In Defense of 3D-Label Stereo

Carl Olsson Johannes Ulén
Centre for Mathematical Sciences
Lund University, Sweden
calle@maths.lth.se ulen@maths.lth.se

Yuri Boykov
Department of Computer Science
University of Western Ontario, Canada
yuri@csd.uwo.ca

Abstract

It is commonly believed that higher order smoothness should be modeled using higher order interactions. For example, 2nd order derivatives for deformable (active) contours are represented by triple cliques. Similarly, the 2nd order regularization methods in stereo predominantly use MRF models with scalar (1D) disparity labels and triple clique interactions. In this paper we advocate a largely overlooked alternative approach to stereo where 2nd order surface smoothness is represented by pairwise interactions with 3D-labels, e.g. tangent planes. This general paradigm has been criticized due to perceived computational complexity of optimization in higher-dimensional label space. Contrary to popular beliefs, we demonstrate that representing 2nd order surface smoothness with 3D labels leads to simpler optimization problems with (nearly) submodular pairwise interactions. Our theoretical and experimental results demonstrate advantages over state-of-the-art methods for 2nd order smoothness stereo. [1]

1. Introduction

Dense stereo matching is one of the core problems of Computer Vision. In recent years considerable progress has been made due to the availability of powerful regularizers for handling ambiguous and noisy data. Perhaps the most common are the first order regularization priors [4, 7, 6]. One reason for their popularity is that when applying move-making algorithms such as α-expansion [4] or fusion moves [8] they often result in submodular moves, allowing efficient computation using min-cut/max-flow algorithms [4].

Many basic optimization methods for stereo use scalar (1D) disparity labels. Such methods often implicitly assume fronto-parallel planes. For example, standard piecewise smooth (e.g. truncated linear or quadratic) pairwise regularization potentials assign higher cost to surfaces with larger tilt [4]. To model surfaces more accurately Birchfeld and Tomasi [1] introduced 3D-labels corresponding to arbitrary 3D planes, but this approach is limited to piecewise planar scenes. To address more general scenes our paper follows the popular trend of using 2nd derivative surface regularization for stereo [9, 15].

There are two ways of modeling such higher order smoothness potentials. Woodford *et al*. [15] retain the scalar disparity labels while using triple-cliques to penalize 2nd derivatives of the reconstructed surface. This encourages near planar smooth disparity maps. The optimization problem is however made substantially more difficult due to the introduction of non-submodular triple interactions.

In contrast, Li and Zucker [9] use 3D-labels corresponding to tangent planes to encode 2nd order disparity map smoothness as pairwise interactions. A similar idea was also recently applied to surface reconstruction from sparse point clouds [10]. Li and Zucker's interaction penalizes two terms. The first term computes the difference of the disparity assignment, and the disparity predicted by the neighboring tangent plane. (We show in this paper that this alone actually corresponds to penalizing 2nd derivative.) The second term penalizes the (squared) angular difference between neighboring tangent plane normals. This term encourages parallel assignments of tangent planes. The interaction is non-submodular and the authors employ a belief propagation algorithm to optimize it. In comparison to Woodford *et al*. [15] that only use a disparity estimate at each pixel, the approach by Li and Zucker [9] requires discretization of a much larger label space. Their discretized 3D-labels are precomputed as locally optimal deformed windows with respect to the SSD measure [5], thereby disregarding the assignments of neighboring pixels. As shown in [15], this specific approach results in disparity maps that are inferior to those of Woodford *et al*. The discussed limitations of [9] may have helped to promote the general perception of triple interactions of scalar disparity labels as a superior approach for modeling 2nd order smoothness.

Bleyer *et al*. [2] use even higher dimensional labels to represent surface models like b-splines. This makes it possible to precompute and penalize 2nd order smoothness via

[1] This work has been funded by the Swedish Research Council (grant 2012-4213), the Swedish Foundation for Strategic Research (SSF) through the programs *Future Research Leaders* and *Wearable Visual Information Systems*, by the European Research Council (GlobalVision grant no. 209480), the Canadian Foundation for Innovation (CFI 10318) and the Canadian NSERC Discovery Program (grant 298299-2012RGPIN).

a unary term. There is however no smoothness interaction between models and therefore it is not possible to combine local models into global ones. The label space is even more complex than that of Li and Zucker [9] and therefore re-estimation of the surfaces is crucial for this approach.

In this paper we propose a new 3D-label stereo algorithm encoding 2nd order smoothness of the disparity map with pairwise interactions. While similar to Li and Zucker's general idea to reconstruct piecewise smooth surfaces from local models (tangent planes), our algorithm resolves many problems, see [15], associated with this approach. We show how to properly measure 2nd derivative of the reconstructed surface using pairwise cliques when the labels are tangent planes. Instead of using a fixed set of locally precomputed tangents, we adaptively generate new surface proposals based on the current surface estimate. We also replace a no-guarantee belief-propagation in [9] with QPBO-based fusion similar to [15]. In contrast to the triple-cliques used by Woodford *et al.* [15] we show that our formulation is submodular when using planar proposals, and verify experimentally that Roof duality [11] labels much more pixels for general proposals with our formulation. Besides being a simpler optimization problem we also demonstrate that the running time is reduced. We show that the use of even higher order labels (that encode higher order derivatives) further extends the class of submodular functions. In addition we present a version of our method that works with depth rather than disparity and, therefore, does not require rectified cameras.

1.1. Optimization Background

Consider the optimization of an arbitrary second order pseudo-boolean function (PBF) of n variables, usually expressed as,

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

where $\mathcal{N}$ is some connectivity. Our goal is to find a minimizer of (1). If $\boldsymbol{L} = \{0, 1\}$ and V_{pq} is submodular this can be efficiently solved [4]. Even with V_{pq} non-submodular roof-duality (RD) [3, 11] can be used. RD will give a partial solution where labeled variables are guaranteed to be correct for an optimal solution and some variables are left *unlabeled*.

Lempitsky *et al.* [8] proposed a way to minimize (1) when $\boldsymbol{L} = \mathsf{R}$. Given two assignments $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ we fuse them into a new one with lower energy by solving

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

where $\cdot$ is element-wise multiplication. If we solve (2) using RD and then set $\boldsymbol{z} = \mathbf{0}$ for all unlabelled variables the autarky results in [11] gives us,

$$E(\boldsymbol{z}\cdot\boldsymbol{x}_0 + (\mathbf{1}-\boldsymbol{z})\cdot\boldsymbol{x}_1) \leq \min\left(E(\boldsymbol{x}_0), E(\boldsymbol{x}_1)\right). \quad (3)$$

Therefore we can iteratively minimize (1) by proposing new solutions and fusing them with the old solution.

The possibility to decrease the energy for each fusion move is an attractive feature, however there is no guarantee on how many variables will be labeled in each fusion move. As will be shown in Section 5.1.1 a large fraction of the variables might be unlabeled. For submodular fusion moves we are guaranteed to label all variables. Minimizing a submodular function is also faster in practice [11].

2. A Second Order Smoothness Prior for Multi-View Stereo

In this section we present a second order smoothness prior for dense multi-view stereo reconstruction. The idea is similar to that of [9]. We will use 3D-labels to represent the prior using pairwise cliques. To each viewing ray we will assign a plane that locally represents the surface geometry close to the ray. The intersection between the ray and the plane will be the estimated 3D point. By interpreting the planes as a tangents of the viewed 3D surface we can encourage smooth solutions by penalizing neighboring 3D-points that deviate largely from neighboring tangents.

2.1. Rectified Cameras and Disparity Maps

We will start by assuming that the cameras have been rectified, since this allows us to work in disparity space. For multiple views this does however place some restrictions on the camera positions that are usually not fulfilled in general image collections [13]. We will therefore relax this condition in Section 2.2.

Let p and q be to two neighboring pixels in the image $\mathcal{I}$ and $\boldsymbol{p}$, $\boldsymbol{q}$ be their image coordinates. The goal of stereo is to estimate the function $\mathcal{D} : \mathcal{I} \mapsto \mathsf{R}$ that gives a disparity value for each pixel in the image. To each pixel p we will assign a tangent plane that locally approximates this function. We can think of these tangents as samples of the disparity function and its derivatives. By the function $\mathcal{T}_{\boldsymbol{p}}\mathcal{D} : \mathcal{I} \mapsto \mathsf{R}$ we will mean the tangent at the point p seen as a function of the whole image, that is

$$\mathcal{T}_{\boldsymbol{p}}\mathcal{D}(\boldsymbol{x}) = \mathcal{D}(\boldsymbol{p}) + \nabla\mathcal{D}(\boldsymbol{p})^T(\boldsymbol{x}-\boldsymbol{p}), \quad (4)$$

where $\mathcal{D}(\boldsymbol{p})$ and $\nabla\mathcal{D}(\boldsymbol{p})$ is the assigned disparity and disparity gradient (with respect to the image coordinate system) at pixel p. We define a pairwise interaction between neighboring pixels as

$$V_{pq} = |\mathcal{T}_{\boldsymbol{p}}\mathcal{D}(\boldsymbol{q}) - \mathcal{D}(\boldsymbol{q})|, \quad (5)$$

That is, V_{pq} measures the curve's deviation from the tangent plane, see Figure 1. Intuitively, if the surface is smooth then

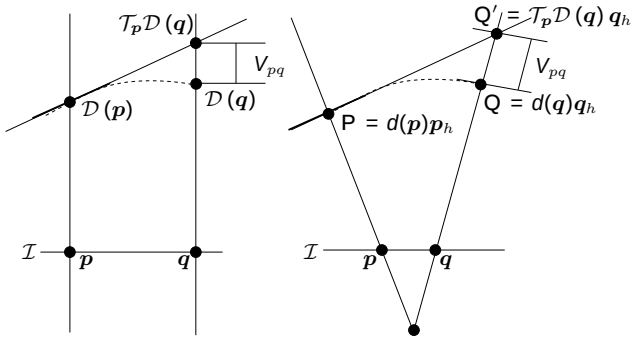


Figure 1. Left: Geometric interpretation of the smoothness term for parallel viewing rays. Right: Smoothness term when the viewing rays are not parallel.

the tangent plane should be a good approximation. Therefore we expect V_{pq} to be small for smooth surfaces. Using the Taylor expansion

$$\mathcal{D}(\boldsymbol{q}) \approx \mathcal{D}(\boldsymbol{p}) + \nabla\mathcal{D}(\boldsymbol{p})^T(\boldsymbol{q}-\boldsymbol{p}) + \frac{1}{2}(\boldsymbol{q}-\boldsymbol{p})^T\nabla^2\mathcal{D}(\boldsymbol{p})(\boldsymbol{q}-\boldsymbol{p}) \quad (6)$$

where $\nabla^2\mathcal{D}(\boldsymbol{p})$ is the Hessian at p, we see that

$$V_{pq} \approx |\frac{1}{2}(\boldsymbol{q}-\boldsymbol{p})^T\nabla^2\mathcal{D}(\boldsymbol{p})(\boldsymbol{q}-\boldsymbol{p})|. \quad (7)$$

That is, V_{pq} measures the second derivative at $\boldsymbol{p}$ in direction $\boldsymbol{q}-\boldsymbol{p}$ of the underlying disparity function.

2.2. Regular Cameras

In many real world situations rectified cameras may not be available. In such cases we work with depth rather than disparity. However directly penalizing 2nd derivative of the depth function is not a good idea. In general the projection of a plane will not yield a linear depth function unless the camera is affine (which can be seen from (11) below). Hence, such an energy would assign a 3D-plane a nonzero penalty. Therefore we will instead measure the deviation from the tangent plane along the viewing ray.

Let $\boldsymbol{p}_h$ and $\boldsymbol{q}_h$ denote the homogeneous coordinates (with third coordinate 1) of the two pixels p and q. We will assume a pinhole camera model where the center camera has been calibrated and normalized to be of form $[\boldsymbol{I}\ \mathbf{0}]$. Given a function $d : \mathcal{I} \mapsto \mathsf{R}_+$ that gives a depth for each pixel the 3D points $\mathbf{P}$ and $\mathbf{Q}$ corresponding to p and q can be computed (in regular Cartesian coordinates) using the simple formulas

$$\mathbf{P} = d(\boldsymbol{p})\boldsymbol{p}_h, \quad (8)$$
$$\mathbf{Q} = d(\boldsymbol{q})\boldsymbol{q}_h. \quad (9)$$

By $(\mathbf{n}_p, a_p)$, where $\mathbf{n}_p \in \mathsf{R}^3$, $\|\mathbf{n}_p\| = 1$ and $a_p \in \mathsf{R}$ we denote the tangent plane at $\boldsymbol{p}$ given by the equation

$$\mathbf{n}_p^T\boldsymbol{x} + a_p = 0. \quad (10)$$

Consider the intersection point $\mathbf{Q}'$ between the viewing ray at $\boldsymbol{q}$ and the tangent plane at $\boldsymbol{p}$. We let $\mathcal{T}_{\boldsymbol{p}}d : \mathcal{I} \mapsto \mathsf{R}_+$ be the depth function of the tangent plane at point $\boldsymbol{p}$, that is, $\mathbf{Q}' = \mathcal{T}_{\boldsymbol{p}}d(\boldsymbol{q})\,\boldsymbol{q}_h$. We can calculate the tangent function using

$$\mathcal{T}_{\boldsymbol{p}}d(\boldsymbol{q}) = -\frac{a_p}{\mathbf{n}_p^T\boldsymbol{q}_h}. \quad (11)$$

(Here we are assuming that the viewing ray is not completely contained in the tangent plane.) Note that even though this function represents a plane in 3D it is usually not linear in $\boldsymbol{q}$. In contrast disparity is inversely proportional to depth and will therefore be linear.

To encourage smooth assignments we use the cost

$$V_{pq} = \|\mathbf{Q}-\mathbf{Q}'\| = |\mathcal{T}_{\boldsymbol{p}}d(\boldsymbol{q}) - d(\boldsymbol{q})|\,\|\boldsymbol{q}_h\|. \quad (12)$$

The geometric interpretation of this expression can be seen in Figure 1. Given the estimated tangent plane at $\boldsymbol{p}$ and the depth at $\boldsymbol{q}$ the interaction computes the distance between the estimated 3D point and the tangent plane along the viewing ray. The smoothness term will penalize deviations from planes and thereby encourage solutions with small second derivatives.

The interaction in (12) is very similar to (5) and the properties that we derive in Section 3 will hold for both interactions.

3. Submodularity of Fusion Moves

In this section we will show that fusion moves [8] with our interactions are often submodular. Given a current disparity function $\mathcal{D}$ and a proposal function $\mathcal{P}$ the fusion move allows pixels to change their labels from the tangents of $\mathcal{D}$ to the tangents of $\mathcal{P}$. In what follows we will use

$$V_{pq}(\mathcal{D},\mathcal{P}) = |\mathcal{T}_{\boldsymbol{p}}\mathcal{D}(\boldsymbol{q}) - \mathcal{P}(\boldsymbol{q})|, \quad (13)$$

to mean the penalty for assigning p the tangent plane from $\mathcal{D}$ and q the tangent plane from $\mathcal{P}$. Note that our energies will also contain a similar term $V_{qp}(\mathcal{P},\mathcal{D})$ and therefore the penalty is symmetric. However for showing submodularity it is sufficient to consider one of them at a time.

3.1. Candidate Planes

We first show that fusion moves where the candidate function $\mathcal{P}$ is a plane result in submodular terms. The result is a simple consequence of the triangle inequality.

Proposition 3.1 *If the proposal $\mathcal{P}$ is a plane then the fusion with any function $\mathcal{D}$ is a submodular move.*

Since $\mathcal{P}$ is a plane we have

$$\mathcal{T}_{\boldsymbol{p}}\mathcal{P}(\boldsymbol{q}) = \mathcal{P}(\boldsymbol{q}) \quad (14)$$

and therefore $V_{pq}(\mathcal{P},\mathcal{P}) = 0$. Furthermore,

$$V_{pq}(\mathcal{D},\mathcal{D}) = |\mathcal{T}_{\boldsymbol{p}}\mathcal{D}(\boldsymbol{q}) - \mathcal{D}(\boldsymbol{q})| \quad (15)$$
$$= |\mathcal{T}_{\boldsymbol{p}}\mathcal{D}(\boldsymbol{q}) - \mathcal{P}(\boldsymbol{q}) + \mathcal{T}_{\boldsymbol{p}}\mathcal{P}(\boldsymbol{q}) - \mathcal{D}(\boldsymbol{q})| \quad (16)$$
$$\leq |\mathcal{T}_{\boldsymbol{p}}\mathcal{D}(\boldsymbol{q}) - \mathcal{P}(\boldsymbol{q})| + |\mathcal{T}_{\boldsymbol{p}}\mathcal{P}(\boldsymbol{q}) - \mathcal{D}(\boldsymbol{q})| \quad (17)$$
$$= V_{pq}(\mathcal{D},\mathcal{P}) + V_{pq}(\mathcal{P},\mathcal{D}) \quad (18)$$

which shows that submodularity,

$$V_{pq}(\mathcal{D},\mathcal{D}) + V_{pq}(\mathcal{P},\mathcal{P}) \leq V_{pq}(\mathcal{P},\mathcal{D}) + V_{pq}(\mathcal{D},\mathcal{P}), \quad (19)$$

holds.

3.2. General Candidates

Next we derive some more general sufficient conditions for submodularity of the fusion move.

Proposition 3.2 *If both $\mathcal{D}$ and $\mathcal{P}$ are convex (or alternatively both concave) between $\boldsymbol{p}$ and $\boldsymbol{q}$ then the interactions V_{pq} and V_{qp} are submodular for the fusion move.*

To see this we first note that if both $\mathcal{D}$ and $\mathcal{P}$ are convex then they are both bounded from below by their tangent planes. Specifically

$$\mathcal{D}(\boldsymbol{q}) \geq \mathcal{T}_{\boldsymbol{p}}\mathcal{D}(\boldsymbol{q}) \quad (20)$$
$$\mathcal{P}(\boldsymbol{q}) \geq \mathcal{T}_{\boldsymbol{p}}\mathcal{P}(\boldsymbol{q}). \quad (21)$$

Using the above we now have

$$V_{pq}(\mathcal{D},\mathcal{D}) + V_{pq}(\mathcal{P},\mathcal{P}) = \quad (22)$$
$$\Big(\mathcal{D}(\boldsymbol{q}) - \mathcal{T}_{\boldsymbol{p}}\mathcal{D}(\boldsymbol{q}) + \mathcal{P}(\boldsymbol{q}) - \mathcal{T}_{\boldsymbol{p}}\mathcal{P}(\boldsymbol{q})\Big) \leq \quad (23)$$
$$\Big(|\mathcal{P}(\boldsymbol{q}) - \mathcal{T}_{\boldsymbol{p}}\mathcal{D}(\boldsymbol{q})| + |\mathcal{D}(\boldsymbol{q}) - \mathcal{T}_{\boldsymbol{p}}\mathcal{P}(\boldsymbol{q})|\Big) = \quad (24)$$
$$V_{pq}(\mathcal{D},\mathcal{P}) + V_{pq}(\mathcal{P},\mathcal{D}). \quad (25)$$

It is easy to see that the same statement is true if both $\mathcal{P}$ and $\mathcal{D}$ are concave. In this case the inequalities in (20) and (21) are switched which means that the signs of (23) are switched.

3.3. A Discontinuity Preserving Energy

To make the energy discontinuity preserving we add a threshold t to the interaction

$$E_{pq}(\mathcal{D},\mathcal{P}) := \min(V_{pq}(\mathcal{D},\mathcal{P}), t). \quad (26)$$

In the case of a plane proposal $\mathcal{P}$ we see that

$$\min(V_{pq}(\mathcal{D},\mathcal{D}), t) \leq \quad (27)$$
$$\min\big(V_{pq}(\mathcal{D},\mathcal{P}) + V_{pq}(\mathcal{P},\mathcal{D}), t\big) \leq \quad (28)$$
$$\min\big(V_{pq}(\mathcal{D},\mathcal{P}), t\big) + \min\big(V_{pq}(\mathcal{P},\mathcal{D}), t\big). \quad (29)$$

$V_{pq}(\mathcal{P},\mathcal{P}) = 0$ as shown earlier resulting in the inequality,

$$E_{pq}(\mathcal{D},\mathcal{D}) + E_{pq}(\mathcal{P},\mathcal{P}) \leq E_{pq}(\mathcal{D},\mathcal{P}) + E_{pq}(\mathcal{P},\mathcal{D}), \quad (30)$$

showing that planar proposals also generate submodular interactions with this energy.

The result in Proposition 3.2 can fail for surfaces of high curvature because of the added threshold. It is possible to add extra constraints on the derivatives (e.g. $V_{pq}(\mathcal{D},\mathcal{D}) + V_{pq}(\mathcal{P},\mathcal{P}) \leq t$) to extend this result. However since we do not explicitly check or enforce these in our implementations we do not pursue this further.

4. General Order Smoothness Priors

In Section 2 we used tangent planes to create our smoothness prior. It is possible to use higher order local models to encode more complex priors. Let $\mathcal{A}_p\mathcal{D}$ be a Taylor approximation of order n, then the interaction

$$V_{pq}(\mathcal{D},\mathcal{D}) = |\mathcal{A}_p\mathcal{D}(\boldsymbol{q}) - \mathcal{D}(\boldsymbol{q})| \quad (31)$$

would be a $n+1$ order smoothness penalty. At the same time we can add a penalty for derivatives of order at most n using only unary terms. For example, if we to each pixel assign a quadratic function instead of a tangent we have an interaction that penalizes 3rd derivatives. In this case 2nd and 1st derivatives can be encoded into the unary term. As in Proposal 3.1 it is easy to see that if our proposals fulfill

$$\mathcal{A}_p\mathcal{P}(\boldsymbol{q}) = \mathcal{P}(\boldsymbol{q}), \quad (32)$$

then the fusion move will be submodular. Hence the n'th order surfaces give submodular interactions. For example if we only use the zero order expansion (constant functions/ fronto-parallel planes) then we find that fusion moves with constant depth proposals are submodular. This is the regular version of α-expansion from [4]. Table 1 shows properties for some different choices of labels.

5. Experiments

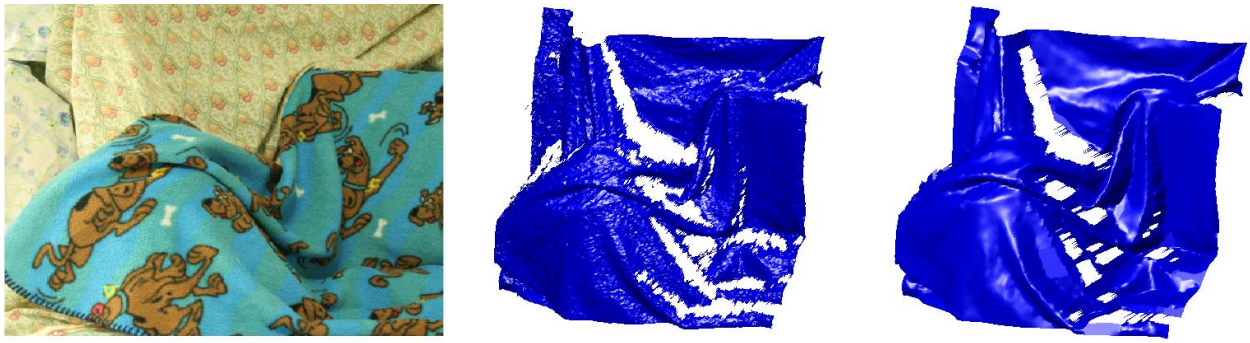
Next we evaluate the proposed framework on a couple of multiple view stereo data sets. The energy that we use is of the standard form

$$E(\mathcal{D}) = \sum_p \sum_{q\in\mathcal{N}(p)} E_{pq}(\mathcal{D},\mathcal{D}) + \mu \sum_p E_p(\mathcal{D}) \quad (33)$$

where E_{pq} is the smoothness term presented in Section 3.3. In all following experiments the neighborhood $\mathcal{N}$ is chosen as 4-connectivity. We will use both the disparity version (Section 2.1) and the depth version (Section 2.2). The parameter μ controls the tradeof between the smoothness and data terms.

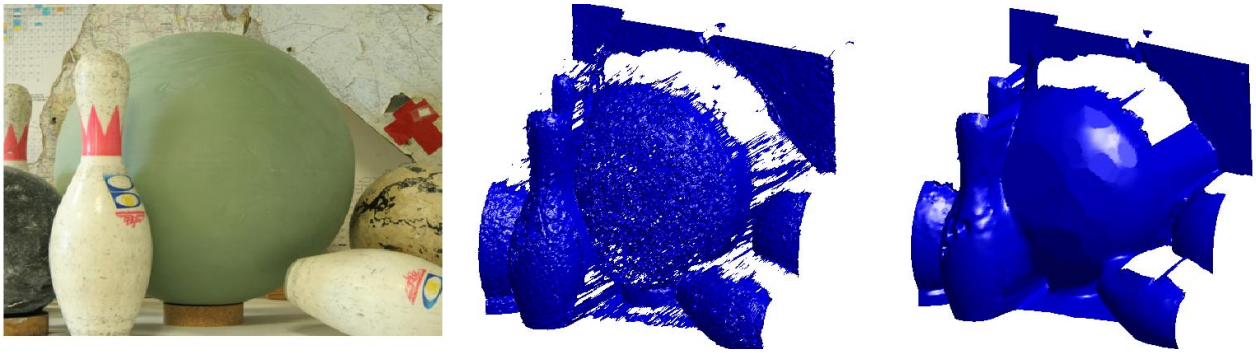
Label	Pairwise Interaction	Unary Term	Submodular Proposals
Depth	1st derivative	Depth	Constant functions
Tangent planes	2nd derivative	Depth, 1st derivative	Constant 1st derivative
2nd order approximation	3rd derivative	Depth, 1st, 2nd derivative	Constant 2nd derivative
⋮	⋮	⋮	⋮

Table 1. Characterization of Pairwise interactions, unary terms and submodular proposals for different types of labels.



(a) (b) (c)

Figure 2. Results for the cloth sequence. (a) - Image, (b) - depth map using only the data term, (c) - depth map computed with regularization.



(a) (b) (c)

Figure 3. Results for the bowling sequence. (a) - Image, (b) - depth map using only the data term, (c) - depth map computed with regularization.

The data term E_p is a unary term that depends on the tangent plane at p. For this term we use normalized cross correlation (NCC) computed at different possible depth\disparities. For each depth we use a planar homography to project one of the neighboring images into the center image. Then we compute NCC of 3×3 patches (if a pixel is outside the image boundary we assign it NCC zero). This way we get a cross correlation function of depth for each pixel. In principle we could make the NCC depend on the tilt of the tangent as well, however storing the samples of such a function would require lots of memory. We compute NCC for every neighboring camera and take mean values over the cameras to obtaining the final result. Occlusion is modeled using the approach of [14].

Since our algorithm may assign depths/disparities that are in between the sampled values we use quadratic interpolation to represent the cost function at every possible depth. We also add an extra cost to assignments of planes which are roughly parallel to the viewing rays. The reason for doing this is that we are unlikely to be able to see many pixels from such planes (and if we do, the data term that we have computed using fronto-planar patches is probably not accurate). We us the extra cost

$$(1 - \mathbf{n}_p^T \mathbf{v}_p)^{2k}, \tag{34}$$

where $\mathbf{n}_p$ is the normal of the plane assigned to p and $\mathbf{v}_p$

is the direction of the viewing ray in p (in the 3D space the viewing ray direction will be $\boldsymbol{p}/\|\boldsymbol{p}\|$ and in disparity space $(0,0,1)$). The constant k is selected large enough so that the penalty effects tangents with high tilt, in our experiments we use $k = 10$.

5.0.1 Proposal Generation

To generate proposals we use similar heuristics to those of [15].

- To generate planar proposals we randomly select a point and a small neighborhood. Using the best local maximum of the normalized cross correlation for each viewing ray we create a 3D cloud to the neighborhood and fit a plane using RANSAC.
- We also generate 2nd order surfaces using a similar RANSAC approach as above.
- We use a filtering process that takes the current assignment, computes the corresponding 3D points, and for each pixel fits a plane to its neighboring 3D points.
- Finally we have a proposal that just increases/decreases the depth/disparity of all proposals with a small random step size.

5.1. Rectified Cameras

We first present results obtained on two of the well known Middlebury stereo sequences [12], Cloth and Bowling see Figures 2-3. For their data sets we computed a depth map for the middle image (nr 4) and used the remaining 6 images to compute the cross correlations needed for the data term. We used the data weight $\mu = 40$ and the threshold $t = 1$ (see Secction 3.3) to generate these results. The effects of the regularization term can be seen by comparing the surface generated from the data term without regularization (b) and the one with regularization (c) (the data term is particularly weak in the bowling data set because of the large texture less region).

5.1.1 Comparison to [15]

In this section we compare our regularization to the one used by Woodford *et al.* in [15], hereafter called GlobalStereo. GlobalStereo also penalize second derivative but use triple cliques with scalar disparity labels. We will show that switching to our regularization leads to simpler problem instances, without reducing the quality of the results. The comparison is performed on the Middlebury data set consisting of stereo pairs of rectified images.

In order to make a fair comparison we use data terms computed as outlined in [15] for both methods in this experiment. Furthermore we do not use any occlusion model here since these are different for the two models.

	Tsukuba	Venus	Teddy	Cones
Our	0.065 %	0.0264 %	0.127 %	0.0847 %
GlobalStereo	30.0 %	30.6 %	27.6 %	27.3 %
GlobalStereo 1op	0 %	0 %	0 %	0.0411 %

Table 2. Unlabelled for the 14 SegPln proposals on Middlebury.

The only remaining difference is therefore the regularization. In [15] this is computed as

$$E_{\text{smooth}}(\boldsymbol{D}) = \sum_{\mathbf{N}\in\mathcal{N}} W(\mathbf{N}) \min\left(|S(\mathbf{N})|, \sigma_s\right). \quad (35)$$

Here $\mathcal{N}$ is a collection of (horizontal and vertical) triples $\{p,q,r\}$ over which the regularization is computed, $S(\mathbf{N})$ is the approximation of second derivative

$$S(\mathbf{N}) = \boldsymbol{D}(\boldsymbol{p}) - 2\boldsymbol{D}(\boldsymbol{q}) + \boldsymbol{D}(\boldsymbol{r}) \quad (36)$$

and $W(\mathbf{N})$ is a weight depending on a segmentation of the image (see [15] for further details, all parameters are chosen as in GlobalStereo). If $\mathbf{N}$ contains pixels from several segments $W(\mathbf{N})$ takes a low value, if all pixels are from the same segment it takes a high value.

To achieve a similar regularization with our method we change $\mathcal{N}$ to normal 4-connectivity. Now $\mathcal{N}$ consist of collection of pairs $\{p,q\}$ and the second derivative is penalized using,

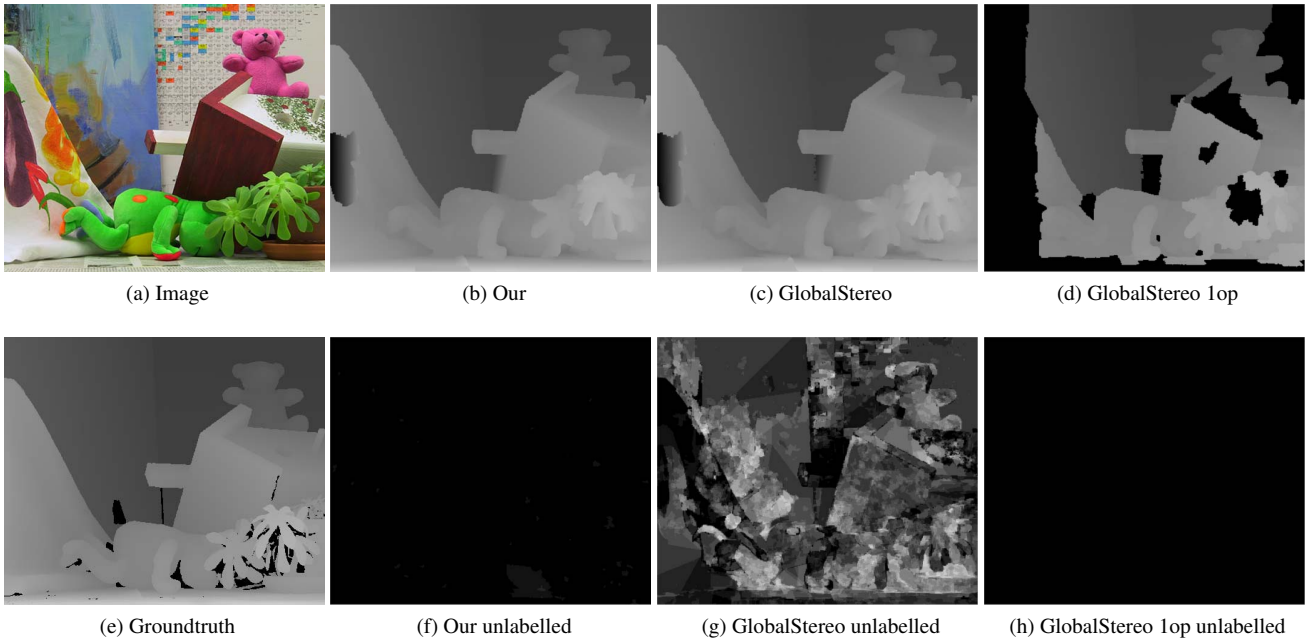
$$S(\mathbf{N}) = E_{pq} + E_{qp}. \quad (37)$$

As a reference we also use the first order priors defined in [15] which we call GlobalStereo 1op. In the fusion moves we use "improve" [11] after running RD to label all unlabelled variables.

GlobalStereo uses three types of proposals. First we consider the *Segpln*-proposals which are 14 piecewise planar proposals generated from a segmentation (see [15]). In Figure 4 we started from the same randomized disparity function with tangents parallel to the image plane. We then fused each SegPln proposal one at a time for both methods. We kept track of how many variables where unlabeled after RD for both methods and presented the numbers in Table 2 and the resulting disparity maps in Figure 4.

Note that the fusion move for our method is only submodular if we fuse one planar function at a time. The SegPln proposals are piecewise planar and the regularization at transitions between planes may not be submodular.

We also test our regularization on the full pipeline of GlobalStereo which uses all three types of proposals (*Segpln*, *SameUni* and *Smooth*). The results on all of the 2003 Middlebury Sequences [12] are given in Table 3 and the running times are given in Table 4.



(a) Image (b) Our (c) GlobalStereo (d) GlobalStereo 1op

(e) Groundtruth (f) Our unlabelled (g) GlobalStereo unlabelled (h) GlobalStereo 1op unlabelled

Figure 4. The Teddy sequence from Middlebury [12]. (b-d) are estimated disparity maps after fusing the 14 SegPln proposals. In (f-h) we present the unlabelled variables summed over all 14 proposals scaled 0–14. A white pixel would mean that fusing a proposal for this pixel failed for every single proposal.

	Tsukuba			Venus			Teddy			Cones			Average
	Non occ	All	Disc	Non occ	All	Disc	Non occ	All	Disc	Non occ	All	Disc	
Our	4.49	5.52	12.3	0.298	0.648	3.99	7.71	11.2	17.8	9.78	15.4	18.3	8.95
GlobalStereo	4.83	5.99	13.9	0.536	0.921	6.39	8.16	11.8	19.3	9.74	15.6	18.4	9.63

Table 3. Scores on Middlebury [12] using the same proposals, lower is better. All values are % of pixels being ≥ 1 pixel incorrect for each of the three classes. The classes are **non occ**luded regions, **all** pixels and regions near depth **disc**ontinuities.

	Tsukuba	Venus	Teddy	Cones	Average
Our	21.3	25.5	29.4	36.5	28.2
GlobalStereo	106	139	143	181	142
GlobalStereo/Our	4.96	5.47	4.87	4.96	5.07

Table 4. Running time for the optimization in seconds using the convergence criteria in GlobalStereo.

5.2. Regular Cameras

In this final section we compute depth maps for a couple of images that are not rectified, see Figures 5-6. Both these images are part or real outdoor data sets, and as a preprocessing step we have removed the background sky. In both cases we use 9 neighboring images to compute cross correlations. Here we used the data weight $\mu = 10$ and the threshold $t = 1$.

6. Conclusions

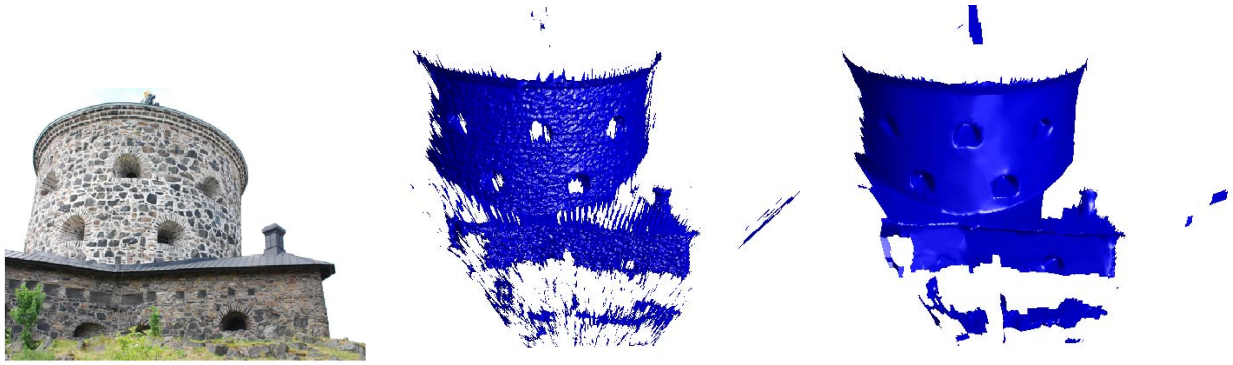
In this paper we advocated a largely overlooked approach to stereo with 2nd order smoothness regularization. In contrast to popular approaches where triple cliques are used for representing 2nd order surface derivatives, we proposed to use pairwise interactions with 3D-labels. We showed that this leads to simpler optimization problems and in many cases (nearly) submodular fusion moves.

References

[1] S. Birchfield and C. Tomasi. Multiway cut for stereo and motion with slanted surfaces. In *International Conference on Computer Vision*, 1999. 1

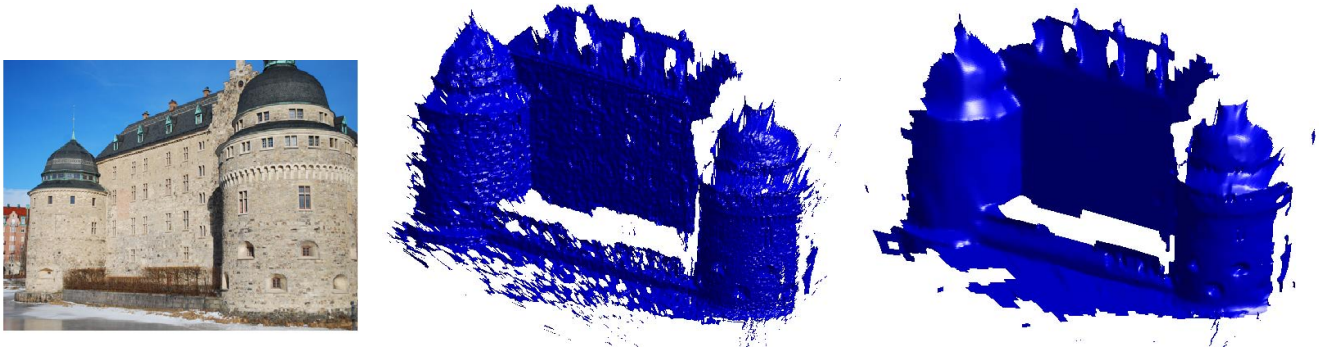
[2] M. Bleyer, C. Rother, and P. Kohli. Surface stereo with soft segmentation. In *IEEE conf on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010. 1

[3] E. Boros and P.L. Hammer. Pseudo-boolean optimization. *Discrete applied mathematics*, 123(1):155–225, 2002. 2



(a) (b) (c)

Figure 5. (a) - Image, (b) - depth map using only the data term, (c) - depth map computed with regularization.



(a) (b) (c)

Figure 6. (a) - Image, (b) - depth map using only the data term, (c) - depth map computed with regularization.

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2001. 1, 2, 4

[5] F. Devernay and O. Faugeras. Computing differential properties of 3-d shapes from stereoscopic images without 3-d models. In *IEEE conf. on Computer Vision and Pattern Recognition*, 1994. 1

[6] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient belief propagation for early vision. *Int. J. Comput. Vision*, 70(1):41–54, October 2006. 1

[7] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *European conf. on Computer Vision*, 2002. 1

[8] V. S. Lempitsky, C. Rother, S. Roth, and A. Blake. Fusion moves for markov random field optimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(8):1392–1405, 2010. 1, 2, 3

[9] G. Li and S.W. Zucker. Differential geometric inference in surface stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(1):72–86, 2010. 1, 2

[10] C. Olsson and Y. Boykov. Curvature-based regularization for surface approximation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 1

[11] C. Rother, V. Kolmogorov, V. S. Lempitsky, and M. Szummer. Optimizing binary mrfs via extended roof duality. In *IEEE conf. on Computer Vision and Pattern Recognition*, 2007. 2, 6

[12] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *IEEE conf on Computer Vision and Pattern Recognition*, volume 1, 2003. 6, 7

[13] Steven M. Seitz. The space of all stereo images. In *Int. conf. Computer Vision*, 2001. 2

[14] Y. Wei and L. Quan. Asymmetrical occlusion handling using graph cut for multi-view stereo. In *IEEE conf. on Computer Vision and Pattern Recognition, 2005. CVPR*, 2005. 5

[15] O.J. Woodford, P.H.S. Torr, I.D. Reid, and A.W. Fitzgibbon. Global stereo reconstruction under second order smoothness priors. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009. 1, 2, 6