DISSERTATION

submitted to the

Combined Faculty for the Natural Sciences and Mathematics

of

Heidelberg University, Germany

for the degree of
Doctor of Natural Sciences

put forward by
Dipl. Math. Paul Swoboda
born in Kozienice (Poland)

Date of oral examination: .....

New Convex Relaxations and Global Optimality in Variational Imaging

Advisor: Prof. Dr. Christoph Schnörr

Zusammenfassung

Variationelle Methoden bilden einen grundlegenden Baustein für die Lösung vieler Probleme der Bildverarbeitung, etwa für das Problem der Segmentierung, der Tiefenschätzung, des optischen Flusses, der Objekterkennung etc. Viele dieser Probleme werden mithilfe eines Markovschen Zufalls-Feldes (MRF) oder eines Label-Zuweisungsproblems mit stetigen Variablen beschrieben. Das Finden einer Maximum A-Posteriori (MAP)-Konfiguration eines für das jeweilige Problem angepassten MRF oder das Finden eines Minimierers eines Label-Zuweisungsproblems ergibt eine Lösung des usprünglichen Problems. In beiden Fällen ist das Lösen eines strukturierten Optimierungsproblem erforderlich.

In dieser Arbeit studieren wir neue Erweiterungen für Markovsche Zufalls-Felder and Label-Zuweisungsprobleme, welche globale statistische Informationen und Bedingungen in das jeweilige Modell integrieren. Zu diesem Zweck schlagen wir handhabbare konvexe Relaxierungen des zugehörgien Optimierungsproblems, sowie Algorithmen, die diese lösen können, vor. Wir schlagen darüberhinaus einen allgemeinen Algorithmus vor, mithilfe dessen man einen Teil der MAP-Konfiguration eines MRF mithilfe von gebräuchlichen Relaxierungen finden kann.

Abstract

Variational methods constitute the basic building blocks for solving many image analysis tasks, be it segmentation, depth estimation, optical flow, object detection etc. Many of these problems can be expressed in the framework of Markov Random Fields (MRF) or as continuous labelling problems. Finding the Maximum A-Posteriori (MAP) solutions of suitably constructed MRFs or the optimizers of the labelling problems give solutions to the aforementioned tasks. In either case, the associated optimization problem amounts to solving structured energy minimization problems.

In this thesis we study novel extensions applicable to Markov Random Fields and continuous labelling problems through which we are able to incorporate statistical global constraints. To this end, we devise tractable relaxations of the resulting energy minimization problem and efficient algorithms to tackle them. Second, we propose a general mechanism to find partial optimal solutions to the problem of finding a MAP-solution of an MRF, utilizing only standard relxations.

Acknowledgments

I would like to thank my advisor Christoph Schnörr for introducing me to the field of mathematical image analysis and the associated optimization problems and for supporting me excellently throughout my whole PhD. Moreover I would like to thank Bogdan Savchynskyy, Jörg H. Kappes, Alexander Shekhovtsov, Thorsten Beier, Stefania Petra, Jan Kuske, Bernhard Schmitzer, Stefania Petra, Vera Tschaikowska, Marco Esquinazi and Warwara Wierna for fruitful discussions and collaboration, which greatly benefitted my research. Thanks also go to Evelyn Wilhelm for supporting me in all administrative aspects.

List of Figures

List of Tables

List of Algorithms

List of Publications

- P. Swoboda, B. Savchynskyy, J. H. Kappes, and C. Schnörr. Partial optimality via iterative pruning for the Potts model. In *SSVM*, 2013
- P. Swoboda, B. Savchynskyy, J. H. Kappes, and C. Schnörr. Partial optimality by pruning for MAP-inference with general graphical models. In *CVPR*, 2014
- P. Swoboda and C. Schnörr. Convex variational image restoration with histogram priors. *SIAM J. Imaging Sciences*, 6(3):1719–1735, 2013
- P. Swoboda and C. Schnörr. Variational image segmentation and cosegmentation with the Wasserstein distance. In *EMMCVPR*, pages 321–334, 2013

Contents

1 Introduction

Markov Random Fields (MRF), also known as graphical models, come from statistics and play a key role in most problems of image analysis. Finding the most likely configuration of a MRF, also called MAP-inference or energy minimization problem for graphical models, is of great importance in computer vision, bioinformatics, communication theory, statistical physics, combinatorial optimization, signal processing, information retrieval and statistical machine learning, see [2, 40, 111] for an overview of applications. MRFs factorize according to a given graphical structure, which gives us a way to build large models from simple building blocks. Often, this structure consists of a data term indicating which configuration each variable should take and pairwise regularization terms, which couple variables, which are expected to be directly related to each other. This structure helps in modelling and optimization.

The counterpart of MAP-inference in the variational community are continuous labelling problems. They usually consist of a dataterm and a regularizer. After discretizing, such problems can be expressed as MRFs with unaries for the data term and, under some conditions, pairwise terms for the regularizer.

Yet, there remain problems connected with the usage of MRFs. Among those, we will study the following two problems: First, finding MAP-solutions is, except for special cases, NP-hard [95]. Second, standard MRFs with low order potentials (pairwise, ternary etc.) are often not expressive enough. Both problems pose questions from the viewpoint of mathematical optimization:

1. How shall we find in practice MAP-solutions of such models?
2. How shall we extend MRFs to enable them to capture more global interactions, while still maintaining computational efficiency?

Analoguous questions can be posed for continuous labelling problems.

In this thesis we will study both problems. Our contribution consists in:

First, we will propose how to find *part* of a MAP-configuration of a standard pairwise MRF while only utilizing approximate solvers for this task. Such solvers, while fast in practice, produce solutions which may in general be far away from the MAP-configuration. The part of the generated solutions which equals the MAP-configuration is called partially optimal or persistent, see Chapter 3.

Second, we will propose global data terms based on Wasserstein distances [108]. These distances measure similarity between histograms, hence can be used to relate global properties of images, like grayvalue or feature distributions, to each other, see Chapter 5. We will use Wasserstein distances to solve denoising problems with statistical information on grayvalue distributions in Section 5.4 and for segmentation and cosegmentation in Section 5.5.

1.1 Notation

The following table summarises notation used throughout this thesis. Specific notation will be introduced at the beginning of the chapter where it will be needed.

	General Notation
$\lvert A\rvert$	the cardinality of a set A
$\mathbb{N}$	natural numbers
$\mathbb{Z}$	whole numbers
$\mathbb{N}_{a,b}$	subset $\{x \in \mathbb{N} \ : \ a \leq x \leq b\}$
$\mathbb{Z}_{a,b}$	subset $\{x \in \mathbb{Z} \ : \ a \leq x \leq b\}$
$\mathbb{R}$	real numbers
$\mathbb{R}_+$	set of non-negative reals $\{x \in \mathbb{R} \ : \ x \geq 0\}$
$\overline{\mathbb{R}}$	reals plus infinity, $\mathbb{R} \cup \{\infty\}$
(a, b)	open interval between a and b, i.e. $\{x \in \mathbb{R} \ : \ a < x < b\}$
$[a, b]$	closed interval between a and b, i.e. $\{x \in \mathbb{R} \ : \ a \leq x \leq b\}$
$A \times B$	Cartesian product of sets A and B
A^n	n-fold Cartesian product of set A, i.e. $\underbrace{A \times \ldots \times A}_{n \text{ times}}$
$\mathcal{P}(A)$	power set $\{B \ : \ B \subset A\}$
$\mathbb{R}^{n \times m}$	set of real-valued $m \times n$-matrices
x_i	i-th entry of vector $x \in \mathbb{R}^n$
$\langle a, b\rangle$	scalar product of $a, b \in \mathbb{R}^n$, i.e. $\sum_{i=1}^n a_i \cdot b_i$
$\lVert x\rVert$	Euclidean norm of x, i.e. $\lVert x\rVert = \sqrt{\langle x, x\rangle}$
e_i	i-th unit vector of $\mathbb{R}^n$ $(0, \ldots, 0, \underbrace{1}_{i\text{-th position}}, 0, \ldots, 0)$
$\mathcal{E}_n$	basis of $\mathbb{R}^n$ consisting of all unit vectors, i.e. $\{e_1, \ldots, e_n\}$ redbesser machen
$f_{\lvert C}$	restriction to $C \subset A$ of a function $f : A \to B$
$\operatorname{sign}(x)$	sign of x, i.e. $\begin{cases} 1, x > 0 \\ 0, x = 0 \\ -1, x < 0 \end{cases}$
	Markov Random Fields
$G = (\mathcal{V}, \mathcal{E})$	graph G with vertex set $\mathcal{V}$ and edge set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$
Λ	local polytope, defined in (2.2.7)
μ	(pseudo-)marginals $\mu \in \Lambda$
	Convex Analysis
$\operatorname{conv}(A)$	convex hull of $A \subset \mathbb{R}^n$, i.e. $\{\sum_{i=1}^{n+1} \alpha_i x^i \lvert \sum_{i=1}^{n+1} \alpha_i = 1, \alpha \geq 0, x^i \in A\}$
Δ_n	n-dimensional simplex, i.e. $\{a \in \mathbb{R}^n_+ \ : \ \sum_{i=1}^n a_i = 1\} = \operatorname{conv}(\mathcal{E}_n)$
$\operatorname{prox}_f(\cdot)$	prox-operator for function f, i.e. $\operatorname{prox}_f(x^0) = \min_x \frac{1}{2}\lVert x^0 - x\rVert + f(x)$
$\chi_A(\cdot)$	indicator function on set A, i.e. $\chi_A(x) = \begin{cases} 0, & x \in A \\ \infty, & x \notin A \end{cases}$
$\sigma_A(\cdot)$	support function of set A, i.e. $\sigma_A(x) = \sup_{y \in A}\langle x, y\rangle$
	Variational Analysis
$\operatorname{BV}(A, B)$	space of funtions $f : A \to B$ of bounded variation, see Definition 4.1.1

$C^k(\Omega)$	space of k-times differentiable functions on Ω
$C_c^k(\Omega)$	space of compactly supported k-times differentiable functions on Ω
Df	derivative operator, also in the weak sense of BV-functions
C'	set of lifted functions, see Definition 4.3.1
C''	convex hull of C', see (4.3.4)
	Measure Theory
$L_p(A)$	space of p-integrable functions on domain A
$(\mathbb{V}, \Sigma)$	measurable space of image values with the associated Borel-σ-Algebra.
$\mathcal{M}_+(A)$	space of finite, nonnegative Borel measures with domain A
$\Pi(\nu_1, \nu_2)$	space of coupling measures with marginals ν_1 and ν_2, see Definition 5.1.2
$\nu_{\llcorner A}$	measure ν restricted to set A
$f_*\nu$	pushforward of measure ν by map f defined by $(f_*\nu)(A) = \nu(f^{-1}(A))$
$\mathcal{K}$	dual admissible set for the Wasserstein distance, see Definition 5.1.4
ν_Ω^I	histogram of $I_{\|\Omega}$, see Definition 5.2.2

2 Markov Random Fields and Inference

In this chapter we give a short introduction to Markov Random Fields, insofar as is needed for this thesis, the associated MAP-inference problem and an overview of solvers used in this thesis for this task.

2.1 Basic Definitions

Markov Random Fields describe models together with a probability distribution, which factorize over a graphical structure.

Definition 2.1.1 (Label set and label space)**.** *Given an undirected graph $G = (\mathcal{V}, \mathcal{E})$, we associate to each node $v \in \mathcal{V}$ a label set X_v. The label space is the Cartesian product of all label sets: $X = \bigotimes_{v \in \mathcal{V}} X_v$.*

Remark 2.1.1. We will be mainly interested in discrete label sets, which means $X_u = \{1, \dots, n\}$ for some $n \in \mathbb{N}$. From now on, we will always assume that $|X_v| > 1$.

For notational convenience we write $X_{uv} = X_u \times X_v$ and for subsets $A \subset \mathcal{V}$ we write $X_A = \bigotimes_{v \in A} x_v$. To each vertex $v \in \mathcal{V}$ we associate a variable $x_v \in X_v$.

Definition 2.1.2 (Labeling)**.** *An assignment $x = \bigotimes_{v \in \mathcal{V}} x_v \in X$ of each variable x_v $\forall v \in \mathcal{V}$ is called a labeling.*

We use the notation $x_{uv} = (x_u, x_v)$ for $uv \in \mathcal{E}$. Notations like $x \in X_A$ implicitly indicate that the vector x only has components x_u indexed by $u \in A$. With $x_{|A} \in X_A$ we denote restriction of the labeling $x \in X_{\mathcal{V}}$ to the set $A \subset \mathcal{V}$. We want to associate to each labeling a cost. The cost should factorize according to the graphical structure described by G. This is done with the use of potentials.

Definition 2.1.3 (Potentials)**.** *To each vertex $v \in \mathcal{V}$ we associate a* unary *potential $\theta_u : X_u \to \mathbb{R}$ and for each edge $(uv) \in \mathcal{E}$ we associate a* pairwise *potential $\theta_{uv} : X_u \times X_v \to \mathbb{R}$.*

Now we connect all definitions to define Markov Random Fields.

Definition 2.1.4 (Markov Random Field)**.** *A* Markov Random Field (MRF) *is a tuple $(G = (\mathcal{V}, \mathcal{E}), X, (\theta_v)_{v \in \mathcal{V}} \cup \theta)_{uv \in \mathcal{E}})$ consisting of a graph G, a label space X and unary and pairwise potentials θ. The probability of a labeling $x \in X$ is*

$$p(x) = \frac{\exp\left(\sum_{v \in \mathcal{V}} \theta_v(x_v) + \sum_{uv \in \mathcal{E}} \theta_{uv}(x_u, x_v)\right)}{Z(\theta)}, \tag{2.1.1}$$

where $Z(\theta)$ is the partition function

$$Z(\theta) = \sum_{x \in X} \exp\left(\sum_{v \in \mathcal{V}} \theta_v(x_v) + \sum_{uv \in \mathcal{E}} \theta_{uv}(x_u, x_v) \right). \tag{2.1.2}$$

Remark 2.1.2. We can also define higher order Markov Random Fields by considering hypergraphs $\mathcal{H} = (V, \mathcal{S})$, where $\mathcal{S} \subset \mathcal{P}(V)$ ranges over some subsets of the vertices and where to each $s \in \mathcal{S}$ we associate a potential $\theta_s : X_s \to \mathbb{R}$. For example, if $|s| = 3$, we call θ_s a ternary potential.

2.2 MAP-Inference Problem

Definition 2.2.1 (MAP-Inference)**.** *The Maximum A-Posteriori inference problem for a Markov Random Field (G, X, θ) (short MAP-inference) consists of finding a most probable labeling, called MAP-solution, which is*

$$x^* \in \operatorname{argmax}_{x \in X_{\mathcal{V}}} p(x) \tag{2.2.1}$$

Such an x^* is also called the mode of the MRF. Finding the mode is equivalent to the energy minimization problem

$$\min_{x \in X} E_{\mathcal{V}}(x) := \sum_{v \in \mathcal{V}} \theta_v(x_v) + \sum_{uv \in \mathcal{E}} \theta_{uv}(x_u, x_v), \tag{2.2.2}$$

Definition 2.2.2 (Marginals)**.** *In the literature, MAP-inference also encompasses the more general problem of determining the marginals of p, that is computing the unary marginals*

$$\mu_v(i) = \sum_{x \in X, x_v = i} p(x) \text{ for all } v \in \mathcal{V} \text{ and } i \in X_v, \tag{2.2.3}$$

and the pairwise marginals

$$\mu_{uv}(i,j) := \sum_{x \in X, x_{uv} = (i,j)} p(x) \text{ for all } uv \in \mathcal{E} \text{ and } (i,j) \in X_{uv}. \tag{2.2.4}$$

Remark 2.2.1. It can be easily seen that finding the mode of an MRF can be accomplished by first computing the marginals and then selecting the most probable configurations in each vertex and edge suitably. Hence, marginalization is more general than mode-finding. In this thesis, we will not compute marginals, but below we will use the space of all possible marginals to linearize problem (2.2.2).

MAP-inference, i.e. finding a solution to (2.2.2), could theoretically be accomplished by enumerating all possible assignments $x \in X$. This would entail enumerating $\prod_{v \in \mathcal{V}} |X_v|$ possibilities, which is exponential in $|\mathcal{V}|$, hence not practicable. Therefore, polyhedral approaches where considered in the literature [112, 113]. To linearize (2.2.2), we first introduce the standard overcomplete representation for unary and pairwise marginals [111].

Definition 2.2.3 (Overcomplete Representation)**.** *To each variable $v \in \mathcal{V}$ and each label $i \in X_v = \{1, \dots, n\}$ we associate e_i, the i-th unit vector. Abusing notation, we define the unary cost in the overcomplete representation as $\langle \theta_v, e_i \rangle := \theta_v(i)$. To each edge $uv \in \mathcal{E}$ and each label combination $(i,j) \in X_{uv}$ we associate the unit vector $e_{i+j\cdot|X_u|}$. Abusing notation, we define the pairwise cost in the overcomplete representation as $\langle \theta_{uv}, e_{i+j\cdot|X_u|} \rangle := \theta_{uv}(i,j)$.*

The set of all marginals belonging to a probability distribution factorizing according to the graph G is

Definition 2.2.4 (Marginal Polytope)**.** *The marginal polytope is the set*

$$\mathcal{M}_{\mathcal{V}} := \operatorname{conv} \left\{ \mu \; : \; \begin{array}{l} \sum_{x_v \in \mathcal{V}} \mu_v(x_v) = 1, \;\; v \in \mathcal{V}, \\ \sum_{x_v \in \mathcal{V}} \mu_{uv}(x_u, x_v) = \mu_u(x_u), \;\; x_u \in X_u, \;\; uv \in \mathcal{E}, \\ \sum_{x_u \in \mathcal{V}} \mu_{uv}(x_u, x_v) = \mu_v(x_v), \;\; x_v \in X_v, \;\; uv \in \mathcal{E}, \\ \mu_v(x_v) \in \{0,1\}, \;\; v \in \mathcal{V}, \;\; x_v \in X_v \\ \mu_{uv}(x_u, x_v) \in \{0,1\}, \;\; (x_u, x_v) \in X_{uv}, \;\; uv \in \mathcal{E} \, . \end{array} \right\} \tag{2.2.5}$$

The extreme points of $\mathcal{M}_{\mathcal{V}}$ are by definition those marginals which correspond to a labeling $x \in X$ via the overcomplete representation. Hence we can now reformulate problem (2.2.2) as the following linear program.

$$\min_{\mu \in \Lambda_{\mathcal{V}}} \langle \theta, \mu \rangle := \sum_{v \in \mathcal{V}} \sum_{x_v \in X_v} \theta_v(x_v) \mu_v(x_v) + \sum_{uv \in \mathcal{E}} \sum_{x_{uv} \in X_{uv}} \theta_{uv}(x_{uv}) \mu_{uv}(x_{uv}) \, . \tag{2.2.6}$$

Slightly abusing notation we will denote the objective function in (2.2.6) as $E_{\mathcal{V}}(\mu)$.

It is unlikely that an efficient algorithm for MAP-inference exists, as the corresponding minimization problem is NP-hard [95]. Hence, it is unlikely, that the marginal polytope can be separated in polynomial time, as this would entail a polynomial time algorithm for MAP-inference [31]. There exist however special cases, where polynomial time algorithms exist [22, 52, 53, 59, 92]. These approaches usually rely on solving a linear relaxation of the MAP-inference problem and showing its tightness for the special cases considered.

To this end, we introduce the local polytope.

Definition 2.2.5 (Local Polytope)**.** *The local polytope is the set*

$$\Lambda_{\mathcal{V}} := \left\{ \mu \; : \; \begin{array}{l} \sum_{x_v \in \mathcal{V}} \mu_v(x_v) = 1, \;\; v \in \mathcal{V}, \\ \sum_{x_v \in \mathcal{V}} \mu_{uv}(x_u, x_v) = \mu_u(x_u), \;\; x_u \in X_u, \;\; uv \in \mathcal{E}, \\ \sum_{x_u \in \mathcal{V}} \mu_{uv}(x_u, x_v) = \mu_v(x_v), \;\; x_v \in X_v, \;\; uv \in \mathcal{E}, \\ \mu_v(x_v) \geq 0, \;\; v \in \mathcal{V}, \;\; x_v \in X_v \\ \mu_{uv}(x_u, x_v) \geq 0, \;\; (x_u, x_v) \in X_{uv}, \;\; uv \in \mathcal{E} \, . \end{array} \right\} \tag{2.2.7}$$

We define Λ_A for $A \subset \mathcal{V}$ similarly.

Remark 2.2.2. It is easily seen that $\mathcal{M}_{\mathcal{V}} \subset \Lambda_{\mathcal{V}}$. For trees, $\mathcal{M}_{\mathcal{V}} = \Lambda_{\mathcal{V}}$ holds. For graphs with cycles, strictness of this inequality can already be seen on models on the complete graph with four variables [112, Fig. 6]. Hence, variables $\mu \in \Lambda_{\mathcal{V}}$ are called *pseudo-marginals*.

The problem of finding

$$\min_{\mu \in \Lambda_{\mathcal{V}}} E_{\mathcal{V}}(\mu) \tag{2.2.8}$$

is called solving the *local polytope relaxation* of (2.2.2).

The local polytope can be described by inequalities whose number is polynomial in the input size of the underlying graph. Hence, solving the local polytope relaxation can be accomplished in polynomial time as well [45]. Note however that $\mu \in \Lambda_{\mathcal{V}}$ may not correspond to any probability distribution. In fact, when an optimal μ^* for (2.2.8) is not integral, i.e. $\mu \notin \{0,1\}^{|\Lambda_{\mathcal{V}}|}$, this is usually due to $\mu \notin \mathcal{M}_{\mathcal{V}}$.

For algorithmic purposes, it has turned out that solving the dual of (2.2.8) can be more advantageous than solving the primal formulation. First we introduce reparametrizations. It is well-known [89] (see also [112]) that representation (2.2.2) of the energy function is not unique. There are other potentials, which keep the energy of all labelings unchanged.

Definition 2.2.6 (Reparametrization)**.** *Potentials*

$$\theta_v^\phi(x_v) := \theta_v(x_v) - \sum_{u:uv \in \mathcal{E}} \phi_{vu}(x_v)\,, \tag{2.2.9}$$

$$\theta_{uv}^\phi(x_u, x_v) := \theta_{uv}(x_u, x_v) + \phi_{vu}(x_v) + \phi_{uv}(x_u) \tag{2.2.10}$$

with some numbers $\phi_{uv}(x_u)$, $uv \in \mathcal{E}$, $x_u \in X_u$ are called reparametrized potentials*. The vector θ^ϕ is called* reparametrization *of θ.*

The dual problem to (2.2.8) can now be compactly written.

Definition 2.2.7 (Dual Formulation of the Local Polytope Problem)**.** *The dual to (2.2.8) is*

$$\begin{array}{rll} \max_{z,\phi} & \sum_{v \in \mathcal{V}} z_v & \\ s.t. & z_v \le \theta_v^\phi(x_v) & \forall v \in \mathcal{V}\,, x_v \in X_v \\ & 0 \le \theta_{uv}^\phi(x_{uv}) & \forall uv \in \mathcal{E}\,, x_{uv} \in X_{uv}\,. \end{array} \tag{2.2.11}$$

Note that each reparametrization gives us a lower bound: For each vertex $v \in \mathcal{V}$ we choose label $x_v \in X_v$ such that $\theta_v^\phi(x_v)$ is minimal and for each edge $uv \in \mathcal{E}$ we choose label combination x_{uv} such that $\theta_{uv}^\phi(x_{uv})$ is minimal. Summing those potential values gives a lower bound for (2.2.2). In case the minimal labels are consistent, i.e. there exists a labeling $x \in X_{\mathcal{V}}$ such that $x_u \in \operatorname{argmin}_i \theta_u^\phi(i)$ $\forall u \in \mathcal{V}$ and $\theta_{uv}^\phi(x_{uv}) = 0$ $\forall uv \in \mathcal{E}$, the lower bound is the value of the energy minimization problem (2.2.2).

2.3 Solving MAP-Inference in Practice

In the preceeding Section 2.2 we have introduced the local polytope relaxation commonly used to approximately solve the underlying MAP-inference problem (2.2.2). However, large-scale problem sizes prohibit usage of off-the-shelf LP-solvers, as those algorithms empirically exhibit quadratic space and time complexity. Hence, approximate solvers were developed to efficiently arrive at points close to the solution of either (2.2.8) or (2.2.11). The following solution paradigms stand out:

- Dual block coordinate ascent methods based on belief propagation, described in Section 2.3.1,
- Proximal minimization algorithms from the field of convex variational analysis, described in Section 4.4.3,
- Dual decomposition [54, 55, 85] methods,
- Max-flow based techniques [11, 12, 30, 47, 49, 51, 53, 56, 82].

We will not detail the latter two methodologies here, since we will only use the first two in this thesis. For a comprehensive overview of solvers for MAP-inference and their respective applicability and merits, see [41].

2.3.1 Dual Block Coordinate Ascent Algorithms

Algorithms from this class historically arose from the belief propagation algorithms [111], also known as message passing. It was observed in [110], that belief propagation algorithms are connected to the dual of the local polytope relaxation (2.2.11). When applied to tree-structured graphs G, those algorithms amount to dynamic programming. In this case, they are exact [59] and can not only output the mode, but also marginals. Unfortunately, early versions of belief propagation algorithms were not convergent on graphs with cycles, but could oscillate [111]. To overcome this defect, monotonically increasing variants of those algorithms were developed [28, 29, 33, 34, 48, 50, 65, 88, 91, 98, 112, 117]. The algorithms directly addressing Problem (2.2.11) rely on performing in some sequence the marginalization (2.3.1) and averaging (2.3.2) operations described below.

Remark 2.3.1. The well-known dual block coordinate ascent algorithm TRW-S [48] will be used in chapter 3. In chapter 5.5 we will extend the dual block coordinate ascent operations described below to include operations for the Wasserstein distance described in chapter 5.

2.3.1.1 Basic Update Operations

The marginalization operation is applied to paiwise reparametrized potentials and consists in finding for fixed label $x_u \in X_u$ the best possible label $x_v \in X_v$ such that $\theta^{\phi}_{uv}(x_{uv})$ is minimal and updating the ϕ-variables by the value of the reparametrized potential.

Definition 2.3.1 (Marginalization Update)**.**

$$\phi_{uv}(x_u) -= \min_{x_v \in X_v} \theta^{\phi}_{v}(x_u, x_v) \tag{2.3.1}$$

Lemma 2.3.1. *Operation* (2.3.1) *increases the lower bound.*

Proof. Note that operation (2.3.1) respects the condition $\theta^{\phi}_{uv} \geq 0$ in (2.2.11). Also, operation (2.3.1) decreases the ϕ-variables, hence increases each component of θ^{ϕ}_{v}, hence we can increase the z_v-variable and thereby increase the dual lower bound. □

The second operation is applied to reparametrized unary potentials and consists of averaging the ϕ-variables, which are connected to the associated node $v \in \mathcal{V}$.

Definition 2.3.2 (Averaging Update)**.** *Let $(\omega_u)_{vu\in\mathcal{E}}$ be some sequence of non-negative weights and $\alpha = \min_{x_v \in X_v} \theta_v^\phi(x_v)$. The averaging operation consists in*

$$\phi_{vu}(x_v) \mathrel{+}= \omega_u \cdot \Big(\theta_v^\phi(x_v) - \alpha\Big) \tag{2.3.2}$$

Lemma 2.3.2. *Let the weights ω in Definition 2.3.2 be such that $\sum_{u:vu\in\mathcal{E}} \omega_u \leq 1$ and $\omega_u \geq 0$ $\forall u : uv \in \mathcal{E}$. Then Operation (2.3.2) increases the lower bound.*

Proof. Note that the right hand side of (2.3.2) is positive, hence the pairwise reparametrized potentials θ_{vu}^ϕ stay feasible. Let $\omega = \sum_{u:vu\in\mathcal{E}} \omega_u$. The new reparametrized unary is equal to

$$\theta_v^\phi(x_v) - \sum_{u:uv\in\mathcal{E}} \omega_u \cdot \Big(\theta_v^\phi(x_v) - \alpha\Big) = (1-\omega)\cdot\theta_v^\phi(x_v) + \omega\cdot\alpha\,. \tag{2.3.3}$$

Due to $\omega \leq 1$ the minimum over the latter is the minimum of θ_v^ϕ, hence the lower bound is not decreased. □

Various message passing algorithms differ in the sequence of steps by which the above two operations are performed, as well as in the choice of weights ω_u. These choices make a great difference in practice. A particularly choice is the TRWS [48, 50] algorithm, detailed in Algorithm 1. The backward iteration is done analoguously, by traversing the nodes in reverse order and exchanging $<$ by $>$ and vice versa.

Unfortunately, applying the operations (2.3.1) and (2.3.2) iteratively need not result in convergence to the optimum of relaxation (2.2.11), see [48, 50]. Instead, they may get stuck in so-called arc-consistent dual solutions, which constitute fixed points of this class of algorithms and where no dual increase is possible any more. However, these fixed points are usually very good in practice.

Remark 2.3.2. The algorithms [34, 88] address optimization of a smoothed formulation of (2.2.8). Specifically, the objective function also includes an entropic term. This results in a smooth dual problem. Replacing the operations $(\min, +)$ by $(+, \cdot)$ in the marginalization operation, leads to dual block coordinate ascent for their smooth dual. Due to smoothnedd, these algorithms converge and have no suboptimal fixed points.

2.3.1.2 Obtaining Labelings from Dual Solutions

One drawback of optimizing the dual formulation (2.2.11) is that no labelings $x \in X$, which is what we seek, or at least pseudo-marginals $\mu \in \Lambda_\mathcal{V}$ are returned, but only a lower bound on the energy of the MAP-solution is computed. To overcome this problem, various *rounding algorithms* have been proposed [48, 87, 114]. One of the best procedures is the rounding approach from [48]. It consists in first choosing an order of the vertices in $\mathcal{V}$, i.e. $\{v_1, \ldots, v_{|\mathcal{V}|}\} = \mathcal{V}$, then traversing all nodes in this order

Algorithm 1: One forward iteration of TRWS [48]

Input : Graph $\mathcal{G} = (\mathcal{V} = \{1, \dots, n\}, \mathcal{E})$, potentials $\theta_u, u \in \mathcal{V}$, $\theta_{uv}, uv \in \mathcal{E}$.

for $u = 1, \dots, n$ **do**
  Receive Messages:
  for $v : uv \in \mathcal{E}, v < u$ **do**
    Compute $\phi_{u,v}(x_u) \mathrel{-}= \min_{x_v \in X_v}\{\theta^\phi_u v(x_u, x_v)\} \quad \forall x_u \in X_u$
  end
  Round Primal Solution:
  $x^*_u = \min_{x \in X_u}\{\theta^\phi_u(x) + \sum_{v:uv\in\mathcal{E}, v<u} \theta^\phi_{uv}(x_u, x^*_v)\}$.
  Send Messages:
  Compute $\delta^*(x_u) = \theta^\phi_u(x_u) - \min_{x'_u \in X_u}\{\theta^\phi_u(x'_u)\}$. Set $\omega = \frac{1}{|\{v:uv\in\mathcal{E}, v>u\}|}$.
  for $v : uv \in \mathcal{E}, v > u$ **do**
    Update $\phi_{(u,v)} \mathrel{+}= \omega \cdot \delta^*$
  end
end

and choosing a minimal label $x^*_v \in \operatorname{argmin}_{x_{v_k} \in X_{v_k}} \theta^\phi_{v_k}(x_v) + \sum_{l<k: v_k v_l \in \mathcal{E}} \theta^\phi_{v_k v_l}(x_{v_k}, x^*_{v_l})$. See also Algorithm 1.

The difference $E_\mathcal{V}(x^*) - \langle \mathbb{1}, z \rangle$ between the energy of the MAP-inference problem (2.2.2) and the dual lower bound obtained by minimizing (2.2.11) is called the *integrality gap* and is always non-negative. A vanishing integrality gap means that we have found an optimal reparametrization θ^ϕ and a MAP-solution $x^* \in X$. While often not zero, the integrality gap is usually small for typical image analysis tasks.

3 Partial Optimality for Markov Random Fields

This chapter is based on publications [101, 102].

3.1 Introduction

In this chapter we consider the MAP-inference (2.2.2) problem. As previously noted, MAP-inference is NP-hard and we cannot expect to obtain MAP-solutions in polynomial time. In practice, the local polytope relaxation (2.2.8) is good and the associated integraily gap is small and close approximate solutions can be obtained efficiently in big MRFs commonly arising in image processing by approximate methods as discussed in Section 2.3 and [40, 105]. However, the obtained rounded solutions do not need to coincide with the MAP-solutions, as long as the integrality gap is strictly larger than zero. If one could prove, that some variables of the solution given by such approximate algorithms belong to an optimal configuration, the value of such approximate methods would be greatly enhanced. In particular, the problem for the remaining variables could be solved by stronger, but computationally more expensive methods to obtain a global optimum as done e.g. in [43].

In particular, natural questions are: (i) Is there a subset $A \subset \mathcal{V}$ and a labeling x^0 of the original NP-hard problem (2.2.2) such that a for minimizer x^* of some relaxation $x^0_v = \mu^*_v \; \forall v \in A$ holds? In other words, is x^* *partially optimal* or *persistent* on some set A? (ii) Given a approximate solution x^*, how can we determine such a set A?

In this chapter we propose a novel polynomial time algorithm to gain such a partially optimal solution for the MAP-inference problem with *general* discrete MRFs from possibly also non-exact solutions of the commonly used local polytope relaxation (2.2.8). Our algorithm is initialized with variables taking integral values in the solution of a convex relaxation of the MAP-inference problem and iteratively prunes those variables, which do not satisfy our criterion for partial optimality. We show that our pruning strategy is in a certain sense theoretically optimal. Also empirically our method outperforms previous approaches in terms of the number of persistently labelled variables. The method is very general, as it is applicable to models with arbitrary factors of an arbitrary order and can employ any solver for the considered relaxed problem. Our method's runtime is determined by the runtime of the convex relaxation solver for the MAP-inference problem. Solving over the local polytope amounts to solving a linear problem for which *any* linear programming (LP) solver can be used and for which dedicated and efficient algorithms exist.

3.1.1 Related Work

We distinguish two classes of approaches to partial optimality.

3.1.1.1 Roof duality based approaches

The earliest paper dealing with persistency is [67], which states a persistency criterion for the stable set problem and verifies it for every solution of a certain relaxation. This relaxation is the same, as used by the roof duality method in [8] and which is also the basis for the well known QPBO-algorithm [8, 82]. The MQPBO method [47] extends roof duality to the multi-label case. The authors transform multi-label problems into quadratic binary ones and solve them via QPBO [8]. However, their transformation is dependent upon choosing a label order and their results are so as well, see the experiments in [101], where the label order is sampled randomly. It is not known how to choose an optimal label order to obtain the maximum number of persistent variables.

The roof duality method has been extended to higher order binary problems in [26, 37, 39]. The generalized roof duality method for binary higher order problems [39] computes partially optimal variables directly for higher order potentials, while Ishikawa's and Fix et al's approaches [26, 37] transform the higher order problem to ones with unary and pairwise terms only. Fix et al's method [26] is an improvement upon Ishikawa's [37].

Windheuser et al [115] proposed *a multi-label higher-order* roof duality method, which is a generalization of both MQPBO [47] to higher order and Kahl and Strandmark's work [39] to the multi-label case. However Windheuser et al neither describe an implementation nor provide experimental validation for the higher order multi-label case.

3.1.1.2 Labeling testing approaches

A different approach, specialized for Potts models, is pursued by Kovtun [57], where possible labelings are tested for persistency by auxiliary submodular problems. The dead-end elimination procedure [23] tests, if certain labels of nodes cannot belong to an optimal solution. It is a local heuristic and does not perform any optimization.

Since for non-binary multi-labeling problems the submodular approximations constructed by approaches of class (i) are provably less tight than the standard local polytope relaxation [94, Prop. 1], we consider class (ii) in this paper. Specifically, based on ideas in [101] to handle the Potts model, we develop a theoretically substantiated approach to recognizing partial optimality for *general* graphical models, together with a competitive comparison to the 5 approaches [26, 37, 39, 47, 57] discussed above, that define the state-of-the-art.

3.1.1.3 Unified study

In addition we point to the recent paper [93], which provides a unified study of most mentioned methods and a systematic way of their analysis. While their persistency

criterion is provably not weaker than ours, due to the general structure of the resulting LP it cannot be applied to large-scale problems in a straightforward manner. Moreover, our approach is directly applicable to higher order models and tighter than the local polytope relaxations, whereas [93] requires generalization to these cases, though such a generalization is presumably possible. We show that our algorithm solves a special case of the maximal presistency problem formulated in [93].

3.1.1.4 Shrinking technique.

The recent work [86] proposes a method for efficient shrinking of the combinatorial search area with the local polytope relaxation. Though the algorithmic idea is similar to the presented one, the method [86] does not provide partially optimal solutions. We refer to Section 3.3 for further discussion.

3.1.2 Contribution and Organization

We propose *a novel* method for computing partial optimality, which is applicable to *general graphical models with arbitrary higher order potentials*. Our algorithm is initialized with variables taking integral values in the solution of a convex relaxation of the MAP-inference problem and iteratively prunes those, which do not satisfy our persistency criterion. We show that our pruning strategy is in a certain sense theoretically optimal. Though the used relaxation can be chosen arbitrarily, for brevity we restrict our exposition and experiments to the local polytope relaxation (2.2.7). Tighter relaxations *provably* yield better results. However even by using the local polytope relaxation we can often achieve *a substantially higher* number of persistent variables than competing approaches, which we confirm experimentally. We also show how our approach can be made invariant against reparametrizations. This improves our partial optimality criterion and we can show equivalence with the all-to-one improving mapping class of partial optimality methods proposed in [93]. Our approach is very general, as it can use *any*, also approximate, solver for the considered convex relaxation. Moreover, the computational complexity of our method is determined mainly by the runtime of the used solver.

The comparison to existing persistency methods is summarized in Table 3.1.

Our code together with the experimental setup is available at `http://paulswoboda.net`.

3.1.2.1 Organization

In Section 3.2 our persistency criterion is presented. The corresponding algorithm and its theoretical analysis are presented in Sections 3.3, 3.4 and 3.5 respectively. Extensions to the higher order case and tighter relaxations are discussed in Section 3.6. Section 3.7 provides experimental validation of our approach and a comparison to the existing methods [26, 37, 39, 47, 57].

Work	non-binary	higher order	non-Potts	Auxiliary problem
Boros & Hammer 2002 [8]	−	−	+	QPBO
Kovtun 2003[57]	+	−	−	submodular
Rother et al. 2007 [82]	−	−	+	QPBO
Kohli et al. 2008 [47]	+	−	+	QPBO
Kovtun 2011 [58]	+	−	+	submodular
Ishikawa 2011 [37]	−	+	+	QPBO
Fix et al. 2011 [26]	−	+	+	QPBO
Kahl & Strandmark 2012 [39]	−	+	+	bi-submodular
Windheuser et al. 2012 [115]	+	+	+	bi-submodular
Swoboda et al. 2013 [101]	+	−	−	local polytope
Shekhovtsov 2014 [93]	+	−	+	general linear program
Ours	+	+	+	**any convex relaxation**

Table 3.1 - Comparison between partial optimality methods. A detailed description is presented in Section 3.1.1.

3.2 Persistency

Assume we have marginals $\mu \in \Lambda_{\mathcal{V}}$. We say that the marginal μ_u, $u \in \mathcal{V}$, is *integral* if $\mu_u(x_u) \in \{0,1\}\ \forall x_u \in X_u$. In this case the marginal corresponds uniquely to a label x_u with $\mu_u(x_u) = 1$. If this *integrality condition* holds for all $u \in \mathcal{V}$ the corresponding vector μ will be denoted as $\delta(x)$. For a wide spectrum of problems however most of the entries of optimal marginals μ^* for the local polytope relaxation will be integral. Unfortunately, there is no guarantee that any of these integral variables will be part of a globally optimal solution to (2.2.6), except in the case of binary variables, that is $X_u = \{0,1\}\ \forall u \in \mathcal{V}$, and unary and pairwise potentials [32].

Let the boundary nodes and edges of a subset of nodes $A \subset \mathcal{V}$ be defined as follows:

Definition 3.2.1 (Boundary and Interior)**.** *For the set $A \subset \mathcal{V}$ the set $\partial\mathcal{V}_A := \{u \in A : \exists v \in \mathcal{V}\backslash A \text{ s.t. } uv \in \mathcal{E}\}$ is called* its boundary*. The respective* set of boundary edges *is defined as $\partial\mathcal{E}_A = \{uv \in \mathcal{E} : u \in A \text{ and } v \in \mathcal{V}\backslash A\}$. The set $A\backslash\partial\mathcal{V}_A$ is called the interior of A.*

An exemplary graph illustrating the concept of interior and boundary nodes can be seen in Figure 3.1.

Definition 3.2.2 (Persistency)**.** *A labeling $x^0 \in X_A$ on a subset $A \subset \mathcal{V}$ is* partially optimal *or* persistent *if x^0 coincides with an optimal solution to (2.2.2) on A.*

In the remainder of this section, we state our novel persistency criterion in Theorem 3.2.1. Taking additionally into account convex relaxation yields a computationally tractable approach in Corollary 3.2.1.

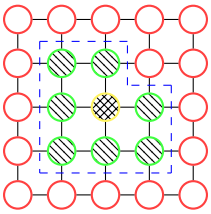


Figure 3.1 - An exemplary graph containing inside nodes (yellow with crosshatch pattern) and boundary nodes (green with diagonal pattern). The blue dashed line encloses the set A. Boundary edges are those crossed by the dashed line.

As a starting point, consider the following sufficient criterion for persistency of $x^0 \in X_A$. Introducing a *concatenation* of labelings $x^0 \in X_A$ and $\tilde{x} \in X_{\mathcal{V}\setminus A}$ as $(x^0, \tilde{x}) := \begin{cases} x^0_v, & v \in A, \\ \tilde{x}_v, & v \in \mathcal{V}\setminus A \end{cases}$, the criterion reads:

Proposition 3.2.1. *The partial labeling $x^0 \in X_A$ is persistent if there holds*

$$\forall \tilde{x} \in X_{\mathcal{V}\setminus A} \;:\; x^0 \in \operatorname*{argmin}_{x \in X_A} E_{\mathcal{V}}((x, \tilde{x}))\,. \tag{3.2.1}$$

Proof. Consider the equation

$$\min_{x \in X_{\mathcal{V}}} E(x) = \min_{\tilde{x} \in X_{\mathcal{V}\setminus A}} \min_{x \in X_A} E_{\mathcal{V}}((x, \tilde{x}))\,. \tag{3.2.2}$$

Let $\tilde{x} \in X_{\mathcal{V}\setminus A}$ be such that it leads to a minimal value on the right hand side of (3.2.2). Then $\tilde{x}$ is part of an optimal solution. By the assumption (3.2.1), x^0 is an optimal solution to the inner minimization problem of (3.2.2), hence $(x^0, \tilde{x})$ is optimal for (2.2.2). □

This means that if we fix *any* labeling $\tilde{x}$ on the complement of A and optimize with respect to x^0 on A, the concatenated labeling $(x^0, \tilde{x})$ has to be optimal. Informally this means that the solution x^0 is independent of what happens on $\mathcal{V}\setminus A$. This criterion however is hard to check directly, as it entails solving NP-hard minimization problems over an exponential number of labelings $\tilde{x} \in X_{\mathcal{V}\setminus A}$.

We relax the above criterion (3.2.1) so that we have to check the solution of only *one* energy minimization problem by modifying the unaries θ_v on boundary nodes so that they bound the influence of *all* labelings on $\mathcal{V}\setminus A$ uniformly.

Definition 3.2.3 (Boundary potentials and energies)**.** *For a set $A \subset \mathcal{V}$ and* a test *labeling $y \in X_A$, we define for each boundary edge $uv \in \partial\mathcal{E}_A$, $u \in \partial\mathcal{V}_A$ the "boundary" potential $\hat{\theta}_{uv,y_u} : X_u \to \mathbb{R}$ as follows:*

$$\hat{\theta}_{uv,y_u}(x_u) := \begin{cases} \max_{x_v \in X_v} \theta_{uv}(x_u, x_v), & y_u = x_u \\ \min_{x_v \in X_v} \theta_{uv}(x_u, x_v), & y_u \neq x_u \end{cases}. \tag{3.2.3}$$

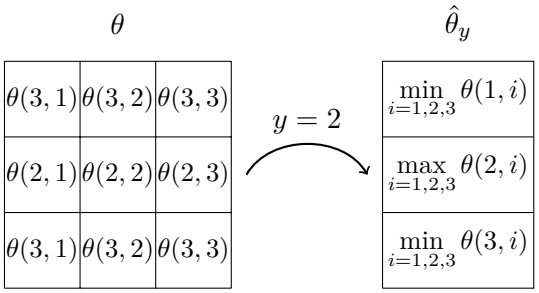


Figure 3.2 - Illustration of a boundary potential $\hat{\theta}_y$ constructed in (3.2.3). The second label comes from the test labeling y, therefore entries are maximized for the second row and minimized otherwise.

Define the energy $\hat{E}_{A,y} \colon X_A \to \mathbb{R}$ *with test labeling* y *as*

$$\hat{E}_{A,y}(x) := E_A(x) + \sum_{uv \in \partial\mathcal{E}_A \colon u \in \partial\mathcal{V}_A} \hat{\theta}_{uv,y_u}(x_u)\,, \tag{3.2.4}$$

where $E_A(x) = \sum_{u \in A} \theta_u(x_u) + \sum_{uv \in \mathcal{E} : u,v \in A} \theta_{uv}(x_{uv})$ *is the energy with potentials with support in* A*.*

Given a test labeling $y \in X_A$, energy (3.2.4) assigns a higher value than the original energy (2.2.2) for all labelings conforming to y and makes it more favourable for all labelings not conforming to y. An illustration of a boundary potential is depicted in Figure 3.2.

As a consequence, if the test labeling y from Definition 3.2.1 minimizes the energy (3.2.4), the proof of the following theorem asserts that changing an arbitrary labeling $x \in X_\mathcal{V}$ as follows: $x'(v) = \begin{cases} y(v), & v \in A \\ x(v), & v \notin A \end{cases}$ will always result in a labeling with not bigger energy (2.2.2), hence y in particular fulfills the conditions (3.2.1) of Proposition 3.2.1 and thus is persistent.

Theorem 3.2.1 (Partial optimality criterion)**.** *A labeling* $x^0 \in X_A$ *on a subset* $A \subseteq \mathcal{V}$ *is persistent if*

$$x^0 \in \operatorname{argmin}_{x \in X_A} \hat{E}_{A,x^0}(x)\,, \tag{3.2.5}$$

where $\hat{E}_{A,x^0}$ *is the augmented energy functional* (3.2.4)*.*

To prove the theorem we need the following technical lemma.

Lemma 3.2.1. *Let* $A \subset \mathcal{V}$ *be given together with* $y \in X_A$*. Let* x^0 *and* x' *be two labelings on* $\mathcal{V}$ *such that* $x^0|_A = y$*. Then it holds for* $uv \in \partial\mathcal{E}_A$*,* $u \in \partial\mathcal{V}_A$ *that*

$$\theta_{uv}(x^0_u, x'_v) + \hat{\theta}_{uv,y}(x'_u) - \hat{\theta}_{uv,y}(x^0_u) \leq \theta_{uv}(x'_u, x'_v)\,. \tag{3.2.6}$$

Proof. The case $x'_u = x^0_u$ is trivial. Otherwise, by Definition 3.2.3, inequality (3.2.6)

is equivalent to

$$\theta_{uv}(x_u^0, x'_v) + \min_{x_v \in X_v} \theta_{uv}(x'_u, x_v) - \max_{x_v \in X_v} \theta_{uv}(x_u^0, x_v) - \theta_{uv}(x'_u, x'_v) \leq 0\,. \quad (3.2.7)$$

Choose x'_v for x_v in the minimization and maximization in (3.2.7) to obtain the result. □

Proof of Theorem 3.2.1. Let

$$\tilde{x} \in \arg \min_{\substack{x \in X_{\mathcal{V}} \\ x|_A = x^0|_A}} E_{\mathcal{V}}(x)\,. \quad (3.2.8)$$

and let $x' \in X_{\mathcal{V}}$ be an arbitrary labeling. Then

$$E_{\mathcal{V}}(\tilde{x}) = E_A(x^0) + E_{\mathcal{V}\setminus A}(\tilde{x}) + \sum_{uv \in \partial\mathcal{E}_A} \theta_{uv}(x_u^0, \tilde{x}_v) \quad (3.2.9)$$

$$= E_A(x^0) + \sum_{uv \in \partial\mathcal{E}_A} \hat{\theta}_{uv,y}(x_u^0) \quad (3.2.10)$$

$$+ E_{\mathcal{V}\setminus A}(\tilde{x}) + \sum_{uv \in \partial\mathcal{E}_A} \left[\theta_{uv}(x_u^0, \tilde{x}_v) - \hat{\theta}_{uv,y}(x_u^0)\right] \quad (3.2.11)$$

$$= \hat{E}_{A,x^0}(x^0) + E_{\mathcal{V}\setminus A}(\tilde{x}) + \sum_{uv \in \partial\mathcal{E}_A} \left[\theta_{uv}(x^0, \tilde{x}_v) - \hat{\theta}_{uv,x^0}(x_u^0)\right] \quad (3.2.12)$$

$$\leq \hat{E}_{A,x^0}(x') + E_{\mathcal{V}\setminus A}(x') + \sum_{uv \in \partial\mathcal{E}_A} \left[\theta_{uv}(x^0, x'_v) - \hat{\theta}_{uv,x^0}(x_u^0)\right] \quad (3.2.13)$$

$$= E_A(x') + \sum_{uv \in \partial\mathcal{E}_A} \hat{\theta}_{uv,x^0}(x'_u) \quad (3.2.14)$$

$$+ E_{\mathcal{V}\setminus A}(x') + \sum_{uv \in \partial\mathcal{E}_A} \left[\theta_{uv}(x_u^0, x'_v) - \hat{\theta}_{uv,x^0}(x_u^0)\right] \quad (3.2.15)$$

$$\leq E_A(x') + E_{\mathcal{V}\setminus A}(x') + \sum_{uv \in \partial\mathcal{E}_A} \theta_{uv}(x'_u, x'_v) = E_{\mathcal{V}}(x'). \quad (3.2.16)$$

The equality (3.2.9) is due to definition of $\tilde{x}$ in (3.2.7). The first inequality (3.2.13) is due to $x^0 \in \operatorname{argmin}_x \hat{E}_{A,x^0}(x)$, as assumed, and of $\tilde{x}$ for (3.2.8). The second inequality (3.2.16) is due to Lemma 3.2.1. Hence x^0 is part of a globally optimal solution, as x' was arbitrary. □

Checking the criterion in Theorem 3.2.1 is NP-hard, because (3.2.5) is a MAP-inference problem of the same class as (2.2.2). By relaxing the minimization problem (3.2.5) one obtains the polynomially verifiable persistency criterion in Corollary 3.2.1.

Corollary 3.2.1 (Tractable partial optimality criterion)**.** *Labeling $x^0 \in X_A$ on $A \subset \mathcal{V}$ fulfilling the condition*

$$\delta(x^0) \in \operatorname{argmin}_{\mu \in \Lambda_A} \hat{E}_{A,x^0}(\mu) \quad (3.2.17)$$

Algorithm 2: Finding persistent variables.

Data: $G = (\mathcal{V}, \mathcal{E})$, $\theta_u : X_u \to \mathbb{R}$, $\theta_{uv} : X_{uv} \to \mathbb{R}$
Result: $A^* \subset \mathcal{V}$, $x^* \in X_{A^*}$

Initialize:
Choose $\mu^0 \in \operatorname{argmin}_{\mu \in \Lambda_\mathcal{V}} E_\mathcal{V}(\mu)$
$A^0 = \{u \in \mathcal{V} \colon \mu_u^0 \in \{0,1\}^{|X_u|}\}$
$t = 0$
repeat
  Set x_u^t such that $\mu_u^t(x_u^t) = 1,\ u \in A^t$
  Choose $\mu^{t+1} \in \operatorname{argmin}_{\mu \in \Lambda_{A^t}} \hat{E}_{A^t, x^t}(\mu)$
  $t = t + 1$
  $W^t = \{u \in \partial\mathcal{V}_{A^{t-1}} \ :\ \mu_u^t(x_u^{t-1}) \neq 1\}$
  $A^t = \{u \in A^{t-1} \colon \mu_u^t \in \{0,1\}^{|X_u|}\} \backslash W^t$
until $A^t = A^{t-1}$;
$A^* = A^t$
Set $x^* \in X_{A^*}$ such that $\mu_u^t(x_u^*) = 1$

is also a solution to (3.2.5)*, hence persistent on* A*.*

Proof. Expression (3.2.17) implies

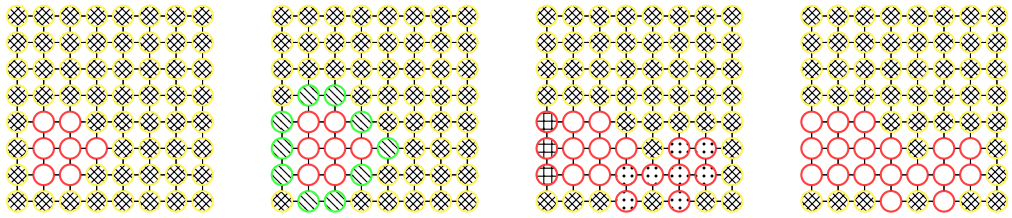
$$\delta(x^0) \in \operatorname{argmin}_{\mu \in \Lambda_A, \mu \in \{0,1\}} \hat{E}_{A,x^0}(\mu) \tag{3.2.18}$$

because $\delta(x^0)$ is integral by definition. As (2.2.2) and (2.2.6) are equivalent and the corresponding labeling x^0 satisfies the conditions of Theorem 3.2.1, x^0 is partially optimal on A. □

3.3 Persistency Algorithm

Now we concentrate on finding a set A and labeling $x \in X_A$ such that the solution of $\min_{\mu \in \Lambda_A} \hat{E}_{A,x}(\mu)$ fulfills the conditions of Corollary 3.2.1. Our approach is summarized in Algorithm 2.

In the initialization step of Algorithm 2 we solve the relaxed problem over $\mathcal{V}$ without boundary labeling and initialize the set A^0 with nodes having an integer label. Then in each iteration t we minimize over the local polytope the energy $\hat{E}_{A^t,x^t}$ defined in (3.2.4), corresponding to the set A^t and boundary labeling coming from the solution of the last iteration. We remove from A^t all variables which are not integral or do not conform to the boundary labeling. In each iteration t of Algorithm 2 we shrink the set A^t by removing variables taking non-integral values or not conforming to the current boundary condition. See Figure 3.3 for an illustration of one iteration of Algorithm 2.



nodes A^t,
nodes $\mathcal{V}\backslash A^t$.

Boundary costs are assigned to boundary nodes on $\partial\mathcal{V}_{A^t}$.

Having solved the inference problem:
nodes are fractional,
nodes disagree with previous labeling,
nodes agree.

Variables are pruned. A new set A^{t+1} is constructed.

Figure 3.3 - Illustration of one iteration of Algorithm 2.

3.3.1 Convergence

Since $\mathcal{V}$ is finite and $|A^t|$ is monotonically decreasing, the algorithm converges in at most $|\mathcal{V}|$ steps. Solving each subproblem in Algorithm 2 can be done in polynomial time. As the number of iterations of Algorithm 2 is at most $|\mathcal{V}|$, Algorithm 2 itself is polynomial as well. In practice only few iterations are needed.

After termination of Algorithm 2, we have

$$\delta(x^*) \in \operatorname{argmin}_{\mu\in\Lambda_{A^*}} \hat{E}_{A^*,x^*}(\mu)\,. \tag{3.3.1}$$

Hence x^* and A^* fulfill the conditions of Corollary 3.2.1, which proves persistency.

3.3.2 Choice of Solver

All our results are independent of the specific algorithm one uses to solve the relaxed problems $\min_{\mu\in\Lambda_A} \hat{E}_{A,y}$, provided it returns an exact solution. However this can be an issue for large-scale datasets, where classical exact LP solvers like e.g. the simplex method become inapplicable. It is important that one can also employ *approximate* solvers, as soon as they provide (i) a proposal for *potentially* persistent nodes and (ii) sufficient conditions for optimality of the found *integral* solutions such as e.g. zero duality gap. These properties have the following precise formulation.

Definition 3.3.1 (Consistent labeling)**.** *A labeling $c \in \bigotimes_{v\in\mathcal{V}} (X_v \cup \{\#\})$ is called* a consistent labeling *for the energy minimization problem (2.2.2), if from $c_v \in X_v$ $\forall v \in \mathcal{V}$ follows that $c \in \operatorname{argmin}_{x\in X_\mathcal{V}} E_\mathcal{V}(x)$.*

We will call an algorithm for solving the energy minimization problem (2.2.2) consistency ascertaining*, if it provides a consistent labeling as its output.*

Consistent labelings can be constructed for a wide range of algorithms, e.g.:

- Dual decomposition based algorithms [42, 55, 85, 88] deliver *strong tree agreement* [109] and dual block coordinate ascent methods [48, 50] and algorithms considering the Lagrangian dual [28, 34, 90] return *strong arc consistency* [112] for some nodes. If one of these properties holds for a node v, we set c_v as the corresponding label. Otherwise we set $c_v = \#$.

- Naturally, any algorithm solving $\min_{\mu \in \Lambda_{\mathcal{V}}} E(\mu)$ exactly is consistency ascertaining with
$c_v = \begin{cases} x_v, & \mu_v(x_v) = 1 \\ \#, & \mu_v \notin \{0,1\}^{|X_v|} \end{cases}.$

Proposition 3.3.1. *Let operations* $\mu \in \operatorname{argmin}(...)$ *in Algorithm 2 be exchanged with*

$$\forall v \in \mathcal{V}, x_v \in X_v,\ \mu_v(x_v) := \begin{cases} 1, & c_v = x_v \\ 0, & c_v \notin \{x_v, \#\}, \\ 1/|X_v|, & c_v = \# \end{cases}$$

where c are consistent labelings returned by a consistency ascertaining algorithm applied to the corresponding minimization problems. Then the output labeling x^* is persistent.

Proof. At termination of Algorithm 2 we have obtained a subset of nodes A^*, a test labeling $y^* \in X_A$, a labeling x^* equal to y^* on A and a consistency mapping $c_u = x^*_u$ for $u \in A^*$. Hence, by Definition 3.3.1, $x^* \in \operatorname{argmin}_{x \in X_A} \hat{E}_{A^*, y^*}$ and x^* fulfills the conditions of Theorem 3.2.1. □

Remark 3.3.1. Note that a bad or early stopped solver, i.e. one which rarely (or even never) returns an optimality certificate or solves a weak relaxation, will also work with Algorithm 2. However it will find smaller (or even empty) partial optimal solutions.

3.3.3 Comparison to the Shrinking Technique (CombiLP) [86]

The recently published approach [86], similar to Algorithm 2, describes how to shrink the combinatorial search area with the local polytope relaxation. However (i) Algorithm 2 solves a series of auxiliary problems on the subsets A^t of integer labels, whereas the method [86] considers nodes, which got fractional labels in the relaxed solution; (ii) Algorithm 2 is polynomial and provides only persistent labels, whereas the method [86] has exponential complexity and either finds an optimal solution or gives no information about persistence.

From the practical point of view, both algorithms have different application scenarios: CombiLP [86] will only work on sparse graphs, as otherwise the combinatorial part, which one has to solve with exact methods, becomes too big, as the boundary $\partial \mathcal{V}_A$ for $A \subsetneq \mathcal{V}$ grows very quickly then. Also, even for sparse graphs, the combinatorial part may not grow too big during the application of the algorithm, as otherwise the combinatorial solver will again not be able to cope with it. Our algorithm does not possess these two disadvantages. From the perspective of running time it does

	CombiLP [86]	Our method
Dense graphs	−	+
Very large-scale	−	+
Big fractional part of LP solution	−	+
Relaxed MAP-inference is solved only once	+	−
Provides a complete solution to Labeling Problem (2.2.2)	+	−

Table 3.2 - Comparison between our method and CombiLP [86].

not matter how big the set $\mathcal{V}\backslash A^t$ becomes during the iterations of Algorithm 2. On the other hand, the subsets of variables to which the method [86] applies a combinatorial solver to achieve global optimality are often smaller than $\mathcal{V}\backslash A^t$ in Algorithm 2, because potentials in CombiLP [86] remain unchanged in contrast to the perturbation (3.2.4). Another advantage of the method [86] is that it needs to solve the (typically) big LP relaxation of the original problem only once, whereas our method does this iteratively, which makes it often slower than CombiLP.

One other possible application scenario which is possible with our method but not with CombiLP [86] is the following: Assume we want to solve an extremely big inference problem, one that does not fit even into memory. To do this, choose a subset $A \subsetneq \mathcal{V}$ of nodes of the graphical model, solve the inference problem on the induced subgraph $G(A)$ with some boundary conditions, and find a partially optimal labeling on it. This is akin to the windowing technique of [93]. By doing so for an overlapping set of subgraphs, one may try to find a labeling for the overall problem on G.

The major differences between CombiLP [86] and our method are summarised in Table 3.2.

3.4 Largest Persistent Labeling

Let $A^0 \subseteq \mathcal{V}$ and $\mu^0 \in \Lambda_{A^0}$ be defined as in Algorithm 2. Subsets $A \subset A^0$ which fulfill the conditions of Corollary 3.2.1 taken with labelings $\mu^0|_A$ can be partially ordered with respect to inclusion $\subset$ of their domains. In this section we will show that the following holds:

- There is a largest set among those, for which there exists a *unique* persistent labeling fufilling the conditions of Corollary 3.2.1.

- Algorithm 2 finds this largest set.

This will imply that Algorithm 2 cannot be improved upon with regard to the criterion in Corollary 3.2.1.

Definition 3.4.1 (Strong Persistency)**.** *A labeling $x^* \in X_A$ is called* strongly persistent *on A, if x^* is the* unique *labeling on A fulfilling the conditions of Theorem 3.2.1.*

Lemma 3.4.1. *Let $x^* \in X_A$ be strongly persistent. Then for any optimal solution x of (2.2.2) we have $x^* = x_{|A}$.*

Proof. This follows from Inequality (3.2.13) being strict in this case. □

Theorem 3.4.1 (Largest persistent labeling)**.** *Algorithm 2 finds a superset A^* of the largest set $A^*_{strong} \subseteq A^* \subset \mathcal{V}$ of strongly persistent variables identifiable by the criterion in Corollary 3.2.1.*

To prove the theorem we need the following technical lemma.

Lemma 3.4.2. *Let $A \subset B \subset \mathcal{V}$ be two subsets of $\mathcal{V}$ and $\mu^A \in \Lambda_A$ marginals on A and $x^A \in X_A$ a labeling fulfilling the conditions of Corollary 3.2.1 uniquely (i.e. x^A is strongly persistent). Let $y^B \in X_B$ be a test labeling such that $y^B|_A = x^A$.*

Then for all marginals $\mu^* \in \operatorname{argmin}_{\mu \in \Lambda_B} \hat{E}_{B,y^B}(\mu)$ on B it holds that $\mu^*_v(x^A_v) = 1$ $\forall v \in A$.

Proof. Similar to the proof of Theorem 3.2.1. Replace $\mathcal{V}$ by B. □

Proof of Theorem 3.4.1. We will use the notation from Algorithm 2. It will be enough to show that for every $\overline{A} \subseteq \mathcal{V}$ such that there exists a strongly persistent labeling $\overline{x} \in X_{\overline{A}}$ we have $\overline{A} \subseteq A^t$ in each iteration of Algorithm 2 and furthermore $\overline{x}_v = x^t_v$ for all $v \in \mathcal{V}_{\overline{A}}$. Hence the union of sets A'_{strong}, for which a strongly persistent labeling exists which fulfills the conditions of Corollary 3.2.1, is a subset of A^t $\forall t$. Also by Lemma 3.4.1 the associated strongly persistent labelings agree where they overlap, hence we are done.

For $t = 0$ apply Lemma 3.4.2 with $A := \overline{A}$ and $B := A^0 (= \mathcal{V})$. Condition $\overline{x} = y^B|_{\overline{A}}$ in Lemma 3.4.2 is assured by Corollary 3.2.1. Hence, Lemma 3.4.2 ensures that for all $\mu^0 \in \operatorname{argmin}_{\mu \in \Lambda_\mathcal{V}} E(\mu)$ it holds that $\mu^0_v(\overline{x}_v) = 1$ for all $v \in \overline{A}$.

Now assume the claim to hold for iteration $t-1$. We need to show that it also holds for t. For this invoke Lemma 3.4.2 with $A := \overline{A}$, $B := A^{t-1}$ and $y^B := x^{t-1}$. The conditions of Lemma 3.4.2 hold by assumption on $t-1$. Lemma 3.4.2 now ensures that for all $\mu^t \in \operatorname{argmin}_{\mu \in \Lambda_{A^{t-1}}} \hat{E}_{A^{t-1},x^{t-1}}(\mu)$ there holds $\mu^t(x^A_v) = 1$ $\forall v \in A$. □

From the proof of Theorem 3.4.1 we can directly conclude the existence and uniqueness of a largest strongly persistent labeling identifiable by Corollary 3.2.1 and a set supporting it.

Corollary 3.4.1. *There exists a unique largest set A^*_{strong}, for which there exists a strongly persistent labeling identifiable by Corollary 3.2.1.*

Also exactly the largest strongly persistent labeling identifiable by Corollary 3.2.1 can be found under a mild uniqueness assumption.

Corollary 3.4.2. *If there is a* unique *solution of* $\min_{\mu\in\Lambda_{A^t}} \hat{E}_{A^t,x^t}(\mu)$ *for all* $t = 0,\dots$ *obtained during the iterations of Algorithm 2, then Algorithm 2 finds the largest subset of persistent variables identifiable by the sufficient partial optimality criterion in Corollary 3.2.1.*

Remark 3.4.1. Above we showed that Algorithm 2 will find a persistent labeling which contains the largest strongly persistent one identifiably by Corollary 3.2.1. The two may differ when the optimization problems solved in the course of Algorithm 2 have multiple optima. The simplest example of such a situation occurs if the relaxation $\min_{\mu\in\Lambda_{\mathcal{V}}} E_{\mathcal{V}}(\mu)$ is tight, but has several integer solutions. Any convex combination of these solutions will form a non-integral solution, hence the strongly persistent labeling is defined on a smaller set than any integral solution of $\min_{\mu\in\Lambda_{\mathcal{V}}} E_{\mathcal{V}}(\mu)$, which is non-strongly persistent. Note however that a labeling obtained by Algorithm 2, also when it is not strongly persistent, comes from *one* globally optimal labeling, i.e. it can be completed to a globally optimal labeling by solving for the remaining variables.

3.5 Reparametrization and Optimality of the Method

The boundary potentials (3.2.3) and hence the persistency approach described above are dependent on reparametrization, see Section 2.2. Below, we will study how to choose an optimal reparametrization θ^{ϕ} for the partial optimality problem.

3.5.1 Optimal Reparametrization

In the context of partial optimality, we call a reparametrization *optimal*, if it gives the largest persistent set.

The only coordinates of the reparametrization vector ϕ, which can potentially influence the solution of the test problem (3.2.5) are $\phi_{v,u}(x_v)$, $u \in \partial\mathcal{V}_A$, $uv \in \partial\mathcal{E}_A$. Reparametrization $\phi_{v,u}(x_v)$, $v \in A$ "inside" A does not influence the solution, because it does not change the augmented energy $\hat{E}_{A,\cdot}$ of any labeling. Similarly, the reparametrization $\phi_{u,v}(x_u)$, $u, v \notin A$ "outside" A does not influence it, because the optimization is performed over A only.

Considering the reparametrized potentials θ^{ϕ} and subtracting $\max_{x_v\in X_v} \theta_{uv}(y_u, x_v)$ in (3.2.3) the boundary potentials $\hat{\theta^{\phi}}_{uv,y_u}(x_u)$ can be equivalently exchanged with

$$\begin{cases} 0, & y_u = x_u \\ \min\limits_{x_v\in X_v} \theta^{\phi}_{uv}(x_u, x_v) - \max\limits_{x_v\in X_v} \theta^{\phi}_{uv}(y_u, x_v), & y_u \neq x_u \end{cases} . \tag{3.5.1}$$

It means that the labelings x not coinciding with y on $\partial\mathcal{V}_A$ will be "encouraged" with (typically negative) value $\Delta^{\phi}_{uv}(x_u) := \min\limits_{x_v\in X_v} \theta^{\phi}_{uv}(x_u, x_v) - \max\limits_{x_v\in X_v} \theta^{\phi}_{uv}(y_u, x_v)$. Intuitively clear that the bigger $\Delta^{\phi}_{uv}(x_u)$ is, the better the proposal labeling $y|_A$ comparing to $x|_A \neq y|_A$ is and hence the greater the found persistent set A^* returned by Algorithm 2 would be. We will prove correctness of this intuition formally, but first let us find *the maximal possible* value of $\Delta^{\phi}_{uv}(x_u)$ w.r.t. the reparametrization

ϕ, where we consider as non-zero only coordinates $\phi_{v,u}(x_v)$, $u \in \partial\mathcal{V}_A$, $uv \in \partial\mathcal{E}_A$, $x_v \in X_v$.

Clearly

$$\Delta^{\phi}_{uv}(x_u) \leq \min_{x_v \in X_v} (\theta^{\phi}_{uv}(x_u, x_v) - \theta^{\phi}_{uv}(y_u, x_v)) \tag{3.5.2}$$

$$= \min_{x_v \in X_v} (\theta_{uv}(x_u, x_v) + \phi_{v,u}(x_v) - \theta_{uv}(y_u, x_v) - \phi_{v,u}(x_v)) \tag{3.5.3}$$

$$= \min_{x_v \in X_v} (\theta_{uv}(x_u, x_v) - \theta_{uv}(y_u, x_v)) \,, \tag{3.5.4}$$

hence, the right-hand-side of this inequality does not depend on the reparametrization, whereas the left-hand-side does. There is indeed such a reparametrization that turns the inequality (3.5.2) into equality and in this way guarantees the largest possible values of $\Delta^{\phi}_{uv}(x_u)$ for all x_u. This, as we show below, *optimal* reparametrization is defined as

$$\phi_{u,v}(x_v) = -\theta_{uv}(y_u, x_v) \,, \tag{3.5.5}$$

which can be seen when plugging (3.5.5) into (3.5.1).

Moreover, since as we mentioned above the reparametrization "outside" and "inside" A^t does not influence the criterion (3.2.3), we can construct a single, equal for all iterations of Algorithm 2 optimal reparametrization ψ according to the rule (3.5.5) as

$$\psi_{u,v}(x_v) = -\theta_{uv}(y_u, x_v), \; u \in \mathcal{V}, \; uv \in \mathcal{E} \,, \tag{3.5.6}$$

where y is arbitrarily extended from A^0 to $\mathcal{V}$. Now we are ready to formulate our main result related to the reparametrization.

Let us denote by $\hat{E}^{\phi}_{A,y}$ the energy with boundary labeling defined as in Definition 3.2.3 w.r.t. the potentials θ^{ϕ}. Then for the reparametrization ψ defined as in (3.5.6) there holds

Lemma 3.5.1. *From*

$$\delta(y) \in \arg\min_{\mu \in \Lambda_A} \hat{E}_{A,y}(\mu) \tag{3.5.7}$$

follows $\delta(y) \in \arg\min_{\mu \in \Lambda_A} \hat{E}^{\psi}_{A,y}(\mu)$*, which means: if* y *satisfies the persistency criterion of Corollary 3.2.1 w.r.t. potentials* θ *then it satisfies it w.r.t. the reparametrized potentials* θ^{ψ}*.*

Proof. From (3.5.2) and (3.5.7) it follows that for all $uv \in \mathcal{E}_A$, $x_u \in X_u$ there holds $\hat{\theta}^{\psi}_{uv,y}(x_u) - \hat{\theta}^{\psi}_{uv,y}(y_u) \geq \hat{\theta}_{uv,y}(x_u) - \hat{\theta}_{uv,y}(y_u)$ and hence

$$\hat{E}^{\psi}_{A,y}(\mu) \quad - \quad \hat{E}^{\psi}_{A,y}(y) \quad \overset{(3.5.2)}{\geq} \quad \hat{E}_{A,y}(\mu) \quad - \quad \hat{E}_{A,y}(y) \quad \geq \quad 0 \tag{3.5.8}$$

for all $\mu \in \Lambda_A$. Thus $\hat{E}^{\psi}_{A,y}(y) \leq \hat{E}^{\psi}_{A,y}(\mu)$, which proves the statement of the lemma. □

Remark 3.5.1. Lemma 3.5.1 holds for *any* polytope containing all integer solutions, i.e. $\Lambda_A \supseteq \mathcal{M}_A$ and hence it holds also when $\Lambda_A = \mathcal{M}_A$. In this case it corresponds

to the non-relaxed persistency criterion provided by Theorem 3.2.1.

Let now $A_y^{\phi,*}$ be *the largest* set containing all strongly persistent variables satisfying Corollary 3.2.1 w.r.t. the reparametrized potentials θ^ϕ and test labeling $y \in X_\mathcal{V}$. Let also A_y^* correspond to the trivial reparametrization $\phi \equiv 0$.

Applying Lemma 3.5.1 to the set A_y^* leads to the following

Theorem 3.5.1. *For any test labeling $y \in X_\mathcal{V}$ there holds $A_y^* \subseteq A_y^{\psi,*}$.*

Proof. Same proof as in Lemma 3.5.1 applied to A_y^*. □

Remark 3.5.2. For Potts models, where $\theta_{uv}(x_u, x_v) = \begin{cases} 0, & x_u = x_v \\ \alpha, & x_u \neq x_v \end{cases}$, the inequality (3.5.2) holds as equality also for trivial reparametrization $\phi_{v,u}(x_v) = 0\ \forall u, v \in \mathcal{V}$, $uv \in \mathcal{E}$, $x_v \in X_v$. For such models Algorithm 2 with trivial reparametrization delivers the same persistent set as with the optimal one (3.5.6).

3.5.2 Optimality of the Method

Theorem 3.4.1 proves optimality of Algorithm 2 w.r.t. the formulated persistency criterion provided by Theorem 3.2.1. However it does not prove optimality of the method with respect to other possible criteria and hence does not guarantee its superiority over other partial optimality techniques. There is however a recent study [93], which provides such *an optimal relaxed persistency criterion* covering *all* existing methods. In what follows we will introduce key notions from [93] and show that our persistency criterion coincides with *the optimal* one provided in [93] for a certain class of persistency methods, those providing only node-persistency, i.e. either eliminating all labels except one in a given node or not eliminating any.

Definition 3.5.1. *A mapping $p\colon X_\mathcal{V} \to X_\mathcal{V}$ is called* (strictly) improving *for the potentials θ if it is idempotent ($p(p(x)) = p(x)$) and for all $x \in X_\mathcal{V}$ such that $p(x) \neq x$ there holds $\langle \theta, \delta(p(x)) \rangle \leq \langle \theta, \delta(x) \rangle$ (resp. $\langle \theta, \delta(p(x)) \rangle < \langle \theta, \delta(x) \rangle$).*

Following [93] we consider only *node-wise* maps of the form $p(x)_v = p_v(x_v)$, where $p_v\colon X_v \to X_v$ are idempotent, i.e. $p_v(p_v(x_v)) = p_v(x_v)$ for all $x_v \in X_v$. This class is already general enough to include nearly all existing techniques.

Improving mappings defines persistency due to the following proposition:

Proposition 3.5.1 (Stat.1[93])**.** *Let p be an improving mapping. Then there exists an optimal solution x of (2.2.2) such that for all $v \in \mathcal{V}$ from $p_v(i) \neq i$ follows $x_v \neq i$. In case p is strictly improving this holds for* any *optimal solution.*

For an idempotent mapping p a linear mapping $P\colon \mathbb{R}^\Sigma \to \mathbb{R}^\Sigma$ satisfying $\delta(p(x)) = P\delta(x)$ for all $x \in X_\mathcal{V}$ is called its *linear extension*. A particular linear extension denoted as $[p]$ is defined as follows. For each p_v we define the matrix $P_v \in \mathbb{R}^{X_v \times X_v}$

by $P_{v,ii'} = \begin{cases} 1, & p_v(i') = i \\ 0, & p_v(i') \neq i \end{cases}$. The linear extension $P = [p]$ is given by

$$\begin{aligned} (P\mu)_v &= \sum\nolimits_{i' \in X_v} P_{v,ii'} \mu_v(i') = P_v \mu_v; \\ (P\mu)_{uv} &= P_u \mu_{uv} P_v^\top . \end{aligned} \tag{3.5.9}$$

Denote by I the identity matrix. From Definition 3.5.1 follows that p is improving iff the value of

$$\min_{x \in X_\mathcal{V}} \langle \theta, (I - [p])\delta(x) \rangle \equiv \min_{x \in X_\mathcal{V}} \langle (I - [p])^\top \theta, \delta(x) \rangle \equiv \min_{\mu \in \mathcal{M}_\mathcal{V}} \langle (I - [p])^\top \theta, \mu \rangle \tag{3.5.10}$$

is zero. If additionally $p(x) = x$ for all minimizers of (3.5.10) then the mapping p is strictly improving.

Problem (3.5.10) is of the same form as energy minimization (2.2.2) and is therefore as hard as Problem (3.5.10). Its relaxation is obtained by letting μ vary in the local polytope $\Lambda_\mathcal{V} \subset \mathbb{R}^\Sigma$, an outer approximation to $\mathcal{M}_\mathcal{V}$.

Definition 3.5.2. *Mapping $p\colon X_\mathcal{V} \to X_\mathcal{V}$ is $\Lambda_\mathcal{V}$-*improving for potentials $\theta \in \mathbb{R}^\Sigma$ *if*

$$\min_{\mu \in \Lambda_\mathcal{V}} \langle (I - [p])^\top \theta, \mu \rangle = 0 \,. \tag{3.5.11}$$

If additionally $[p]\mu = \mu$ for all minimizers μ of (3.5.11) then p is strictly $\Lambda_\mathcal{V}$-improving.

Compared to (3.5.10), only the polytope was changed to $\Lambda_\mathcal{V} \supset \mathcal{M}_\mathcal{V}$. This implies the following simple fact:

Proposition 3.5.2. *If mapping p is (strictly) $\Lambda_\mathcal{V}$-improving then it is (strictly) improving.*

The method presented in this work can be interpreted as considering *all-to-one* node-wise mappings p having the form

$$p_v(i) = \begin{cases} y_v, & \text{if } v \in A \\ i, & \text{if } v \notin A \end{cases} \tag{3.5.12}$$

for a fixed *test labeling* y. All labels in the nodes $v \in A \subset \mathcal{V}$ are mapped to y_v. Among all all-to-one (strictly) $\Lambda_\mathcal{V}$-improving mappings the one with the largest set A will be called *maximal*.

Corollary 3.2.1 determines $\Lambda_\mathcal{V}$-improving mappings, as stated by

Lemma 3.5.2. *The relaxed persistency criterion provided by Corollary 3.2.1 with the reparametrization given by (3.5.6) is equivalent to Definition 3.5.2 with the improving mapping p defined as in (3.5.12) for a given test labeling y.*

Proof. For future references we write down potentials θ^ψ with ψ defined by (3.5.6)

explicitly:

$$\theta_u^{\psi}(x_u) = \theta_u(x_u) + \sum_{v \in \mathrm{nb}(u)} \theta_{uv}(x_u, y_v)\,, \tag{3.5.13}$$
$$\theta_{uv}^{\psi}(x_u, x_v) = \theta_{uv}(x_u, x_v) - \theta_{uv}(x_u, y_v) - \theta_{uv}(y_u, x_v)\,.$$

In what follows we will show that the criteria (3.2.17) and (3.5.11) coincide. Both of them represent the local polytope relaxation of specially constructed energy minimization problems. To prove that the relaxations coincide it is sufficient to prove that the non-relaxed energies are equal.

First we write down the non-relaxed test problem (3.2.5) with potentials θ^{ψ} as

$$\arg\min_{x \in X_{\mathcal{V}}} \sum_{v \in \mathcal{V}} \beta_v(x_v) + \sum_{uv \in \mathcal{E}} \beta_{uv}(x_u, x_v) + \sum_{uv \in \partial\mathcal{E}_A\colon\, u \in \partial\mathcal{V}_A} \hat{\theta}^{\psi}_{uv,y_u}(x_u) \tag{3.5.14}$$

with potentials β equal to θ^{ψ} on A and vanishing outside it, i.e.

$$\beta_u(x_u) = \begin{cases} \theta_u(x_u) + \sum\limits_{v \in \mathrm{nb}(u)} \theta_{uv}(x_u, y_v), & u \in A \\ 0, & u \in \mathcal{V} \backslash A \end{cases} \tag{3.5.15}$$

$$\beta_{uv}(x_u, x_v) \begin{cases} \theta_{uv}(x_u, x_v) - \theta_{uv}(x_u, y_v) - \theta_{uv}(y_u, x_v), & u, v \in A \\ 0, & \text{otherwise}\,. \end{cases} \tag{3.5.16}$$

Border potentials $\hat{\theta}^{\psi}$ for $uv \in \mathcal{E},\ u \in \mathcal{V}_A,\ v \in \mathcal{V}\backslash A$ and $x_u \neq y_u$ read:

$$\hat{\theta}^{\psi}_{uv,y_u}(x_u) = \min_{x_v \in X_v} \theta^{\psi}_{uv}(x_u, x_v) \tag{3.5.17}$$
$$= \min_{x_v \in X_v} \left(\theta_{uv}(x_u, x_v) - \theta_{uv}(x_u, y_v) - \theta_{uv}(y_u, x_v)\right) \tag{3.5.18}$$
$$= -\,\theta_{uv}(x_u, y_v) + \min_{x_v \in X_v} \left(\theta_{uv}(x_u, x_v) - \theta_{uv}(y_u, x_v)\right); \tag{3.5.19}$$

for $x_u = y_u$:

$$\hat{\theta}^{\psi}_{uv,y_u}(y_u) = \max_{x_v \in X_v} \theta^{\psi}_{uv}(y_u, x_v) \tag{3.5.20}$$
$$= \max_{x_v \in X_v} \left(\theta_{uv}(y_u, x_v) - \theta_{uv}(y_u, y_v) - \theta_{uv}(y_u, x_v)\right) \tag{3.5.21}$$
$$= -\,\theta_{uv}(y_u, y_v)\,. \tag{3.5.22}$$

Note that (3.5.17) turns into (3.5.20) when $x_u = y_u$, hence it is sufficient to use only expression (3.5.17).

The non-relaxed version of condition (3.5.11) defining $\Lambda_{\mathcal{V}}$-improving all-to-one mapping with the labeling proposal y can be formulated as checking whether

$$y \in \arg\min_{x \in X_{\mathcal{V}}} \sum_{v \in \mathcal{V}} \gamma_v(x_v) + \sum_{uv \in \mathcal{E}} \gamma_{uv}(x_u, x_v) + \sum_{u \in \partial\mathcal{E}_A} \hat{\gamma}_{uv,y_u}(x_u) \tag{3.5.23}$$

with potentials γ defined as:

$$\gamma_u(x_u) = \begin{cases} \theta_u(x_u) - \theta_u(y_u), & u \in A \\ 0, & u \in \mathcal{V} \backslash A \end{cases} \tag{3.5.24}$$

$$\gamma_{uv}(x_u, x_v) = \begin{cases} \theta_{uv}(x_u, x_v) - \theta_{uv}(y_u, y_v), & u, v \in A \\ 0, & \text{otherwise}\,. \end{cases} \tag{3.5.25}$$

and the border term

$$\hat{\gamma}_{uv,y_u}(x_u) = \min_{x_v \in X_v} \left(\theta_{uv}(x_u, x_v) - \theta_{uv}(y_u, x_v)\right). \tag{3.5.26}$$

Comparing (3.5.24), (3.5.25) and (3.5.26) to (3.5.15), (3.5.16) and (3.5.17) respectively it can be seen that they can be transformed to each other by several operations, which *equally* change energies of all labelings and thus do not influence the criterions provided by Theorem 3.2.1 and [93, eq.(14)]. These operations are:

1. Subtract $\theta_u(y_u)$ from $\beta_u(x_u)$ for all $u \in \mathcal{V}_A$, $x_u \in X_u$.
2. Subtract $\theta_{uv}(y_u, y_v)$ from $\beta_{uv}(x_u, x_v)$ for all $uv \in \mathcal{E}_A$, $(x_u, x_v) \in X_u \times X_v$.
3. Reparametrize β with the reparametrization vector ϕ defined as

$$\phi_{u,v}(x_u) = \begin{cases} -\theta_{uv}(x_u, y_v), & u \in A \\ 0, & u \in \mathcal{V} \backslash A\,. \end{cases} \tag{3.5.27}$$

□

The following theorem states that our method provably delivers the best results among the methods providing node-persistency:

Theorem 3.5.2. *Under conditions of Corollary 3.4.2, Algorithm 2 with the reparametrizations given by (3.5.6) finds the maximal strict $\Lambda_{\mathcal{V}}$-improving all-to-one mapping for a given proposal labeling x^0.*

Proof. Under condition of Corollary 3.4.2 (i.e. when on each iteration there is a unique solution μ^t) Lemma 3.5.2 guarantees equivalence of our criterion (Corollary 3.2.1 with reparametrization ψ) to Definition 3.5.2 for the strict $\Lambda_{\mathcal{V}}$-improving all-to-one mapping. Theorem 3.4.1 states that Algorithm 2 delivers the largest set A^* satisfying this criterion, which in turn proves the theorem. □

3.6 Extensions

3.6.1 Higher Order Models

Assume now we are not in the pairwise case anymore but have an energy minimization problem over a *hyper*graph $G = (\mathcal{V}, \mathcal{E})$ with $\mathcal{E} \subset \mathcal{P}(\mathcal{V})$ a set of subsets of $\mathcal{V}$:

$$\min_{x \in X_{\mathcal{V}}} E_{\mathcal{V}}(x) := \sum_{e \in \mathcal{E}} \theta_e(x_e)\,. \tag{3.6.1}$$

All definitions, our persistency criterion and Algorithm 2 admit a straightforward generalization. Analoguously to Definition 3.2.1 define for a subset of nodes $A \subset \mathcal{V}$ the boundary nodes as

$$\partial\mathcal{V}_A := \{u \in A \colon \exists v \in \mathcal{V}\backslash A, \exists e \in \mathcal{E} \text{ s.t. } u, v \in e\} \tag{3.6.2}$$

and the boundary edges as

$$\partial\mathcal{E}_A := \{e \in \mathcal{E} \colon \exists u \in A, \exists v \in \mathcal{V}\backslash A \text{ s.t. } u, v \in e\}\,. \tag{3.6.3}$$

The equivalent of boundary potential in Definition 3.2.3 for $e \in \partial\mathcal{E}_A$ is

$$\hat{\theta}_{e,y}(x) := \begin{cases} \max\limits_{\tilde{x} \in X_e \colon \tilde{x}_{|A\cap e} = x_{|A\cap e}} \theta_e(\tilde{x}), & x_{|A\cap e} = y_{|A\cap e} \\ \min\limits_{\tilde{x} \in X_e \colon \tilde{x}_{|A\cap e} = x_{|A\cap e}} \theta_e(\tilde{x}), & x_{|A\cap e} \neq y_{|A\cap e} \end{cases}\,. \tag{3.6.4}$$

Now Theorem 3.2.1, Corollary 3.2.1 and Algorithm 2 can be directly translated to the higher order case.

3.6.2 Tighter Relaxations

Essentially, Algorithm 2 can be applied also to tighter relaxations than Λ_A, e.g. when one includes cycle inequalities [96]. One merely has to replace the local polytope Λ_A for $A \subset \mathcal{V}$ by the tighter feasible convex set:

Proposition 3.6.1. *Let the polytopes $\tilde{\Lambda}_A \supseteq \mathcal{M}_A$ satisfy $\tilde{\Lambda}_A \subseteq \Lambda_A \ \forall A \subseteq \mathcal{V}$. Use $\tilde{\Lambda}_{A^t}$ in place of Λ_{A^t} in Algorithm 2 and let $\tilde{A}^*$ be the corresponding persistent set returned by the modified algorithm. Let $A^*_{strong} \subseteq A^*$ be the largest subset of strongly persistent variables identifiable by Corollary 3.2.1 subject to the relaxations $\tilde{\Lambda}_A$ and Λ_A. Then $A^*_{strong} \subseteq \tilde{A}^*_{strong}$.*

Remark 3.6.1. For approximate dual solvers for tighter relaxations like [97, 99] there are analogues of strict arc-consistency, hence these are also consistency-ascertaining solvers as in Definition 3.3.1 and we can also use these algorithms in Algorithm 2 with the obvious modifications.

Optimal reparametrization for tighter relaxations and higher order models is beyond the scope of this work.

3.7 Numerical Experiments

We tested our approach with initial and optimal reparametrizations (described in Section 3.5) on several datasets from different computer vision and machine learning benchmarks, 47 problem instances overall, see Table 3.3. We describe each dataset and the corresponding experiments in detail below.

3.7.1 Competing methods

We compared our method to **MQPBO** [47, 94], **Kovtun**'s method [57], Generalized Roof Duality (**GRD**) by Kahl and Strandmark [39], Fix et al's [26] and Ishikawa's Higer Order Clique Reduction (**HOCR**) [37] algorithms. For the first two methods we used our own implementation, and for the other the freely available code of Strandmark [100]. We were unable to compare to the method of Windheuser et al. [115], because the authors do not give a description for implementing their method in the higher order case and only provide experimental evaluation for problems with pairwise potentials, where their method coincides with MQPBO [47].

3.7.2 Implementation details

We employed TRWS as an approximate solver for Algorithm 2 and strong tree agreement as a consistency mapping (see Proposition 3.3.1) for most of the pairwise problems. We stop TRWS once it has either arrived at (i) tree-agreement; (ii) a small duality gap of 10^{-5}; (iii) when number of nodes with tree agreement did not increase over the last 100 iterations or (iv) overall 1500 iterations. For the higher-order models `protein-interaction`, `cell-tracking` and `geo-surf` we employed CPLEX [36] as an exact linear programming solver. We have run Algorithm 2 with boundary potentials computed as in (3.2.3) for all problems and with boundary potentials computed with the optimal reparametrization as in (3.5.1) for the pairwise problems.

3.7.3 Datasets and Evaluation

We give a brief characterization of all datasets below and in Table 3.3. We also report the obtained total percentage of persistent variables of our and competing methods in Table 3.4. The percentage of partial optimality is computed as follows: Suppose we have found a persistent labeling on set $A \subset \mathcal{V}$. Then the percentage is $1 - \frac{\sum_{u \notin A} \log |X_u|}{\sum_{u \in \mathbb{V}} \log |X_u|}$. Note that by this formulation we take into account the size of the label space for each node. For an uniform label space the above formula equals $\frac{|A|}{|\mathcal{V}|}$. The latter measure was used in [102].

The problem instances `teddy`, `venus`, `family`, `pano`, `Potts` and `geo-surf` were made available by [40], while the datasets `side-chain` and `protein-interaction` were made available by [2].

The problem instances `teddy` and `venus` come from **the disparity estimation for stereo vision** [105]. None of the competing approaches was able to find even a single persistent variable for these datasets, presumably because of the large number of labels, whereas we labeled over one third of them as persistent in `teddy`, though none in `venus`.

Instances named `pano` and `family` come from the **photomontage** dataset [105]. These problems have more complicated pairwise potentials than the disparity estimation problems, but less labels. For both datasets we found significantly more

Experiment	#Instances	#Labels	#Vertices	Order
teddy	1	60	168749	2
venus	1	20	166221	2
family	1	5	425631	2
pano	1	7	514079	2
Potts	12	≤12	≤424720	2
side-chain	21	≤483	≤1971	2
protein -interaction	8	2	14440	3
cell-tracking	1	2	41134	9
geo-surf	1	7	837	3

Table 3.3 - Short summary of experiments.

persistent variables than MQPBO, in particular, we were able to label more than a third of the variables in `pano`.

We also chose 12 relatively big energy minimization problems with grid structure and **Potts** interaction terms. The underlying application is a color segmentation problem previously considered in [101]. Our general approach reproduces results of [101] for the specific Potts model.

We considered also `side-chain` prediction problems in **protein folding** [116]. The datasets consist of pairwise graphical models with $32-1971$ variables and $2-483$ labels. The problems with fewer variables are densely connected and have very big label spaces, while the larger ones are less densely connected and have label space up to 81 variables.

The `protein interaction` models [38] aim to find the subset of proteins, which interact with each other. Roof-duality based methods, i.e. Fix et at, GRD, HOCR [26, 37, 39] gave around a quarter of persistent labels. This is the only dataset where our methods gives worse results. Note that for higher-order models we do not provide an optimal reparametrization and hence our method is not provably better then the competitors. We consider this as a direction for future work.

The **cell tracking** problem consists of a binary higher order graphical model [44]. Given a sequence of microscopy images of a growing organism, the aim is to find the lineage tree of all cells. For implementation reasons we were not able to solve `cell-tracking` dataset with Ishikawa's [37] method. However Fix [26] reports that his method outperforms Ishikawa's method [37]. Other methods are not applicable even theoretically.

Last, we took the higher order multi-label **geometric surface labeling problems** (denoted as `geo-surf` in Table 3.3) from [35]. The only instance having an integrality gap has 968 variables with 7 labels each and has ternary terms. Note that MQPBO cannot handle ternary terms, Fix et al's [26] Ishikawa's [37] methods and the generalized roof duality method by Strandmark and Kahl [39] cannot handle more than 2 labels. Hence we report our results without comparison.

Exemplary pictures comparing the pixels optimally labelled between Kovtuns's

Experiment	MQPBO [47]	Kovtun [57]	GRD [39]	Fix [26]	HOCR [37]	Ours original [102]	Ours optimal
teddy	0	†	†	†	†	**0.3820**	**0.3820**
venus	0	†	†	†	†	0	0
family	0.0432	†	†	†	†	0.0044	**0.0611**
pano	0.1247	†	†	†	†	0.2755	**0.3893**
Potts	0.1839	0.7475	†	†	†	**0.9220**	**0.9220**
side-chain	0.0247	†	†	†	†	0.1747	**0.2558**
protein -interaction	†	†	**0.2603**	0.2545	0.2545	0.0008	†
cell-tracking	†	†	†	0.1771	†	**0.2966**	†
geo-surf	†	†	†	†	†	**0.0743**	†

Table 3.4 - Percentage of persistent variables obtained by methods [47],[57],[39],[26],[37] and our methods with boundary potentials computed as in (3.2.4) (Ours original) and as in (3.5.1) (Ours optimal). Notation † means inapplicability of the method.

method [57] and our method for some Potts-models can be seen in Figure 3.4.

3.7.4 Runtime

The runtime of our algorithm mainly depends on the speed of the underlying solver for the local polytope relaxation. Currently there seems to be no general rule regarding the runtime of our algorithm, neither in the number of Algorithm 2-iterations nor in the number of TRWS [48]-iterations. We show three iteration counts for instances of the Potts dataset in Figure 3.5.

3.8 Conclusion

We have presented a novel method for finding persistent variables for undirected graphical models. Empirically it outperforms all tested approaches with respect to the number of persistent variables found on every single dataset. Our method is general: it can be applied to graphical models of arbitrary order and type of potentials. Moreover, there is no fixed choice of convex relaxation for the energy minimization problem and also approximate solvers for these relaxations can be employed in our approach.

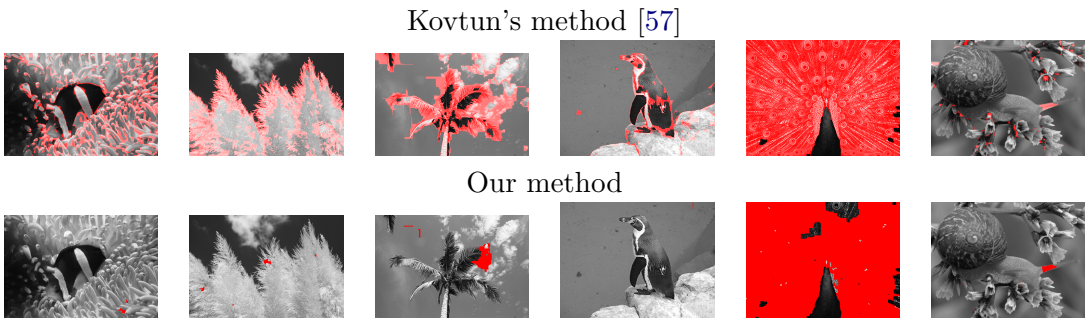


Figure 3.4 - Comparison between Kovtun's method [57] and our method. The red area denotes pixels which could not be labelled persistently. Contrary to ours the Kovtun's method allows to eliminate separate labels, which is denoted by different intensity of the red color: the more intensive is red, the less labels were eliminated.

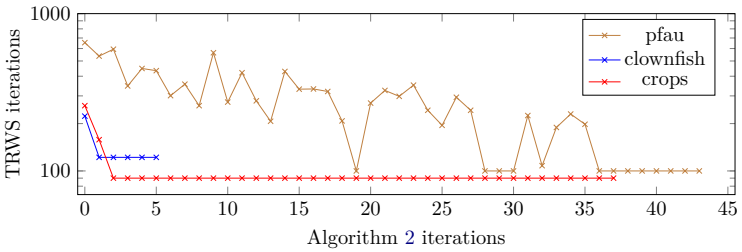


Figure 3.5 - Iterations needed by TRWS [48] in Algorithm 2 for three instances from the `Potts` dataset.

4 Continuous Variational Image Labeling

In this chapter we will introduce two key variational imaging problems from a continuous perspective. These problems will require the introduction of the space of functions of bounded variation. Finally, we will discuss the relationship to MRFs and proximal algorithms for solving convex relaxations.

We assume throughout this chapter that $\Omega \subset \mathbb{R}^k$ is the domain on which the problems we are interested in live. Associated to Ω is the set of all measurable subsets of Ω formed by the Borel σ-algebra. In typical imaging applications $\Omega = [0,1]^2$.

The two key problems that we will treat are

1. The *minimal partition problem*

$$\min_{(\Omega_1,\dots,\Omega_k)} \left\{ \sum_{i=1}^{k} \int_{\Omega_i} d_i(x)\ dx + \mathrm{Per}(\Omega_i) \right\}, \tag{4.0.1}$$

   where d_i are cost functions denoting which label $x \in \Omega$ shall take, $\mathrm{Per}(\Omega_i)$ is the perimeter of Ω_i and $(\Omega_1,\dots,\Omega_k)$ partition Ω [15, 19, 61, 73], and

2. the *real-valued labeling problem*

$$\min_{u:\Omega\to\mathbb{R}} \left\{ \int_{\Omega} f(u(x),x)\ dx + |Du| \right\}, \tag{4.0.2}$$

   where f is a cost function describing which value $u(x)$ shall take in $x \in \Omega$ [72].

We will study the associated function spaces in which the above problems have minimizers and are well-defined. Also we will study reformulation and convex relaxation techniques which make the above problems solvable.

This chapter contains a short summary of necessary results from [6] and [72].

4.1 Functions of Bounded Variation

Both of the key problems above will be stated for functions of bounded variation, which we introduce below.

Definition 4.1.1 (Functions of Bounded Variation)**.** *Let $u \in L_1(\Omega)$. We say that u is a* function of bounded variation in Ω *if the distributional derivatives of u are representable by a finite Radon measure in Ω, i.e. if*

$$\int_{\Omega} u \frac{\partial \phi}{\partial x_i}\ dx = -\int_{\Omega} \phi\ dD_i u \quad \forall \phi \in C_c^1(\Omega), \quad i = 1,\dots,k\,. \tag{4.1.1}$$

The space of all functions of bounded variation is denoted by $\mathrm{BV}(\Omega)$.

When functions are smooth, their associated level sets are smooth as well. For functions of bounded variation this is not true anymore. Their level sets are however of finite perimeter, when we extend the definition suitably.

Definition 4.1.2 (Perimeter)**.** *Let $A \subset \Omega$ be a measurable subset. The perimeter of A in Ω is the variation of $\mathbb{1}_A$ in Ω, i.e.*

$$\operatorname{Per}(A, \Omega) := \sup \left\{ \int_A \operatorname{div} \phi \; dx \; : \phi \in C_c^1(\Omega)^k, \|\phi\|_\infty \leq 1 \right\}. \tag{4.1.2}$$

In the remainder, we will often write $\operatorname{Per}(A)$ instead of $\operatorname{Per}(A, \Omega)$, when Ω is clear from the context. For sets with C^1-boundary, the above definition gives the volume of the boundary. See [6, Section 3.3] for a detailed discussion.

The integral over the perimeter of all levelsets of a functions is also finite for functions of bounded variation. This is expressed by the famous Coarea formula.

Theorem 4.1.1 (Coarea Formula)**.** *If $u \in \mathrm{BV}(\Omega)$, the level set $\{u > t\}$ has finite perimeter in Ω for a.e. $t \in \mathbb{R}$ and*

$$\begin{aligned} |Du|(B) &= \int_{-\infty}^{\infty} |D\chi_{\{u>t\}}|(B) \; dt \\ Du(B) &= \int_{-\infty}^{\infty} D\chi_{\{u>t\}}(B) \; dt \,. \end{aligned} \tag{4.1.3}$$

As is common in the image analysis literature, we define

$$TV(u) = |Du|(\Omega) \tag{4.1.4}$$

to be the total variation. For a comprehensive treatment of functions of bounded variations, see [6].

4.2 Minimal Partition Problems and Perimeter

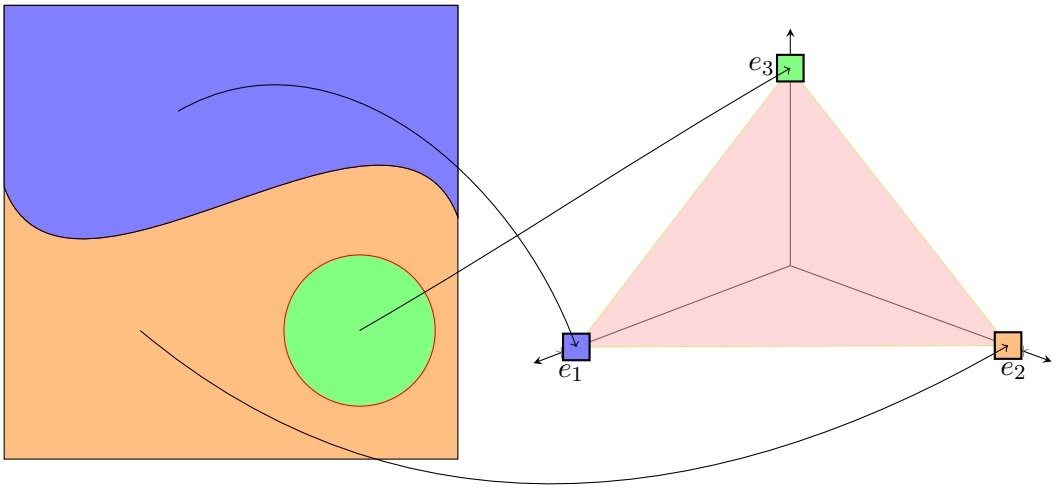
In segmentation problems we are interested in a partition of Ω.

Definition 4.2.1 (Partition)**.** *A partition of a set Ω is a tuple $(\Omega_1, \ldots, \Omega_k)$ of subsets of Ω such that $\Omega_i \cap \Omega_j = \varnothing$ for $i \neq j$ and $\Omega_1 \cup \ldots \cup \Omega_k = \Omega$.*

Usually, we want to find sets Ω_i forming a partition such that their shape is regular. A common way to accomplish this task is to penalize by the perimeter $\operatorname{Per}(\Omega_i, \Omega)$. The minimal partition problem is

Definition 4.2.2 (Minimal Partition Problem)**.** *Given $d_1, \ldots, c_k \in L_1(\Omega)$, the minimal partition problem is*

$$\begin{array}{ll} \min_{(\Omega_1,\ldots,\Omega_k)} & \sum_{i=1}^k \operatorname{Per}(\Omega_i) + \int_{\Omega_i} d_i(x) \; dx \\ s.t. & (\Omega_1, \ldots, \Omega_k) \text{ is a partition} \end{array} \tag{4.2.1}$$



Partition $(\Omega_1, \Omega_2, \Omega_3)$ of Ω Values taken by indicator function $u : \Omega \to \mathcal{E}_3$

Figure 4.1 - Lifted representation of an image

The above problem (4.2.1) weighs local data terms favoring a specific class against a geometric term favoring smoothness of the contour of the area occupied by the class. Problem (4.2.1) has the drawback that it entails minimization over sets, which is not amenable to algorithms. A common remedy is to study problem (4.2.1) in terms of labeling functions.

Definition 4.2.3 (Labeling functions)**.** *The space of labeling functions is*

$$\mathrm{BV}(\Omega, \mathcal{E}_k) = \left\{ u \in \mathrm{BV}(\Omega)^k \ : u(x) \in \mathcal{E}_k \ \textit{a.e.}\ x \in \Omega \right\}, \tag{4.2.2}$$

A partition $(\Omega_1, \ldots, \Omega_k)$ *corresponds uniquely to* $u \in \mathrm{BV}(\Omega, \mathcal{E}_k)$ *via the relation*

$$x \in \Omega_i \quad \Leftrightarrow \quad u(x) = e_i \,. \tag{4.2.3}$$

The relationship between partitions and labeling functions is illustrated in figure 4.1.

By the coarea formula we see that the level set of the i-th component is in $\mathrm{BV}(\Omega)$, if and only if set Ω_i is of finite perimeter. Hence (4.2.3) is well-defined. Moreover, we can reformulate the minimal partition problem (4.2.1) in terms of labeling functions.

Theorem 4.2.1. *The minimal partition problem* (4.2.1) *is equivalent to*

$$\min_{u \in \mathrm{BV}(\Omega, \mathcal{E}_k)} \sum_{i=1}^{k} \mathrm{TV}(u_i) + \int_\Omega d_i(x) u_i(x) \, dx \,. \tag{4.2.4}$$

Proof. Follows from Definition 4.2.3 and the Coarea formula (4.1.3). □

Unfortunately, $BV(\Omega, \mathcal{E}_k)$ is not a convex set and therefore problem (4.2.4) is not a convex problem anymore. Hence we let functions take values in the simplex Δ_k,

which is the convex hull of $\mathcal{E}_k$:

$$\mathrm{BV}(\Omega, \Delta_k) = \left\{ u \in \mathrm{BV}(\Omega)^k \ : \ u(x) \in \Delta_k \text{ a.e. } x \in \Omega \right\} , \tag{4.2.5}$$

The resulting relaxation is

$$\min_{u \in \mathrm{BV}(\Omega, \Delta_k)} \sum_{i=1}^{k} \mathrm{TV}(u_i) + \int_\Omega d_i(x) u_i(x) \ dx \, . \tag{4.2.6}$$

Note that the space $\mathrm{BV}(\Omega, \Delta_k)$ is convex and consequently (4.2.6) is convex also. In the case of $k = 2$, i.e. two labels, relaxation (4.2.6) is exact [19]. More specifically, we can threshold u^* optimal for (4.2.6) at every value $t \in (0, 1)$ and the level sets $(\{u_1 > t\}, \{u_1 \leq t\})$ will give a minimal partition for (4.2.1).

4.3 Functional Lifting for Real-Valued Labeling

We will now study the second of the key problems (4.0.2). First, we note that minimization in (4.0.2) should occur over $\mathrm{BV}(\Omega, \mathbb{R})$, as this ensures that the term $|Du|$ is well-defined. For (4.0.2) the space of functions is convex, in comparison to the minimal partition problem (4.2.1), whereas the dataterm $f(u(x), x)$ may not be convex. We will discuss the concept of *functional lifting*, also known as *calibration* which allows us to still manage the non-convexity by introducing an extra dimension. This technique was introduced in [5] and is commonly applied to many optimization problems.

Definition 4.3.1 (Lifted Functions)**.** *Let*

$$C' = \left\{ \phi \in \mathrm{BV}\left(\Omega \times \mathbb{R}, \{0, 1\}\right) \ : \ \begin{array}{c} \lim\limits_{\gamma \to -\infty} \phi(\cdot, \gamma) = 1, \\ \lim\limits_{\gamma \to \infty} \phi(\cdot, \gamma) = 0, \\ D_\gamma \phi(\cdot, \gamma) \leq 0 \end{array} \right\} . \tag{4.3.1}$$

Every function $u : \Omega \to \mathbb{R}$ corresponds uniquely to a function $\phi \in C'$ via the relation

$$- D_\gamma \phi = \mathcal{H}^k \llcorner \mathit{graph}(u) \, , \tag{4.3.2}$$

where $\mathcal{H}^k \llcorner \mathit{graph}(u)$ is the restriction of the k-dimensional Hausdorff measure to the graph of u.

The above definition states in words that for each $x \in \Omega$, the lifted function ϕ is the decreasing jump function with jump point at value $u(x)$. In other words, the extra dimension introduced by the lifting represents the range of u. An illustration of the lifting from Definition 4.3.1 can be seen in figure 4.2

Minimization problem (4.0.2) can be restated in terms of the lifted function as

$$\min_{\phi \in C'} \int_{\Omega \times \mathbb{R}} f(\gamma, x) \cdot (-D_\gamma \phi(x, \gamma)) + |D_x \phi(x, \gamma)| \ dx \ d\gamma \, . \tag{4.3.3}$$

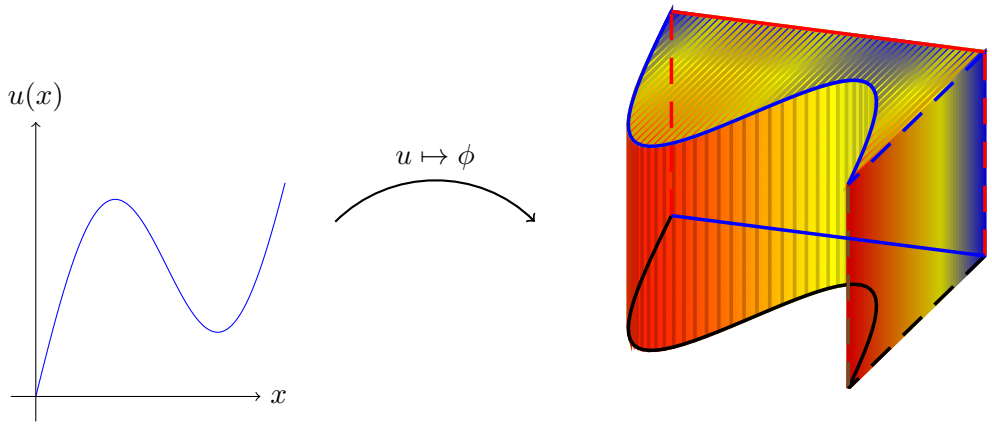


Figure 4.2 - Ordered lifted representation of an image

Note that (4.3.3) is now convex in the objective. It also corresponds to the original problem (4.0.2) via relation (4.3.2). Finally, noting that $\phi \in C' \Leftrightarrow u \in \mathrm{BV}(\Omega, \mathbb{R})$ gives equivalence between (4.0.2) and (4.3.3).

However, note that we have replaced the nonconvexity in the data term of (4.0.2) by the non-convexity of C' in (4.3.3). By taking the convex hull of C', which is given by

$$C'' = \left\{ \phi \in \mathrm{BV}\left(\Omega \times \mathbb{R}, [0,1]\right) \ : \ \begin{array}{c} \lim\limits_{\gamma \to -\infty} \phi(\cdot, \gamma) = 1, \\ \lim\limits_{\gamma \to \infty} \phi(\cdot, \gamma) = 0, \\ D_\gamma \phi(\cdot, \gamma) \leq 0 \end{array} \right\}. \tag{4.3.4}$$

we obtain a convex minimization problem. This relaxation is exact.

Theorem 4.3.1 (Exactness of functional lifting)**.** *Problems*

$$\min_{u \in \mathrm{BV}(\Omega, \mathbb{R})} \int_\Omega f(u(x), x) + |Du(x)| \ dx \tag{4.3.5}$$

and

$$\min_{\phi \in C''} \int_{\Omega \times \mathbb{R}} f(\gamma, x) \cdot (-D_\gamma \phi(x, \gamma)) + |D_x \phi(x, \gamma)| \ dx \ d\gamma \tag{4.3.6}$$

are equivalent. By thresholding an optimal lifted function ϕ of the second problem and using relation (4.3.2) we obtain an optimal function u to the first problem.

Proof. See [72]. □

4.4 Discretized Variational Problems and MRFs

For numerical treatment, we have to discretize the domain Ω into finitely many points $v \in \mathcal{V}$. We will present two approaches below.

The minimal partition problem (4.2.1) has a discrete value space. For the real-valued labeling problem (4.0.2) we have to discretize the value domain also.

4.4.1 Discretization of the Value Domain

For the real-valued labeling problem we discretize the extra dimension introduced by the lifting (4.3.1) by $\mathbb{Z}_{a,b}$. In this case we implicitly assume that the data term in (4.0.2) forces the optimizer to have values in $\mathbb{Z}_{a,b-1}$. We still denote the lifted sets by C' and C''.

Definition 4.4.1 (Discrete Lifted Functions)**.** *Let*

$$C' = \left\{ \phi : \Omega \times \mathbb{Z}_{a,b} \to \{0,1\} \ : \ \begin{array}{c} \phi(\cdot, a) = 1, \ \phi(\cdot, b) \equiv 0, \\ D_\gamma \phi(\cdot, \gamma) \leq 0 \end{array} \right\}. \tag{4.4.1}$$

where the discrete gradient is $D_\gamma \phi(x, \gamma) = \phi(x, \gamma + 1) - \phi(x, \gamma)$*. Every function* $u : \Omega \to \mathbb{Z}_{a,b}$ *corresponds uniquely to a function* $\phi \in C'$ *via the relation*

$$-D_\gamma \phi(x, \gamma) = \begin{cases} 1, & u(x) = \gamma \\ 0, & \textit{otherwise} \end{cases}, \tag{4.4.2}$$

Alternatively, we can model the value set $\mathbb{Z}_{a,b-1}$ in a MRF by the discrete label set $X_v = \{1, \ldots, k := b - 1 - a\}$ for every $v \in \mathcal{V}$. The bijective relationship between such a lifting defined in terms of indicator functions $u : \Omega \to \mathcal{E}_k$ and the one defined by Definition 4.4.1 is achieved via mapping $A : \mathcal{E}_k \to \{(b_0, \ldots, b_k) \in \{0,1\}^{k+1} \ : \ b_i \geq b_{i+1}\}$ via $A(e_y) = (1, \ldots, 1, 0, \ldots, 0)$, where the 1/0-transition occurs after the y-th position.

4.4.2 Discretization of Ω

There are two main approaches to discretizing the domain Ω.

4.4.2.1 Discretization on a Grid

When $\Omega = [0,1]^2$, discretization is usually performed by defining a grid. We denote the resulting points by $x_{i,j}$, $i, j = 0, \ldots, n-1$. Neighboring grid elements are connected and pairwise potentials are defined by $\mathbb{1}_{\{x_{i,j+1} \neq x_{i,j}\}}$ and $\mathbb{1}_{\{x_{i+1,j} \neq x_{i,j}\}}$ for the Potts-model, analoguously for TV. Note that this discretization might introduce metrification artifacts [60], which can be alleviated by introducing a more complicated neighborhood system on the grid [10].

4.4.2.2 Discretization by Superpixels

Alternatively, we can use superpixels to presegment the image domain Ω into meaningful small regions. A common method is SLIC [3]. Neighboring superpixels $uv \in \mathcal{E}$ are connected via pairwise potentials defined by $\alpha_{uv} \mathbb{1}_{\{x_u \neq x_u\}}$ for the Potts-model, analoguously for TV, where α_{uv} is the length of the border between superpixel u and v.

4.4.3 Proximal Splitting Algorithms

To solve discretization of the convex relaxations (4.2.6) and (4.3.6), proximal splitting algorithms are commonly used [15, 61, 71, 73]. For solving the MAP-inference problem with the local polytope relaxation, proximal splitting methods were proposed as well [64, 78, 90].

Remark 4.4.1. We will use proximal splitting algorithms in connection with the Wasserstein distance, which is described in Chapter 5. Specifically, we will use proximal splitting techniques for image histogram regularization in Section 5.4 and in connection with segmentation and cosegmentation in Section 5.5.

The starting point is the prox-operator, which will be the basic building blocks for the proximal splitting algorithms.

Definition 4.4.2 (Proximity Operator)**.** *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be a function. The prox-operator of f at $x^0 \in \mathbb{R}^n$ is*

$$\operatorname{prox}_{f_i}(x^0) = \operatorname{argmin}_{x \in \mathbb{R}^N} \frac{1}{2}\|x - x^0\|^2 + f_i(x) \tag{4.4.3}$$

Note that the prox-operator might not be well-defined. The argmin in (4.4.3) might be empty or have more than one element. However, the prox-operator is well-defined, when the function f is proper convex and lower-semicontinuous.

Definition 4.4.3 (Proper Functions)**.** *A function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is called proper if $\exists x^0 \in \mathbb{R}^n$ with $f(x^0) < \infty$.*

Definition 4.4.4 (Convex Functions)**.** *A function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is convex, if*

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y) \tag{4.4.4}$$

holds for all $x, y \in \mathbb{R}^n$ and $0 \leq \alpha \leq 1$.

Definition 4.4.5 (Lower Semicontinuity)**.** *A function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is lower semicontinuous (lsc) if*

$$\liminf_{x \to x^0} f(x) \leq f(x^0) \tag{4.4.5}$$

holds for all $x^0 \in \overline{\mathbb{R}}$.

Proposition 4.4.1. *Let $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ be proper convex and lsc. Then the prox-operator prox_f is well-defined for all $x \in \mathbb{R}^n$.*

Proof. See [81, Example 10.2] □

In the sequel we will always assume that all functions are proper convex lsc.

Note that computing the proximity operator is at least as hard as computing $\min_{x \in \mathbb{R}^n} f(x)$ for any function f, as the algorithm given by $x^{k+1} = \operatorname{prox}_f(x^k)$ would result in a sequence of points converging to $\operatorname{argmin}_{x \in \mathbb{R}^n} f(x)$ [79]. Also, for the minimization problems interesting us in this thesis, the prox-operator will not be efficiently computable. However, we can often decompose function $f : \mathbb{R}^N \to \mathbb{R}$ into

Algorithm 3: Douglas Rachford algorithm.

Data: $f_1, f_2 : \mathbb{R}^n \to \mathbb{R}$, $\epsilon \in (0,1)$, $\gamma > 0$, $y^0 \in \mathbb{R}^n$
for $n = 1, \ldots$ **do**
  $x^n = \text{prox}_{\gamma f_2}(y^n)$
  $\lambda_n \in [\epsilon, 2-\epsilon]$
  $y^{n+1} = y^n + \lambda_n \left(\text{prox}_{\gamma f_1}(2x^n - y^n) - x^n\right)$
end

several functions $f_1, \ldots, f_k : \mathbb{R}^N \to \mathbb{R}$ such that $f = f_1 + \ldots + f_k$. It is assumed that for each function f_i, $i = 1, \ldots, k$, the prox-operator is efficiently computable. Usually, this means that evaluating the prox-operator amounts to evaluating a closed form expression or some other efficient operation, e.g. applying the Fourier transformation or soft-thresholding. Proximal splitting algorithms work by iteratively evaluating prox-operators for the functions $f_1, \ldots, f_k$ and combining the resulting points. One classic proximal splitting scheme is the Douglas-Rachford algorithm detailed in Algorithm 3.

Theorem 4.4.1. *Assume that $f_1, f_2 : \mathbb{R}^n \to \mathbb{R}$ are proper convex lsc functions. Then the sequence x^n generated by the Douglas-Rachford algorithm 3 converges to some element of* $\text{argmin}_{x \in \mathbb{R}^x} f_1(x) + f_2(x)$.

Proof. See [21]. □

One drawback of the Douglas-Rachford algorithm is that in can only handle splitting into two functions. Classically, this is overcome by replicating variables as follows: Assume that we have a function $f : \mathbb{R}^n \to \mathbb{R}$ split into $f = f_1 + \ldots f_k$, where $k \in \mathbb{N}$. Define $g_1, g_2 : (\mathbb{R}^n)^k \to \overline{\mathbb{R}}$,

$$g_1(x^1, \ldots, x^k) = \begin{cases} 0, & x^1 = \ldots = x^k \\ \infty, & \text{otherwise} \end{cases} \tag{4.4.6}$$

and

$$g_2(x_1, \ldots, x_k) = f_1(x^1) + \ldots + f_k(x^k)\,. \tag{4.4.7}$$

To any minimizer x^* of $f_1 + \ldots + f_k$ corresponds a minimizer $(x^{*,1}, \ldots, x^{*,k})$ of function $g_1 + g_2$ with $x^{*,1}, \ldots, x^{*,k} = x^*$ by construction.

The proximal operator for g_1 is $\begin{pmatrix} \sum_{i=1}^k x_i \\ \vdots \\ \sum_{i=1}^k x_i \end{pmatrix}$, i.e. k-times the averaging operation, while the proximal operator for g_2 is the direct sum of the proximal operators of $f_1, \ldots, f_k$.

Other notable proximal algorithms include [9, 16, 77]. For an overview of proximal optimization algorithms see [20, 69].

5 The Wasserstein Distance for Variational Imaging

In this chapter we first introduce the Wasserstein distance in Section 5.1. To utilize the Wasserstein distance in imaging, we will describe how to construct image histograms in Sections 5.2 and 5.3. Based on this formalism, we will treat two applications of the Wasserstein distance in image analysis:

First, regularizing the grayvalue distribution of an image in Section 5.4 and

Second, using Wasserstein distances as dataterms for segmentation and cosegmentation in Section 5.5.

Sections 5.4 and 5.5 of this chapter are based on publications [103] and [104] respectively.

5.1 Wasserstein Distances

Definition 5.1.1 (Measurable space)**.** *A tuple* $(\mathbb{V}, \Sigma)$ *consisting of an underlying space* $\mathbb{V}$ *and a* σ*-algebra over* $\mathbb{V}$*, i.e.* $\Sigma \subset 2^{\mathbb{V}}$ *is called a measurable space. Each element of* Σ *is called measurable.*

Usually, we will omit the set of measurable subsets Σ for brevity. See [7] for an introduction to the associated measure theory.

We will encounter the following two examples.

Example 5.1.1 (Borel σ-algebra). $\mathbb{R}^k$, the k-dimensional Euclidean space can be equipped with the Borel σ-algebra, which is derived from the topology of $\mathbb{R}^k$, see [7].

Example 5.1.2 (Discrete Measurable Space). The discrete set $\{0, \dots, k-1\}$ can be equipped with the power set as σ-algebra $\Sigma = \mathcal{P}(\mathbb{V})$

For an image $I : \Omega \to \mathbb{V}$, the image value set $\mathbb{V}$ will be either the measurable set from Example 5.1.1 or 5.1.2. The first choice corresponds to natural images, for gray-value images we can use $(\mathbb{R}, \mathcal{B}(R))$ and $(\mathbb{R}^3, \mathcal{B}(\mathbb{R}^3))$ for color images. The second choice corresponds to labeling problems with discrete value space. The Wasserstein distance will live on the image value set $\mathbb{V}$.

We also assume we are given a measurable similarity function $c : \mathbb{V} \times \mathbb{V} \to \mathbb{R}$ which will denote similarity of any two points in $\mathbb{V}$. Given two measures $\nu_1, \nu_2 : \Sigma \to \mathbb{R}_+$ with $\nu_1(\mathbb{V}) = \nu_2(\mathbb{V})$, the Wasserstein distance $W(\nu_1, \nu_2) \in \mathbb{R}$ of these two measures is computed by evaluating the cost of an optimal rearrangement of ν_1 onto ν_2 with regard to the similarity function c on $\mathbb{V}$. Specifically, consider the space of all *coupling measures* of ν_1 onto ν_2, that is all measures on $\mathbb{V} \times \mathbb{V}$ with marginals ν_1 and ν_2:

Definition 5.1.2 (Coupling Measure)**.** *Let ν_1, ν_2 be two measures on $\mathbb{V}$. The space of coupling measures is defined by*

$$\Pi(\nu_1, \nu_2) = \left\{ \pi \text{ a measure on } \mathbb{V} \times \mathbb{V} : \begin{array}{ll} \pi(A \times \mathbb{V}) = \nu_1(A) & \\ \pi(\mathbb{V} \times B) = \nu_2(B) & \end{array} \quad \forall A, B \in \Sigma \right\}. \tag{5.1.1}$$

Measures in Π are also known as *rearrangements* or *transport plans* in the literature. The Wasserstein distance is defined as the infimum over all possible rearrangements with regard to the cost c:

Definition 5.1.3 (Wasserstein Distance)**.** *Let ν_1 and ν_2 be two measures. The Wasserstein distance is defined by*

$$W(\nu_1, \nu_2) = \inf_{\pi \in \Pi(\nu_1, \nu_2)} \int_{\mathbb{V} \times \mathbb{V}} c \, d\pi \,. \tag{5.1.2}$$

The Wasserstein distance is also known as the *optimal transport* or earth mover's distance in the literature. It can be shown that under mild assumptions on c the infimum is attained and the distance is finite, see [108]. The Wasserstein distance is a metric on the space of probability measures whenever c is a metric on $\mathbb{V}$, hence it gives a reasonable distance for measures when c is properly chosen.

As the Wasserstein distance can be computed via a linear program, it also has a dual formulation.

Definition 5.1.4 (Dual Wasserstein Distance)**.** *Let $c : \mathbb{V} \times \mathbb{V} \to \mathbb{R}$ be a cost function and let ν_1, ν_2 be two measures on $\mathbb{V}$. Let the dual admissible set, called dual Kantorovich set from now on, be*

$$\mathcal{K} := \left\{ \psi, \psi' \in L_1(\mathbb{V}) : \ \psi(y^1) + \psi'(y^2) \leq c(y^1, y^2) \quad \forall y^1, y^2 \in \mathbb{V} \right\}. \tag{5.1.3}$$

The dual formulation of the Wasserstein distance is

$$W(\nu_1, \nu_2) = \sup_{(\psi, \psi') \in \mathcal{K}} \int_{\mathbb{V}} \psi \, d\nu_1 + \int_{\mathbb{V}} \psi' \, d\nu_2 \tag{5.1.4}$$

Theorem 5.1.1 (Equivalence of Primal and Dual Wasserstein Distance)**.** *Assume the cost $c : \mathbb{V} \times \mathbb{V} \to \mathbb{R}$ is lower semicontinuous such that*

$$c(y^1, y^2) \geq a(y^1) + b(y^2) \quad \forall y^1, y^2 \in \mathbb{V} \tag{5.1.5}$$

for some $a, b \in L^1(\mathbb{V})$ upper semicontinuous (i.e. $-a, -b$ are lower semicontinuous. Then the values of the primal (5.1.2) and dual (5.1.4) formulation of the Wasserstein distance are equal.

Proof. See [108, Theorem 5.10]. □

The dual formulation of the Wasserstein distance can be advantageous, as the number of variables is smaller: While the coupling measure lives in the space $\mathbb{V}^2$, the variables needed in (5.1.4) are functions defined on $\mathbb{V}$.

Remark 5.1.1. The duality in Theorem 5.1.1 is also known as Kantorovich duality .

The minimization problem (5.1.2) 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.

Theorem 5.1.2. *Let $\mathbb{V}$ be a Polish space, let $c : \mathbb{V} \times \mathbb{V} \to \mathbb{R} \cup \{\infty\}$ be a lower semicontinuous function, and let W be the associated Wasserstein distance with regard to c. Let (Φ, λ) be a probability space and let ν_1 and ν_2 be two measurable functions defined on Φ with values in the space of probability distributions over $\mathbb{V}$. Assume that $c(v_1, v_2) \geq a(v_1) + b(v_2)$, where $a \in L_1(d\nu_1(\phi)d\lambda(\phi))$, $b \in L_1(d\nu_2(\phi)d\lambda(\phi))$. Then*

$$W\left(\int_\phi \nu_1(\phi)d\lambda(\phi), \int_\phi \nu_2(\phi)d\lambda(\phi)\right) \leq \int_\phi W\left(\nu_1(\phi), \nu_2(\phi)\right) d\lambda(\phi)\,. \tag{5.1.6}$$

Proof. See Theorem 4.8 in [108] □

Finally, the Wasserstein distance offers much flexibility in modelling similarity of measures by choosing an appropriate cost function c in (5.1.2).

A comprehensive treatment of Wasserstein distances can be found in [108].

5.1.1 Hoeffding-Fréchet Bounds

In this section we investigate $\mathbb{V} = \mathbb{R}$. In this scenario, we can represent measures by distribution functions (d.f.s) and derive explicit formulas for the Wasserstein distance under some conditions. The resulting formulation is known as Hoeffding-Fréchet bound.

Theorem 5.1.3 ([76, Thm. 3.1.1])**.** *Let F_1, F_2 be two real d.f.s and F a d.f. on $\mathbb{R}^2$. Then F has marginals F_1, F_2, if and only if*

$$\max\{F_1(\gamma_1) + F_2(\gamma_2) - 1, 0\} \leq F(\gamma_1, \gamma_2) \leq \min\{F_1(\gamma_1), F_2(\gamma_2)\} \tag{5.1.7}$$

By (5.1.2) the Wasserstein distance with marginal d.f.s F_1, F_2 can be computed by solving the optimal transport problem. In terms of d.f.s this can be written as follows.

Corollary 5.1.1. *Let F_1, F_2 be two real d.f.s. The Wasserstein distance can be written as*

$$W(dF_1, dF_2) = \min_F \int_{\mathbb{R}^2} c(dF_1, dF_2)\; dF, \qquad \text{s.t. } F \text{ respects the conditions (5.1.7)} \tag{5.1.8}$$

where dF_i shall denote the measure associated to the d.f. F_i, $i = 1, 2$.

Under convexity assumptions on the cost c, the Wasserstein distance additionally can be computed in closed form.

Proposition 5.1.1. *Let F_1, F_2 be two d.f.s. Let the cost function $c : \mathbb{R}^2 \to \mathbb{R}$ be $c(y^1, y^2) = \Phi(y^1 - y^2)$, where Φ is convex. Then the Wasserstein distance can be computed by*

$$W(F_1, F_2) = \int_0^1 c\left(F_1^{-1}(s), F_2^{-1}(s)\right)\, ds\,. \tag{5.1.9}$$

Proof. See [76]. □

This result will help us in designing efficient algorithms.

5.2 Histogram Construction for Images

In the preceeding section 5.1 we have presented the Wasserstein distance, which gives us the possibility to compare histogram to each other. In this section, we will show how to construct histograms for given images or for subsets of the image domain.

Let $\Omega \subset \mathbb{R}^2$ be the image domain, typically $\Omega = [0,1]^2$. We will denote images by $I, I_1, I_2 : \Omega \to \mathbb{V}$ and always assume that all images are measurable mappings. To be able to define the Wasserstein distance of image histograms, we first equip the image domain Ω with additional structure.

Definition 5.2.1 (Measure space)**.** *A measure space is a triple $(\Omega, \Sigma, \lambda)$, where (Ω, Σ) is a measurable space, see Definition 5.1.1, and $\mathcal{M}_+(\Omega) \ni \lambda : \Sigma \to \mathbb{R}_+$ is a measure.*

See again [7] for an introduction to the associated measure theory.

Example 5.2.1 (Lebesgue measure). The Lebesgue measure $\lambda : \mathcal{B}(\mathbb{R}^k) \to \mathbb{R}_+$ is defined on the Borel σ-algebra and assigns to each rectangle $[a_1, b_1] \times \ldots \times [a_k, b_k]$ the value $(b_1 - a_1) \cdot \ldots \cdot (b_k - a_k)$ and is extended uniquely to $\mathcal{B}(\mathbb{R}^k)$.

Example 5.2.2 (Counting Measure). The counting measure $\lambda : \mathcal{P}(\{1, \ldots, n\}) \to \mathbb{R}_+$ is defined on the power set of any discrete set and is defined by $\lambda(A) = |A|$ $\forall A \subset \{1, \ldots, n\}$.

In what follows, we will assume that either $\Omega = [0,1]^2$ or $\Omega = \{1, \ldots, n\}$. The first choice corresponds to image analysis in the continuous domain, the second choice corresponds to a discretized image domain used for the numerical implementation. If Ω is discrete and we work on an image grid, we also consider $\Omega = \{1, \ldots, n_1\} \times \{1, \ldots, n_2\}$, which is equivalent to the discrete choice with $n = n_1 \cdot n_2$. In both cases, the associated measures are taken from Examples 5.2.1 and 5.2.2 respectively.

Definition 5.2.2 (Image Histogram)**.** *Let $(\Omega, \Sigma, \lambda)$ be a measure space and let $(\mathbb{V}, \Sigma_{\mathbb{V}})$ be a measurable space. Let $I : \Omega \to \mathbb{V}$ be a measurable mapping. Let $\Theta \subset \Omega$ be a measurable subset of the image domain. The image histogram ν_Θ^I of I restricted to Θ is defined as the pushforward*

$$\nu_\Theta^I = (I_{|\Theta})_* \nu_{\llcorner \Theta}\,, \tag{5.2.1}$$

where $\nu_{\llcorner\Theta}$ is the measure ν restricted to subset Θ. Specifically, the measure for measurable set $A \subset \mathbb{V}$ is

$$\nu^I_\Theta(A) = \nu(I^{-1}(A) \cap \Theta) = \int_\Theta \mathbb{1}_{\{I(x)\in A\}}(x) d\nu(x)\,. \tag{5.2.2}$$

If subset $\Theta = \Omega$, we also omit the subscript and just write ν^I.

In the discrete case, i.e. $\Omega = \{1,\dots,n\}$, $\mathbb{V} = \{1,\dots,m\}$, evaluating the image histogram ν^I_Θ amounts to

$$\nu^I_\Theta(A) = |\{x \in \Theta \,:\, I(x) \in A\}|\,. \tag{5.2.3}$$

The general setup however allows to state the model in a continuous setting and makes notation easier.

Example 5.2.3. In Section 5.4 we will consider grayvalue histograms and we will want to find grayvalue images having a grayvalue distribution close to a known one. In the histogram construction, the subset Θ will be Ω, i.e. the histogram will be taken over the whole domain of the image. The unknown will be the image $I : \Omega \to \mathbb{R}$ and we will minimize the functional

$$\min_{I:\Omega\to\mathbb{R}} R(I) + W(\nu^I_\Omega, \nu_{\text{prior}})\,, \tag{5.2.4}$$

where $R(I)$ is a regularization and data term as in ordinary image denoising, and ν_{prior} is the a-priori known grayvalue distribution.

Example 5.2.4. In Section 5.5 we will consider images $I : \Omega \to \mathbb{V}$ taking values in some feature space, e.g. $\mathbb{V} = \mathbb{R}^3$ denoting the color space. For segmentation, the image I is fixed, but the subset Θ, describing the object to be segmented, will be unknown. This leads to the following minimization problem

$$\min_{\Theta\in\Sigma} R(\Theta) + W(\nu^I_\Theta, \nu_{\text{prior}})\,, \tag{5.2.5}$$

where $R(\Theta)$ is a regularizer of the shape given by Θ and ν_{prior} is the given appearance distribution of the sought-after object.

Note that in both examples 5.2.3 and 5.2.4 the histogram construction follows the same mathematical principle, while the tasks at hand are unrelated to each other. This demonstrates the broad range of application scenarios covered by the Wasserstein distance and our variational framework.

5.3 Linear Histogram Construction

In the preceeding Section 5.2 we have presented an approach to construct histograms from given images I and subsets of the image gomain Θ. However, the transformations $I \mapsto \nu^I_\Theta$ for given Θ and $\Theta \mapsto \nu^I_\Theta$ for given image I are both nonlinear, hence difficult to use in a variational setting. In this section we will show how to change the image

value domain to be able to linearize the histogram construction. The procedures will be akin to the lifting techniques in Sections 4.2 and 4.3.

5.3.1 Discrete Value Domain

Assume that $\mathbb{V} = \{1, \dots, k\}$ is discrete. Then we can view $I : \Omega \to \mathbb{V}$ equivalently as $I : \Omega \to \mathcal{E}_k$.

Proposition 5.3.1 (Linear Histogram Construction)**.** *Let an image $I : \Omega \to \mathcal{E}_k$. Then the appearance histogram ν^I from I by the approach from Definition 5.2.2 corresponds to*

$$\nu^I(A) = \int_{\Omega \times A} \langle \mathbb{1}_{\{y \in A\}}, I(x) \rangle \, d\nu(x, y) \,. \tag{5.3.1}$$

Also procedure (5.3.1) is linear in I.

With this construction, we can use the Wasserstein distance in variational problems, as it is convex in both of its arguments, hence the whole approach results in a convex function.

Note that this procedure is akin to the convexification of the minimal partition problem in Section 4.2.

5.3.2 Ordered Value Domains

Assume that the value domain is either $\mathbb{R}$ or subset of $\mathbb{Z}$. As in Section 4.3 we will lift the image $I : \Omega \to \mathbb{V}$ to (abusing notation) $I : \Omega \times \mathbb{V} \to \{0, 1\}$ with $I(x, -\infty) = 1$ and $I(x, \infty) = 0$ and $D_\gamma I(x, \gamma) \leq 0$ with the same correspondence as in (4.3.2). The histogram of the lifted image I is

Definition 5.3.1 (Linear Histogram Construction)**.** *Define the image histogram of lifted image $I : \Omega \times \mathbb{V} \to \{0, 1\}$ for any $\Theta \subset \Omega$ by*

$$\nu^I_\Theta(A) = \int_{\Omega \times \mathbb{V}} \langle \mathbb{1}_A, -D_\gamma I(x, \gamma) \rangle \, dx \, d\gamma \,. \tag{5.3.2}$$

Proposition 5.3.2. *Histogram construction of the original image as in Definition 5.2.2 and of the lifted image as in Definition 5.3.1 lead to the same histogram ν^I_Θ.*

5.4 Convex Variational Image Restoration With Histogram Priors

In this section we present a novel variational approach to image restoration (e.g., denoising, inpainting, labeling) that enables to complement established variational approaches with a histogram-based prior based on the Wasserstein distance enforcing closeness of the solution to some given empirical measure. By minimizing a single objective function, the approach utilizes simultaneously two different sources of information for restoration: spatial context in terms of some smoothness prior and non-spatial statistics in terms of the novel prior utilizing the Wasserstein distance between probability measures. We study the combination of the functional lifting technique with two different relaxations of the histogram prior and derive a jointly convex variational approach. Mathematical equivalence of both relaxations is established and cases where optimality holds are discussed. Additionally, we present an efficient algorithmic scheme for the numerical treatment of the presented model. Experiments using the basic total-variation based denoising approach as a case study demonstrate our novel regularization approach.

The work of this section is based on publication [103].

5.4.1 Introduction

A broad range of powerful variational approaches to low-level image analysis tasks exist, like image denoising, image inpainting or image labeling [61, 68]. It is not straightforward however to incorporate *directly* into the restoration process statistical prior knowledge about the image class at hand. Particularly, handling global statistics as part of a single *convex* variational approach has not been considered so far.

We introduce a class of variational approaches of the form

$$\inf_I F(I) + \lambda R(I) + \kappa W(\nu^I, \nu^0), \tag{5.4.1}$$

where $F(I) + \lambda R(I)$ is any energy functional consisting of a data fidelity term $F(I)$ and a regularization term $R(I)$, $W(\nu^I, \nu^0)$ denotes the histogram prior in terms of the Wasserstein distance between the histogram corresponding to the minimizing image $I : \Omega \to \mathbb{R}$ to be determined and some given histogram ν^0 and $\lambda > 0$ and $\kappa > 0$ are parameters weighing the influence of each term. We require $R(I)$ to be convex. As a case study, we adopt for $R(I) = \mathrm{TV}(I)$, the Total Variation (4.1.4), and $F(I) = \int_\Omega f(I(x), x)dx$, where f can also be a nonconvex function. The basic Rudin Osher Fatemi (ROF) denoising approach of [84] is included in this approach with $f(I(x), x) = (I(x) - I_0(x))^2$, where I_0 is the image to be denoised.

Note that minimizing the second term $R(I)$ in (5.4.1) entails spatial regularization whereas the third Wasserstein term utilizes statistical information that is not spatially indexed in any way. As an illustration, consider the academical example in figure 5.1. Knowing the grayvalue distribution of the original image helps us in regularizing the noisy input image. We tackle the corresponding main difficulty in two different, mathematically plausible ways: by convex relaxations of (5.4.1) in order to obtain a

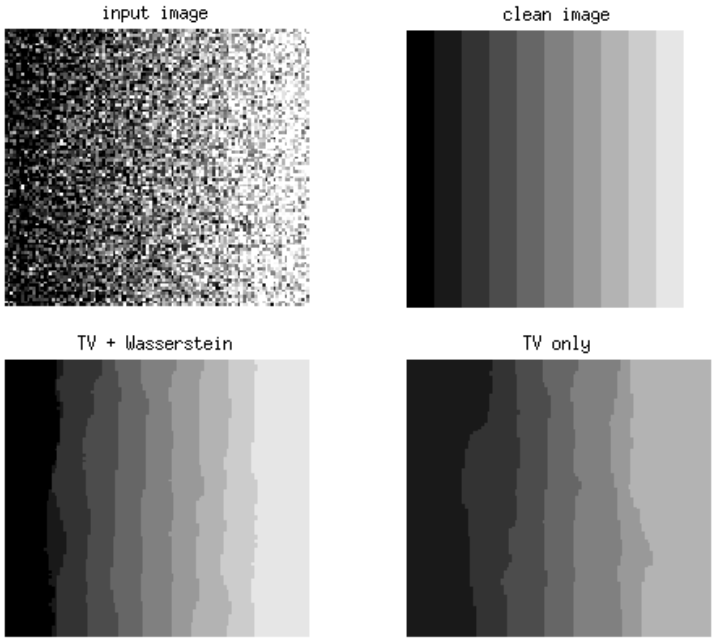


Figure 5.1 - Denoising experiment of a noisy image (upper row, left side) taking into account statistical prior information through convex optimization (lower row, left side) infers the correct image structure and outperforms hand-tuned established variational restoration (lower row, right side). Enforcing global image statistics to be similar to those of the clean image (upper row, right side) gives our approach an advantage over methods not taking such information into account.

computationally tractable approach. Comparing these two relaxations – one may be tighter than the other one – reveals however mathematical equivalence. Preliminary numerical experiments demonstrate that the relaxation seems to be tight enough so as to bias effectively variational restoration towards given statistical prior information.

5.4.2 Related Work

Image regularization by variational methods is a powerful and commonly used tool for denoising, inpainting, labeling and many other applications. As a case study in connection with (5.4.1), we consider one of the most widely used approaches for denoising, namely the Rudin, Osher and Fatemi (ROF) model from [84]:

$$\min_{I \in \mathrm{BV}(\Omega)} \|I - I_0\|^2 + \lambda \mathrm{TV}(I), \tag{5.4.2}$$

where U_0 is the input image, TV denotes the Total Variation and $\mathrm{BV}(\Omega)$ is the space of functions of bounded variation with domain $\Omega \subset \mathbb{R}^d$ and values in $\mathbb{R}$. The minimization problem (5.4.2) is convex and can be solved to a global optimum efficiently by various first-order proximal splitting algorithms even for large problem sizes, e.g. by the Douglas-Rachford, Algorithm 3, or alternatively by Primal-Dual methods [16] or other proximal minimization algorithms for nonsmooth convex optimization [9, 77].

We can also use more general data terms instead of the quadratic term in (5.4.2). We have shown in Section 4.3 how a possibly non-convex data term $\int_\Omega f(I(x), x)dx$ can be optimized to optimality. Still this data function is local and does not take into account global statistics.

In the case that some prior knowledge is encoded as a histogram, the Wasserstein distance and the associated Optimal Transport are a suitable choice for penalizing deviance from prior knowledge. More generally the Wasserstein distance can be used as a distance on histograms over arbitrary metricized spaces.

Regarding utilization of the Wasserstein distance in variational imaging, recent applications include [17, 70] in connection with segmentation and [25] for texture synthesis.

The authors of [75] propose an approach to contrast and color modification. Given a prior model of how the color or grayvalues are distributed in an image, the authors propose a variational formulation for modifying the given image so that these statistical constraints are met in a spatially regular way. While their algorithm is fast, high runtime performance is achieved by minimizing a non-convex approximation of their original energy. In contrast, we directly minimize a convex relaxation of the original energy, hence we may hope to obtain lower energies and not to get stuck in local minima.

Our variational approach employing the Wasserstein distance as a histogram-based prior through *convex* relaxation appears to be novel.

5.4.3 Contribution

We present

- a variational model with a histogram-based prior for image restoration (Section 5.4.4),
- two convex relaxations of the original problem together with discussions of cases where optimality holds (Sections 5.4.7 and 5.4.8),
- a proof of equivalence for the two presented relaxations (Section 5.4.9),
- an efficient numerical implementation of the proposed variational model (Section 5.4.10),
- experimental validation of the proposed approach (Section 5.4.11).

5.4.4 Problem and Mathematical Background

We introduce the original non-convex model, consider different ways to write the Wasserstein distance and introduce the functional lifting technique for rewriting the resulting optimization problem to show well-posedness and to make it amenable for global optimization.

5.4.5 Problem Statement

Let the image domain be $\Omega = [0,1]^2$ and let $I^0 : \Omega \to \mathbb{R}$ be a given noisy grayvalue image. Let the desired grayvalue histogram be ν^0. Our aim is to obtain an image I from I^0 such that I is at the same time close to I^0, denoised, and such that its grayvalue histogram ν^I is similar to ν^0. To this end consider the following energy minimization problem:

$$\min_{I \in \mathrm{BV}(\Omega)} E(I) = \int_\Omega f(I(x), x)dx + \lambda \mathrm{TV}(I) + \kappa \min_{\pi \in \Pi(\nu^I, \nu^0)} \langle c, \pi \rangle \,. \tag{5.4.3}$$

By minimizing (5.4.3) we obtain a solution u which remains faithful to the data by the fidelity term f, is spatially coherent by the Total Variation term and has global grayvalue statistics similar to ν^0 by the Wasserstein term.

Taking into account the dual Wasserstein distance (5.1.4), the energy in problem (5.4.3) can be reformulated as

$$E_{dual}(I) = \int_\Omega f(I(x), x)dx + \lambda \mathrm{TV}(I) + \kappa \sup_{(\psi,\psi') \in \mathcal{K}} \left(\int_{\mathbb{R}} \psi d\nu^I + \int_{\mathbb{R}} \psi' d\nu^0 \right) , \tag{5.4.4}$$

where $\mathcal{K}$ is the dual Kantorovich set. This leads to saddle point problem via the Lagrangian

$$L(I, \psi, \psi') = \int_\Omega f(I(x), x)dx + \lambda \mathrm{TV}(I) + \kappa \left(\int_{\mathbb{R}} \psi d\nu^I + \int_{\mathbb{R}} \psi' d\nu^0 \right) , \tag{5.4.5}$$

and

$$\min_{I\in \mathrm{BV}\,\Omega} E(I) = \min_{I\in \mathrm{BV}(\Omega)} \sup_{(\psi,\psi')\in D} L(I,\psi,\psi') \tag{5.4.6}$$

holds.

Note however that both energies (5.4.3) and (5.4.4) are not convex due to the nonlinear transformation $I \to \nu^I$ and the possible nonconvexity of f. Hence we have to convexify energy (5.4.3), which will be done below.

5.4.6 Functional Lifting

While the Wasserstein distance (5.1.2) is convex in both of its arguments, see Theorem 5.1.2, the energies in (5.4.3) and (5.4.4) are not convex due to the nonconvex transformation $I \mapsto \nu^I$ in the first argument of the Wasserstein term and the possible nonconvexity of f. To overcome the nonconvexity of both the data term and the transformation in the first argument of the Wasserstein distance we lift the function I as in Section 4.3. Instead of I we consider the lifted image $I' \in C'$ as in Definition 4.3.1, which allows us both to linearize the fidelity term and to convexify the Wasserstein distance.

Consider the Lagrangian with lifted primal part

$$L'(I',\psi,\psi') = \begin{array}{l} -\int_\Omega \int_{\mathbb{R}} f(\gamma,x) D_\gamma I'(x,\gamma)\,dx + \lambda \int_{\mathbb{R}} \mathrm{TV}(I'(\cdot,\gamma))d\gamma \\ +\kappa \left(\int_{\mathbb{R}} \psi d\nu^{I'} - \int_{\mathbb{R}} \psi' d\nu^0 \right) . \end{array} \tag{5.4.7}$$

For a pair (I, I') as in Definition 4.3.1 the identity

$$L(I,\psi,\psi') = L'(I',\psi,\psi') \tag{5.4.8}$$

holds true by the coarea formula, see [6]. Consequently, we have

$$\inf_{I\in \mathrm{BV}(\Omega)} \sup_{(\psi,\psi')\in\mathcal{K}} L(I,\psi,\psi') = \inf_{I'\in C'} \sup_{(\psi,\psi')\in\mathcal{K}} L'(I',\psi,\psi') . \tag{5.4.9}$$

Note that L' is convex in I' and concave in (ψ,ψ'), hence is easier to handle from an optimization point of view.

Theorem 5.4.1. *Let $\Omega \subset \mathbb{R}^2$ be bounded, let $f(x,\gamma)$ be continuous and let the cost c of the Wasserstein distance fulfill the conditions from Theorem 5.1.1. Then there exists a minimizer $\overline{I'}$ of $\inf_{I'\in C'} \sup_{(\psi,\psi')\in\mathcal{K}} L'(I',\psi,\psi')$.*

Proof. We first show that the set C' is compact in the weak* topology in BV. By theorem 3.23 in [6], C' is precompact. It then remains to prove that C' is closed in the weak*-topology. Thus let (I'_n) in C' converge weakly* to I', which means that (I'_n) converges strongly in L^1_{loc} and $D_\gamma I'_n$ converges weakly*. $D_\gamma I'_n(\cdot,\gamma) \le 0$ means

$$\int_{\Omega\times\mathbb{R}} w D_\gamma I'_n \ge 0 \quad \forall w \in C_c(\Omega\times\mathbb{R}) . \tag{5.4.10}$$

This property is preserved under weak*-convergence by definition. $I'(x,\gamma) \in \{0,1\}$ a.e.

as convergence in L^1 implies pointwise convergence of some subsequence. Obviously $I_n'(\cdot, -\infty) = 1$ and $I_n'(\cdot, \infty) = 0$ are naturally preserved in the limit.

The first term in the energy (5.4.7) is lower semicontinuous by assumption. The TV-term is lower-semicontinuous by Theorem 5.2 in [6].

The Wasserstein term in (5.4.7) has the form $\sup_{\{(\psi,\psi')\in\mathcal{K}\}} \int_{\mathbb{R}} \psi \, d\nu^{I'} - \int_{\mathbb{R}} \psi' \, d\nu^0$ and can thus be written as

$$\sup_{\{(\psi,\psi')\in\mathcal{K},\psi,\psi'\in C_c(\mathbb{R})\}} -\frac{1}{|\Omega|} \int_{\mathbb{R}} \int_{\Omega} \psi(\gamma) dx \, D_\gamma I'(x,\gamma) - \int_{\mathbb{R}} \psi'(\gamma) \, d\nu^0(\gamma) \,. \tag{5.4.11}$$

Hence it is a supremum of linear functionals and lsc as well.

As a supremum of positive sums of lsc terms, $\sup_{(\psi,\psi')\in\mathcal{K}\cap C_c(\mathbb{R})^2} L'(\cdot,\psi,\psi')$ is lsc as well. A minimizing sequence therefore has a weakly*-convergent subsequence due to compactness of C'. The limit is a minimizer by the lower semicontinuity of the energy. □

As we have shown above, the proposed lifted model is well-posed, which means that the minimizer is attained under mild technical conditions. Then by (5.4.9) and (5.4.6) also the original energy is well-posed.

Remark 5.4.1. We have considered a spatially continuous formulation, as discretizations thereof suffer less from grid bias [46, 60] than purely discrete formulations. Thus, proving existence of a solution of the spatially continuous model substantiates our approach from a modelling point of view.

Remark 5.4.2. As discussed in Section 5.4.1, we merely consider total variation based regularization as a case study, but this restriction is not necessary. More general regularizers can be used as well as long as they are convex, e.g. quadratic or Huber functions. Then all the statements still hold, see [73]. In the present paper however, we rather focus on the novel prior based on the Wasserstein distance.

5.4.7 Relaxation as a Convex/Concave Saddle Point Problem

Optimizing energies (5.4.3) and (5.4.4) is not tractable, as it is a nonconvex problem. Also solving (5.4.9) is not tractable, as the set C' is nonconvex. The latter can be overcome by considering the convex hull C'' of C'

$$C'' := \operatorname{conv} C' \tag{5.4.12}$$

which leads to a relaxation as a convex/concave saddle point problem of the minimization problem (5.4.9), which is solvable computationally.

Proposition 5.4.1. *The Lagrangian L' from (5.4.7) is convex/concave and*

$$\min_{I\in BV(\Omega)} E(I) \geq \min_{I'\in C''} \sup_{(\psi,\psi')\in\mathcal{K}} L'(I',\psi,\psi'). \tag{5.4.13}$$

If

$$\min_{I\in C} \sup_{(\psi,\psi')\in\mathcal{K}} L(I,\psi,\psi') = \sup_{(\psi,\psi')\in\mathcal{K}} \min_{I\in C} L(u,\psi,\psi') \tag{5.4.14}$$

holds, then the above relaxation is exact.

Proof. Note that C'' is a convex set, in particular it is the convex hull of C'. L' is also convex in I', therefore the right side of (5.4.13) is a convex/concave saddle point problem. For fixed (ψ, ψ') we have the following equality:

$$\min_{I \in BV(\Omega)} L(I, \psi, \psi') = \min_{I' \in C''} L'(I', \psi, \psi'), \tag{5.4.15}$$

which is proved in [72].

$$\begin{array}{rcl} \min_{I \in BV(\Omega)} E(I) & = & \min_{I \in BV(\Omega)} \sup_{(\psi,\psi') \in \mathcal{K}} L(I, \psi, \psi') \\ & \overset{(*)}{\geq} & \sup_{(\psi,\psi') \in \mathcal{K}} \min_{I \in BV(\Omega)} L(I, \psi, \psi') \\ & \overset{(**)}{=} & \sup_{(\psi,\psi') \in \mathcal{K}} \min_{I' \in C''} L'(I', \psi, \psi'), \end{array} \tag{5.4.16}$$

where $(*)$ is always fulfilled for minimax problems and $(**)$ is a consequence of (5.4.15). This proves (5.4.13). If (5.4.14) holds, then $(*)$ above is actually an equality and the relaxation is exact. □

Remark 5.4.3. The relaxation presented above will provide us with a model for numerically obtaining solutions in section 5.4.11 for the model (5.4.3). This relaxation technique also works with histograms of dimension > 1, see [13] for lifting techniques for vector valued functions, but the exactness of the functional lifting as done in (5.4.7) may not hold any more. Also it is computationally more expensive.

5.4.8 Relaxation with Hoeffding-Fréchet Bounds

A second relaxation can be constructed using Hoeffding-Fréchet bounds introduced in Section 5.1.1. Using Corollary 5.1.1, where we replace the distribution functions F_1 by the distribution function of $\nu^{I'}$, which is $\int_\Omega \int_{-\infty}^{\gamma} -D_{\gamma'} I'(x, \gamma')\, d\gamma'\, dx$, and the lifting technique from Definition 4.3.1 we arrive at the following relaxation:

$$\begin{array}{rl} \min_{I', F} & \int_\Omega \int_{\mathbb{R}} -f(\gamma, x) D_\gamma I'(x, \gamma) dx + \lambda \int_{\mathbb{R}} \mathrm{TV}(I'(\cdot, \gamma) d\gamma + \kappa \int_{\mathbb{R}^2} c\, dF, \\ s.t. & F_{I'}(\gamma) = \frac{1}{|\Omega|} \int_\Omega \int_{-\infty}^{\gamma} -D_{\gamma'} I'(x, \gamma')\, d\gamma'\, dx, \\ & F_{\nu^0}(\gamma) = \nu^0((-\infty, \gamma]), \\ & F_{I'}(\gamma_1) + F_{\nu^0}(\gamma_2) - 1 \leq F(\gamma_1, \gamma_2) \leq \min\{F_{I'}(\gamma_1), F_{\nu^0}(\gamma_2)\} \\ & I' \in C'' \end{array} \tag{5.4.17}$$

The minimization problem (5.4.17) is a relaxation of (5.4.3). Just set

$$I'(x, \gamma) = \begin{cases} 1, & u(x) < \gamma \\ 0, & u(x) \geq \gamma \end{cases}$$

and let F be the d.f. of the optimal transport measure with marginals $\nu^{I'}$ and ν^0.

Remark 5.4.4. It is interesting to know, when relaxation (5.4.17) is exact. By the

coarea formula [118] we know that

$$\begin{aligned} & \int_\Omega \int_{\mathbb{R}} -f(\gamma,x) D_\gamma I'(x,\gamma)\; d\gamma\; dx + \lambda \int_{\mathbb{R}} \mathrm{TV}(I'(\cdot,\gamma)) d\gamma \\ = \; & \int_0^1 \int_\Omega f(I'_\alpha(x),x) dx d\alpha + \lambda \int_0^1 \mathrm{TV}(I'_\alpha) d\alpha\,, \end{aligned} \tag{5.4.18}$$

where I'_α corresponds to the thresholded function $I'_\alpha = \mathbb{1}_{\{I'>\alpha\}} \in C'$ via relation (4.3.2). However such a formula does not generally hold for the optimal transport: Let $I'_\alpha = \mathbb{1}_{\{I'>\alpha\}}$ and let F_α be the d.f. of the optimal coupling with marginal d.f.s $F_{I'_\alpha}$ and F_{ν^0}. Then

$$F = \int_0^1 F_\alpha \; d\alpha \tag{5.4.19}$$

has marginal d.f.s $\int_0^1 F_{I'_\alpha} d\alpha$ and F_{ν^0}, but it may not be optimal.

A simple example for inexactness can be constructed as follows: Let the data term be $f \equiv 0$ and let $\nu^0 = \frac{1}{2}(\delta_0 + \delta_1)$ and let the cost for the Wasserstein distance be $c(\gamma_1,\gamma_2) = \lambda|\gamma_1 - \gamma_2|$. Every constant function with $I(x) = \text{const} \in [0,1]$ will be a minimizer if λ is small and κ is big enough. The objective value will be $\frac{\lambda}{2}$. But relaxation (5.4.17) is inexact in this situation: Choose $I'(x,\gamma) = \frac{1}{2} \quad \forall \gamma \in (0,1)$ and the relaxed objective value will be 0.

Remark 5.4.5. The above remark was concerned with an example, where a convex combination of optimal solutions to the non-relaxed problem is a unique solution of the relaxed problem with lower objective value.

By contrast, in Section 5.4.11 two different academical examples are shown, which illustrate the behaviour of our relaxation (5.4.17) in situations when the non-relaxed solution is unique, see Figures 5.2. Then exactness may hold or not, depending on the geometry of level sets of solutions. No easy characterization seems to be available for the exactness of model (5.4.17).

5.4.9 Relationship between the two Relaxations

Both relaxations from Sections 5.4.7 and 5.4.8 seem to be plausible but seemingly different relaxations. Their different nature reveals itself also in the conditions for which exactness was established. While the condition in Proposition 5.4.1 depends on the gap introduced by interchanging the minimum and maximum operation, relaxation (5.4.17) is exact if a coarea formula holds for the optimal solution. It turns out, however, that both equations are equivalent, hence both optimality conditions derived in Sections 5.4.7 and 5.4.8 can be used to ensure exactness of a solution to either one of the relaxed minimization problems.

Theorem 5.4.2. *The optimal values of the two relaxations* (5.4.13) *and* (5.4.17) *are equal.*

Proof. It is a well known fact that

$$\min_{x\in X} \max_{y\in Y} \langle Kx, y\rangle + G(x) - H^*(y) \tag{5.4.20}$$

and

$$\min_{x\in X} H(Kx) + G(x) \tag{5.4.21}$$

are equivalent, where $G : X \to [0,\infty]$ and $H^* : Y \to [0,\infty]$ are proper, convex, lsc functions, H^*, defined by $H^*(y) = \sup_x \langle x, y\rangle - H(x)$, is the convex conjugate of H and X and Y are two real vector spaces, see [80] for details.

To apply the above result choose

$$G(I') = \int_0^1 \int_\Omega -D_\gamma I'(x,\gamma)\cdot f(\gamma,x)dx + \lambda \int_0^1 \mathrm{TV}(I'(\cdot,\gamma))d\gamma + \chi_{C''}(I'), \tag{5.4.22}$$

$$H^*(\psi,\psi') = \kappa \int_0^1 \psi' d\nu^0 + \chi_{\mathcal{K}}(\psi,\psi') \tag{5.4.23}$$

and

$$\begin{aligned} K : C'' &\to \mathcal{M}_+(\mathbb{R})^2, \\ K(I') &= (\kappa\nu^{I'}, 0) \end{aligned} \tag{5.4.24}$$

(5.4.20) corresponds with the above choices to the saddle point relaxation (5.4.13).

Recall that $H = (H^*)^*$ if H is convex and lsc, i.e. H is the Legendre-Fenchel bidual of itself, see [80]. Hence, for positive measures $\nu, \tilde{\nu}$, the following holds true:

$$\begin{aligned} H(\nu,\tilde{\nu}) &= \sup_{\psi,\psi'}\{\textstyle\int_0^1 \psi d\nu - \int_0^1 \psi' d\tilde{\nu} - H^*(\psi,\psi')\} \\ &= \sup_{(\psi,\psi')\in\mathcal{K}}\{\textstyle\int_0^1 \psi d\nu - \int_0^1 \psi' d\tilde{\nu} - \kappa\int_0^1 \psi' d\nu^0\} \\ &= \sigma_{\mathcal{K}}(\nu, \tilde{\nu} + \kappa\nu^0) \\ &\overset{(*)}{=} W(\nu, \tilde{\nu} + \kappa\nu^0) \end{aligned} \tag{5.4.25}$$

where $\sigma_A(x) = \sup_{a\in A}\langle a, x\rangle$ is the support function of the set A and κ is the weight for the Wasserstein term in (5.4.3). To prove $(*)$, we invoke Theorem 5.1.1, which states that

$$\sigma_{\mathcal{K}}(\nu,\tilde{\nu}) = \sup_{(\psi,\psi')\in\mathcal{K}} \int_0^1 \psi d\nu - \int_0^1 \psi'\tilde{\nu} = \min_{\pi\in\Pi(\nu,\tilde{\nu})} \int_{\mathbb{R}^2} c(\gamma_1,\gamma_2)d\pi(\gamma_1,\gamma_2) = W(\nu,\tilde{\nu}), \tag{5.4.26}$$

and we have infinity for measures which do not have the same mass.

Thus, the energy in (5.4.21) can be written as

$$G(I') + H(\kappa\nu^{I'}, 0) = G(I') + W(\kappa\nu^{I'}, \kappa\nu^0) = G(I') + \kappa W(\nu^{I'}, \nu^0)\,. \tag{5.4.27}$$

This energy is the same as in relaxation (5.4.17), which concludes the proof. □

5.4.10 Optimization

We present six experiments and the numerical method used to compute them.

5.4.10.1 Implementation

First, we discretize the image domain Ω to be $\{1,\dots,n_1\}\times\{1,\dots,n_2\}$ and use forward differences as the gradient operator. Second, we discretize the values image I' can take to $\{0,1,\dots,k-1,k\}$. Hence we consider images taking only finitely many grayvalues. The resulting space of images I' can be identified with

$$C_d = \left\{ I' : \Omega\times\{0,1,\dots,k\}\to\mathbb{R} \;:\; \begin{array}{l} I'(\cdot,k)=0,\; I'(\cdot,0)=1, \\ I'(\cdot,l)\le I'(\cdot,l-1) \end{array} \right\}. \tag{5.4.28}$$

The dual Kantorovich set for the discretised problem is then

$$\mathcal{K}_d = \left\{ \psi,\psi' : \{0,1,\dots,k\}\to\mathbb{R} \;:\; \psi(y^1)+\psi'(y^2)\le c(y^1,y^2)\;\forall y^1,y^2 \right\}. \tag{5.4.29}$$

After computing a minimizer I'^* of the discretized energy, we threshold it at the value 0.5 to obtain $I'^* = \mathbb{1}_{\{I'^*>0.5\}}$ and then calculate $I^\star$ by the discrete analogue of relation (4.3.2).

For computing a minimizer of the discretized optimization problem

$$\min_{I'\in C_d}\max_{(\psi,\psi')\in\mathcal{K}_d} E'(I',\psi,\psi') \tag{5.4.30}$$

it is expedient to use first order algorithms like the Douglas Rachford Algorithm 3 or [9, 16, 77] as the dimensionality of the problem is high. To use such algorithms it is necessary to split the function $\max_{(\psi,\psi')\in\mathcal{K}_d} L'(I',\psi,\psi')$ into a sum of terms, whose proximity operators can be computed efficiently. Hence consider the following equivalent minimization problem:

$$\min_{I',g\in(\mathbb{R}^{n_1\times n_2\times k\times 2})} \langle\tilde{f},I'\rangle + \|g\|_1 + \chi_{\{(u,v):\,\nabla u=v\}}(I',g) + \chi_{C_d}(I') + W(\nu^{I'},\nu^0)\,, \tag{5.4.31}$$

where $\tilde{f}$ comes from the local cost factor in (5.4.7).

- The proximity operator of the term $\|g\|_1$ is the soft-thresholding operator.
- $\operatorname{prox}_{\chi_{\{(u,v):\,\nabla u=v\}}}(I',g)$ can be efficiently computed with Fourier transforms, see for example [77].
- $\operatorname{prox}_{\chi_{C_d}}$ can be computed by projection onto the set of non-increasing sequences. To compute this projection, we employ the algorithm proposed in [15], Appendix D. It is trivially parallelisable and converges in a finite number of iterations.
- Finally, the proximity operator for the Wasserstein distance can be computed efficiently in some special cases, as discussed in the next Section 5.4.10.2.

We can either use [77] to minimize (5.4.31) directly, which is equivalent to using the Douglas-Rachford described in Algorithm 3 on a suitably defined product space as in Section 5 and absorbing the linear term in the functions in (5.4.31) arbitrarily.

5.4.10.2 Wasserstein Proximation for $c(\gamma_1, \gamma_2) = |\gamma_1 - \gamma_2|$ by soft-thresholding

In general, computing the proximity operator for the Wasserstein distance can be expensive and requires solving a quadratic program. However, for the real line and convex costs, we can compute the proximity operator more efficiently, as shown in (5.1.9). One algorithm for the cost function $c(\gamma_1, \gamma_2) = |\gamma_1 - \gamma_2|$ is presented below.

The proximation for the weighted Wasserstein distance is

$$\operatorname{argmin}_{I'} \frac{1}{2}\|I'^0 - I'\|_2^2 + \lambda W(\nu^{I'}, \nu^0). \tag{5.4.32}$$

For the special case we consider here, there is a simple expression for the Wasserstein distance:

Proposition 5.4.2 ([76])**.** *For two measures ν^1, ν^2 on the real line and $c(y^1, y^2) = |y^1 - y^2|$, the Wasserstein distance is*

$$W(\nu^1, \nu^2) = \int_{\mathbb{R}} |F_{\nu^1}(\gamma) - F_{\nu^2}(\gamma)| \, d\gamma \tag{5.4.33}$$

Due to $D_\gamma I'(x, y) \leq 0$ and $I'(x, 0) = 1$, we can also write $F_{\nu^{I'}}(\gamma)$ as

$$F_{\nu^{I'}}(\gamma) = \frac{1}{|\Omega|} \int_\Omega \int_{-\infty}^{\gamma} -D_{\gamma'} I'(x, \gamma') \, d\gamma' \, dx = \frac{1}{|\Omega|} \int_\Omega 1 - I'(x, \gamma) dx \,. \tag{5.4.34}$$

Next we show how to solve in closed form the proximity operator for the Wasserstein distance in the present case.

Proposition 5.4.3. *Given I'^0, $\lambda > 0$, the optimal $\tilde{I}'$ for the proximity operator*

$$\tilde{I}' = \operatorname{argmin}_{I'} \frac{1}{2}\|I' - I'^0\|_2^2 + \lambda W(F_{\nu^{I'}}, F_{\nu^0}) \tag{5.4.35}$$

is determined by

$$\tilde{I}'(x, \gamma) = I'(x, \gamma) + c_\gamma \,, \tag{5.4.36}$$

where

$$c_\gamma = \operatorname{shrink}\left(\left[-\frac{1}{|\Omega|}\int_\Omega I'^0(x, \gamma) dx - F_{\nu^0}(\gamma) + 1\right], \frac{\lambda}{|\Omega|}\right) + \frac{1}{|\Omega|}\int_\Omega I'^0(x, \gamma) dx + F_{\nu^0}(\gamma) - 1 \tag{5.4.37}$$

and shrink *denotes the soft-thresholding operator defined componentwise by*

$$\operatorname{shrink}(a, \lambda)_i = \max\{|a_i| - \lambda, 0\} \cdot \operatorname{sign}(a_i) \tag{5.4.38}$$

for $a \in \mathbb{R}^n$, $\lambda > 0$.

Proof. By proposition 5.4.2 and the characterisation of $F_{\nu^{I'}}$ in (5.4.34), proximation (5.4.32) reads

$$\operatorname{argmin}_{I'} \frac{1}{2}\|I'^0 - I'\|_2^2 + \lambda \int_{\mathbb{R}} |1 - \left(\frac{1}{|\Omega|}\int_\Omega I'(x, \gamma) dx\right) - F_{\nu^0}(\gamma)| d\gamma. \tag{5.4.39}$$

Note that (5.4.39) is an independent optimization problem for each γ. Thus, for each γ we have to solve the problem

$$\operatorname{argmin}_{I'(\cdot,\gamma)} \frac{1}{2}\|I'^0(\cdot,\gamma) - I'(\cdot,\gamma)\|_2^2 + \lambda|1 - \left(\frac{1}{|\Omega|}\int_\Omega I'(x,\gamma)dx\right) - F_{\nu^0}(\gamma)|. \quad (5.4.40)$$

It can be easily verified that the solution to problem (5.4.40) is $I'^0(\cdot,\gamma) + c_\gamma$, where $c_\gamma \in \mathbb{R}$ and

$$c_\gamma \in \operatorname{argmin}_{c\in\mathbb{R}} \frac{1}{2}|\Omega|c^2 + \lambda|\frac{1}{|\Omega|}\int_\Omega I'^0(x,\gamma)dx + c + F_{\nu^0}(\gamma) - 1| \quad (5.4.41)$$

and hence

$$c_\gamma = \operatorname{shrink}\left(-\frac{1}{|\Omega|}\int_\Omega I'^0(x,\gamma) - F_{\nu^0}(\gamma) + 1, \frac{\lambda}{|\Omega|}\right) + \frac{1}{|\Omega|}\int_\Omega I'^0(x,\gamma)dx + F_{\nu^0}(\gamma) - 1. \quad (5.4.42)$$

□

For the discretized problem one just needs to replace integration with summation to obtain the proximation operator. Concluding, the cost for the Wasserstein proximal step is linear in the size of the input data.

Remark 5.4.6. We have seen in Proposition 5.4.2 that $W_1(\nu^{I'}, \nu^0) = \|HI' - (1 - F\nu^0)\|_1$, where H is an operator corresponding to a tight frame, i.e. $HH^* = |\Omega|^{-1}$, hence it is also possible to derive Proposition 5.4.3 by known rules for proximity operators involving composition with tight frames and translation.

5.4.11 Numerical Experiments

We want to show experimentally

1. that computational results conform to the mathematical model,
2. that the convex relaxation is reasonable.

Note that we do not claim to achieve the best denoising or inpainting results and we do not wish to compete with other state-of-the-art methods here. We point out again that the Wasserstein distance can be used together with other variational approaches to enhance their performance, e.g. with nonlocal total variation based denoising, see [27].

Remark 5.4.7. As detailed in Section 4.3, we lift our functional, so that it has one additional dimension, thereby increasing memory requirements and runtime of our algorithm. Non-convex approaches like [75] do not have such computational requirements. Still, the viability of the lifting approach we use was demonstrated in [73] for our variational model without the Wasserstein term. Also all additional operations our algorithm requires can be done very fast on recent graphic cards, hence the computational burden is tractable.

(a) The gray area is to be inpainted with partly black and white, with slightly more white.

(b) The circle in the middle has been inpainted with slightly more white as demanded by the Wasserstein term.

(c) The gray area is the area to be inpainted with a given Wasserstein prior favoring the gray area to be half black and half white.

(d) Inpainting result: we obtain a non-integral solution visualized by gray color.

Figure 5.2 - Examples illustrating tightness and failure of tightness of our relaxation (5.4.17).

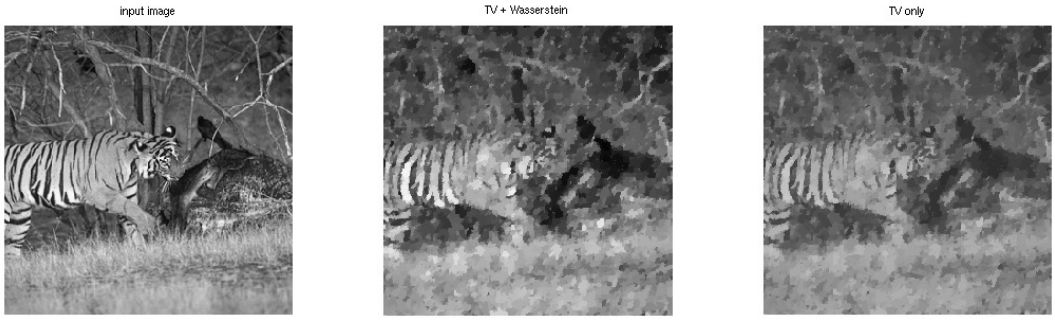
We have generally chosen the parameters λ, κ by hand to obtain reasonable results, if not stated differently.

In the **first experiment** we compare total variation denoising and total variation denoising with the Wasserstein term for incorporating prior knowledge. The data term is $f(s,x) = (I^0(x) - s)^2$, where I^0 is the noisy image in figure 5.1. The cost for the Wasserstein distance is $c(y^1, y^2) = \kappa|y^1 - y^2|,\ \kappa > 0$. To ensure a fair comparison, the parameter λ for total variation regularization *without* the Wasserstein term was *hand-tuned in all experiments* to obtain best results. The histogram was chosen to match the noiseless image. See Figure 5.1 for the results.

Note the trade-off one always has to make for pure total variation denoising: If one sets the regularization parameter λ high, the resulting grayvalue histogram of the recovered image will be similar to the noisy input image and generally far away from the histogram of ground truth. By choosing lower data fidelity and higher regularization strength we may obtain a valid geometry of the image, however then the grayvalue histogram tends to be peaked at one mode, as total variation penalizes scattered histograms and tries to draw the modes closer to each other, again letting the recovered grayvalue histogram being different from the desired one. By contrast, the Wasserstein prior in (5.4.3) guarantees a correct grayvalue histogram also with strong spatial regularization.

The **second** set of experiments illustrates where exactness of our relaxation may hold or fail, depending on the geometry of the level sets of solutions, see the Figures 5.2. The gray area is to be inpainted with a Wasserstein prior favoring the gray area to be partly black and partly white. Note that both settings illustrate cases, when the global Wasserstein term is indispensable, as otherwise there would be completely no control over how much of the area to be inpainted ends up being white or black. While our relaxation is not exact for the right experiment in Figure 5.2, thresholding at 0.5 still gives a reasonable result.

The **third** experiment is a more serious denoising experiment. Notice that again



(a) Tiger denoising experiment with the original image on the left, the image denoised with the Wasserstein term in the middle and the standard ROF-model on the right.

(b) Detailed view of the tiger denoising experiment revealing that contrast is better preserved when the Wasserstein term is used.

Figure 5.3 - Tiger denoising experiment

pure total variation denoising does not preserve the white and black areas well, but makes them gray, while the approach with the Wasserstein distance preserves the contrast better, see Figure 5.3.

In the **fourth experiment** we compare image inpainting with a total variation regularization term without prior knowledge and with prior knowledge, see Figure 5.4 for the results. The region where the data term is zero is enclosed in the blue rectangle. Outside the blue rectangle we employ a quadratic data term as in the first experiment. Total variation inpainting without the Wasserstein term does not produce the results we expected, as the total variation term is smallest, when the gray color fills most of the area enclosed by the blue rectangle. Heuristically, this is so because the total variation term weighs the boundary length multiplied by the difference between the gray value intensities, and a medium intensity minimizes this cost. Thus the TV-term tends to avoid interfaces, where high and low intensities meet, preferring smaller intensity changes, which can be achieved by interfaces with gray color on one side. Note that also the regularized image with the Wasserstein term lacks symmetry. This is also due to the behaviour of the TV-term described above.

In the **fifth** experiment we consider inpainting again. Yevgeni Khaldei, the

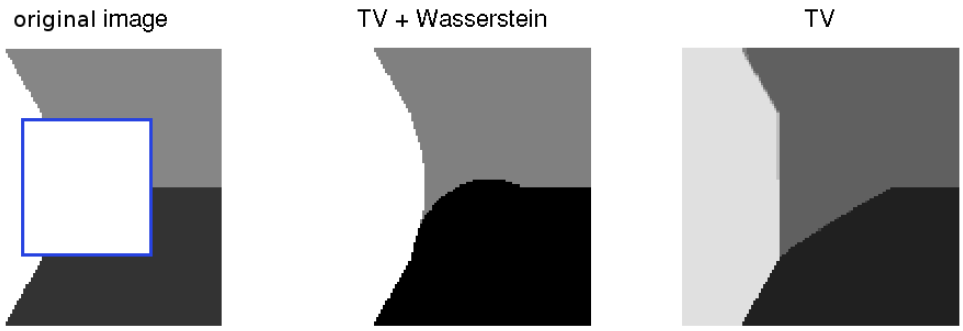


Figure 5.4 - Inpainting experiment with the original image and the inpainting area enclosed in a blue rectangle on the left, the inpainting result with the Wasserstein term in the middle and the result where only the TV-regularizer is used on the right. By enforcing the three regions to have the same size with the Wasserstein term, we obtain a better result than with the Total Variation term alone.

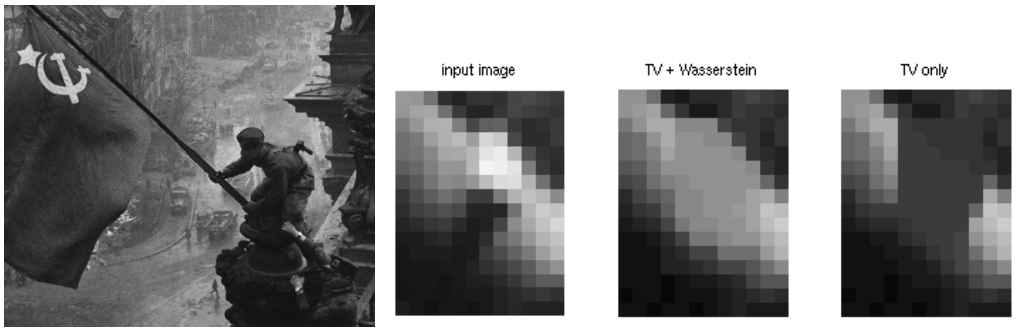


Figure 5.5 - Here we want to inpaint the area occupied by the watch of the soldier, see the second left image. Our approach, on the second right image gives better results again than the approach with TV alone.

photographer of the iconic picture shown on the left of Figure 5.5 had to remove the second watch. Trying to inpaint the wrist with a TV-regularizer and a Wasserstein term results in the middle picture, while only using a TV-regularizer results in the right picture. Clearly using the Wasserstein term helps.

In the **sixth** experiment we have a different setup. The original image is on the left of Figure 5.6. The histogram ν^0 was computed from a patch of clouds, which did not include the plane. The data term is $f(x, y) = \lambda \min(|I^0(x) - y|^2, \alpha)$, where $\alpha > 0$ is a threshold, so the data term does not penalize great deviances from the input image too strongly. The Wasserstein term penalizes the image of the plane whose appearance differs from the prior statistics. The TV-regularizer is weighted weaker than in the previous examples, because we do not want to smooth the clouds.

Note that unlike in ordinary inpainting applications, we did not specify the location of the plane beforehand, but the algorithm figured it out on its own. The total variation term finally favors a smooth inpainting of the area occupied by the plane. In essence we have combined two different tasks: Finding out where the plane is and inpainting that area occupied by it. See Figure 5.6 for results.

Figure 5.6 - *Unsupervised* inpainting using empirical measures as priors. Objects not conforming to the prior statistics are removed *without* labeling image regions.

5.4.12 Conclusion

We have presented in this section a novel method for variational image regularization, which takes into account global statistical information in one model. By solving a relaxation of the nonconvex problem we obtain regularizd images which conform to some global image statistics, which sets our method apart from standard variational methods. Moreover, the additional computational cost for the Wasserstein term we introduced is negligible, however our relaxation is not tight anymore as in models without the latter term. In our experiments the relaxation was seen to be tight enough for good results.

Our future work will consider extensions of the present approach to multidimensional input data and related histograms, e.g. based on color, patches or gradient fields. The theory developed in this section regarding the possible exactness of solutions does not carry over without modifications to such more complex settings. Moreover, it is equally important to find ways related to our present work to minimize such models efficiently.

5.5 Segmentation and Cosegmentation with the Wasserstein distance

In this section 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 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.

The work of this section is based upon publication [104].

5.5.1 Introduction

In this chapter we will treat the segmentation problem (4.2.1), which is

$$E(\Omega_1, \dots, \Omega_k) = \frac{1}{2} \sum_{i=1}^{k} \mathrm{Per}(\Omega_i; \Omega) + \sum_{i=1}^{k} \int_{\Omega_i} d^i(x) dx \,, \tag{5.5.1}$$

See [15, 61, 73] for treatments of this problem, including relaxations, discretizations and extensions of the minimization problem (5.5.1). In the case of two classes this is the well-known Continuous Cut segmentation model, see [18]. This model can be exactly solved by variational methods, see [19].

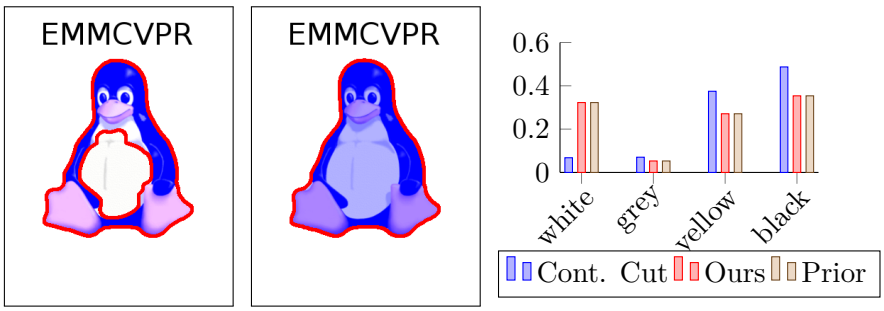
Often the potential functions $d^i(x) = -\log(p^i(I(x)))$ are chosen as the negative log-likelihood of some probability density p^i modelling the data. Using such potentials d^i poses in general the following problems:

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 5.7 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.



(a) Cont. Cut (b) Ours (c) Foreground histograms

***Figure 5.7* - 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).

We propose to solve the first and second of the stated problems by introducing the Wasserstein distance on global appearance measures. By using such a global term, we force each of the subsets Ω_i of the partition $(\Omega_1, \dots, \Omega_k)$ to have an appearance measure which is near a prespecified one. To approach the third problem, we use the Wasserstein distance to measure closeness between appearance measures of the common object in the two images and ensures that they are similar.

5.5.2 Related Work

5.5.2.1 Segmentation

Foreground/background segmentation with the Wasserstein distance was already proposed in the two papers [70] and [17].

Peyré et al. introduce in [70] 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 [17] boils down to the Continuous Cut model. The novelty is the computation of the local costs d^1 and d^2 from (5.5.1). 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 [17] 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 [17].

5.5.2.2 Cosegmentation

Rother et al. introduce in [83] 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 [106] 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 [106] 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 [106].

Another approach to cosegmentation is presented in [107], 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 section.

5.5.2.3 Numerical Follow-up Work

Recently, the authors in [74] have proposed a novel algorithm to solve the variational problems we have proposed faster.

5.5.3 Contribution

We present

- A new variational model for supervised segmentation with global appearance-based data-terms, see Section 5.5.4,
- a new variational model for unsupervised cosegmention of two images based on the similarity of the appearance measures of the respective cosegmentations, see Section 5.5.5,
- convex relaxations for both models together with efficient numerical schemes to minimize them, see Sections 5.5.6 and 5.5.7,
- experimental validation of the proposed approach, see Section 5.5.8.

5.5.4 Variational Model for Supervised Segmentation

We will combine into a single variational problem the spatial regularization from the minimal partition problem (5.5.1), appearance measures from subsets of the image domain constructed by (5.2.2) and the Wasserstein distance (5.1.2) for comparing the resulting measures to obtain a new model for segmenting images.

We assume in this setting that one image $I : \Omega \to \mathbb{V}$ and k probability measures ν^i over $\mathbb{V}$ are given. For a partition $(\Omega_1, \dots, \Omega_k)$ of Ω we enforce the measures $\nu^I_{\Omega_i}$ to be similar to the prespecified measures ν^i by using the Wasserstein distance (5.1.2) with k different similarity functions $c_1, \dots, c_k$.

Replacing the data term with the potential functions d^i in the minimal partition problem (5.5.1) by the Wasserstein distance yields

$$E_{seg}(\Omega_1, \dots, \Omega_k) = \frac{1}{2} \sum_{i=1}^{k} \mathrm{Per}(\Omega_i, \Omega) + \sum_{i=1}^{k} W\left(\nu^I_{\Omega_i}, |\Omega_i| \cdot \nu^i\right). \tag{5.5.2}$$

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 (5.1.1) of coupling measures Π is empty for measures of differing masses.

Minimizing (5.5.2) over all partitions $(\Omega_1, \dots, \Omega_k)$ of Ω results in partitions, which have regular boundaries due to the perimeter term, and the appearance measures of the partition $\nu^I_{\Omega_i}$ being similar to the given appearance measures ν^i. Note that the measures $\nu^I_{\Omega_i}$ depend on the partition through Ω_i.

As for the minimal partition problem in Section 4.2, we replace the sets Ω_i by indicator functions $u^i = \mathbb{1}_{\Omega_i}$ and minimize over them.

Proposition 5.5.1. *Let $u^i = \mathbb{1}_{\Omega_i}$. Then (5.5.2) is equal to*

$$J_{seg}(u) = \frac{1}{2} \sum_{i=1}^{k} \int_\Omega |Du^i| \, dx + \sum_{i=1}^{k} W\Big(\int_\Omega u^i(x) \delta_{I(x)} dx, \int_\Omega u^i(x) dx \cdot \nu^i \Big). \tag{5.5.3}$$

Minimizing (5.5.2) *over all partitions* $(\Omega_1, \dots, \Omega_k)$ *such that each* Ω_i *has a finite perimeter is equivalent to minimizing* (5.5.3) *over* $u \in \mathrm{BV}(\Omega, \mathcal{E}_k)$.

The functional $J_{seg}(\cdot)$ from (5.5.3) is convex, as the Total Variation term is convex and the Wasserstein term is so as well by Theorem 4.8 in [108]. However $\mathcal{E}_k$ is a nonconvex set, so taken together minimizing $\min_{u \in \mathrm{BV}(\Omega, \mathcal{E}_k)} J_{seg}(u)$ is not a convex problem. Thus, as in Section 4.2 and 5.3, we take the convex hull $\mathrm{BV}(\Omega, \Delta_k)$ of $\mathrm{BV}(\Omega, \mathcal{E}_k)$ to make the whole energy convex.

$$\inf_{u \in \mathrm{BV}(\Omega, \Delta_k)} J_{seg}(u)\,. \tag{5.5.4}$$

Remark 5.5.1. It is possible to introduce additional local costs $d^i : \Omega \to \mathbb{R}$ without compromising convexity of (5.5.3), i.e. to minimize

$$\inf_{u \in \mathrm{BV}(\Omega, \Delta_k)} J_{seg}(u) + \sum_{i=1}^{k} \int_\Omega d^i(x) u^i(x) dx\,. \tag{5.5.5}$$

Numerically it comes at a negligible cost to do so. However we chose not to use local costs to demonstrate most directly the power of the global Wasserstein cost.

Remark 5.5.2. (5.5.3) is the Continuous Cut model when we choose $k = 2$, two points $v_1, v_2 \in \mathbb{V}$ and $\nu^1 = \delta_{v_1}$ and $\nu^2 = \delta_{v_2}$, as then we can replace the Wasserstein distance by multiplication with a local data term. The resulting model is the minimal partition problem (5.5.1) for two classes. In [19] it is shown that a global minimizer of the non-relaxed problem can be obtained by thresholding.

5.5.5 Variational Model for Unsupervised Cosegmentation

Let two images $I_1, I_2 : \Omega \to \mathbb{V}$ be given and let c be some similarity function for the Wasserstein distancee. 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 $\nu^{I_1}_{\Omega_1}$ and $\nu^{I_2}_{\Omega_2}$ be the appearance measures of the common object in images I_1 and I_2 respectively. We know that both appearance measures should be similar. Therefore we will use the Wasserstein distance $W\left(\nu^{I_1}_{\Omega_1}, \nu^{I_2}_{\Omega_2}\right)$ as a penalization term for enforcing similarity of the appearance measures $\nu^{I_1}_{\Omega_1}$ and $\nu^{I_2}_{\Omega_2}$.

Consider the energy

$$E_{coseg}(\Omega_1, \Omega_2) = \sum_{i=1}^{2} \mathrm{Per}(\Omega_i, \Omega) + W\left(\nu^{I_1}_{\Omega_1}, \nu^{I_2}_{\Omega_2}\right) + \sum_{i=1}^{2} P \cdot |\Omega \backslash \Omega_i| \tag{5.5.6}$$

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 [106] and is needed to avoid the empty cosegmentation. Minimizing (5.5.6) 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

$\nu_{\Omega_1}^{I_1}$ and $\nu_{\Omega_2}^{I_2}$ are similar. Note that neither $\nu_{\Omega_1}^{I_1}$ nor $\nu_{\Omega_2}^{I_2}$ are known but completely depend on the segmentation.

The main difference between the segmentation model (5.5.2) and the cosegmentation model (5.5.6) 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 5.5.4 and Proposition 5.5.1, we can establish a similar correspondence between (5.5.6) and a suitable convex formulation in the space of indicator functions.

Proposition 5.5.2. *Let $u^i = \mathbb{1}_{\Omega_i}$. Then (5.5.6) is equal to*

$$J_{coseg}(u^1, u^2) = \begin{array}{l} \sum_{i=1}^{2} \int_\Omega |Du^i|\, dx + W\left(\int_\Omega u^1(x)\delta_{I_1(x)}dx, \int_\Omega u^2(x)\delta_{I_2(x)}dx\right) \\ + \sum_{i=1}^{2} P \cdot \int_\Omega (1 - u^i(x))\, dx \end{array}. \tag{5.5.7}$$

Minimizing $E_{coseg}(\Omega_1, \Omega_2)$ (5.5.6) 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 5.5.4, J_{coseg} is convex, whereas the space of $\{0,1\}$-valued functions is not. Relaxing to functions $u^i \in \mathrm{BV}(\Omega, [0,1])$, $i = 1, 2$ yields a convex relaxation.

Note that due to aggregating the appearance in the two measures $\nu_{\Omega_1}^{I_1}$ and $\nu_{\Omega_2}^{I_2}$ in a *translation-, rotation- and deformation-invariant* way, the resulting cosegmentation energy also exhibits these properties.

Remark 5.5.3. (5.5.6) 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 work.

5.5.6 Numerical Implementation with Proximal Algorithms

It is common to solve convex large-scale non-smooth problems with first order algorithms like [9, 16, 24]. 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 (5.5.3) and 3 such functions with an additional linear term for the cosegmentation functional (5.5.7). We use the Algorithm 3 and the technique from Section 5 to handle an arbitrary number of functions.

In practice our image domain is discrete. Here we assume discretization of Ω as in Section 4.4 The gradient operator will be approximated by forward differences.

We can rewrite the energy function (5.5.3) for the segmentation problem as follows by splitting variables for the gradient operator:

$$J_{seg}(u, g) = \chi_{\{\nabla u = g\}} + \chi_{\{u \in \Delta_k\}} + \|g\| + \sum_{i=1}^{k} W_{seg}^i(u^i)\,, \tag{5.5.8}$$

where $W^i_{seg}(u) = W\left(\sum_{x\in\Omega} u(x)\delta_{I(x)}, \left(\sum_{x\in\Omega} u(x)\right)\nu^i\right)$ are the Wasserstein terms in (5.5.3). The energy (5.5.7) for the cosegmentation problem can be split as follows:

$$J_{coseg}(u,g) = \sum_{i=1}^{2}\left\{\chi_{\{\nabla u^i = g^i\}} + \|g^i\|\right\} + \langle d, u\rangle + \chi_{\{u\in[0,1]^{|\Omega|}\}} + W_{coseg}(u^1,u^2)\,, \quad (5.5.9)$$

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 (5.5.7) and $\langle d, u\rangle$ takes care of the balloning term.

We solve (5.5.3) and (5.5.7) with [77], which is equivalent to using the Douglas-Rachford described in Algorithm 3 on a suitably defined product space as in Section 5 and absorbing the linear term arbitrarily. For this we need to evaluate efficiently the proximity operators for each convex function in (5.5.8) and (5.5.9). Proximity operators for all the convex functions in (5.5.8) and (5.5.9) except for the Wasserstein term can be computed very efficiently by standard methods:

- $\text{prox}_{\chi_{\{\nabla u = g\}}}(u^0, g^0)$ 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 [66].
- $\text{prox}_{\|g\|}(g^0)$ amounts to computing the shrinkage operator.

The Wasserstein proximity operator can be computed more efficiently with the technique detailed below.

5.5.6.1 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 $|\Omega|+|\mathbb{V}|^2$ variables. However due to symmetry we can significantly reduce the size of the quadratic program to $|\mathbb{V}|^2$ 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 $|\Omega| \gg |\mathbb{V}|$.

In the following we only discuss the segmentation case. Dimensionality reduction for the cosegmentation case works analoguously.

Due to the representation of the Wasserstein distance (5.1.2), 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 (5.5.8) can be written equivalently as

$$\begin{aligned} \operatorname{argmin}_{\{u,\pi\}} \quad & \textstyle\sum_{x\in\Omega}(u(x)-u^0(x))^2 + \int_{\mathbb{V}\times\mathbb{V}} c(v_1,v_2)\,d\pi(v_1,v_2) \\ \text{s.t.} \quad & \pi(\mathbb{V}\times A) = \textstyle\sum_{\{x\in I^{-1}(A)\}} u(x) \quad \forall A\subset\mathbb{V} \\ & \pi(B\times\mathbb{V}) = (\textstyle\sum_{x\in\Omega} u(x))\,\nu^i(B) \quad \forall B\subset\mathbb{V} \\ & \pi \geq 0 \end{aligned} \tag{5.5.10}$$

Note that the Wasserstein distance term above is invariant to permutations of values inside each set $\{I^{-1}(v)\}\ \forall v\in\mathbb{V}$. 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 $v\in\mathbb{V}$ and let $\nu^0 = \sum_{x\in\Omega} u^0(x)\delta_{I(x)}$. Consider the problem

$$\begin{aligned} \operatorname{argmin}_{\pi\in\mathcal{P}(\mathbb{V}\times\mathbb{V})} \quad & \textstyle\int_{\mathbb{V}} n_v\cdot\left(\pi(\mathbb{V}\times\{v\}) - \nu^0(\{v\})\right)^2 dv + \int_{\mathbb{V}\times\mathbb{V}} c(v_1,v_2)\,d\pi(v_1,v_2) \\ \text{s.t.} \quad & \pi(B\times\mathbb{V}) = \pi(\mathbb{V}\times\mathbb{V})\cdot\nu^1(B) \quad \forall B\subset\mathbb{V} \\ & \pi\geq 0 \end{aligned} \tag{5.5.11}$$

The relation between the two minimization problems (5.5.10) and (5.5.11) is:

Lemma 5.5.1. *The minimization problems* (5.5.10) *and* (5.5.11) *are equivalent in the following sense: For $I(x)=v\in\mathbb{V}$ the optimal solutions $\hat{u}$ of* (5.5.10) *and $\hat{\pi}$ of* (5.5.11) *correspond to each other via the relation*

$$\hat{u}(x) = u^0(x) + \frac{\hat{\pi}(\mathbb{V}\times\{v\}) - \nu^0(\{v\})}{n_v}\,. \tag{5.5.12}$$

Lemma 5.5.1 allows for efficiently solving (5.5.10) via (5.5.11) and (5.5.12).

5.5.7 Numerical Implementation with Message Passing

The proximal algorithms proposed in Section 5.5.6 can solve problems with large image domain and medium size Wasserstein distance term. The evaluation of the prox-operator for the Wasserstein distance is equal to a min-cost flow problem with quadratic costs, for which no fast practical schemes are known. Hence, the need to evaluate the prox-operator for the Wasserstein distance prohibits cost matrices of large dimension.

On the other hand, very efficient min-cost flow solvers exist for computing the Wasserstein distance, however the quadratic term in the prox-operator prohibits application of those. Hence, to address larger scale problem we propose to use message passing algorithms as introduced in Section 2.3.1, as these do not need to evaluate prox-operators. As a case study, we will investigate the cosegmentation problem (5.5.7). For this purpose, we extend the sequential message passing algorithm [48, 50] presented in Algorithm 1 for MAP-inference in pairwise MRFs.

The message passing scheme requires us to consider the discretized problem directly. Hence, we assume that the image domain has been discretized: $\Omega=\{1,\dots,n\}$. The TV-term in the discretized model amounts to Potts-terms and the ballooning term

results in unaries. Hence, excluding the Wasserstein term, problem (5.5.7) amounts to two separate MRFs $\mathcal{G}^i = (\mathcal{V}^i, \mathcal{E}^i, \theta^i)$, $i = 1, 2$, which we can solve with the standard local polytope relaxation. The respective pairwise marginals in the local polytope relaxation (2.2.7) will be denoted by μ^1 and μ^2 for the first and the second MRF problem.

The overall problem is

$$\begin{array}{rl} \min_{\mu^1,\mu^2} & \sum_{i=1}^{2} \left\{ \sum_{u\in\mathcal{V}^i} \langle \theta_u, \mu_u^i \rangle + \sum_{uv\in\mathcal{E}^i} \langle \theta_{uv}, \mu_{uv}^i \rangle \right\} + W(\nu^{I_1}_{\{u:\mu_u^1(0)=1\}}, \nu^{I_2}_{\{u:\mu_u^2(0)=1\}}) \\ \text{s.t.} & \mu^i \in \Lambda_{\mathcal{G}^i} \cap \{0,1\}^{\dim(\Lambda_{\mathcal{G}^i})}, \quad i = 1, 2 \end{array} \tag{5.5.13}$$

where $W(\cdot, \cdot)$ is the Wasserstein distance of the histograms, as in Definition 5.1.3 and $\nu^{I_1}_{\{u:\mu_u^1(0)=1\}}$ is the foreground area in image 1 as defined by the lifting process in Definition 5.2.2, similarly for image 2. Similarly to (5.5.7), the unaries $\theta_u^i = \begin{pmatrix} 0 \\ P \end{pmatrix}$ $i = 1, 2$ take into account the ballooning term and the pairwise potentials $\theta_{uv}^i = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $i = 1, 2$ the perimeter term. The label set is $X_u^i = \{0, 1\}$, $i = 1, 2$, label 0 denoting foreground and 1 background.

We can linearize the Wassestein distance term as follows: Define $\theta_W \in \mathbb{R}^{(n+1)\times(n+1)}$ by

$$\theta_W(u, v) = \begin{cases} c(I(u), I(v)), & u, v \leq n \\ \infty, & u < v = n + 1 \\ \infty, & v < u = n + 1 \\ 0, & u = v = n + 1 \end{cases}. \tag{5.5.14}$$

Then

$$W(\nu^{I_1}_{\{u:\mu_u^1(0)=1\}}, \nu^{I_2}_{\{u:\mu_u^2(0)=1\}}) = \left\{ \begin{array}{rll} \min_{\pi\in\mathbb{R}_+^{(n+1)\times(n+1)}} & \langle \theta_W, \pi \rangle & \\ & \text{s.t.} & \sum_{v=1}^{n+1} \pi(u, v) = \mu_u^1(0) \quad \forall u = 1, \ldots, n \\ & & \pi(u, v + 1) = \mu_u^1(1) \quad \forall u = 1, \ldots, n \\ & & \sum_{u=1}^{n+1} \pi(u, v) = \mu_v^2(0) \quad \forall v = 1, \ldots, n \\ & & \pi(n + 1, v) = \mu_v^2(1) \quad \forall v = 1, \ldots, n \end{array} \right\} \tag{5.5.15}$$

holds. Problem (5.5.15) is a linear assignment problem, which is a special case of the minimum cost flow problem on the complete bipartite graph with $n + 1$ edges in each component [4].

We propose two variants of message passing to optimize (5.5.15): The first solves the underlying min-cost flow problem for W with message passing, the second one relies on a dedicated combinatorial min cost flow solver.

5.5.7.1 Min-Cost Flow via Message Passing

Consider the relaxation

$$\begin{aligned}
\min_{\mu^1,\mu^2,\pi^1,\pi^2} \quad & \textstyle\sum_{i=1}^{2} \Big\{ \sum_{u\in\mathcal{V}^i} \langle \theta_u, \mu_u^i \rangle + \sum_{uv\in\mathcal{E}^i} \langle \theta_{uv}^i, \mu_{uv}^i \rangle + \langle \tfrac{1}{2}\theta_W, \pi^i \rangle \Big\} \\
\text{s.t.} \quad & \mu^i \in \Lambda_{\mathcal{G}^i}, \quad i = 1,2 \\
& \textstyle\sum_{v=1,\dots,n+1} \pi^1(u,v) = 1 \quad \forall u = 1,\dots,n \\
& \textstyle\sum_{u=1,\dots,n+1} \pi^2(u,v) = 1 \quad \forall v = 1,\dots,n \\
& \textstyle\sum_{v\in\mathcal{V}^2} \pi^1(u,v) = \mu_u^1(0), \quad \pi(u,n+1) = \mu_u^1(1) \quad \forall u \in \mathcal{V}^1 \\
& \textstyle\sum_{u\in\mathcal{V}^1} \pi^2(u,v) = \mu_v^2(0), \quad \pi(n+1,v) = \mu_v^2(1) \quad \forall v \in \mathcal{V}^2 \\
& \pi^1(u,v) = \pi^2(u,v)
\end{aligned} \tag{5.5.16}$$

Combining

$$\sum_{v=1}^{n+1} \pi^1(u,v) = \mu_u^1(0), \quad \pi^1(u,v+1) = \mu_u^1(1) \quad \forall u = 1,\dots,n \tag{5.5.17}$$

$$\sum_{u=1}^{n+1} \pi^2(u,v) = \mu_v^2(0), \quad \pi^2(n+1,v) = \mu_v^2(1) \quad \forall v = 1,\dots,n \tag{5.5.18}$$

$$\pi^1 = \pi^2 \tag{5.5.19}$$

$$\tag{5.5.20}$$

leads to

$$\sum_{v=1}^{n+1} \pi^i(u,v) = \mu_u^1(0), \quad \pi^i(u,v+1) = \mu_u^1(1) \quad \forall u = 1,\dots,n \tag{5.5.21}$$

$$\sum_{u=1}^{n+1} \pi^i(u,v) = \mu_v^2(0), \quad \pi^i(n+1,v) = \mu_v^2(1) \quad \forall v = 1,\dots,n \tag{5.5.22}$$

$$\tag{5.5.23}$$

for $i = 1,2$, hence (5.5.16) is a relaxation of (5.5.13).

Let the reparametrization analoguously to (2.2.10) be given by

$$\begin{aligned}
\theta_u^{i,\phi}(x_u) &= \theta_w^i(x_u) + \textstyle\sum_{v:uv\in\mathcal{E}^i} \phi_{uv}(x_u) + \phi_{uW}^i(x_u) \\
\theta_{uv}^{i,\phi}(x_{uv}) &= \theta_{uv}^i(x_{uv}) - \phi_{uv}(x_u) - \phi_{vu}(x_v) \\
\theta_W^{1,\phi}(u,v) &= \tfrac{1}{2}\theta_W(u,v) + \phi_W(u,v) + \begin{cases} \phi_{uW}^1(0), & u \le n \\ \phi_{uW}^1(1), & u = n+1 \end{cases} \\
\theta_W^{2,\phi}(u,v) &= \tfrac{1}{2}\theta_W(u,v) - \phi_W(u,v) + \begin{cases} \phi_{vW}^2(0), & v \le n \\ \phi_{vW}^2(1), & v = n+1 \end{cases}
\end{aligned} \tag{5.5.24}$$

The additional variables ϕ_W correspond to the constraint $\pi^1 = \pi^2$ and ϕ_{uW}^1 to $\sum_{v=1}^{n+1} \pi^1(u,v) = \mu_u^1(0)$, $\pi^1(u,v+1) = \mu_u^1(1)$, analoguously for ϕ_{vW}^2.

Then the dual problem reads

$$\begin{aligned} \max_{\phi} \quad & \sum_{i=1}^{2} \left\{ \sum_{v \in \mathcal{V}^i} \min_{x_u \in X_u} \{\theta^{i,\phi}\} \right\} \\ & + \sum_{u=1}^{n+1} \left\{ \min_{v=1,\dots,n+1} \{\tfrac{1}{2}\theta_W^{1,\phi}(u,v)\} \right\} + \sum_{v=1}^{n+1} \left\{ \min_{u=1,\dots,n+1} \{\tfrac{1}{2}\theta_W^{2,\phi}(u,v)\} \right\} \\ \text{s.t.} \quad & \theta_{uv}^{i,\phi} \geq 0 \quad \forall uv \in \mathcal{E}^i, i = 1,2 \end{aligned} \tag{5.5.25}$$

Problem (5.5.25) is solved via Algorithm 4.

Due to lemmata 2.3.1 and 2.3.2 we only have to prove that lines 6-8 and 15-24 of Algorithm 4 increase the dual lower bound. Lines 25-34 increase the dual bound analoguously.

Proposition 5.5.3. *Lines 6-8 and 15-24 of Algorithm 4 increase the dual lower bound* (5.5.25).

Proof. (i) In lines 6-8 the dual lower bound is non-decreasing Note that the change is ≥ 0, hence $\min_{v=1,\dots,n+1}\{\frac{1}{2}\theta_W^{1,\phi}(u,v)\}$ increases in (5.5.25). Moreover $\sum_{v=1}^{n+1}\left\{\min_{u=1,\dots,n+1}\{\frac{1}{2}\theta_W^{2,\phi}(u,v)\}\right\}$ stays the same in (5.5.25), as for each $v = 1,\dots,n+1$, the minimum $\min_{u=1,\dots,n+1}\{\frac{1}{2}\theta_W^{2,\phi}(u,v)\}$ is not decreased.

(ii) In line 15, there are two cases: 1. $\theta_W^{2,\phi}(u,v) = \min_{u=1,\dots,n+1}\theta_W^{2,\phi}$ or 2. $\theta_W^{2,\phi}(u,v) > \min_{u=1,\dots,n+1}\theta_W^{2,\phi}$. In the first case, the change $\theta_W^{2,\phi}(u,v) - \min_{v' \neq v}\theta_W^{2,\phi}(u,v')$ is non-positive, hence the reparametrization $\theta_W^{1,\phi}(u)$ may decrease. On the other hand, $\min_{v'=1,\dots,n+1}\theta_W^{2,\phi}(u,v')$ increases by the same amount, therefore $\min_{v'=1,\dots,n+1}\theta_W^{1,\phi}(u,v') + \min_{u'=1,\dots,n+1}\theta_W^{2,\phi}(u',v)$ is non-decreasing, hence the dual lower bound is non-decreasing. In the second case, the change $\theta_W^{2,\phi}(u,v) - \min_{v' \neq v}\theta_W^{2,\phi}(u,v')$ is non-negative, hence $\min_{v'=1,\dots,n+1}\theta_W^{1,\phi}(u,v')$ is non-decreasing. On the other hand, $\min_{u'=1,\dots,n+1}\theta_W^{2,\phi}(u',v)$ stays constant.

(iii) In lines 18-24 $\delta \geq 0$. Hence, the reparametrization of all connected factors is increased. On the other hand, due to the choice of ω, $\theta_W^{1,\phi} = \alpha$ after line 24, hence the lower bound for $\theta_u^{1,\phi}$ stays constant. □

5.5.7.2 Min-Cost Flow via Combinatorial Min-Cost Flow Solver

We will use a minimum cost flow solver to first solve the Wasserstein distance W and second, to compute new reparametrizations of W in (5.5.15). By this approach, we avoid the copy of the assignment matrix as in Section 5.5.7.1. We connect the Potts-MRFs with the min-cost flow problem similarly as in Section 5.5.7.1:

$$\begin{aligned} \sum_{v=1}^{n} \pi(u,v) &= \mu_u^1(1) \\ \pi(u,n+1) &= \mu_u^1(0)\,, \end{aligned} \tag{5.5.26}$$

and similary for μ^2.

Algorithm 4: One forward iteration of message passing for (5.5.16)

Input : Graphs $\mathcal{G}^i = (\mathcal{V}^i = \{1, \dots, n\}, \mathcal{E}^i)$, potentials $\theta^i_u, u \in \mathcal{V}^i$, $\theta^i_{uv}, uv \in \mathcal{E}^i$, $i = 1, 2$, Wasserstein cost $\theta_W \in \mathbb{R}^{(n+1)\times(n+1)}$.

for $i = 1, 2$ **do**
  for $u = 1, \dots, n$ **do**
    `// Receive message from pairwise potentials`
    for $v : uv \in \mathcal{E}^i, v < u$ **do**
      Compute $\phi^i_{uv}(x_u) \mathrel{-}= \min_{x_v \in X_v}\{\theta^{i,\phi}_{uv}(x_u, x_v)\} \quad \forall x_u \in X_u$
    end
    `// Receive message from Wasserstein potential`
    Compute $\alpha = \min_{v=1,\dots,n+1}\{\theta^{i,\phi}_W(u, v)\}$
    $\phi^i_{uW}(0) \mathrel{-}= \min_{v=1,\dots,n}\{\theta^{i,\phi}_W(u, v)\} - \alpha$
    $\phi^i_{uW}(1) \mathrel{-}= \theta^{i,\phi}_W(u, n+1) - \alpha$
    `// Send message to pairwise potentials`
    Compute $\delta^*(x_u) = \theta^{i,\phi}_u(x_u) - \min_{x'_u \in X_u}\{\theta^{i,\phi}_u(x'_u)\}$. Set $\omega = \frac{1}{1+|\{v:uv\in\mathcal{E}, v>u\}|}$.
    for $v : uv \in \mathcal{E}^i, v > u$ **do**
      Update $\phi^i_{(uv)} \mathrel{+}= \omega \cdot \delta^*$.
    end
  end
end
`// Process` $\theta^{1,\phi}_W$
`// Receive messages from` $\theta^{2,\phi}_W$
for $v = 1, \dots, n$ **do**
  Compute $\phi_W(u, v) \mathrel{+}= \theta^{2,\phi}_W(u, v) - \min_{v' \neq v} \theta^{2,\phi}_W(u, v')$.
end
`// Send messages`
Set $\omega = \frac{1}{n+2}$.
Set $\delta(v) = \theta^{1,\phi}_W(u, \cdot) - \min_{v=1,\dots,n+1}\{\theta^{1,\phi}_W(u, v)\}$.
$\phi_W(u, \cdot) \mathrel{-}= \omega \cdot \delta$.
for $u = 1, \dots, n$ **do**
  $\phi_{uW}(0) \mathrel{+}= \omega \cdot \min_{v=1,\dots,n}\{\delta(v)\}$.
  $\phi_{uW}(1) \mathrel{+}= \omega \cdot \delta(n+1)$.
end
`// Process` $\theta^{2,\phi}_W$
`// Receive messages from` $\theta^{1,\phi}_W$
for $u = 1, \dots, n+1$ **do**
  Compute $\phi_W(u, v) \mathrel{-}= \theta^{1,\phi}_W(u, v) - \min_{u' \neq u} \theta^{1,\phi}_W(u', v)$.
end
`// Send messages`
Set $\omega = \frac{1}{n+2}$.
Set $\delta(u) = \theta^{2,\phi}_W(u, \cdot) - \min_{u=1,\dots,n+1}\{\theta^{2,\phi}_W(u, v)\}$.
$\phi_W(\cdot, v) \mathrel{-}= \omega \cdot \delta$.
for $v = 1, \dots, n$ **do**
  $\phi_{vW}(0) \mathrel{+}= \omega \cdot \min_{u=1,\dots,n}\{\delta(u)\}$.
  $\phi_{vW}(1) \mathrel{+}= \omega \cdot \delta(n+1)$.
end

The whole inference problem amounts to

$$\begin{array}{rl} \min_{\mu^1,\mu^2,\pi} & \sum_{i=1}^{2}\{\sum_{u\in\mathcal{V}^i}\langle\theta_u,\mu_u^i\rangle+\sum_{uv\in\mathcal{E}^i}\langle\theta_{uv}^i,\mu_{uv}^i\rangle\}+\langle\theta_W,\pi\rangle \\ \text{s.t.} & \mu^i\in\Lambda_{\mathcal{G}^i},\quad i=1,2 \\ & \sum_{v\in\mathcal{V}^2}\pi(u,v)=\mu_u^1(0),\quad \pi(u,n+1)=\mu_u^1(1)\quad\forall u\in\mathcal{V}^1 \\ & \sum_{u\in\mathcal{V}^1}\pi(u,v)=\mu_v^2(0),\quad \pi(n+1,v)=\mu_v^2(1)\quad\forall v\in\mathcal{V}^2 \end{array} \tag{5.5.27}$$

Introduce again dual variables ϕ_{uW}^i, $i=1,2$ for the constraints (5.5.26) and dual variables ϕ_{uv}^i, $i=1,2$ for the constraints defining $\Lambda_{\mathcal{G}^i}$. Then define the reparametrization similarly as in (2.2.10) to be

$$\begin{aligned} \theta_u^{i,\phi}(x_u) &= \theta_u^i(x_u)+\textstyle\sum_{v:uv\in\mathcal{E}^1}\phi_{uv}^i(x_u)+\phi_{uW}^i(x_u) \\ \theta_{uv}^{i,\phi}(x_{uv}) &= \theta_{uv}^i(x_{uv})-\phi_{uv}^i(x_u)-\phi_{vu}^i(x_v) \\ \theta_W^\phi(u,v) &= \theta_W(u,v)-\begin{cases}\phi_{uW}^1(u)(0)+\phi_{vW}^2(v)(0), & u,v\le n\\ \phi_{uW}^1(u)(0)+\phi_{vW}^2(v)(1), & u<v=n+1\\ \phi_{uW}^1(u)(1)+\phi_{vW}^2(v)(0), & v<u=n+1\\ \phi_{uW}^1(u)(1)+\phi_{vW}^2(v)(1), & u=v=n+1\end{cases} \end{aligned} \tag{5.5.28}$$

The dual problem reads

$$\begin{array}{rl} \max_{\phi} & \sum_{i=1}^{2}\left\{\sum_{v\in\mathcal{V}^i}\min_{x_u\in X_u}\{\theta^{i,\phi}\}\right\}+\min_{\pi\in\mathbb{R}_+^{(n+1)\times(n+1)}:\pi\mathbb{1}=\mathbb{1},\pi^\top\mathbb{1}=\mathbb{1}}\langle\theta_W^\phi,\pi\rangle \\ \text{s.t.} & \theta_{uv}^{i,\phi}\ge 0\quad\forall uv\in\mathcal{E}^i, i=1,2 \end{array} \tag{5.5.29}$$

Problem (5.5.29) is solved via Algorithm 5.

Proposition 5.5.4. *Algorithm 5 increases the dual lower bound* (5.5.29).

Proof. Due to lemmata 2.3.1 and 2.3.2 it only remains to prove that (5.5.33) increases (5.5.29). The update operations (5.5.33) result in positive changes to ϕ_{uW}^i by construction of (5.5.32) as well. Hence, $\min_{x_u\in X_u}\{\theta_u^{i,\phi}(x_u)\}$ increases by definition of reparametrization (5.5.28).

Second, by (5.5.33), denote by $\tilde{\theta}_W^\phi$ the potential θ_W^ϕ after executing Algorithm 5. It holds that $\theta_W^\phi(u,v)\ge-\lambda^*(u)+\lambda'^*(v)$. It follows that

$$\begin{array}{rl} & \min_{\pi\in\mathbb{R}^{(n+1)\times(n+1)}:\pi\mathbb{1}=\mathbb{1},\pi^\top\mathbb{1}=\mathbb{1}}\sum_{u,v=1}^{n+1}\pi(u,v)\cdot\tilde{\theta}_W^\phi(u,v) \\ \ge & \min_{\pi\in\mathbb{R}^{(n+1)\times(n+1)}:\pi\mathbb{1}=\mathbb{1},\pi^\top\mathbb{1}=\mathbb{1}}\sum_{u,v=1}^{n+1}\pi(u,v)\cdot(-\lambda^*(u)+\lambda'^*(v)) \\ = & \sum_{u=1}^{n+1}(-\lambda^*(u))+\sum_{v=1}^{n+1}\lambda'^*(v)) \end{array} \tag{5.5.30}$$

The last equality is due to $\sum_{v=1}^{n+1}\pi(u,v)=1=\sum_{u=1}^{n+1}\pi(u,v)$. Therefore the last line above is independent of π. □

Remark 5.5.4. One can see that (5.5.32) is a totally unimodular problem and can be solved with a minimum cost network flow solver. Therefore, Algorithm 5 can be implemented efficiently. In particular, we use the primal network simplex solver from the LEMON project [62] to compute all needed minimum cost flow problems.

Algorithm 5: One forward iteration of message passing for (5.5.27)

Input : Graphs $\mathcal{G}^i = (\mathcal{V}^i = \{1, \dots, n\}, \mathcal{E}^*)$, potentials $\theta^i_u, u \in \mathcal{V}^i$, $\theta^i_{uv}, uv \in \mathcal{E}^i$, $i = 1, 2$, Wasserstein cost $\theta_W \in \mathbb{R}^{(n+1)\times(n+1)}$.

for $i = 1, 2$ **do**
  for $u = 1, \dots, n$ **do**
    Receive Messages:
    for $v : uv \in \mathcal{E}^i, v < u$ **do**
      Compute $\phi^i_{uv}(x_u) \mathrel{-}= \min_{x_v \in X_v} \{\theta^{i,\phi}_u v(x_u, x_v)\} \quad \forall x_u \in X_u$
    end
    Send Messages:
    Compute $\delta^*(x_u) = \theta^{i,\phi}_u(x_u) - \min_{x'_u \in X_u} \{\theta^{i,\phi}_u(x'_u)\}$. Set $\omega = \frac{1}{1+|\{v:uv\in\mathcal{E}, v>u\}|}$.
    `// Message to pairwise potentials`
    for $v : uv \in \mathcal{E}^i, v > u$ **do**
      Update $\phi^i_{(uv)} \mathrel{+}= \omega \cdot \delta^*$.
    end
    `// Message to Wasserstein potential`
    Update $\phi^i_{(uW)} \mathrel{+}= \omega \cdot \delta^*$.
  end
end
`// Message from Wasserstein potential`
Compute

$$\begin{aligned} \pi^* \in \operatorname{argmin}_{\pi \in \mathbb{R}^{(n+1)\times(n+1)}_+} \quad & \langle \theta^\phi_W, \pi \rangle \\ \text{s.t.} \quad & \pi \mathbb{1} = \mathbb{1}, \quad \pi^\top \mathbb{1} = \mathbb{1} \end{aligned} \tag{5.5.31}$$

$$\begin{aligned} \lambda^*, \lambda'^* \in \operatorname{argmin}_{\lambda, \lambda' \in \mathbb{R}^{(n+1)}_+} \quad & \langle \mathbb{1}, \lambda \rangle - \langle \mathbb{1}, \lambda' \rangle \\ \text{s.t.} \quad & \theta^\phi_W(i,j) + \lambda(i) - \lambda'(j) \begin{cases} = 0, & \pi^*(i,j) = 1 \\ \geq 0, & \pi^*(i,j) = 0 \end{cases} \end{aligned} \tag{5.5.32}$$

Set

$$\begin{aligned} \phi^1_u(1) &\mathrel{-}= \max_{v=0,\dots,n} \frac{1}{2}\theta^\phi_W(u,v) + \lambda^*(u) - \lambda'^*(v) \\ \phi^1_u(0) &\mathrel{-}= \frac{1}{2}\theta^\phi_W(i,j) + \lambda^*(i) - \lambda'^*(j) \\ \phi^2_v(1) &\mathrel{-}= \max_{u=0,\dots,n} \frac{1}{2}\theta^\phi_W(i,j) + \lambda^*(u) - \lambda'^*(v) \\ \phi^2_v(0) &\mathrel{-}= \frac{1}{2}\theta^\phi_W(i,j) + \lambda^*(u) - \lambda'^*(v) \end{aligned} \tag{5.5.33}$$

5.5.8 Numerical 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 $\mathbb{V}$ 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 section, 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 [27] for the latter.

5.5.8.1 Segmentation

In our experimental setting we assume that we have probability measures ν^1, ν^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 (5.5.5), e.g. by $d^i(x) = -\log(p^i(I(x)))$, where p^i is the density of ν^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 [15, 19, 61, 73] 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 ν^1, ν^2 from them. In a preprocessing step, we clustered the color values of the image by the k-means method [63]. The number of prototypes was set to 50. The quadratic problem in the prox-step (5.5.11) 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 5.8. 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.

5.5.8.2 Cosegmentation

For cosegmentation we first subdivide the image into superpixels with SLIC [3]. 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.

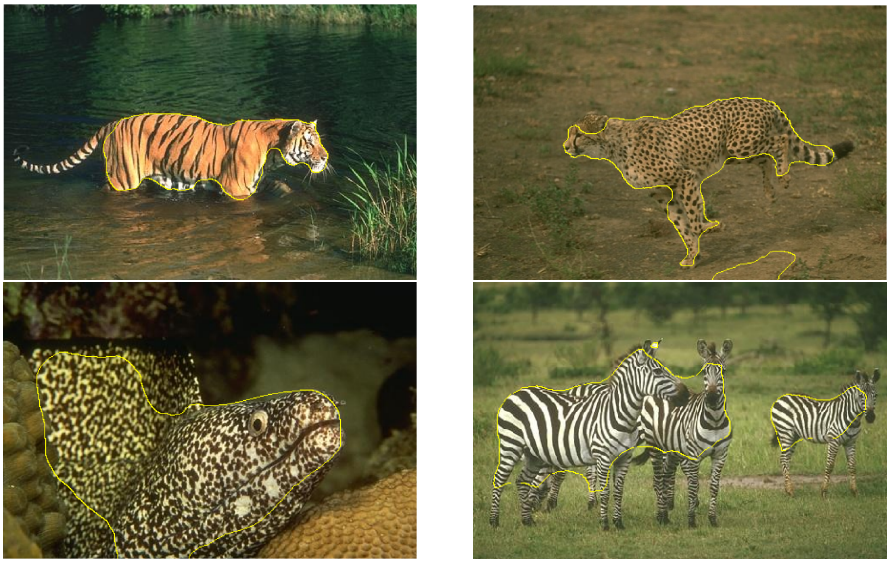


Figure 5.8 - Supervised segmentation experiments with global segmentation-dependent data term using the Wasserstein distance. Note that because the results correspond to *global* optima of a single convex functional, undesired parts of the partition are solely due to the – in our case: simple color – features and the corresponding prior appearance measures.

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 5.5.6.1.

Four experiments can be seen in figure 5.9. The foreground objects were taken from the dataset [14]. 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.

5.5.8.3 Comparison of Algorithms for the Cosegmentation Problem

The proximal algorithm described in Section 5.5.6 is slow due to the need to evaluate the proximal term for the Wasserstein distance. This amounts to a minimum cost flow problem with quadratic costs, for which no practically fast algorithm is known. We solve this problem with the quadratic program solver from the MOSEK [1] software suite. Still, for medium-sized problems, evaluating the Wasserstein proximal term takes more than one minute.

Hence we show below runtimes of the faster message-passing algorithms from Section 5.5.7. In Figure 5.10 we compare the two message-passing algorithms regarding iteration agains energy, while in Figure 5.11 we compare energies against the time taken to compute them. We ran the message passing algorithms until the dual lower bound was not increased anymore. Note that the x-axis is in log-scale for easier comparison in both Figures 5.10 and 5.11.

Figure 5.9 - 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 knowledge at all was used in these unsupervised experiments.

Remark 5.5.5. The message passing algorithm utilizing a dedicated min-cost flow solver from Section 5.5.7.2 beats the one using message passing for the Wasserstein distance from Section 5.5.7.1. This is due to the fact that the min-cost flow solver propagates information much faster: It need not coordinate between two copies of the assignment, each of which only incorporates part of the constraint, as the message-passing algorithm from Section 5.5.7.1 does.

Remark 5.5.6. While the proximal algorithm from Section 5.5.6 is currently not competitive with regard to runtime, this may change upon implementation of an efficient quadratic minimum cost solver. Also the work [74] presents an alternative primal-dual algorithm which is faster then our proximal algorithm for problems with Wasserstein distances.

5.5.9 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. Future work will focus on extending the proposed models by also including more spatial information. It is obvious, that e.g. cosegmented objects should have a similar spatial layout or shape. For example, it should be possible to find a registration between them. For segmentation, we would like to include more spatial information as well into our prior knowledge, e.g. we would like to encode the spatial arrangement of the objects we want to segment.

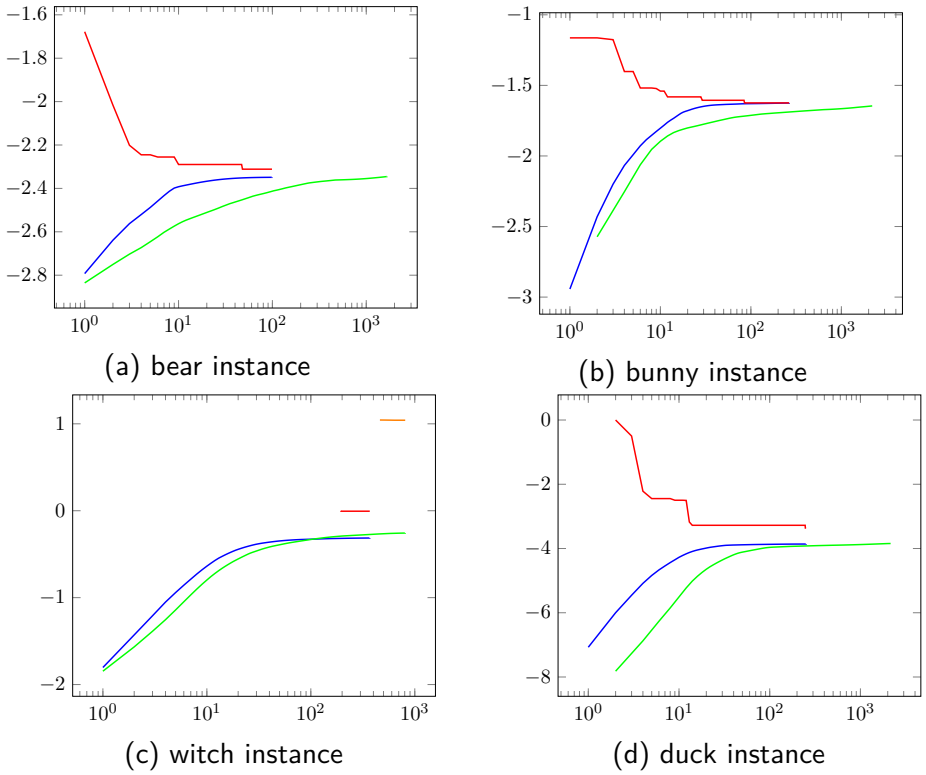
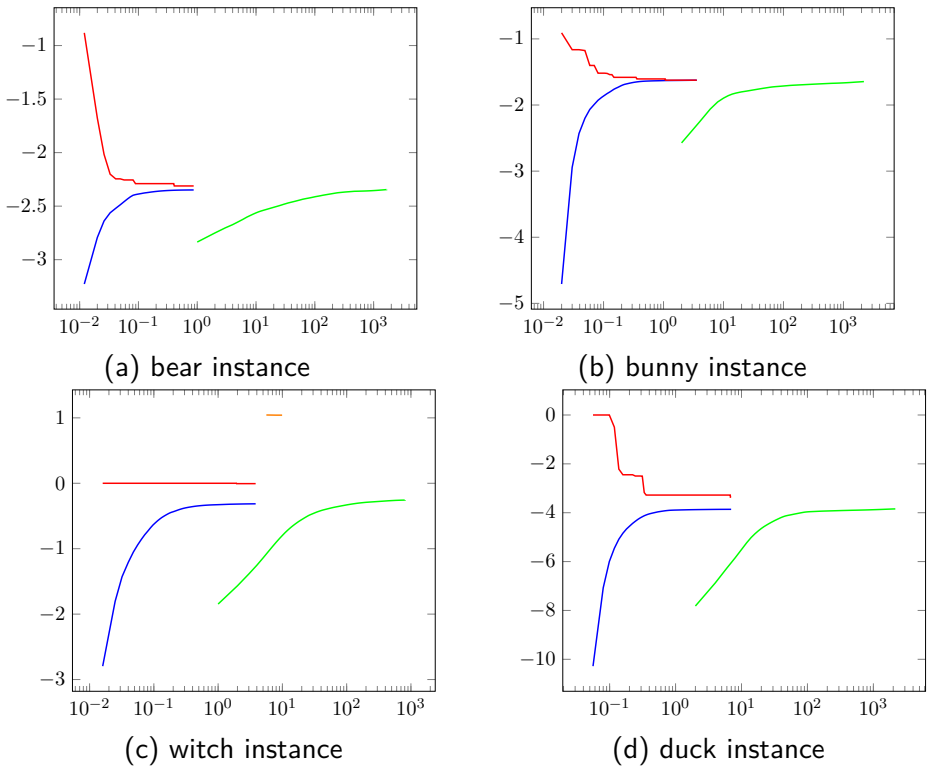


Figure 5.10 - Convergence plot comparing energies (vertical scale) against iterations (horizontal scale) for the presented algorithms for the four cosegmentation problem instances from Figure 5.9. The red line denotes the primal bound obtained by message passing utilizing a min-cost flow solver, see Section 5.5.7.2, the blue line its lower bound. The green line denotes the lower bound of message passing without utilization of a min-cost flow solver, as in Section 5.5.7.1. The rounding of this algorithm did not produce feasible primal solutions, except for the witch instance, were it is indicated by the orange line.



(a) bear instance (b) bunny instance

(c) witch instance (d) duck instance

Figure 5.11 - Convergence plot comparing energies (vertical scale) against runtime in seconds (horizontal scale) for the presented algorithms for the four cosegmentation problem instances from Figure 5.9. Colors have the same meaning as in Figure 5.10.

6 Conclusion

In this thesis we presented two extensions of MAP-inference.

First, we addressed in Chapter 3 the partial optimality problem and proposed a way to utilize standard suboptimal solvers for MAP-inference to obtain part of globally optimal solutions. The runtime of our proposed method is dependent on the used approximate solver. Advances in this area will lead to faster inference also for partial optimality. Also solvers based on tighter relaxations will benefit our persistency approach and lead to a larger number of persistently labelled variables.

Second, we extended standard labeling problems corresponding to pairwise MRFs with a higher order term based on the Wasserstein distance in Chapter 5. We covered three applications scenarios: denoising (in Section 5.4), segmentation and cosegmentation (in Section 5.5). Those tasks all either benefitted or were made possible by the Wasserstein distance. It is noteworthy, that the additional Wasserstein distance terms for those problems are based on a common framework. It is therefore reasonable to predict that the Wasserstein distance can be used in many more applications.

Bibliography

[1] The MOSEK optimization software. http://www.mosek.com/.

[2] The probabilistic inference challenge (PIC2011). http://www.cs.huji.ac.il/project/PASCAL/, 2011.

[3] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Trans. Patt. Anal. Mach. !ntell.*, 34(11), 2012.

[4] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[5] G. Alberti, G. Bouchitte, and G. Dal Maso. The calibration method for the Mumford-Shah functional and free-discontinuity problems. *Journal: Calc. Var. Partial Differential Equations*, 16:299–333, 2003.

[6] L. Ambrosio, N. Fusco, and D. Pallara. *Functions of Bounded Variation and Free Discontinuity Problems (Oxford Mathematical Monographs)*. Oxford University Press, USA, May 2000.

[7] P. Billingsley. *Probability and Measure*. Wiley Series in Probability and Statistics. Wiley, 1995.

[8] E. Boros and P. L. Hammer. Pseudo-Boolean optimization. *Discrete Applied Mathematics*, 123(1–3):155–225, 2002.

[9] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learning*, 3(1):1–122, 2010.

[10] Y. Boykov and V. Kolmogorov. Computing geodesics and minimal surfaces via graph cuts. In *ICCV*, pages 26–33. IEEE Computer Society, 2003.

[11] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137, Sept 2004.

[12] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(11):1222–1239, November 2001.

[13] E. S. Brown, T. F. Chan, and X. Bresson. A convex relaxation method for a class of vector-valued minimization problems with applications to mumford-shah segmentation. *CAM Report*.

[14] J. Wang M. Gelautz P. Kohli P. Rott C. Rhemann, C. Rother. alpha matting evaluation website.

[15] A. Chambolle, D. Cremers, and T. Pock. A convex approach to minimal partitions. *SIAM J. Imag. Sci.*, 5(4):1113–1158, 2012.

[16] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.

[17] T. F. Chan, S. Esedoglu, and K. Ni. Histogram based segmentation using Wasserstein distances. In Fiorella Sgallari, Almerico Murli, and Nikos Paragios, editors, *SSVM*, volume 4485 of *Lecture Notes in Computer Science*, pages 697–708. Springer, 2007.

[18] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Trans. Imag. Proc.*, 10(2):266–277, 2001.

[19] T.F. Chan, S. Esedoglu, and M. Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM J. Appl. Math.*, 66(5):1632–1648, 2006.

[20] P.-C. Combettes, P. L. Combettes. Proximal splitting methods in signal processing. In R.S.; Combettes-P.L.; Elser V.; Luke D.R.; Wolkowicz H. (Eds.) Bauschke, H.H.; Burachik, editor, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. Springer, 2011.

[21] P. L Combettes and J.-C. Pesquet. A douglas–rachford splitting approach to nonsmooth convex variational signal recovery. *Selected Topics in Signal Processing, IEEE Journal of*, 1(4):564–574, 2007.

[22] M. C. Cooper and S. Zivny. Tractable triangles and cross-free convexity in discrete optimisation. *J. Artif. Intell. Res. (JAIR)*, 44:455–490, 2012.

[23] J. Desmet, M. D. Maeyer, B. Hazes, and I. Lasters. The dead-end elimination theorem and its use in protein side-chain positioning. *Nature*, 356(6369):539–542, April 1992.

[24] J. Eckstein and D. P. Bertsekas. On the DouglasâĂŤRachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55:293–318, 1992.

[25] S. Ferradans, G-S. Xia, G. Peyré, and J-F. Aujol. Optimal transport mixing of gaussian texture models. In *Proc. SSVM'13*, 2013.

[26] A. Fix, A. Gruber, E. Boros, and R. Zabih. A graph cut algorithm for higher-order Markov random fields. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 1020–1027, Washington, DC, USA, 2011. IEEE Computer Society.

[27] G. Gilboa and S. Osher. Nonlocal operators with applications to image processing. *Multiscale Modeling & Simulation*, 7(3):1005–1028, 2008.

[28] A. Globerson and T. Jaakkola. Fixing max-product: Convergent message passing algorithms for MAP LP-relaxations. In John C. Platt, Daphne Koller, Yoram Singer, and Sam T. Roweis, editors, *NIPS*. Curran Associates, Inc., 2007.

[29] A. Globerson and T. Jaakkola. Convergent propagation algorithms via oriented trees. *CoRR*, abs/1206.5243, 2012.

[30] I. Gridchyn and V. Kolmogorov. Potts model, parametric maxflow and k-submodular functions. In *ICCV*, 2013.

[31] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*, volume 2 of *Algorithms and Combinatorics*. Springer, 1988.

[32] P.L. Hammer, P. Hansen, and B. Simeone. Roof duality, complementation and persistency in quadratic 0-1 optimization. *Math. Programming*, 28:121–155, 1984.

[33] T. Hazan and A. Shashua. Convergent message-passing algorithms for inference over general graphs with convex free energies. In D. A. McAllester and P. Myllymäki, editors, *UAI*, pages 264–273. AUAI Press, 2008.

[34] T. Hazan and A. Shashua. Norm-product belief propagation: Primal-dual message-passing for approximate inference. *IEEE Transactions on Information Theory*, 56(12):6294–6316, 2010.

[35] D. Hoiem, A. A. Efros, and M. Hebert. Recovering surface layout from an image. *Int. J. Comput. Vision*, 75(1):151–172, October 2007.

[36] ILOG, Inc. ILOG CPLEX: High-performance software for mathematical programming and optimization, 2014. See http://www.ilog.com/products/cplex/.

[37] H. Ishikawa. Transformation of general binary MRF minimization to the first-order case. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(6):1234–1249, June 2011.

[38] A. Jaimovich, G. Elidan, H. Margalit, and N. Friedman. Towards an integrated protein-protein interaction network: A relational markov network approach. *Journal of Computational Biology*, 13(2):145–164, 2006.

[39] F. Kahl and P. Strandmark. Generalized roof duality. *Discrete Applied Mathematics*, 160(16-17):2419–2434, 2012.

[40] J. H. Kappes, B. Andres, F. A. Hamprecht, C. Schnörr, S. Nowozin, D. Batra, S. Kim, B. X. Kausler, J. Lellmann, N. Komodakis, and C. Rother. A comparative study of modern inference techniques for discrete energy minimization problem. In *CVPR*, 2013.

[41] J. H. Kappes, B. Andres, F. A. Hamprecht, C. SchnÃűrr, S. Nowozin, D. Batra, S. Kim, B. X. Kausler, T. Kröger, J. Lellmann, N. Komodakis, B. Savchynskyy, and C. Rother. A comparative study of modern inference techniques for structured discrete energy minimization problems. *International Journal of Computer Vision*, 115(2):155–184, 2015.

[42] J. H. Kappes, B. Savchynskyy, and C. Schnörr. A bundle approach to efficient MAP-inference by Lagrangian relaxation. In *CVPR 2012*, 2012.

[43] J. H. Kappes, M. Speth, G. Reinelt, and C. Schnörr. Towards efficient and exact MAP-inference for large scale discrete computer vision problems via combinatorial optimization. In *CVPR*, 2013.

[44] B. X. Kausler, M. Schiegg, B. Andres, M. S. Lindner, U. KÃűthe, H. Leitte, J. Wittbrodt, L. Hufnagel, and F. A. Hamprecht. A discrete chain graph model for 3d+t cell tracking with high misdetection robustness. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *ECCV (3)*, volume 7574 of *Lecture Notes in Computer Science*, pages 144–157. Springer, 2012.

[45] L. G. Khachiyan. A polynomial algorithm in linear programming. *Doklady Akademii Nauk SSSR*, 244:1093–1096, 1979.

[46] M. Klodt, T. Schoenemann, K. Kolev, M. Schikora, and D. Cremers. An experimental comparison of discrete and continuous shape optimization methods. In *Proceedings of the 10th European Conference on Computer Vision: Part I*, ECCV '08, pages 332–345, Berlin, Heidelberg, 2008. Springer-Verlag.

[47] P. Kohli, A. Shekhovtsov, C. Rother, V. Kolmogorov, and P. Torr. On partial optimality in multi-label MRFs. In *Proceedings of the 25th international conference on Machine learning*, ICML '08, pages 480–487, New York, NY, USA, 2008. ACM.

[48] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1568–1583, October 2006.

[49] V. Kolmogorov. Generalized roof duality and bisubmodular functions. *Discrete Applied Mathematics*, 160(4-5):416–426, 2012.

[50] V. Kolmogorov. Reweighted message passing revisited. *ArXiv e-prints*, 2013.

[51] V. Kolmogorov and C. Rother. Minimizing nonsubmodular functions with graph cuts-a review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(7):1274–1279, July 2007.

[52] V. Kolmogorov, J. Thapper, and S. Zivny. The power of linear programming for general-valued csps. *SIAM J. Comput.*, 44(1):1–36, 2015.

[53] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2):147–159, 2004.

[54] N. Komodakis, N. Paragios, and G. Tziritas. MRF optimization via dual decomposition: Message-passing revisited. In *ICCV*, pages 1–8, 2007.

[55] N. Komodakis, N. Paragios, and G. Tziritas. MRF energy minimization and beyond via dual decomposition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(3):531–552, 2011.

[56] N. Komodakis and G. Tziritas. Approximate labeling via graph cuts based on linear programming. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(8):1436–1453, August 2007.

[57] I. Kovtun. Partial optimal labeling search for a NP-hard subclass of (max,+) problems. In Bernd Michaelis and Gerald Krell, editors, *DAGM-Symposium*, volume 2781 of *Lecture Notes in Computer Science*, pages 402–409. Springer, 2003.

[58] I. Kovtun. Sufficient condition for partial optimality for (max, +) labeling problems and its usage. *Control Systems and Computers*, (2):35–42, 2011. Special issue.

[59] S. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems (with discussion). *Journal of the Royal Statistical Society series B*, 50:157–224, 1988.

[60] J. Lellmann, B. Lellmann, F. Widmann, and C. Schnörr. Discrete and Continuous Models for Partitioning Problems. *Int. J. Comp. Vision*, 104(3):241–269, 2013.

[61] J. Lellmann and C. Schnörr. Continuous multiclass labeling approaches and algorithms. *SIAM J. Imag. Sci.*, 4(4):1049–1096, 2011.

[62] LEMON – library for efficient modeling and optimization in networks. `http://lemon.cs.elte.hu/`.

[63] J. MacQueen. Some methods for classification and analysis of multivariate observations. Proc. 5th Berkeley Symp. Math. Stat. Probab., Univ. Calif. 1965/66, 1, 281-297 (1967)., 1967.

[64] A. Martins, M. Figueiredo, P. Aguiar, N. Smith, and E. Xing. Ad3: Alternating directions dual decomposition for map inference in graphical models. *Journal of Machine Learning Research*, 16:495–545, 2015.

[65] T. Meltzer, A. Globerson, and Y. Weiss. Convergent message passing algorithms - a unifying view. In *Proceedings of the Twenty-Fifth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-09)*, pages 393–401, Corvallis, Oregon, 2009. AUAI Press.

[66] C. Michelot. A finite algorithm for finding the projection of a point onto the canonical simplex of rn. *J. Optim. Theory Appl.*, 50(1):195–200, July 1986.

[67] G. L. Nemhauser and L. E. Trotter. Vertex packings: Structural properties and algorithms. *Mathematical Programming*, 8:232–248, 1975. 10.1007/BF01580444.

[68] N. Paragios, Y. Chen, and O. Faugeras, editors. *The Handbook of Mathematical Models in Computer Vision*. Springer, 2006.

[69] N. Parikh and S. Boyd. Proximal algorithms. 2013.

[70] G. Peyré, J. Fadili, and J. Rabin. Wasserstein active contours. In *Proc. ICIP'12*, 2012.

[71] T. Pock, D. Cremers, H. Bischof, and A. Chambolle. An algorithm for minimizing the mumford-shah functional. In *ICCV*, pages 1133–1140. IEEE, 2009.

[72] T. Pock, D. Cremers, H. Bischof, and A. Chambolle. Global solutions of variational models with convex regularization. *SIAM J. Imaging Sciences*, 3(4):1122–1145, 2010.

[73] T. Pock, T. Schoenemann, G. Graber, H.Bischof, and D. Cremers. A convex formulation of continuous multi-label problems. In *ECCV (3)*, pages 792–805, 2008.

[74] J. Rabin and N. Papadakis. Convex color image segmentation with optimal transport distances. In J.-F. Aujol, M. Nikolova, and N. Papadakis, editors, *Scale Space and Variational Methods in Computer Vision*, volume 9087 of *Lecture Notes in Computer Science*, pages 256–269. Springer International Publishing, 2015.

[75] J. Rabin and G. Peyré. Wasserstein regularization of imaging problem. In *ICIP*, pages 1541–1544, 2011.

[76] S. T. Rachev and L. Rüschendorf. *Mass Transportation Problems. Vol. I, Theory*. Springer-Verlag, New York, 1998.

[77] H. Raguet, J. Fadili, and G. Peyré. A Generalized Forward-Backward Splitting. *SIAM J. Imag. Sci.*, 6(3):1199–1226, 2013.

[78] P. D. Ravikumar, A. Agarwal, and M. J. Wainwright. Message-passing for graph-structured linear programs: Proximal methods and rounding schemes. *Journal of Machine Learning Research*, 11:1043–1080, 2010.

[79] R. T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14(5):877–898, 1976.

[80] R. T. Rockafellar. *Convex Analysis*. Princeton Landmarks in Mathematics,. Princeton University Press, Princeton, princeton paperbacks edition, 1997.

[81] R.T. Rockafellar and R.J.B. Wets. *Variational Analysis*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2009.

[82] C. Rother, V. Kolmogorov, V. S. Lempitsky, and M. Szummer. Optimizing binary MRFs via extended roof duality. In *CVPR*, 2007.

[83] C. Rother, T. Minka, A. Blake, and V. Kolmogorov. Cosegmentation of image pairs by histogram matching - incorporating a global constraint into MRFs. In *CVPR*, pages 993–1000, Washington, DC, USA, 2006. IEEE.

[84] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.

[85] B. Savchynskyy, J. H. Kappes, S. Schmidt, and C. Schnörr. A study of Nesterov's scheme for Lagrangian decomposition and MAP labeling. In *CVPR*, pages 1817–1823. IEEE, 2011.

[86] B. Savchynskyy, J. H. Kappes, P. Swoboda, and C. Schnörr. Global MAP-optimality by shrinking the combinatorial search area with convex relaxation. In *NIPS*, 2013.

[87] B. Savchynskyy and S. Schmidt. Getting feasible variable estimates from infeasible ones: MRF local polytope study. In *Advanced Structured Prediction*. MIT Press, 2014.

[88] B. Savchynskyy, S. Schmidt, J. H. Kappes, and C. Schnörr. Efficient MRF energy minimization via adaptive diminishing smoothing. In *UAI*, 2012.

[89] M. Schlesinger. Syntactic analysis of two-dimensional visual signals in the presence of noise. *Kibernetika*, 12(4):113–130, 1976.

[90] S. Schmidt, B. Savchynskyy, J. H. Kappes, and C. Schnörr. Evaluation of a first-order primal-dual algorithm for MRF energy minimization. In *EMMCVPR*, volume 5681 of *LNCS*, pages 89–103. Springer, 2011.

[91] T. Schoenemann and V. Kolmogorov. Generalized sequential tree-reweighted message passing. *Advanced Structured Prediction*, 2012.

[92] N. N. Schraudolph and D. Kamenetsky. Efficient exact inference in planar Ising models. In Daphne Koller, Dale Schuurmans, Yoshua Bengio, and Léon Bottou, editors, *NIPS*, pages 1417–1424. Curran Associates, Inc., 2008.

[93] A. Shekhovtsov. Maximum persistency in energy minimization. In *CVPR*, page 8, 2014.

[94] A. Shekhovtsov, V. Kolmogorov, P. Kohli, V. Hlavac, C. Rother, and P. Torr. LP-relaxation of binarized energy minimization. Research Report CTU–CMP–2007–27, Czech Technical University, 2008.

[95] S. E. Shimony. Finding maps for belief networks is np-hard. *Artif. Intell.*, 68(2):399–410, 1994.

[96] D. Sontag. *Approximate Inference in Graphical Models using LP Relaxations*. PhD thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2010.

[97] D. Sontag, D. K. Choe, and Y. Li. Efficiently searching for frustrated cycles in MAP inference. In *UAI*, pages 795–804. AUAI Press, 2012.

[98] D. Sontag and T. Jaakkola. Tree block coordinate descent for MAP in graphical models. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics (AI-STATS)*, volume 8, pages 544–551. JMLR: W&CP, 2009.

[99] D. Sontag, T. Meltzer, A. Globerson, T. Jaakkola, and Y. Weiss. Tightening lp relaxations for map using message passing. In David A. McAllester and Petri Myllymäki, editors, *UAI*, pages 503–510. AUAI Press, 2008.

[100] P. Strandmark. Generalized roof duality. http://www.maths.lth.se/matematiklth/personal/petter/pseudoboolean.php, 2012.

[101] P. Swoboda, B. Savchynskyy, J. H. Kappes, and C. Schnörr. Partial optimality via iterative pruning for the Potts model. In *SSVM*, 2013.

[102] P. Swoboda, B. Savchynskyy, J. H. Kappes, and C. Schnörr. Partial optimality by pruning for MAP-inference with general graphical models. In *CVPR*, 2014.

[103] P. Swoboda and C. Schnörr. Convex variational image restoration with histogram priors. *SIAM J. Imaging Sciences*, 6(3):1719–1735, 2013.

[104] P. Swoboda and C. Schnörr. Variational image segmentation and cosegmentation with the Wasserstein distance. In *EMMCVPR*, pages 321–334, 2013.

[105] R. Szeliski, R Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. F. Tappen, and C. Rother. A comparative study of energy minimization methods for Markov random fields with smoothness-based priors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(6):1068–1080, 2008.

[106] S. Vicente, V. Kolmogorov, and C. Rother. Cosegmentation revisited: models and optimization. In *Proceedings of the 11th European conference on Computer vision: Part II*, ECCV'10, pages 465–479, Berlin, Heidelberg, 2010. Springer-Verlag.

[107] S. Vicente, C. Rother, and V. Kolmogorov. Object cosegmentation. In *CVPR*, pages 2217–2224. IEEE, 2011.

[108] C. Villani. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer, 1 edition, November 2008.

[109] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. MAP estimation via agreement on trees: message-passing and linear programming. *IEEE Trans. Inf. Theor.*, 51(11):3697–3717, November 2005.

[110] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. A new class of upper bounds on the log partition function. *CoRR*, abs/1301.0610, 2013.

[111] M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1-2):1–305, January 2008.

[112] T. Werner. A linear programming approach to max-sum problem: A review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(7):1165–1179, July 2007.

[113] T. Werner. Revisiting the linear programming relaxation approach to gibbs energy minimization and weighted constraint satisfaction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(8):1474–1488, 2010.

[114] T. Werner. How to compute primal solution from dual one in MAP inference in MRF? *Intl. Jr. on Control Systems and Computers*, (2), April 2011. Published by the National Academy of Sciences of Ukraine.

[115] T. Windheuser, H. Ishikawa, and D. Cremers. Generalized roof duality for multi-label optimization: Optimal lower bounds and persistency. In *ECCV*, Firenze, Italy, October 2012.

[116] C. Yanover, O. Schueler-Furman, and Y. Weiss. Minimizing and learning energy functions for side-chain prediction. *Journ. of Comput. Biol.*, 15(7):899–911, 2008.

[117] J. Yarkony, C. C. Fowlkes, and A. T. Ihler. Covering trees and lower-bounds on quadratic assignment. In *CVPR*, pages 887–894. IEEE, 2010.

[118] W.P. Ziemer. *Weakly Differentiable Functions*. Springer, 1989.

Index