COMPARING LINEAR AND CONVEX RELAXATIONS FOR STEREO AND MOTION

Thomas Schoenemann
Lund University, Lund, Sweden
University of Pisa, Pisa, Italia

Keywords: Motion estimation, Convex relaxation.

Abstract: We provide an analysis of several linear programming relaxations for the problems of stereo disparity estimation and motion estimation. The problems are cast as integer linear programs and their relaxations are solved approximately either by block coordinate descent (TRW-S and MPLP) or by smoothing and convex optimization techniques. We include a comparison to graph cuts. Indeed, the best energies are obtained by combining move-based algorithms and relaxation techniques. Our work includes a (slightly novel) tight relaxation for the typical motion regularity term, where we apply a lifting technique and discuss two ways to solve the arising task. We also give techniques to derive reliable lower bounds, an issue that is not obvious for primal relaxation methods, and apply the technique of (Desmet et al., 1992) to a-priori exclude some of the labels. Moreover we investigate techniques to solve linear and convex programming problems via accelerated first order schemes which are becoming more and more widespread in computer vision.

1 INTRODUCTION

First order methods for correspondence problems, in particular stereo and motion, have received a lot of attention over the past few years. The arising optimization problems are largely solved today, and it is time to compare the different techniques. In doing so, we focus on a topic that has been widely ignored for computer vision: in a primal relaxation scheme, what can we truly conclude about the associated lower bound? Most standard approaches require a very high precision to give a guaranteed lower bound, which is impractical.

The addressed models are discrete multi-label problems and can be cast as Markov Random Fields with binary terms. As unary data terms do not impose algorithmical challenges we focus on the regularity terms, considering ones that can be expressed in terms of absolute differences of indicator functions: linear potentials and the Potts model.

Subsequently, we consider two types of linear programming relaxations: the standard marginal polytope relaxation associated to inference in Markov Random Fields, and specialized linear relaxations where absolute differences are modeled by a constraint system. For the latter, we consider a simple variant and a much tighter relaxation based on lifting.

The problem with these relaxations is that they are convex, but neither strictly convex nor differentiable, which makes solving them exactly very cost intensive. In this paper we evaluate two common strategies to deal with this problem: for the marginal polytope relaxation we use block-coordinate descent methods (Kolmogorov, 2006; Globerson and Jaakkola, 2007; Sontag et al., 2010) that may not find the exact relaxation value, but usually get very close to it.

For the specialized relaxations we consider smoothing techniques that approximate the linear relaxations via strictly convex and differentiable functions. These functions are convex relaxations of the original discrete optimization problem. To solve the smoothed problems, we apply optimization techniques that are becoming increasingly popular in Computer Vision but have not been applied to the exact considered problems yet. A basic scheme consists of combining accelerated first order schemes (Nesterov, 2004) with the projection on a product of simplices (Michelot, 1986), plus a smoothing of non-smooth functions (Nesterov, 2005). This scheme has been used in (Lellmann et al., 2009; Goldlücke and Cremers, 2010). In a second scheme, we apply the well-known augmented Lagrangian method (Bertsekas, 1999, chap. 4.2) to the linear programming formulations.

Previously (Szeliski et al., 2008), energy minimization methods have been compared in a discrete context including stereo, but excluding motion. In contrast, we focus on correspondence problems and

Schoenemann T. (2012).
COMPARING LINEAR AND CONVEX RELAXATIONS FOR STEREO AND MOTION.
In *Proceedings of the 1st International Conference on Pattern Recognition Applications and Methods*, pages 5-14
DOI: 10.5220/0003710400050014

lifting techniques. We start with a review of common approaches for stereo and motion. The optimization techniques underlying the most relevant works will be reviewed in a later section.

Stereo. To get an overview of methods for stereo disparity estimation, the taxonomy given in (Scharstein and Szeliski, 2002) is still a very good starting point. A quite robust approach that guarantees global optima for certain first-order priors was given by (Ishikawa, 2003) and later formulated in (Pock et al., 2008) as a continuous problem. (Bhusnurmath and Taylor, 2008) address second order priors via customized linear programming and a convexification of the data term.

To further improve the robustness of these models, typically non-convex regularization terms are used, and the resulting optimization problems become hard. (Kleinberg and Tardos, 1999) cast the Potts model as a linear program, and (Zach et al., 2008) solve this by coordinate descent on a modified functional. The top-performing methods are currently based on belief propagation (Klaus et al., 2006; Yang et al., 2009) or the related tree reweighted message passing (Kolmogorov, 2006). (Meltzer et al., 2005) show how to obtain global optima by running several stages of these techniques.

Finally there are move-based algorithms like expansion moves (Boykov et al., 2001). In a less direct approach, fusion moves have been considered (Woodford et al., 2008).

Motion. We focus on non-parametric methods to the estimation of optical flow. Traditionally here one minimizes a flow-field energy with a non-convex data term and a convex regularizer. This trend was started by (Horn and Schunck, 1981) and much refined subsequently. In particular, robust data terms and regularizers based on absolutes have been considered (Memin and Perez, 1998; Papenberg et al., 2006). Also, different optimization techniques have been explored (Papenberg et al., 2006; Bruhn, 2006; Zach et al., 2007) and second order priors were considered (Trobin et al., 2008). All of these methods rely on repeated convexification of the energy. Some of them are among the best performing known today.

In a more recent line of work, one instead works with a discretized set of possible motion vectors. One way to address the arising discrete optimization problem is given by move-based algorithms (Boykov et al., 2001; Glocker et al., 2008). The remaining techniques can all be viewed as linear or convex programming methods. (Shekhovtsov et al., 2008) introduce a clever application of tree reweighted message passing (Kolmogorov, 2006), where horizontal and vertical displacements are separate variables.

(Goldstein et al., 2009) propose a lifting technique to solve the non-linearized motion problem in terms of indicator variables for level sets. Below, we give a different solution with indicator variables for labels - the advantage being that we only have easy-to-handle simplex constraints.

Recently (Goldlücke and Cremers, 2010) proposed a very memory efficient technique for large scale non-linearized motion estimation, using a convexification of non-convex terms. However, as we show later in this work, the optimal energy is always zero and the corresponding minimizers are meaningless. Hence, this method requires a careful initialization and must not be run until convergence.

2 THE PROBLEMS

In this paper we consider multi-label problems with unary and binary terms, i.e. of the general form

$$E(\mathbf{y}) = \sum_{p \in \mathcal{V}} D_p(y_p) + \sum_{(p,q) \in \mathcal{N}} V_{p,q}(y_p, y_q) \qquad (1)$$

where $\mathbf{y} : \mathcal{V} \to \{1, \ldots, L\}$, p, q denote locations defined by a set of points $\mathcal{V} \subseteq \mathcal{R}^2$ and $\mathcal{N}$ is a neighborhood system, usually the 8-neighborhood in this work. We require that $\mathcal{N}$ is asymmetric, i.e. if $(p,q) \in \mathcal{N}$ then $(q,p) \notin \mathcal{N}$. Further, it will be convenient to abbreviate $\mathcal{N}(p) = \{q \,|\, (p,q) \in \mathcal{N} \vee (q,p) \in \mathcal{N}\}$.

2.1 Stereo

In stereo, one is given two images showing the same scene and wants to find out where the points in the first image are to be found in the second image. It is known that each point can be found along a certain *epipolar* line. For simplicity, in this work we assume that these lines coincide with scan-lines in the image.

Given two images $I_1, I_2 : \Omega \to \mathcal{R}$, where $\Omega \subset \mathcal{R}^2$, in a continuous formulation one is then searching for a disparity map $d : \Omega \to \mathcal{R}$ that describes the displacements. In our first model, the optimal disparity map is defined as the optimum of

$$\int_\Omega |I_1(x,y) - I_2(x + d(x,y), y)|^p \, dx\, dy \\ + \lambda \int_\Omega |\nabla d(x,y)| \, dx\, dy \,,$$

where p is some positive power and $\lambda > 0$ a weighting factor. The first term is the data term, and we simply choose $p = 1$ in this work. The second term is called a *regularizer*, and we will meet one other choice below.

Most current top-performing methods are based on discretizing the continuous model and, more importantly, the range of the function d. That is, we now assume that $d : \mathcal{V} \to \mathcal{L}$, where $\mathcal{V}$ is the set of pixel locations and $\mathcal{L} \subset \mathcal{R}$ is a finite set. For this work we assume that the points in $\mathcal{L}$ are induced by an equidistant spacing.

Instead of integrals, in the data term one is now considering sums over the pixels in the image. Discretizing the regularity term is more intricate. We will consider the approximation

$$|\nabla d(p)| \approx \sum_{q \in \mathcal{N}(p)} \frac{|d(p) - d(q)|}{\|p - q\|} \quad .$$

Methods derived from the continuous model will typically take a Euclidean vector norm of the gradient here (Goldstein et al., 2009; Goldlücke and Cremers, 2010), known as total variation. By defining

$$D(p = (x,y), l) = |I_1(x,y) - I_2(x+l,y)|$$

we are finally ready to phrase the first discrete model considered in this work:

$$\sum_p D(p, d(p)) + \lambda \sum_{(p,q) \in \mathcal{N}} \frac{|d(p) - d(q)|}{\|p - q\|} \quad . \qquad (SA)$$

For the second model we directly consider the discrete formulation which amounts to the Potts model:

$$\sum_p D(p, d(p)) + \lambda \sum_{(p,q) \in \mathcal{N}} \frac{1 - \delta(d(p), d(q))}{\|p - q\|} \quad , \qquad (SP)$$

where $\delta(\cdot,\cdot)$ is the Kronecker delta.

2.2 Motion

Motion is a generalization of stereo to arbitrary two-dimensional displacements, denoted by two functions $u : \Omega \to \mathcal{R}$ and $v : \Omega \to \mathcal{R}$ for the displacement in horizontal and vertical direction, respectively. The continuous formulation of the model we will consider looks as follows:

$$\int_\Omega |I_1(x,y) - I_2(x + u(x,y), y + v(x,y))| \, dx\,dy + \lambda \int_\Omega \Big[|\nabla u(x,y)| + |\nabla v(x,y)| \Big] \, dx\,dy \; .$$

Here, we have convex regularizers but a non-convex data term which makes the problem difficult. Recently there has been increasing interest (Shekhovtsov et al., 2008; Goldstein et al., 2009; Goldlücke and Cremers, 2010) in methods that discretize the ranges of the displacement functions. That is, we are now considering $u : \mathcal{V} \to \mathcal{L}_H$ and $v : \mathcal{V} \to \mathcal{L}_V$, where $\mathcal{L}_H$ and $\mathcal{L}_V$ are finite subsets of $\mathcal{R}$. The discrete model looks as follows:

$$\sum_{p \in \mathcal{V}} |I_1(x,y) - I_2(x + u(x,y), y + v(x,y))| \qquad (MA)$$
$$+ \lambda \left[\sum_{(p,q) \in \mathcal{N}} \frac{|u(p) - u(q)|}{\|p - q\|} + \sum_{(p,q) \in \mathcal{N}} \frac{|v(p) - v(q)|}{\|p - q\|} \right] .$$

The resulting optimization problem is much harder than (SA) for stereo. Since with a Potts potential it would be even more difficult, we will not consider a variant (MP) here.

2.3 Why Lower Bounds?

Above two essential problems of computer vision have been cast as energy minimization problems. In practice it is desirable but not strictly necessary to find the global minimum of these problems: the functions are designed in a way that any solution with energy in a certain band around the optimal one should be an acceptable solution. Hence, if one is able to provide a tight lower bound, any method that finds a solution close to this bound will give valuable insights into the suitability of the models for relevant real-world instances of the problem.

2.4 Excluding Labels

In the absence of a regularity term (i.e. $\lambda = 0$) the above problems are easy to solve: the optimal label can be determined independently for each pixel. As soon as there is a regularity term, even with a very small λ, this no longer works. However, with the help of the dead-end elimination algorithm of (Desmet et al., 1992) it is still possible to at least *exclude* some of the labels. We state this (adapted to our context) in the following proposition:

Proposition 1. *Consider the general energy* (1) *and assume $V_{p,q}(\cdot,\cdot) \geq 0$ everywhere. For any site p, pick a $k_p \in \arg\min_l D_p(l)$. Now, if for any label l*

$$D_p(l) > D_p(k_p) + \sum_{q:(p,q) \in \mathcal{N}} \max_l V_{p,q}(k_p, l) \quad , \qquad (2)$$

then $y_p \neq l$ in any optimal solution $\mathbf{y}$.

Proof: Assume that $\mathbf{y}^*$ is an optimal solution satisfying $y_p^* = l$ and denote its energy c^*. Now define $\hat{\mathbf{y}}$ as

$$\hat{y}_v = \begin{cases} k_p & \text{if } v = p \\ y_v^* & \text{else .} \end{cases}$$

Then the cost of $\hat{\mathbf{y}}$ are

$$\hat{c} = c^* + D_p(k) - D_p(l) + \sum_{q \in \mathcal{N}(p)} [V_{p,q}(k_p, y_q^*) - V_{p,q}(l, y_q^*)] \; .$$

Exploiting the condition (2) on $D_p(l)$, it follows that

$$\begin{aligned}\hat{c} &< c^* + \sum_{q\in\mathcal{N}(p)} [V_{p,q}(k_p, y_q^*) - \max_{l'} V_{p,q}(k_p, l') \\ &\qquad - V_{p,q}(l, y_q^*)] \\ &\leq c^* \ .\end{aligned}$$

So $\hat{\mathbf{y}}$ has a lower energy than $\mathbf{y}^*$. A contradiction. □

For (SP) up to 25% of all labels can be excluded, and we exploit this information to initialize the convex relaxation methods.

3 LINEAR PROGRAMMING RELAXATIONS

In this paper, we evaluate strategies to solve the problems (SA), (SP) and (MA) by first casting them as integer linear programs and then (approximately) solving the arising linear programming relaxations. Hence, we are dealing with *constrained* optimization. All formulations are based on indicator functions to express which pixel has which label.

For stereo, all considered methods introduce a binary variable $x_d^p \in \{0,1\}$ for every $p \in \mathcal{V}$ and every label $d \in \mathcal{L}$, where a value of 1 indicates that $y_p = d$. The variables are grouped into a vector $\mathbf{x_S}$. Among other things, the constraint system expresses that every pixel must have exactly one label:

$$\mathbf{x_S} \in \mathcal{S}_S = \left\{ \hat{\mathbf{x}} \,|\, \sum_{d\in\mathcal{L}} \hat{x}_d^p = 1 \quad \forall p \in \mathcal{V} \right\} \ .$$

Together with the constraint $\mathbf{x_S} \geq 0$ that is naturally satisfied for binary variables such a constraint system is known as a *product of simplices*, an important fact for several of the methods we consider below.

For motion, there are two different approaches. In one of them there is a binary variable $x_{h,v}^p \in \{0,1\}$ for every pixel $p \in \mathcal{V}$ and combination of displacements $h \in \mathcal{L}_H, v \in \mathcal{L}_V$. All these variables are grouped into a vector $\mathbf{x_{H\times V}}$. The constraint that every pixel have exactly one label is now expressed as

$$\mathbf{x_{H\times V}} \in \mathcal{S}_{H\times V} = \left\{ \hat{\mathbf{x}} \,|\, \sum_{h,v} \hat{x}_{h,v}^p = 1 \quad \forall p \in \mathcal{V} \right\} \ .$$

In the other approach there are separate binary variables $x_h^{p,H}$ and $x_v^{p,V}$ for every pixel $p \in \mathcal{V}$ to express the horizontal and vertical displacements of a pixel. They are grouped into a vector $\mathbf{x_{H\|V}}$. Now every pixel must have exactly one horizontal and exactly one vertical label, expressed as

$$\mathbf{x_{H\|V}} \in \mathcal{S}_{H\|V} = \left\{ \hat{\mathbf{x}} \,|\, \sum_h \hat{x}_h^{p,H} = \sum_v \hat{x}_v^{p,V} = 1 \quad \forall p \in \mathcal{V} \right\} .$$

In both methods we again have products of simplices. All of these problems are based on binary variables. However, these often make the arising constrained optimization problems very hard. As a consequence, $x \in \{0,1\}$ is often relaxed to the constraint $x \in [0,1]$ and the resulting problem is in all considered cases a linear programming relaxation (when using smoothing this turns into a non-linear but still convex problem).

Before we proceed to present the considered methods, we note that the data terms of (SA) and (SP) can now be expressed as a linear function $\mathbf{c_S}^T \mathbf{x_S}$, those for motion as a linear function $\mathbf{c_{H\times V}}^T \mathbf{x_{H\times V}}$, where

$$\begin{aligned} c_d^p &= |I_1(p) - I_2(p + (d\ 0)^T)| \\ c_{h,v}^p &= |I_1(p) - I_2(p + (h\ v)^T)| \ . \end{aligned}$$

3.1 Marginal Polytope Formulations

All considered problems are inference problems for Markov Random Fields with unary and pairwise terms. There is a standard integer linear programming formulation (and associated LP-relaxation) for such problems, called the *marginal polytope* formulation.

Here we detail it by directly choosing (SA) and (SP) as applications. Problem (1) is here re-phrased as the integer linear program

$$\begin{aligned} \min_{\mathbf{z}} \quad & \sum_p \sum_{l\in\mathcal{L}} D_p(l) \cdot z_l^p + \sum_{(p,q)\in\mathcal{N}} \sum_{l_p,l_q} V_{p,q}(l_p,l_q) \cdot z_{l_p,l_q}^{p,q} \\ \text{s.t.} \quad & \sum_{l_q} z_{l_p,l_q}^{p,q} = z_{l_p}^p \quad \forall p \in \mathcal{V}, l_p \in \mathcal{L} \qquad (3) \\ & \sum_{l_p} z_{l_p,l_q}^{p,q} = z_{l_q}^q \quad \forall q \in \mathcal{V}, l_q \in \mathcal{L} \\ & z_l^p \in \{0,1\} \quad \forall p \in \mathcal{V}, l \in \mathcal{L} \ . \end{aligned}$$

For general $V_{p,q}(\cdot,\cdot)$ this integer programming problem is NP-hard, so one aims at solving the linear programming relaxation, i.e. relaxing $z_l^p \in \{0,1\}$ to $z_l^p \in [0,1]$. However, actually storing all pairwise variables $z_{l_p,l_q}^{p,q}$ is only practicable for a very small number of labels and hence not applicable for stereo. Hence, using a standard linear programming solver is not an option here. Instead there are specialized block coordinate descent algorithms which we introduce below.

Many of these methods have already been tested on the problem of stereo, e.g. (Kolmogorov, 2006). For motion we note the work of (Shekhovtsov et al., 2008) who show how to efficiently solve the problem (MA) with these techniques. They build on variables $\mathbf{x_{H\|V}}$, so there are no longer any unary terms: the data term is now binary, too. This results in savings in memory and run-time compared to the straightforward solution with variables $\mathbf{x_{H\times V}}$.

3.2 Specialized Linear Programming

We now turn to specialized linear programming relaxations which take into account that the regularity terms are functions of absolute differences, i.e. terms of the form $|y_p - y_q|$ where $y_p, y_q \in \mathcal{L}$. At first glance this does not apply to the Potts model. However, as exploited by (Kleinberg and Tardos, 1999; Zach et al., 2008), the Potts model can be written as a sum of absolute differences in terms of the indicator variables:

$$1 - \delta(y_p, y_q) \;=\; \tfrac{1}{2} \sum_{l \in \mathcal{L}} |x_l^p - x_l^q| \quad . \tag{4}$$

Finally, minimizing terms of the form $|y_p - y_q|$ is equivalent to the linear program (see e.g. (Dantzig and Thapa, 1997))

$$\begin{aligned} \min_{a_+^{p,q}, a_-^{p,q}} \quad & a_+^{p,q} + a_-^{p,q} \\ \text{s.t.} \quad & y_p - y_q \;=\; a_+^{p,q} - a_-^{p,q} \quad , \\ & a_+^{p,q} \geq 0, \; a_-^{p,q} \geq 0 \quad . \end{aligned} \tag{5}$$

By itself this problem is trivial and the optimum 0. However, this construction also holds when integrated into larger linear programs (as e.g. derived below), provided the auxiliary variables $a_+^{p,q}, a_-^{p,q}$ are not used for other constraints.

While the Potts model (4) can now be directly translated to a linear program, for (SA) and (MA) this is more difficult: the models here involve differences between labels, i.e. in terms of the variables $\mathbf{y}$, whereas it is often convenient to state the problem in terms of the binary indicator variables $\mathbf{x}$. Below we detail two ways on the problem of stereo, then briefly discuss the transfer to motion estimation.

Simple Relaxations. A simple way, used in (Lellmann et al., 2009; Goldlücke and Cremers, 2010), to express the absolute difference $|y_p - y_q|$ in terms of indicator variables is given by

$$|y_p - y_q| \;=\; \Big| \sum_{l \in \mathcal{L}} l \cdot (x_l^p - x_l^q) \Big|$$

While for binary variables this expression holds exactly, we will see later that the associated relaxations are quite weak. The linear program corresponding to (SA) is now

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{a}_\pm} \quad & \mathbf{c_S}^T \mathbf{x_S} + \sum_{p,q \in \mathcal{N}} \frac{\lambda}{\|p-q\|} [a_+^{p,q} + a_-^{p,q}] \\ \text{s.t.} \quad & \sum_{l \in \mathcal{L}} l \cdot (x_l^p - x_l^q) = a_+^{p,q} - a_-^{p,q} \quad \forall (p,q) \in \mathcal{N}, \\ & \mathbf{x_S} \in \mathcal{S}_S \,, \mathbf{x_S} \geq \mathbf{0} \,, \mathbf{a}_\pm \geq \mathbf{0} \quad . \end{aligned}$$

Relaxations based on Lifting. A technique that gives much stronger relaxations is called *lifting* and was used by (Ishikawa, 2003) to show that (SA) can be globally optimized via graph cuts - the linear programming relaxation is here equivalent to the integer linear program. In the considered setting, lifting is closely related to *level sets*, i.e. it can be expressed using variables

$$u_l^p \;=\; \begin{cases} 1 & \text{if } y_p \leq l \\ 0 & \text{else} \end{cases} \;=\; \sum_{l' \leq l} x_{l'}^p \quad .$$

With these variables we can now express the original absolute differences as sums of absolute differences of the level variables:

$$|y_p - y_q| \;=\; \sum_{l \in \mathcal{L}} \left| u_l^p - u_l^q \right| \quad ,$$

and obtain the linear program

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{a}_\pm} \quad & \mathbf{c_S}^T \mathbf{x_S} + \sum_{p,q \in \mathcal{N}} \frac{\lambda}{\|p-q\|} [a_+^{p,q} + a_-^{p,q}] \\ \text{s.t.} \quad & \sum_{l' \leq l} [x_l^p - x_l^q] = a_{+,l}^{p,q} - a_{-,l}^{p,q} \quad \forall (p,q) \in \mathcal{N}, l \in \mathcal{L}, \\ & \mathbf{x_S} \in \mathcal{S}_S \,, \mathbf{x_S} \geq \mathbf{0} \,, \mathbf{a}_\pm \geq \mathbf{0} \quad . \end{aligned}$$

Motion Relaxation. Both introduced techniques can be transferred to the problem of motion estimation when using the variables $\mathbf{x_{H \times V}}$. For simplicity we state the models in terms of absolute differences and leave the construction à la (5) to the reader. The simple relaxation can be cast as

$$\begin{aligned} \min_{\mathbf{x_{H\times V}}} \quad & \mathbf{c_{H\times V}}^T \mathbf{x_{H\times V}} \\ & + \sum_{(p,q) \in \mathcal{N}} \frac{\lambda}{\|p-q\|} \Big| \sum_{h,v} v \cdot [x_{h,v}^p - x_{h,v}^q] \Big| \\ & + \sum_{(p,q) \in \mathcal{N}} \frac{\lambda}{\|p-q\|} \Big| \sum_{h,v} h \cdot [x_{h,v}^p - x_{h,v}^q] \Big| \\ \text{s.t.} \quad & \mathbf{x_{H\times V}} \in \mathcal{S}_{H\times V} \;, \;\; \mathbf{x_{H\times V}} \geq \mathbf{0} \quad . \end{aligned}$$

The lifted version is stated here as

$$\begin{aligned} \min_{\mathbf{x_{H\times V}}, \mathbf{u}} \quad & \mathbf{c_{H\times V}}^T \mathbf{x_{H\times V}} \\ & + \sum_{(p,q) \in \mathcal{N}} \frac{\lambda}{\|p-q\|} \sum_h |u_h^{p,H} - u_h^{q,H}| \\ & + \sum_{(p,q) \in \mathcal{N}} \frac{\lambda}{\|p-q\|} \sum_v |u_v^{p,V} - u_v^{q,V}| \\ \text{s.t.} \quad & u_h^{p,H} = u_{h-1}^{p,H} + \sum_v x_{h,v}^p \\ & u_v^{p,V} = u_{v-1}^{p,V} + \sum_h x_{h,v}^p \\ & \mathbf{x_{H\times V}} \in \mathcal{S}_{H\times V} \;, \;\; \mathbf{x_{H\times V}} \geq \mathbf{0} \quad . \end{aligned}$$

Note that here it is straightforward to eliminate the variables $\mathbf{u}$ as they are defined by acyclic equality constraints.

3.3 Quadratic Programming Formulations

Recently (Goldlücke and Cremers, 2010) proposed to model problems with product label spaces such as (MA) as a quadratic integer programming problem with an indefinite cost function. They build on the separate variable version $\mathbf{x}_{\mathbf{H}\|\mathbf{V}}$ and (in the discrete analogon) express the cost function as

$$\sum_{p \in \mathcal{V}} \sum_{h,v} c^p_{h,v} \cdot x^{p,H}_h \cdot x^{p,V}_v \quad .$$

These are sums of second order non-convex terms and the authors propose to convexify the problem by replacing each term by its lower convex envelope:

$$\sum_{p \in \mathcal{V}} \sum_{h,v} c^p_{h,v} \cdot \max\{0, x^{p,H}_h + x^{p,V}_v - 1\} \quad . \tag{6}$$

The maximization could again be translated to a linear programming relaxation. This relaxation would naturally fit into our survey. However, we have already stated in the introduction that it does not make sense, and will now substantiate this:

Lemma 1. *Suppose* $|\mathcal{L}_H| \geq 3$ *and* $|\mathcal{L}_V| \geq 3$. *Then the minimal value of* (6) *s.t.* $\mathbf{x}_{\mathbf{H}\|\mathbf{V}} \in \mathcal{S}_{H\|V}, \mathbf{x}_{\mathbf{H}\|\mathbf{V}} \geq \mathbf{0}$ *is* 0.

Proof: Consider

$$\begin{aligned} x^{p,H}_h &= 1/|\mathcal{L}_H| \;\; \forall p \in \mathcal{V}, h \in \mathcal{L}_H \\ x^{p,V}_v &= 1/|\mathcal{L}_V| \;\; \forall p \in \mathcal{V}, v \in \mathcal{L}_V \quad . \end{aligned}$$

Then $x^{p,H}_h + x^{p,V}_v \leq 2/3$ and hence $\max\{0, x^{p,H}_h + x^{p,V}_v - 1\} = 0$ for all $p \in \mathcal{V}, h \in \mathcal{L}_H, v \in \mathcal{L}_V$. □

The given $\mathbf{x}_{\mathbf{H}\|\mathbf{V}}$ is identical for all pixels, and since a typical regularization term for stereo and motion will be at its minimum for such a configuration, this is the global minimum even for regularized functionals. The solution is meaningless: it assigns equal probability to all possible labelings.

4 ALGORITHMS

There are standard software packages for (nearly) exactly solving linear programming relaxations. However, in practice they do not scale well: e.g. (Meltzer et al., 2005) found they could handle images of size up to 39×39 for the marginal polytope relaxation in stereo with 30 disparities.

For the specialized relaxations the situation is a bit better: we solved the motion relaxation for a 75×75 image with displacements between -8 and 8 in steps of 1 in both horizontal and vertical directions ($= 289$ labels) using a standard solver. This required roughly 3 GB and several hours computing time, so in practice it is still too inefficient.

Instead we consider approximate techniques to solve the relaxations, either using block coordinate descent or using smoothing techniques. In practice these methods get very close to the actual relaxation value.

4.1 The Marginal Polytope

A number of popular algorithms to (approximately) solve the marginal polytope relaxations are based on block coordinate descent on dual problem formulations. Here we explore two of them.

TRW-S. (Wainwright et al., 2005) proposed an approach where the blocks of the dual take the form of trees. One then alternates the estimation of min-marginals in the trees (using the inward-outward algorithm) with reparameterization steps. The original formulation was not monotonically improving the dual bound, but the issue was solved by TRW-S (Kolmogorov, 2006). We will later see that in practice this methods gets very close to the optimum.

The running times per iteration are in general quadratic in the number of labels. However, for many kinds of potentials (e.g. Potts and absolute differences) there are more efficient ways to perform inward-outward - see e.g. (Felzenszwalb and Huttenlocher, 2006).

MPLP. Recently (Globerson and Jaakkola, 2007) introduced a new message passing algorithm derived from a non-standard dual of the marginal polytope algorithm. On this formulation they perform block coordinate descent where (for pairwise MRFs) a block considers all dual variables associated to a particular edge. See also (Sontag et al., 2010) for a very good explanation and further background on the method. The method is very efficient and monotonically increases the dual bound. It allows the same kind of optimizations as TRW-S.

4.2 Specialized Relaxations

For the specialized relaxations we tried block coordinate descent on the primal relaxation, using standard LP-solvers. This resulted in heavily suboptimal solutions and the running times were still too high. As a consequence, for these problems we apply smoothing techniques. Note that the smoothed problems are themselves valid relaxations of the discrete optimization problem: the cost for integral solutions are the same as in the discrete problem.

Smoothing. (Nesterov, 2005) shows how a large class of non-differentiable functions can be closely approximated by differentiable functions with Lipschitz-continuous gradients. Here, very efficient numerical schemes can be applied (Nesterov, 2004). In our context, the employed approximation

$$|x| \approx \begin{cases} |x| - \frac{\varepsilon}{2} & \text{if } |x| \geq \varepsilon \\ \frac{x^2}{2\varepsilon} & \text{else} \end{cases} ,$$

where $\varepsilon > 0$ is a small constant. We scale this function so that it gives exact values for $x = \pm 1$. This expression is substituted wherever absolutes occur in the problem formulation. One obtains a convex function to be optimized over a product of simplices, which is done by combining (Nesterov, 2004) with the algorithm (Michelot, 1986) to re-project on the simplices.

Augmented Lagrangians Instead of smoothing the absolutes, one can work directly with the linear program and apply the very general augmented Lagrangian method (Bertsekas, 1999, chap. 4.2), which is applicable to many constrained optimization problems. Here we consider

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{c}^T \mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{x} \in \mathcal{S} \end{array}$$

where $\mathbf{A}$ is a matrix and $\mathbf{x}, \mathbf{c}, \mathbf{b}$ are vectors of suitable dimensions and $\mathcal{S}$ denotes a "simple" set, i.e. the projection on $\mathcal{S}$ can be handled efficiently (in our case $\mathcal{S}$ is a product of simplices for the indicator variables times simple variable bounds for the auxiliary variables). The augmented Lagrangian method is based on minimizing

$$\min_{\mathbf{x} \in \mathcal{S}} \mathbf{c}^T \mathbf{x} + \nu^T (\mathbf{A}\mathbf{x} - \mathbf{b}) + \gamma \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 \quad (7)$$

Indeed, since $\mathbf{A}\mathbf{x} - \mathbf{b} = \mathbf{0}$ for any feasible $\mathbf{x}$, for any $\nu \in \mathcal{R}^n$ and any $\gamma > 0$ this is a convex relaxation of the original linear program (the cost for feasible solutions remain unchanged). Moreover, the function has a Lipschitz-continuous gradient, so the accelerated first order method (Nesterov, 2004) with back-projection on $\mathcal{S}$ can be applied.

In practice one also updates the constant γ and the multiplier ν. With sufficient inner and outer iterations this method will eventually find the optimum of the original problem, but this takes too long for our applications.

We iterate the following steps: First solve (7) with moderate precision (we use the same scheme as for the smoothing of absolutes above). Then use the found solution to update $\nu \leftarrow \nu + \gamma(\mathbf{A}\mathbf{x_0} - \mathbf{b})$, increase γ by 1.25 and proceed. When applying this to the lifted version of the motion relaxation it proved much more robust to actually include the level variables.

5 DERIVING BOUNDS

The principle of duality (e.g. (Bertsekas, 1999)) implies that any method that maintains dual feasible solutions will provide a feasible lower bound on the optimal objective value in every iteration. Among the considered methods, only the ones for the marginal polytope satisfy this property.

The methods for the specialized relaxations all solve the primal problem and provide a generally valid lower bound only when the respective problem is solved with very high precision, which is impractical. However, a valid lower bound can be derived from a fundamental property of differentiable convex functions that simultaneously allows to decide if the global optimum has been found:

Lemma 2. *Consider the problem of computing* $f^* = \min_{\mathbf{x} \in Q} f(\mathbf{x})$ *where* $f : \mathcal{R}^n \to \mathcal{R}$ *is a convex and differentiable function and* Q *is an arbitrary closed set. Then for any* $\mathbf{x_0} \in \mathcal{R}^n$

$$f^* \geq f(\mathbf{x_0}) - \nabla f(\mathbf{x_0})^T \mathbf{x_0} + \min_{\mathbf{x} \in Q} \nabla f(\mathbf{x_0})^T \mathbf{x} . \quad (8)$$

Moreover, if Q *is convex then* $\mathbf{x_0}$ *is a global optimum if and only if the right-hand-side of* (8) *is equal to* $f(\mathbf{x_0})$.

Proof: Since f is convex, its tangent hyperplane at $\mathbf{x_0}$ is a lower bound on the problem (Nesterov, 2004), i.e. $f(\mathbf{x}) \geq f(\mathbf{x_0}) + \nabla f(\mathbf{x_0})^T (\mathbf{x} - \mathbf{x_0})$. The first statement follows from taking the minimum on both sides. For the second part, it is well known that $\mathbf{x_0}$ is a global optimum of a differentiable convex optimization problem if and only if there is no local descent direction, i.e. $\mathbf{x_0} \in \arg\min_{\mathbf{x} \in Q} \nabla f(\mathbf{x_0})^T (\mathbf{x} - \mathbf{x_0})$. □

For the considered problems the minimization is over a product of simplices, which is straightforward to compute.

Obtaining Integral Solutions. To obtain integral solutions, for the primal methods we assign each pixel p the label corresponding to the label variable with maximal value. For the dual methods calculating the objective (i.e. the lower bound) entails solving minimization problems for the unary and pairwise terms. For MPLP, we take the minimizers of the unary part to generate an integral solution. TRW-S comes with a more refined integer solution generation method: it proceeds in a fixed order over the pixels, always taking into account the labels of the already fixed pixels.

6 EXPERIMENTS

For testing the considered methods we focus on two different aspects. To examine the tightness of the modeled relaxations the methods are run for a large number of iterations. Then we examine many iterations are needed to get useful integral solutions from inexactly solved relaxations. We also test combinations of relaxation methods and move-based algorithms: the integral solutions obtained from the former serve as initializations for the latter. We compare this strategy to purely move-based algorithms started from the 0-displacement.

In all experiments we chose an $8-$connectivity and found that double precision is necessary. We use a 3 GHz Core2 Duo machine with a Tesla 2050 GPU. For the augmented Lagrangians the sparse matrices are computed on-the-fly in each iteration. We are comparing Kolmogorov's TRW-S code[1] against our own implementation of MPLP, noting that the latter is only moderately optimized and really designed for higher order terms.

Stereo. (SA) is tested on the well-known Venus and Tsukuba[2]. Here, only negligibly few labels can be excluded by Prop. 1. Moreover, since the problem can be solved globally using Ishikawa's lifting method (Ishikawa, 2003), our main point here is to show that the simple relaxation provides a very weak lower bound: on both tested problems its optimal value is barely more than half of Ishikawa's global optimum. And the obtained thresholded solutions are very noisy.

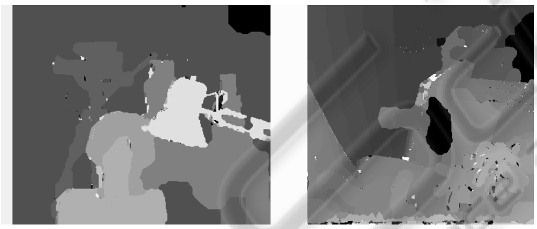

Figure 1: Best (SP)-energy disparity maps for the considered sequences.

For the **Potts model** (SP) between 10% and 25% of all label constellations could be excluded a priori, and exploiting this knowledge reduces the number of required iterations for the smooth approximations by a third. An evaluation of all methods is given in table 1. Here, the augmented Lagrangian (AL) method provides the tightest lower bounds, the smooth approximations are significantly lower but lead to better integral solutions. TRW-S is a generally good performing method. In terms of memory, expansion moves are lowest (140 MB on Teddy), followed by the convex

Table 1: **Stereo:** Comparison of lower bounds and integral energies for (SP) with $\lambda = 15$ on the Tsukuba and Teddy sequences. TRW-S was run for 1500 iterations where it was near convergence. MPLP (being slower) was run for 500 iterations, where it was progressing only slowly. "Refined" denotes a subsequent expansion move process. "low. bd." reports lower bounds. Note that TRW-S finds tight lower bounds much earlier than MPLP.

Tsukuba			
method	**low. bd.**	**integral**	**refined**
TRW-S	347222	347274	**347234**
MPLP	347150	348037	347238
conv ($\varepsilon = 10^{-3}$)	346908	347256	347235
conv ($\varepsilon = 10^{-4}$)	347193	**347251**	347236
Lag. 12×500	**347224**	347294	347236
Expansion Move	–	347833	347833

Teddy			
method	**low. bd.**	**integral**	**refined**
TRW-S	664952	666388	665834
MPLP	664278	669431	665357
conv ($\varepsilon = 10^{-3}$)	663073	666472	665153
conv ($\varepsilon = 10^{-4}$)	664818	**666317**	**665141**
Lag. 12×500	**664991**	666357	665178
Expansion Move	–	666631	666631

relaxation (340 MB), TRW-S (560) and our MPLP (950). With 1.7 GB AL needs too much – all transitions require explicit storage.

In practice, TRW-S gives comparable integral solutions already after 150 iterations (250 sec.), and already after 35 it beats the value MPLP finds after 500 iterations. Hence, TRW-S is clearly superior to MPLP on this problem.

To save running time it is better to choose a large ε (e.g. 0.01) for the smooth approximations. Then 500 iterations (2200 sec) suffice, and on the GPU this should be much faster. Our MPLP implementation has similar running times, but AL cannot compete with them.

Motion. For motion it is both common and sensible to pre-smooth the considered images, so we apply three iterations of binomial smoothing (but see next section). We consider two sequences, shown in Fig. 2, where the entire image is moving. For the upper we take displacements in $[-8,8] \times [-8,8]$ in steps of 1, for the lower we take a range of $[-5,5] \times [-5,5]$ in steps of 0.5. Up to 4% of the labels could be excluded via Proposition 1.

Table 2 shows that the TRW-S method generally gives the best results. MPLP gets close, but already after 95 iterations TRW-S achieves the bound MPLP finds after 500. The two methods need nearly equally long for an iteration[3].

[1] http://research.microsoft.com/en-us/downloads/dad6c31e2c04471fb724ded18bf70fe3/

[2] http://vision.middlebury.edu/stereo/.

[3] We use edges of type "general" for TRW-S everywhere. Memory and run-time can be improved (Shekhovtsov et al., 2008) by changing some of them to type "Potts".

Table 2: **Motion:** Comparison of lower bounds and integral energies for (MA) with $\lambda = 1$ on the Flower Garden and Person Walking sequences. TRW-S and MPLP were run for 2500 iterations (close to convergence), with TRW-S using edges of type "general". Note that TRW-S finds tight lower bounds much earlier than MPLP.

Flower Garden			
method	**low. bd.**	**integral**	**refined**
TRW-S	134532	**135702**	134919
MPLP	134385	137171	**134896**
conv($\varepsilon = 10^{-2}$)	132463	137212	134993
conv($\varepsilon = 10^{-3}$)	134368	136973	134930
aug-lag 12×500	**134564**	137314	134967
Expansion Move	–	136180	136180

Person Walking			
method	**low. bd.**	**integral**	**refined**
TRW-S	**107358**	**109014**	107937
MPLP	107298	110079	**107900**
conv($\varepsilon = 10^{-2}$)	105538	117761	108115
conv($\varepsilon = 10^{-3}$)	107268	116821	108128
aug-lag 12×500	107069	121581	108931
Expansion Move	–	111730	111730

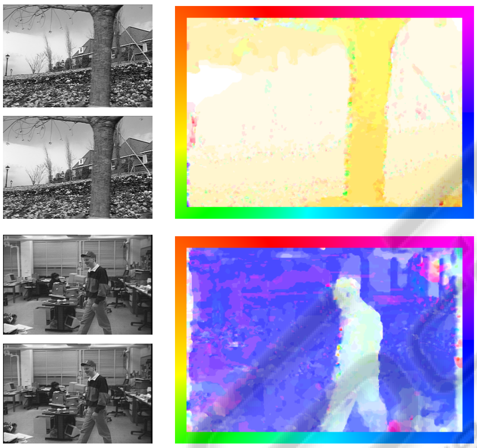

Figure 2: Input sequences (www.cs.brown.edu/ black/) and flow fields (color-coded) corresponding to the best (MA)-energies.

In one case AL gives the best bound, but this seems to be an exception (see below). The smooth approximation needs a small ε to give tight bounds. However, this method and MPLP need the least memory of all relaxation-based methods (500 MB compared to 3 GB for TRW-S and 1 GB for AL) and is furthermore easily implemented in parallel on the GPU. Again, expansion moves need much less memory than all relaxation-based methods: 200 MB.

Concerning the optimization of run-times, again TRW-S wins: it can be terminated after 10 minutes (on Flower Garden) and still gives a quite tight lower bound. For the smooth approximations, we found that with a choice of $\varepsilon = 0.01$ 5000 iterations give comparable integral solutions and still useful lower bounds and runs in 30 min. on a GPU. AL gave useful results after 6×200 iterations, which took roughly 1 hour.

Weak Data Terms. Finally, we consider problems with more difficult data terms, where we introduce noise on the stereo images and no longer pre-smooth those for motion. For stereo, there are very little changes. For motion we draw three conclusions: Firstly, AL now breaks down, requiring many more iterations to produce competitive bounds. Secondly, TRW-S is now even more superior to MPLP - it needs 50 iterations to produce the same bound as MPLP after 500. Thirdly, the lower bounds are now much looser, up to 12% away from the best known integral solutions. Moreover, the thresholded solutions absolutely need the refinement process.

7 CONCLUSIONS

We have considered several relaxation-based techniques for the problems of motion and stereo with discretized displacement sets. In summary, the fastest and most memory saving methods available are not relaxation-based at all: the expansion moves win. However, combining them with convex relaxations gives better results as well as tight lower bounds.

TRW-S is generally the best relaxation-based strategy to get good solutions very fast, and is clearly superior to MPLP. Smooth approximations often need less memory, are parallelizable and work about equally good. Augmented Lagrangians need very much memory and, although they give very tight bounds in some cases, they do not perform good in general. To facilitate further research in this area, the source code associated to this paper will be published.

ACKNOWLEDGEMENTS

We thank Fredrik Kahl for helpful discussions. This work was funded by the European Research Council (GlobalVision grant no. 209480).

REFERENCES

Bertsekas, D. (1999). *Nonlinear Programming, 2nd edition*. Athena Scientific.

Bhusnurmath, A. and Taylor, C. J. (2008). Solving stereo matching problems using interior point methods. In *Fourth International Symposium on 3D Data Processing, Visualization and Transmission, 3DPVT*.

Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *IEEE*

Transactions on Pattern Analysis and Machine Intelligence (PAMI), 23(11):1222–1239.

Bruhn, A. (2006). *Variational optic flow computation: Accurate modelling and efficient numerics*. PhD thesis, Universität des Saarlandes, Saarbrücken, Germany.

Dantzig, G. and Thapa, M. (1997). *Linear Programming 1: Introduction*. Springer Series in Operations Research. Springer.

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

Felzenszwalb, P. and Huttenlocher, D. (2006). Efficient belief propagation for early vision. *International Journal on Computer Vision (IJCV)*, 70(1).

Globerson, A. and Jaakkola, T. (2007). Fixing max-product: Convergent message passing algorithms for MAP relaxations. In *Conference on Neural Information Processing Systems (NIPS)*, Vancouver, Canada.

Glocker, B., Komodakis, N., Tziritas, G., Navab, N., and Paragios, N. (2008). Dense image registration through MRFs and efficient linear programming. *Medical Image Analysis*, 12:731–741.

Goldlücke, B. and Cremers, D. (2010). Convex relaxation for multilabel problems with product label spaces. In *European Conference on Computer Vision (ECCV)*, Iraklion, Greece.

Goldstein, T., Bresson, X., and Osher, S. (2009). Global minimization of Markov random fields with applications to optical flow. Technical Report 09-77, UCLA CAM report.

Horn, B. and Schunck, B. (1981). Determining optical flow. *Artificial Intelligence*, 17:185–203.

Ishikawa, H. (2003). Exact optimization for Markov Random Fields with convex priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 25(10):1333–1336.

Klaus, A., Sormann, M., and Karner, K. (2006). Segment-based stereo matching using adaptive cost aggregation and dynamic programming. In *International Conference on Pattern Recognition (ICPR)*, Hong Kong, China.

Kleinberg, J. and Tardos, E. (1999). Approximation algorithms for classification problems with pairwise relationships: metric labeling and Markov Random Fields. In *Symposium on Foundations of Computer Science*.

Kolmogorov, V. (2006). Convergent tree-reweighted message passing for energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 28(10):1568–1583.

Lellmann, J., Becker, F., and Schnörr, C. (2009). Convex optimization for multi-class image labeling by simplex-constrained total variation. In *IEEE International Conference on Computer Vision (ICCV)*, Kyoto, Japan.

Meltzer, T., Yanover, C., and Weiss, Y. (2005). Globally optimal solutions for energy minimization in stereo vision using reweighted belief propagation. In *IEEE International Conference on Computer Vision (ICCV)*, Beijing, China.

Memin, E. and Perez, P. (1998). Dense estimation and object-based segmentation of the optical flow with robust techniques. *IEEE Transactions on Image Processing (TIP)*, 7(5):703–719.

Michelot, C. (1986). A finite algorithm for finding the projection of a point onto the canonical simplex of $\mathbb{R}^n$. *Journal on Optimization Theory and Applications*, 50(1).

Nesterov, Y. (2004). *Introductory lectures on convex optimization*. Applied Optimization. Kluwer Academic Publishers.

Nesterov, Y. (2005). Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152.

Papenberg, N., Bruhn, A., Brox, T., Didas, S., and Weickert, J. (2006). Highly accurate optic flow computation with theoretically justified warping. *International Journal on Computer Vision (IJCV)*, 67(2):141–158.

Pock, T., Schoenemann, T., Cremers, D., and Bischof, H. (2008). A convex formulation for continuous multi-label problems. In *European Conference on Computer Vision (ECCV)*, Marseille, France.

Scharstein, D. and Szeliski, R. (2002). A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal on Computer Vision (IJCV)*, 47(1–3):7–42.

Shekhovtsov, A., Kovtun, I., and Hlaváč, V. (2008). Efficient MRF deformation model for non-rigid image matching. *Computer Vision and Image Understanding*, 112(1):91–99.

Sontag, D., Globerson, A., and Jaakkola, T. (2010). Introduction to dual decomposition for inference. In *Optimization for Machine Learning*. MIT Press.

Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., and Rother, C. (2008). A comparative study of energy minimization methods for Markov random fields with smoothness-based priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 30(6):1068–1080.

Trobin, W., Pock, T., Cremers, D., and Bischof, H. (2008). An unbiased second-order prior for high-accuracy motion estimation. In *Pattern Recognition (Proc. DAGM)*, Munich, Germany.

Wainwright, M., Jaakkola, T., and Willsky, A. (2005). MAP estimation via agreement on (hyper-)trees: Message-passing and linear programming approaches. *IEEE Tansactions on Information Theory*, 51(11):3697–3717.

Woodford, O., Torr, P., Reid, I., and Fitzgibbon, A. (2008). Global stereo reconstruction under second order smoothness priors. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, Alaska.

Yang, Q., Wang, L., Yang, R., Stewénius, H., and Nistér, D. (2009). Stereo matching with color-weighted correlation, hierarchical belief propagation and occlusion handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31(3):492–504.

Zach, C., Gallup, D., Frahm, J.-M., and Niethammer, M. (2008). Fast global labeling for real-time stereo using multiple plane sweeps. In *Vision, Modeling and Visualization Workshop (VMV)*, Konstanz, Germany.

Zach, C., Pock, T., and Bischof, H. (2007). A duality based approach for realtime TV-L1 optical flow. In *Pattern Recognition (Proc. DAGM)*, Heidelberg, Germany.