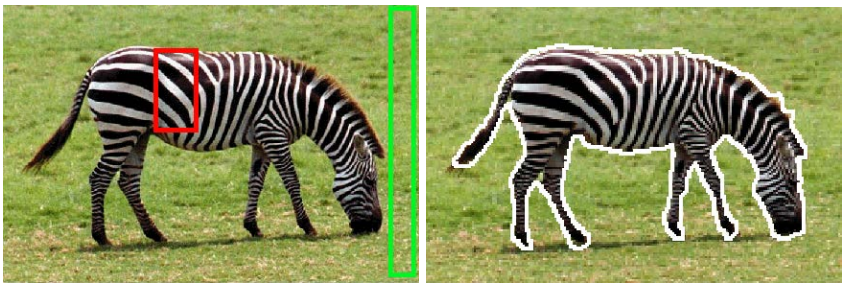
We will then use the cumulative histogram version of our data term, since it is well adapted to gray-valued images and it is more robust to noise. Notice that the l^1 data term between cumulative histograms of grayscale images corresponds to the l^1 Wasserstein distance. A segmentation of a natural image is presented in Figure 2. The animal is accurately segmented, even if the tail is missed.

(a) Initial image (b) Segmentation

Fig. 2. Segmentation of a natural image, with $\beta = 0.5$, $\lambda = 0.15$ and 10 bins. The animal is mainly found.

3.2 Color Images

We show in Figure 3 the segmentation of a zebra, using color histograms. Here again the global model produces accurate results. Next we present in Figure 4 the segmentation of a boat. In this example, the size of the two wanted partitions is unbalanced, as the boat is small compared to the rest of the image. A slight increase of the β parameter then allows one to obtain a more accurate estimation, as illustrated in the image (c).



(a) Initial image (b) Segmentation

Fig. 3. Segmentation of a natural image, with $\beta = 0.5$, $\lambda = 0.2$ and 7^3 bins. The animal is mainly found.

(a) Initial image (b) Segmentation with $\beta = 0.5$ (c) Segmentation with $\beta = 0.59$

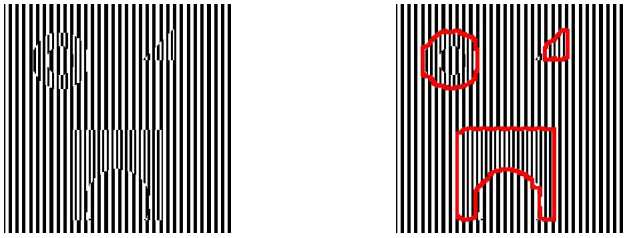
Fig. 4. Segmentation of a boat, with $\beta = 0.5$, $\lambda = 0.13$ and 5^3 bins. The boat is found, but choosing $\beta = 0.59$ gives a more accurate segmentation (c) than the one obtained with the default parameter (b).

3.3 Structure Tensor

The proposed model is also designed to deal with texture, using for instance the structure tensor [6] defined as follows. If I is a grayscale image, the structure tensor is the symmetric matrix $(\nabla I)(\nabla I)^t$. Since the tensor is symmetric, it has only 3 independent dimensions. A $3D$ histogram is therefore sufficient to represent the distribution of the structure tensor values.

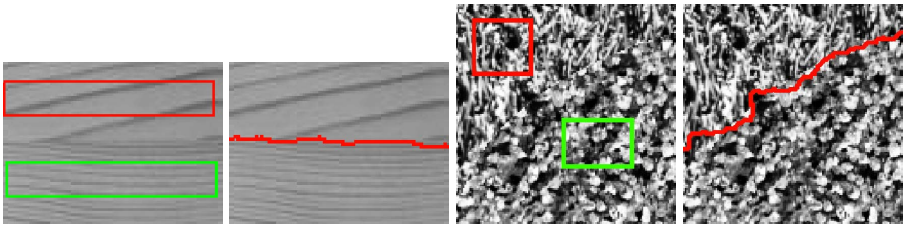
We can therefore apply the algorithm on the three channels of the structure tensor or enhance the model with a 4 dimensional histogram (3 dimensions for

the structure tensor and 1 for the grayscale image I). Here we present some examples just involving the components of the structure tensor. We choose to linearly normalize all channels between 0 and 1 before taking the histograms. We give an example on a synthetic image with two regions that have the same histograms of gray levels, and with the same orientations. The results are given in Figure 5. The model here again performs well, even if the ring is recovered as a disc. We also present a segmentation of two images with two textures in Figures 6 where our global model is able to estimate accurate segmentations of the different textures.



(a) Initial image (b) Segmentation

Fig. 5. Segmentation of a synthetic image, with $\beta = 0.5$, $\lambda = 0.025$ and 3^3 bins



(a) Initial image (b) Segmentation (c) Initial image (d) Segmentation

Fig. 6. Segmentation of an image of wood, with $\beta = 0.5$, $\lambda = 0.075$ and 7^3 bins, and segmentation of an image composed of two textures, with $\beta = 0.5$, $\lambda = 0.1$ and 7^3 bins.

4 Comparison with Local and Non Convex Approaches

In this section, we present comparisons with algorithms of the two main class of approaches that can be found in the literature of the histogram-based segmentation problem. We first compare our approach with an approach based on local histogram comparisons through the convex Potts model. As will be shown in the numerical computations, such a method cannot compete with the global framework introduced in this paper. We also present some comparisons with a recent non-convex approach based on the Wasserstein distance between global histograms. The different experiments demonstrate the capacity of our method to deal with global constraint while being independent of the initialization.

Comparisons with a Local Histogram-Based Approach

In order to illustrate the importance of global histogram comparisons over local ones, we now introduce a local histogram segmentation algorithm based on the Potts model. A pointwise estimation of the probability of a pixel to belong to a class can be formulated for instance with the functional:

$$J_{pointwise}(u) = TV(u) + \langle u(x), h_0(I(x)) - h_1(I(x)) \rangle_\Omega \,, \tag{14}$$

where the data model will enforce $u(x) = 1$ when $h_1(I(x)) > h_0(I(x))$. In the aim of having a more robust data term, we can realize a local estimation of the probability of a pixel to belong to a class. This can be formulated in the same way with the functional:

$$J_{local}(u) = TV(u) + \left\langle u(x), \| \, h_0 - h_{V(x)} \, \|_{l^1} - \| \, h_1 - h_{V(x)} \, \|_{l^1} \right\rangle_\Omega \,, \tag{15}$$

where $h_{V(x)}$ is the histogram estimated on a neighborhood $V(x)$ of $x \in \Omega$. Notice that in these two functionals, the data term can be computed pointwise (once the map of distances with local histograms has been calculated for the local model). This is a main difference with our data term which values depend on the whole state u. On the other hand, it is important to underline that such functionals can be globally minimized on $u \in BV(\Omega, \{0; 1\})$, using convexification approaches (see[18,23,8,5,29]) for more details). We illustrate the need to use global histogram comparisons instead of local ones on the two examples given in Figures 7 and 8.

(a) Image data (b) Groundtruth (c) Pointwise (d) Local (e) Global

Fig. 7. Segmentation of a binary image with three regions. The respective probability of 0 are 0.65, 0.5, 0.95. We fix two histograms $h_0 = [0.8, 0.2]$ and $h_1 = [0.55, 0.45]$. The local estimation on neighborhoods of size 9×9 with $\lambda = 0.5$, gives a classification where the estimated histograms on the labeled images are not as close as the segmentation given in (c) with our model (with parameters $\beta = 0.5, \lambda = 0.5$).

The synthetic data presented in Figure 7 (a) is composed of three different regions of the same area. Each region is filled with independent realizations of a binomial distribution with the probability of 0 depending on the region: this probability is equal to 0.65 for the left region, 0.55 for the middle one and 0.95 for the right region. We fix two histograms, $h_0 = [0.8, 0.2]$ (i.e. the frequency of 0 is 0.8 and the frequency of 1 is 0.2) and $h_1 = [0.55, 0.45]$. Hence, the first histogram (h_0) exactly matches the expected histogram of the union of the left

and the right regions, while the second one (h_1) matches the one of the middle region. The ground truth segmentation is illustrated in image (b) of Figure 7.

When realizing a pointwise estimation of the data term, the pixels with value 0 are always more probable in the class defined by h_0. With no regularization the black pixels will therefore be classified in the h_0 class, and the white pixels in the h_1 class. Since in each model the probability of 0 is greater than the one of 1, increasing the regularization weight will increase the number of pixels labeled in h_0. This kind of classification is obtained with the functional (14) and illustrated in Figure 7 (c).

If we consider local histograms through the functional (15), the problem will be partially solved. Indeed, if the neighborhood used to compute the local histograms is sufficiently large, the data estimation will be good enough and the central region will be classified in h_1. However, if the metric used to compare histograms is the l^1 one, then the local histograms computed in the first region will be closer to h_1 and will thus be misclassified, as illustrated in image (d).

With both local models, the global histograms of the segmented regions are not close to the given reference histograms. This is the main difference with our modeling, as we aim at globally segmenting the image, in the sense that the histograms of the global segmented zones should be close to the reference distributions. We show in figure 7 (e) the aforementioned example, its segmentation with local histograms and the one with our model.

A second comparison is presented in Figure 8. The reference color distributions are here not homogeneous in the sense that the histogram of the whole region is not similar to the ones computed in local neighborhoods. The histograms are estimated in the regions given in the image (a). Here the orange colors are more probable in the region related to the butterfly, so in small neighborhoods the flowers are classified as the butterfly, and the darker regions are segmented as being in the background. This example illustrates again the importance of global histogram comparisons to get a global segmentation of an image. Indeed, our global model (c) is able to recover the butterfly , whereas the local approach (b) completely fails. Local approaches are therefore only relevant when the local histograms correctly approximate the global ones.

Comparisons with a Shape Gradient Approach

We finally illustrate the advantage of having a convex model that does not depend on the initialization. We compare our results with the ones obtained with the Wasserstein Active Contour method proposed in [21][1]. Such approach consists in deforming a level set function in order to minimize globally the Wasserstein distance between the reference histograms and the one induced by the segmentation. To make the level set evolve, this formulation requires complex shape gradients computations. In Figure 9, we present the results obtained with this approach on the synthetic three regions example (build as in Figure 7).

[1] We want to thank the authors of [21], Gabriel Peyré, Jalal Fadili and Julien Rabin, for their code.

(a) Original image (b) Local model (c) Global model

Fig. 8. Segmentation of an image where local histograms (9×9 neighborhoods) are different from the global ones. The segmentation fails for the local histogram model (with $\lambda = 10$) as it classifies the orange areas in the first class and the darker ones in the second class. The global histograms on the segmented zones are not close to the given ones, contrary to our model parameterized with $\beta = 0.5$ and $\lambda = 0.22$. The histograms were composed of 4^3 bins.

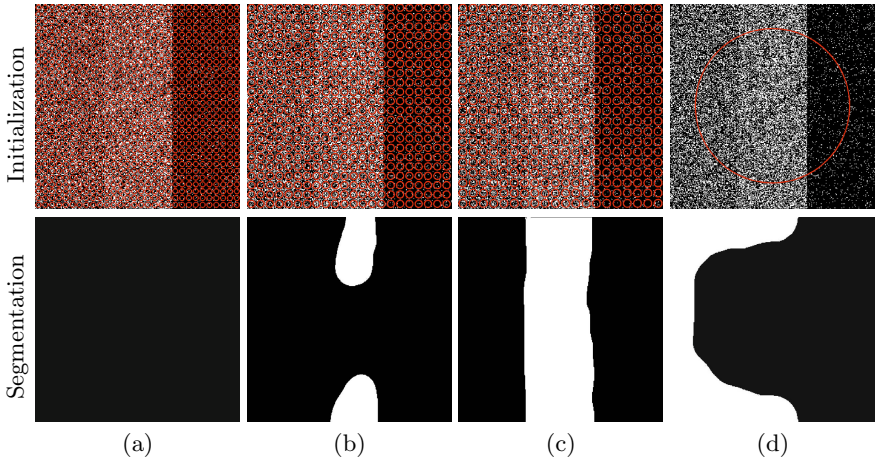
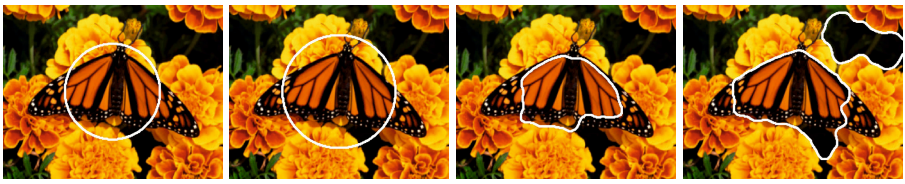


Fig. 9. We illustrate on the three regions synthetic example the problems arising with an algorithm that is very sensitive to the chosen initialization. The Wasserstein active contours method [21] have been initialized in different ways (first row, a-d). As illustrated in the second row, it leads to very different segmentations, even with really similar initializations (a-c). When carefully parameterized, such approach can lead to good segmentations as illustrated in column (c), which is very close to the one obtained with our global approach (see Figure 7 (c)).

An experiment on a natural image is finally shown in Figure 10. Even if this model can give good segmentations that are close to the ones we obtained in Figures 7 (c) and 8 (c), we illustrate that its initialization may be a critical step as really different segmentations are obtained with very similar initializations.



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

Fig. 10. The Wasserstein active contours method [21] have been initialized in two different ways (a-b), the corresponding segmentations being presented in (c-d). When carefully parameterized, it leads to a segmentation close to the one obtained with our global approach (see Figure 8 (c)).

5 Conclusions and Perspectives

In this paper, we have presented a general convex framework based on global histograms to segment images into two parts, and we experimented a special case defined by the l^1 data term. Our approach gives good results in the set of experiments realized on gray images, color images and also when using the structure tensor. The additional use of the cumulative histograms also permits to be robust to noise. Comparisons with local histogram data terms and non-convex approaches also demonstrate the capacity of our model to deal with complex configurations.

Even if the convex reformulation changes the nature of the problem by normalizing the data terms with respect to their prior area, the value of the area ratio parameter β is not an issue. Namely, it has been set to 0.5 in almost all of our experiments, which shows that our model is robust in practice.

Future works will be to study the use of other descriptors for texture. We will also be interested in investigating the connections between minimizers of the original problem and the ones of the relaxed problem for different norms in order to make the best possible choice when finally thresholding the estimated solution.

References

1. Aubert, G., Barlaud, M., Faugeras, O., Jehan-Besson, S.: Image segmentation using active contours: Calculus of variations or shape gradients? SIAM Applied Mathematics 63, 2003 (2002)
2. Aubert, G., Kornprobst, P.: Mathematical Problems in Image Processing. Applied Mathematical Sciences, vol. 147. Springer (2002)
3. Aujol, J.-F., Aubert, G., Blanc-Féraud, L.: Wavelet-based level set evolution for classification of textured images. IEEE Transactions on Image Processing 12, 1634–1641 (2003)
4. Ayed, I., Chen, H., Punithakumar, K., Ross, I., Li, S.: Graph cut segmentation with a global constraint: Recovering region distribution via a bound of the bhattacharyya measure. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3288–3295. IEEE (2010)

5. Brown, E., Chan, T.F., Bresson, X.: Completely Convex Formulation of the Chan-Vese Image Segmentation Model. In: IJCV, pp. 1–19 (2011)
6. Brox, T., Rousson, M., Deriche, R., Weickert, J.: Unsupervised segmentation incorporating colour, texture, and motion. In: Petkov, N., Westenberg, M.A. (eds.) CAIP 2003. LNCS, vol. 2756, pp. 353–360. Springer, Heidelberg (2003)
7. Brox, T., Weickert, J.: Level set segmentation with multiple regions. IEEE Transactions on Image Processing 15(10), 3213–3218 (2006)
8. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. JMIV 40, 120–145 (2011)
9. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Transactions on Image Processing 10(2), 266–277 (2001)
10. Collins, M.D., Xu, J., Grady, L., Singh, V.: Random walks based multi-image segmentation: Quasiconvexity results and gpu-based solutions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012), pp. 1656–1663. IEEE (2012)
11. Cremers, D., Rousson, M., Deriche, R.: A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape. International Journal of Computer Vision 72, 215 (2007)
12. Delfour, M.C., Zolésio, J.-P.: Shapes and geometries: analysis, differential calculus, and optimization. Society for Industrial and Applied Mathematics, Philadelphia (2001)
13. Gorelick, L., Schmidt, F.R., Boykov, Y., Delong, A., Ward, A.: Segmentation with non-linear regional constraints via line-search cuts. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part I. LNCS, vol. ECCV 2012, pp. 583–597. Springer, Heidelberg (2012)
14. Herbulot, A., Jehan-Besson, S., Duffner, S., Barlaud, M., Aubert, G.: Segmentation of vectorial image features using shape gradients and information measures. Journal of Mathematical Imaging and Vision 25(3), 365–386 (2006)
15. Kim, J., Fisher, J.W., Yezzi, A., Cetin, M., Willsky, A.S.: A nonparametric statistical method for image segmentation using information theory and curve evolution. IEEE Transactions on Image Processing 14, 1486–1502 (2005)
16. Mumford, D., Shah, J.: Optimal approximation by piecewise smooth functions and associated variational problems. Comm. Pure Appl. Math. 42, 577–685 (1989)
17. Ni, K., Bresson, X., Chan, T.F., Esedoglu, S.: Local histogram based segmentation using the wasserstein distance. International Journal of Computer Vision 84(1), 97–111 (2009)
18. Nikolova, M., Esedoglu, S., Chan, T.F.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM Journal on Applied Mathematics 66(5), 1632–1648 (2006)
19. Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. JCP 79(1), 12–49 (1988)
20. Paragios, N., Deriche, R.: Geodesic active regions for supervised texture segmentation. In: ICCV (1999)
21. Peyré, G., Fadili, J., Rabin, J.: Wasserstein active contours. In: IEEE International Conference on Image Processing (ICIP 2012) (2012)
22. Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In: IEEE International Conference on Computer Vision (ICCV 2011), pp. 1762–1769 (2011)
23. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: Global solutions of variational models with convex regularization. SIAM JIS 3, 1122–1145 (2010)

24. Punithakumar, K., Yuan, J., Ben Ayed, I., Li, S., Boykov, Y.: A convex max-flow approach to distribution-based figure-ground separation. SIAM Journal on Imaging Sciences 5(4), 1333–1354 (2012)
25. Rother, C., Minka, T., Blake, A., Kolmogorov, V.: Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2006), vol. 1, pp. 993–1000. IEEE (2006)
26. Rousson, M., Brox, T., Deriche, R.: Active unsupervised texture segmentation on a diffusion based feature space. In: CVPR (2003)
27. Vese, L.A., Chan, T.F.: A multiphase level set framework for image segmentation using the mumford and shah model. International Journal of Computer Vision 50, 271–293 (2002)
28. Vicente, S., Kolmogorov, V., Rother, C.: Joint optimization of segmentation and appearance models. In: IEEE International Conference on Computer Vision (ICCV 2009), pp. 755–762. IEEE (2009)
29. Yildizoglu, R., Aujol, J.-F., Papadakis, N.: Active contours without level sets. In: IEEE International Conference on Image Processing (ICIP 2012) (2012)
30. Yuan, Y., Ukwatta, E., Tai, X.C., Fenster, A., Schnörr, C.: A fast global optimization-based approach to evolving contours with generic shape prior. In: submission in IEEE TPAMI, also UCLA Tech. Report CAM 12-38 (2012)
31. Zhu, S.C., Lee, T.S., Yuille, A.L.: Region competition: unifying snakes, region growing, energy/bayes/mdl for multi-band image segmentation. In: IEEE International Conference on Computer Vision (ICCV 1995), pp. 416 –423 (June 1995)

A Continuous Shape Prior for MRF-Based Segmentation

Dmitrij Schlesinger

Dresden University of Technology

Abstract. The competition between discrete (MRF based) and continuous (PDE based) formulations has a very long history, especially in context of segmentation. Obviously, both have their advantages and drawbacks. Therefore the choice of a discrete or continuous framework is often driven by a particular application or (even more often) by personal preferences of a particular researcher. In this work we present a model for binary segmentation, where discrete and continuous parts are combined in a well founded and simple way. We discuss the properties of the proposed model, give a basic inference algorithm and validate it on a benchmark database.

1 Introduction

The discussion about properties, advantages and drawbacks of discrete and continuous approaches is not new (see e.g. [1]). In most cases however the questions are posed in a quite ultimate way: "which framework is better" with respect to a particular property (precision, modeling capabilities, computational efficiency etc.). In this work we try to combine the advantages of continuous and discrete approaches for segmentation. To start with, we would like to discuss/recall some relevant properties. The first issue is the modeling. In a continuous framework it is much easier to express a-priory assumptions that relate to low-level features, like e.g. boundary length, curvature etc. The main drawback of discrete methods in this respect is the presence of well known metrication artifacts. Of course, it is possible to avoid these effects (to some extent) in the discrete framework too [2,3]. However, it leads to higher-order MRF-s or to a complicate graph structure, that makes the modeling less transparent and the related tasks hard to optimize. On the other hand, some semi-global properties (such as e.g. scene layout [4] or star-convexity [5]) is more convenient to express in the discrete framework. Another important related property here is the ability to deal with multi-label segmentation. Again, this is possible in continuous approaches as well [6]. However, it leads to the complex coupling constraints that is again hard to optimize.

An important topic is the ability to model shapes that is nowadays almost obligatory for segmentation (especially in the unsupervised case). Many different shape priors were studied in the continuous optimization community in the past. The most elaborated (in our opinion) techniques are based on the Level Set representation (see e.g. [7] and references therein). In the discrete domain this topic is not as elaborated so far. In [8] it is shown that it is in principle possible in part-shape based segmentation. However, again the resulting model (although of second order only) has a very complex neighborhood structure and a lot of free parameters, that restricts the applicability.

A. Heyden et al. (Eds.): EMMCVPR 2013, LNCS 8081, pp. 350–361, 2013.
© Springer-Verlag Berlin Heidelberg 2013

Another branch of questions deals with inference. The continuous functionals used for inference are often convex and can be optimized globally. Even if not, convex approximations/relaxations can be used. In the discrete case most of inference tasks are NP-complete. In particular, if the Maximum A-posteriori decision is employed as the decision strategy, relaxations are often used for approximations, thus in fact, discrete optimizations are substituted by corresponding continuous ones.

The main drawback of continuous methods is the lack of sound statistic interpretation, which makes it impossible to learn unknown parameters. For discrete models in contrast, the learning can be posed in a well founded statistic way using e.g. the Maximum Likelihood principle or applying the ideas of discriminative learning, such as e.g. Structural SVM [9].

The main aim of the paper is to show that the advantages of discrete and continuous models can be combined in a well founded and simple way. This work is inspired by [10,11] at most. In these works however, a very simple shape prior is used, which is obviously not suitable for real segmentation tasks. We use a more general class of Level Set like functions to represent shapes. In short, the proposed model consists of two parts – a discrete and a continuous one. The former is a standard MRF consisting of the Ising model for the a-priori probability distribution for segmentations and a Gaussian Mixture Model for appearances (similar to e.g. [12,13]). Like in [10], the prior probability distribution is parametrized by a Level Set like function, which assigns additional unary potentials to each pixel. The proposed model has the following properties that are desired for segmentation:

- It is very simple and generic and has few free parameters.
- The modeling of the low-level features is expressed in a transparent way using continuous framework.
- The segmentation is posed in a sound statistic way using discrete formulation.
- Basic algorithms for inference and learning are based on standard techniques widely used in discrete and continuous optimization respectively.
- The method is able to work in fully unsupervised manner and gives promising results.

2 Approach

2.1 Model

In this work we consider binary segmentation for simplicity although all consideration can be easily generalized to the multi-label case. The model architecture is illustrated in Fig. 1. Let $G = (R, E)$ be a graph over the pixel grid, i.e. each node $r \in R$ corresponds to a pixel, the set of edges E corresponds to the 4-neighborhood structure. At the same time the nodes are embedded in a continuous space $\Omega \subset \mathbb{R}^2$, i.e. each pixel has its coordinates, which we denote by $r \in \Omega$ as well to omit notational clutter. The segmentation is a mapping $y : R \to \{0, 1\}$ that assigns a label (0 for the background

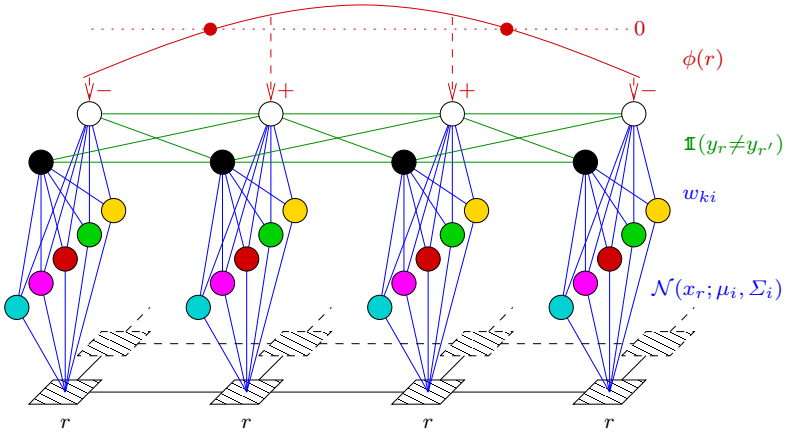


Fig. 1. Model overview (best viewed in color). The shape function $\phi(r)$ (red) assigns additional unary potentials to the foreground labels (white) at each position of the underlying MRF (green). The conditional probability distributions $p(x_r|y_r)$ for observations are Gaussian Mixtures (blue) with a common set of Gaussians (colored circles).

and 1 for the foreground) to each node r. An image is also a mapping $x : R \to C$ that assigns a color value $c \in C$ to each pixel. We denote by y_r and x_r the label and the color value in pixel r correspondingly. The probability distribution for pairs (x, y) is defined by[1]

$$p(x, y; \phi) = p(y; \phi) \cdot p(x|y) = \\ = \frac{1}{Z(\phi)} \exp\Big[-\alpha \sum_{rr' \in E} \mathbb{1}(y_r \neq y_{r'}) + \sum_r y_r \cdot \phi(r)\Big] \cdot \prod_{r \in R} p(x_r|y_r). \qquad (1)$$

The prior probability distribution consists of two parts (energy terms). The first one is the Ising model that penalizes different labels in neighboring pixels by a penalty $\alpha > 0$. The second one is a shape function $\phi : \Omega \to \mathbb{R}$, which supports/suppresses the foreground label in each pixel by assignment an additional unary energy term to it.

As usual for MRF-s, the conditional probability distribution $p(x|y)$ is assumed to be conditionally independent. We use Gaussian Mixture models with common Gaussians for $p(x_r|y_r)$ like in [11]. In contrast however, we use general multivariate Gaussians instead of the isotropic ones as we are not interested in real-time performance. To summarize, the probability to observe a color $c \in C$ for a label $k \in \{0, 1\}$ is

$$p(c|k) = \sum_i w_{ki} \cdot \mathcal{N}(c; \mu_i, \Sigma_i) \qquad (2)$$

with the label-specific weights w_{ki} and Gaussians $\mathcal{N}(\cdot)$ with mean values μ_i and covariance matrices Σ_i.

[1] The parameters are separated from the random variables by semicolon.

2.2 Inference and Learning

Assuming for a moment that the shape function is known, the segmentation is posed as a Bayesian decision task with the Hamming distance as the loss. It leads to the maximum marginal decision strategy

$$y_r^* = \arg\max_{k\in\{0,1\}} p(y_r{=}k|x;\phi) \quad \forall r \in R. \tag{3}$$

Although most successful segmentation methods are based on the maximum a-posteriori decisions, marginal based inference is becoming increasingly popular for MRF-based approaches due to several reasons (see e.g. [14]). First of all, it follows from a more reasonable loss function. Besides, as we will see later, we need marginals for other tasks as well, namely for the shape estimation and unsupervised learning of the appearance characteristics.

The most interesting part is the estimation of the shape function ϕ. We consider it primarily as a continuous function in order to be able to use elaborated techniques from continuous optimization. Consequently, we *do not* consider ϕ as a random variable, mainly because it is not possible to introduce probability measures for function spaces. Therefore, in our model the shape function is a parameter of the probability distribution. Remember that we are mainly interested in fully unsupervised segmentation. Taking all this into account, a reasonable choice is to estimate the shape function e.g. according to the Maximum Likelihood principle, i.e. $\ln p(x;\phi) \to \max_\phi$. However, it is easy to see that such a formulation for (1) has a trivial solution. For those pixels that would be assigned to the foreground according to the appearance model only (i.e. $p(x_r|1) > p(x_r|0)$) the optimum with respect to ϕ is reached at $\phi(r) = \infty$, for other pixels $\phi(r) = -\infty$ holds. It is indeed expected because we did not introduce any requirements (prior assumptions etc.) for shape so far.

It is a common technique in Machine Learning to enhance an objective function (e.g. the Likelihood in our case) by a *regularizer*. In Machine Learning this trick is used mainly in order to resolve ambiguities and/or increase robustness of the learning. Here we use the same ideas in order to be able to express our prior assumptions about the shape. Hence, we pose the shape estimation as the following optimization task:

$$F(\phi) = \ln\Big[\sum_y p(x,y;\phi)\Big] + \mathcal{R}(\phi) \to \max_\phi, \tag{4}$$

where the first addend is the log-likelihood and the second one is a regularizer. Numerous choices are possible for the latter. We use a simple one to keep the whole approach as general as possible:

$$\mathcal{R}(\phi) = -\int_\Omega \big(\lambda_1 \cdot \|\nabla\phi\|^2 + \lambda_2 \cdot \triangle\phi^2\big)d\omega, \tag{5}$$

First of all we would like to require that the shape function is smooth. At the same time penalizing gradients only is not appropriate in our context because in this case the shape function tends to be "flat" inside the segments. As the consequence, it does not influence the underlying MRF good enough. Therefore we penalize both gradients and Laplacians to facilitate the shape function to be "as linear as possible".

The derivative of the subject (4) with respect to the value of the shape function at a given position r yields[2]

$$\frac{\partial F}{\partial \phi(r)} = p(y_r{=}1|x;\phi) - p(y_r{=}1;\phi) + \frac{\partial \mathcal{R}(\phi)}{\partial \phi(r)}. \tag{6}$$

The first term is the posterior marginal probability for the pixel r to be the foreground. These probabilities are estimated using Gibbs Sampling. For the second one (prior marginal probabilities) we use the following approximation. In [11] it is proposed just to binarize the shape function at zero level and substitute real prior marginals by $p(y_r{=}1;\phi) \approx \mathbb{1}(\phi(r) > 0)$. In our case it does not work well, because the shape is much "weaker" as in [11] and non-parametric. However, similar observations can be used. We performed a couple of experiments and found out that the prior marginals can be well approximated (for reasonable values of α, λ_1 and λ_2) by a sigmoid function

$$p(y_r{=}1;\phi) \approx \frac{\exp\big(\beta\phi(r)\big)}{\exp\big(\beta\phi(r)\big)+1}, \tag{7}$$

with a constant $\beta > 1$, which makes the sigmoid function stronger compared to the independent case $\alpha = 0$. The last term in (6) is the Gâteaux derivative of (5) and is obtained (after discretization) by convolution with the corresponding mask.

For unsupervised learning of the Gaussian mixtures (2) we follow standard techniques and use the Expectation-Maximization Algorithm. Note that in the Expectation step again the marginal posterior label probabilities should be computed.

2.3 Discussion

At this place we would like to discuss some properties of our model and relations to other ones. The discrete part is a standard MRF. Without the shape prior it is even simpler as e.g. GrabCut [13] due to the constant edge strengths α. Obviously, the discrete part can be extended to admit more elaborated features. At this stage however we do not follow this way because we would like to keep the model as simple as possible and therefore as generic as possible. In particular, we would like to show that even such simple MRF is able to produce reasonable results if it is "weakly supported" by other parts (e.g. by a continuous shape prior in our case). Nonetheless, we are mainly interested in an unsupervised segmentation that requires generative models.

If we simplify our model in another way doing the discrete part independent (i.e. $\alpha = 0$), we observe the following analogy. As the log-likelihood in (4) can be computed explicitly, the model becomes a continuous one, where the data terms are obtained by marginalization over labels in each pixel. The optimization (4) reads then

$$\sum_r \ln\left[\frac{\exp(\phi(r))}{\exp(\phi(r))+1}\cdot p(x_r|1) + \frac{1}{\exp(\phi(r))+1}\cdot p(x_r|0)\right] + \mathcal{R}(\phi) \to \max_\phi. \tag{8}$$

Suppose, the shape function is very "strong" (which is mainly the case in continuous optimization, as Level Set functions are supposed to be close to the distance transform),

[2] We omit detailed derivation here, because they are standard and quite straightforward.

i.e. the sigmoid function in (8) is very close to the Heaviside function $H(\phi(r)) = \mathbb{1}(\phi(r)>0)$. In other words, we substitute the marginalization by binarization. Then the model becomes a standard Level Set based continuous model for segmentation

$$\sum_r \Big[H(\phi(r)) \cdot \ln p(x_r|1) + \big(1 - H(\phi(r))\big) \cdot \ln p(x_r|0) \Big] + \mathcal{R}(\phi) \to \max_{\phi}. \qquad (9)$$

We would like to stress that the optimization (4)-(6) is in fact a standard task extensively studied in the continuous optimization in the past. The difference is only that the data terms are not given explicitly but their derivatives (first two summands in (6)). Therefore the whole spectrum of methods (like e.g. successive over-relaxation, scale-space methods etc.) can be used for efficient optimization. Moreover, elaborated regularizers can be of course used instead of the simple one (5), such as e.g. TV-norm.

2.4 Implementation Details

We implemented the method in a multi-threaded manner like in [11][3]. The overall system architecture is given in Fig. 2. Basically, it consists of three blocks that work in parallel. The core of the algorithm is the "Gibbs Sampling" block that permanently computes marginal posterior label probabilities for the current appearances and shape function. The "Appearances" block observes these probabilities, performs maximization steps of the EM-Algorithm and computes the unary potentials $\ln p(x_r|k)$ according to (2). The "Shape" block also takes the marginal posterior probabilities as the input and performs gradient steps according to (6). The output of this block is the shape function $\phi(r)$ (i.e. the shape dependent unary potentials in (1)).

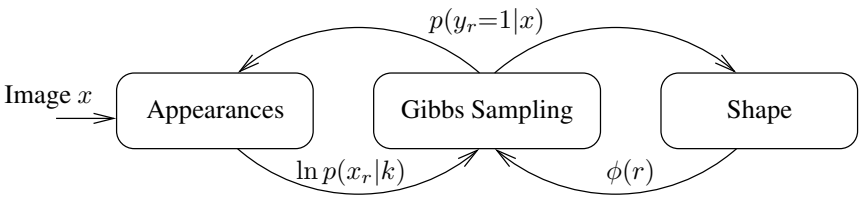


Fig. 2. The system architecture

We do not use any stopping criteria. In our experiments we just limit the time for the system to work. After the time is elapsed, the final segmentation is obtained by thresholding the current marginal label probabilities at 0.5 level.

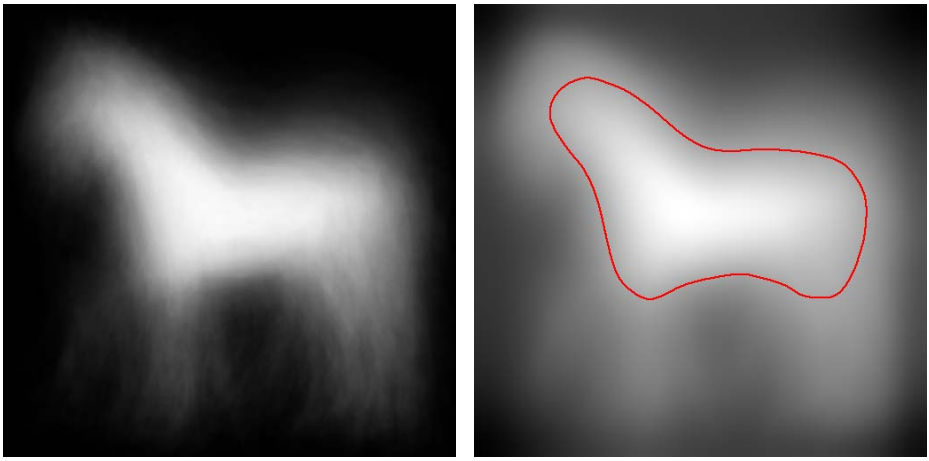
The most slow procedure is the learning of appearances. We did not use acceleration techniques as in [11] in order to avoid approximations. Hence, for each pixel the marginalization over the Gaussians is necessary that involves exponentiation and taking the logarithm many times. The fastest procedure is Gibbs Sampling that can be easily implemented in a very efficient manner. The speed of the shape estimation lies in between. To summarize, for a typical run of 5 minutes the system was able to perform

[3] The source code will be available soon at [23].

about 33 thousands iterations (scans over the whole image) of Gibbs Sampling, about 7 thousands gradient steps for the shape learning and only about 250 iterations (M-steps) for appearances. Obviously the computational efficiency can be essentially improved. We plan to address this issue in the nearest future.

3 Experiments

We validate our model on the Weizmann Horse Database [15]. First of all we would like to note that our simple generic shape prior does not fit "horses" good enough. The obvious problems are legs, pigtails and manes. The next topic is the following. As we perform fully unsurprised segmentation, the problem is highly ill-posed, i.e. there can be many objects in the scene (despite of horses) that are "compact" and fit well to our shape prior. At the same time the optimization in (4)-(6) is non-convex due to the likelihood. Therefore we need a reasonable initialization for the shape function. We obtained it by just considering the position dependent probability distributions of labels, estimated from ground truths over the whole database. For each pixel we count how many times it is labeled as foreground according to the ground truth (to do this, images were rescaled to a predefined size). The probability map obtained in such a manner is given in Fig. 3(a). After that we fit the shape function into these probabilities, i.e. we perform optimization (4)-(6), where the posterior marginals are substituted by the probability map obtained as described above. The resulting initial shape function is shown in Fig. 3(b). The initialization for each image is obtained by rescaling this initial shape function to the original image size.



(a) Probability map. **(b)** Shape function (zero-level is shown in red).

Fig. 3. Initialization for the shape function (scaled to fit into the gray-value range)

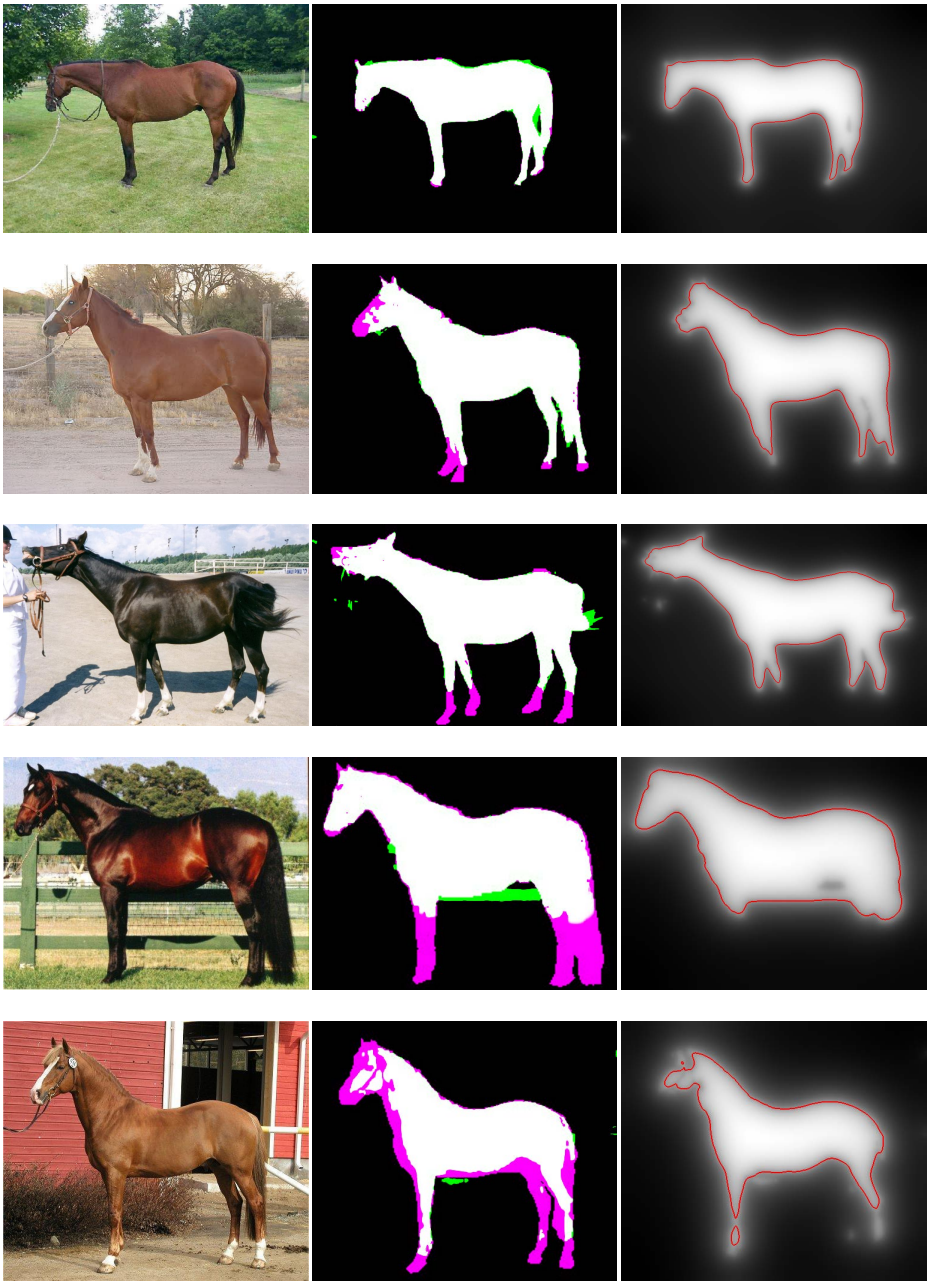


Fig. 4. Some selected results. Left: original images; middle: obtained segmentation overlaid with ground truth (green – false positive, magenta – false negative), right: obtained shape function, zero level is shown in red. Best viewed in color.

In Fig. 4 some selected results are presented. If the coloring properties are distinctive enough, the produced segmentation is almost perfect (the first row, less that 1% misclassified pixels). There are also many "good" segmentations (like in the second and the third rows, about 2-3% misclassified pixels), where the errors have a local nature. Note, that isolated segmentation errors (see the third row) are often not presented in the zero-level set of the found shape function. Obviously, if a particular task consists of the retrieval of compact connected segments that need not necessarily be precise, the zero-level can serve as the segmentation result.

In most cases the method found segmentations that we would consider rather as "satisfactory" ones (fourth and fifth rows) – characteristic horse shapes are clearly visible at most, the obtained segmentations are however not precise enough (about 8-9% misclassification). Finally, there are cases, for which the method does not work at all (see Fig. 5), i.e. it produces segmentations that have nothing in common with what is desired.

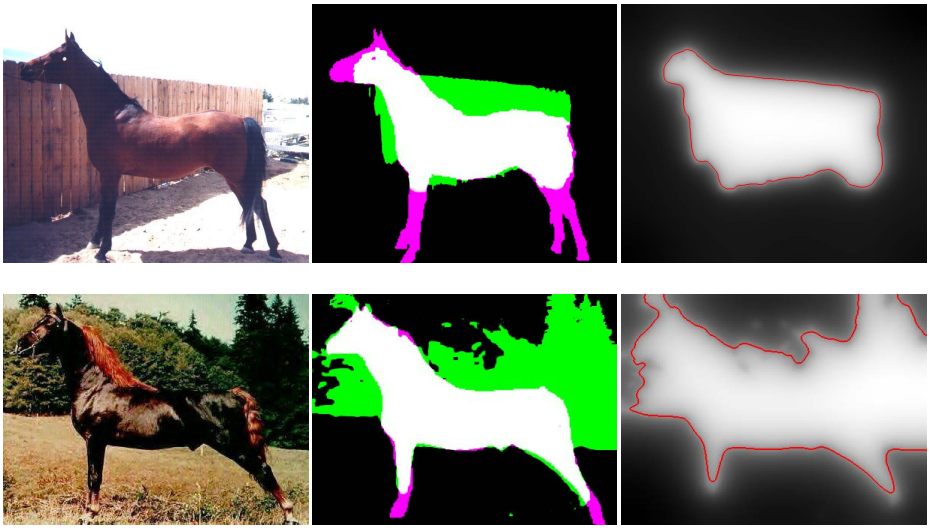


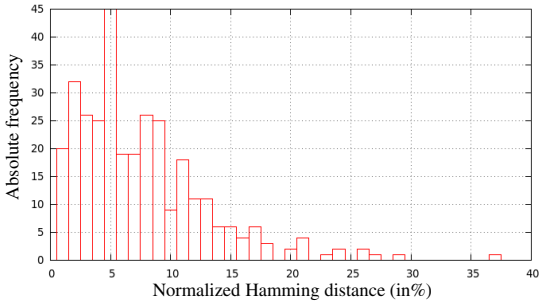
Fig. 5. Fail cases (see description for Fig. 4)

The quantitative results are presented in Fig. 6(a). For each image we compute the Hamming distance normalized to the image size. We prefer to give a complete histogram of the Hamming distances to give a better feeling, how many images in the database were segmented correctly. As it is seen from the figure there were only 12 images (out of 328 in total), where more than 20% of pixels were misclassified. 237 images were segmented with less than 10% of inaccuracy. The most frequent error value (for 45 images) is 5%. The average normalized Hamming distance is 7.8%.

Unfortunately, it is not easy to compare the proposed model with other ones in a straightforward manner. The obvious reason is that there are many algorithms in the literature that are very complex, too different and therefore not comparable. Elaborated segmentation techniques usually use high-level knowledge about the object to be segmented. For example in [16,17,18] template based techniques are used, where the

templates are learned in advance and match characteristic shape fragments during inference. In [19] a complex hierarchic model is presented. In [20] a kernelized structural SVM learning framework is employed. We would like also to mention [14], where relatively simple generic features (although a lot of them) are used in a CRF framework and marginal based inference and learning are exploited. On this background our model looks like an extreme oversimplification. On the other hand, really simple generic algorithms seem to be not able to cope with real segmentation tasks. So for example GrabCut initialized in a reasonable manner has only 85.5% accuracy[4] (data from [20]). To conclude, our 92.2% accuracy were really surprising. We summarize the results found in the literature in Fig. 6(b).

We should admit that at the moment the used optimization techniques (as well as our implementation) are very far from being computationally efficient. Processing of one image takes about 5 minutes on a standard quad-core computer thus the processing of the whole database takes about a day. As a consequence we were not able to tune free parameters of the method carefully. We performed a couple of experiments for only 3-4 images taken randomly from the database and decided for values that seemed reasonable. We also would like to note that for many images 5 minutes were obviously not enough, i.e. the method had not enough time to converge. Consequently, we hope that the segmentation accuracy can be further improved by using efficient implementation.



(a) Histogram of the normalized Hamming distances.

Approach	Accuracy
ObjCut [16]	96.0
Levin [17]	95.5
Borenstein [18]	93.6
Zhang [19]	95.4
Bertelli [20]	94.3
Domke [14]	92.0
Our	92.2
GrabCut [13]	85.5
Co-segmentation [21]	80.1
MNcut [22]	51.0

(b) Comparison to other approaches.

Fig. 6. Quantitative evaluation

Looking at the initialization (see Fig. 3) it might be well asked, whether our results are caused by the proposed model or by a suitable chosen initialization. In order to verify that the model really does its job, we performed the following check. We binarized the initial shape function at the zero-level and used this binary mask as a segmentation result. For each image we compute the normalized Hamming distance between the corresponding ground truth and the binary mask and average over the database. The obtained accuracy was 85.1%. Of course, a reasonable initialization is necessary for the

[4] To be in line with other sources we summarize the results in term of segmentation accuracy, i.e. the percentage of correctly classified pixels.

method. However, it is able to significantly change the initialization and considerably improve the segmentation accuracy.

It is also interesting to see, what are influences of the different model parts to the results. To investigate this we performed two additional tests – one without the shape function at all (i.e. $\phi \equiv 0$) and another one with independent MRF (i.e. $\alpha = 0$). Without the shape the model almost always produces a reasonable result – more or less compact segments, which have however nothing in common with horses. The accuracy in this case was about 83%. The situation without the Ising prior is slightly better – about 85%. In this case horses can be seen in obtained segmentations, which are however very noisy.

4 Conclusion

In this work we presented a model for segmentation, which consists of two parts: a discrete MRF and a continuous shape prior. We show that a proper combination of seemingly very different frameworks leads to promising results remaining at the same time quite simple. We gave a basic algorithm for inference that again consists of two parts, each one being standard in the corresponding framework. To conclude: discrete and continuous methods need not compete, they should rather work together.

There are numerous ways for the further research. The main motivation of the work was the intention to combine advantages of different methods. In particular we use a continuous shape prior in order to be able to use elaborated techniques from continuous optimization. In this work however we applied only a very basic method that is extremely inefficient in practice. Therefore this paper is rather a "work in progress". Other choices for the continuous part are to be evaluated.

Our treatment of "inference" (see sec. 2.2) differs from a commonly used one. Usually, *parameters* of a probability distribution are not dependent on a particular observation, i.e. they represent some intrinsic properties. They should be learned on a training dataset and used (remaining thereby unchanged) during inference. If something is image-dependent, then it is rather a *random variable*. In our model however, it is hardly possible to interpret the shape function as a random variable because of the infinite dimensionality. Therefore it is also not possible to consider the regularizer (5) as the prior probability distribution of shapes. Hence, we are obligated to consider it as a parameter. We plan to investigate the related questions more carefully in the future.

References

1. Cremers, D., Pock, T., Kolev, K., Chambolle, A.: Convex relaxation techniques for segmentation, stereo and multiview reconstruction. In: Markov Random Fields for Vision and Image Processing. MIT Press (2011)
2. Shekhovtsov, A., Kohli, P., Rother, C.: Curvature prior for MRF-based segmentation and shape inpainting. In: Pinz, A., Pock, T., Bischof, H., Leberl, F. (eds.) DAGM and OAGM 2012. LNCS, vol. 7476, pp. 41–51. Springer, Heidelberg (2012)
3. Schoenemann, T., Kahl, F., Masnou, S., Cremers, D.: A linear framework for region-based image segmentation and inpainting involving curvature penalization. International Journal of Computer Vision 99(1), 53–68 (2012)

4. Hoiem, D., Efros, A.A., Hebert, M.: Recovering surface layout from an image. IJCV 75(1), 151–172 (2007)
5. Veksler, O.: Star shape prior for graph-cut image segmentation. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 454–467. Springer, Heidelberg (2008)
6. Cremers, D., Sochen, N., Schnörr, C.: A multiphase dynamic labeling model for variational recognition-driven image segmentation. IJCV 66(1), 67–81 (2006)
7. Cremers, D., Osher, S.J., Soatto, S.: Kernel density estimation and intrinsic alignment for shape priors in level set segmentation. International Journal of Computer Vision 69, 335–351 (2006)
8. Flach, B., Schlesinger, D.: Modelling composite shapes by gibbs random fields. In: CVPR, pp. 2177–2182 (2011)
9. Nowozin, S., Lampert, C.: Structured Learning and Prediction in Computer Vision. Foundations and Trends in Computer Graphics and Vision, vol. 6 (2010)
10. Flach, B., Schlesinger, D.: Combining shape priors and MRF-segmentation. In: da Vitoria Lobo, N., Kasparis, T., Roli, F., Kwok, J.T., Georgiopoulos, M., Anagnostopoulos, G.C., Loog, M. (eds.) S+SSPR 2008. LNCS, vol. 5342, pp. 177–186. Springer, Heidelberg (2008)
11. Schlesinger, D.: A real-time MRF based approach for binary segmentation. In: Pinz, A., Pock, T., Bischof, H., Leberl, F. (eds.) DAGM and OAGM 2012. LNCS, vol. 7476, pp. 387–396. Springer, Heidelberg (2012)
12. Boykov, Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In: ICCV, vol. 1, pp. 105–112 (2001)
13. Rother, C., Kolmogorov, V., Blake, A.: “GrabCut": interactive foreground extraction using iterated graph cuts. ACM Trans. Graph. 23(3), 309–314 (2004)
14. Domke, J.: Learning graphical model parameters with approximate marginal inference. CoRR abs/1301.3193 (2013)
15. http://www.msri.org/people/members/eranb
16. Kumar, M.P., Torr, P.H.S., Zisserman, A.: Objcut: Efficient segmentation using top-down and bottom-up cues. IEEE Trans. Pattern Anal. Mach. Intell. 32(3), 530–545 (2010)
17. Levin, A., Weiss, Y.: Learning to combine bottom-up and top-down segmentation. Int. J. Comput. Vision 81(1), 105–118 (2009)
18. Borenstein, E., Malik, J.: Shape guided object segmentation. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 969–976 (June 2006)
19. Zhang, L., Ji, Q.: Image segmentation with a unified graphical model. IEEE Trans. Pattern Anal. Mach. Intell. 32(8), 1406–1425 (2010)
20. Bertelli, L., Yu, T., Vu, D., Gokturk, B.: Kernelized structural svm learning for supervised object segmentation. In: CVPR, pp. 2153–2160 (2011)
21. Joulin, A., Bach, F., Ponce, J.: Discriminative clustering for image co-segmentation. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1943–1950 (June 2010)
22. Cour, T., Benezit, F., Shi, J.: Spectral segmentation with multiscale graph decomposition. In: Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 1124–1131. IEEE (2005)
23. http://wwwpub.zih.tu-dresden.de/~ds24/rtsegm2/rtsegm2.html

Author Index