Local Refinement for Stereo Regularization

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

Anders Eriksson
School of Computer Science
University of Adelaide
anders.eriksson@adelaide.edu.au

***Abstract*—Stereo matching is an inherently difficult problem due to ambiguous and noisy texture. The non-convexity and non-differentiability makes local linear (or quadratic) approximations poor, thereby preventing the use of standard local descent methods. Therefore recent methods are predominantly based on discretization and/or random sampling of some class of approximating surfaces (e.g. planes). While these methods are very efficient in generating a rough surface estimate, via either fusion of proposals or label propagation, the end result is usually not as smooth as desired. In this paper we show that, if the objective function is decomposed correctly, local refinement of candidate solutions can be performed using an ADMM approach. This allows searching over more general function classes, thereby resulting in visually more appealing smooth surface estimations.**

I. Introduction

In the last decade considerable progress has been made in dense stereo matching, largely due to the availability of powerful regularizers for handling ambiguous and noisy data. The most common are the first order regularization priors [1], [2], [3]. One reason for their popularity is that robust move-making algorithms such as α-expansion [1] or fusion moves [4] capable of modifying large numbers of pixels simultaneously can be applied. Such moves are essential for avoiding poor local solutions.

First order 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 with respect to the camera [1]. To model surfaces more accurately Birchfeld and Tomasi [5] introduced 3D-labels corresponding to arbitrary 3D planes. However, this approach is limited to piecewise planar scenes. To address more general scenes recent papers use 2nd derivative regularization [6], [7], [8]. There are two ways of modeling such higher order smoothness potentials. Woodford et al. [8] 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, [6], [7] use 3D-labels corresponding to tangent planes to encode 2nd order smoothness as pairwise interactions. It is shown in [6] that in contrast to the triple-cliques used by Woodford et al. [8] the 3D-label formulation is often submodular (or near submodular) making fusion moves easier to solve optimally using standard methods like Roof duality [9].

While these methods are very efficient in generating a rough surface estimate the end result is usually not as smooth as desired. Since proposals are often planar or piecewise planar [8] approximations of the surface, a very large number of tangents have to be generated in order to achieve a smooth result. In this paper we focus on local methods for minimizing the 2nd order stereo energy. Such approaches can be used to generate proposals (in the form of locally optimal surfaces) which can then be fused with the current solution.

The energies considered have three features that make local optimization difficult; First, the parametrization of the pairwise interaction is non-linear resulting in non-convex least squares terms. Second, the interaction is truncated to preserve discontinuities in the scene. This makes local optimization difficult since the Taylor approximation can be an arbitrarily bad approximation of the function (even locally). And third, the data term is based on photo consistency of local patches, making it highly non-convex with lots of local minima, see Figure 1. In this paper we show that with the right parametrization all of these difficulties can be addressed within an ADMM framework [10]. We show that the use of local optimization in the proposal generation process results in smoother and more visually appealing surface estimates.

II. Stereo and Fusion Moves

A. Energy

The goal of stereo reconstruction is to compute a depth estimate for every pixel in an image. Doing so requires that every pixel in the image is matched to a corresponding pixel in another image. Due to ambiguous texture this matching is rarely unique and as a result the stereo problem is most often ill posed. Resolving these ambiguities requires adding knowledge of the types of surfaces that we can expect to see in natural scenes, in the form of regularization.

The problem is therefore typically formulated as an energy minimization problem of the form

$$\min_z E(z) \tag{1}$$

where

$$E(z) = \sum_{\boldsymbol{p}} \sum_{\boldsymbol{q}\in\mathcal{N}(\boldsymbol{p})} E_{\boldsymbol{pq}}(z) + \sum_{\boldsymbol{p}} E_{\boldsymbol{p}}(z). \tag{2}$$

The function z represents the sought depth map and $\mathcal{N}(\boldsymbol{p})$ is a predefined neighborhood to $\boldsymbol{p}$. Throughout the paper we will think of our assignments as samples of the underlying depth function. The value $z(\boldsymbol{p})$ represents the assignment at pixel $\boldsymbol{p}$. The term $E_{\boldsymbol{p}}$ is a data term that only depends on the assignment at the particular pixel $\boldsymbol{p}$. Typically this term

is based on some measure of photo consistency. For a given assignment $z(\boldsymbol{p})$ its value can be computed by backprojecting into another image and comparing pixel appearance. In our work we will use normalized cross correlation between a 3×3 patch with center at $\boldsymbol{p}$ in the original image and a patch with center at the backprojection point.

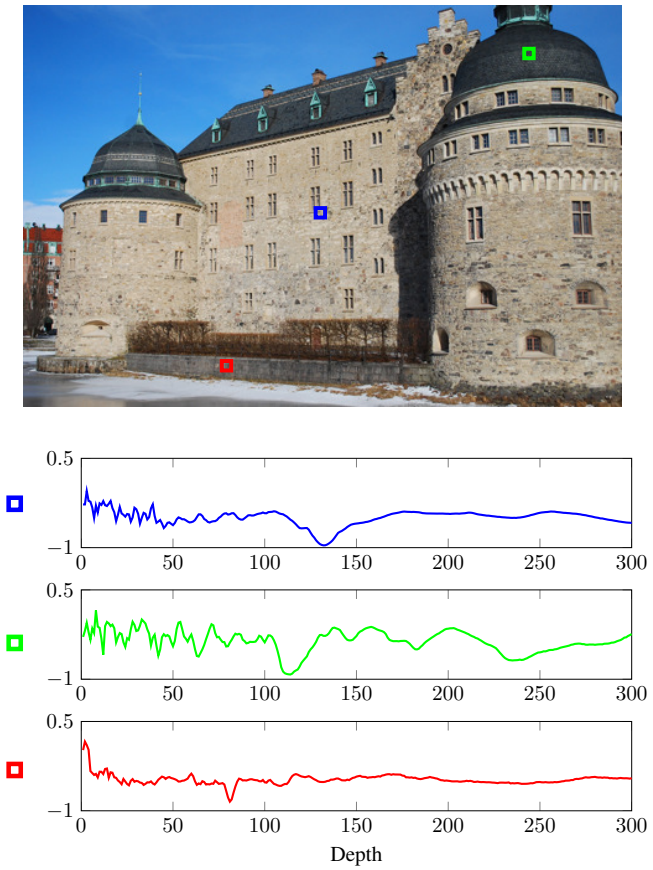


Fig. 1: The dataterm $E_{\boldsymbol{p}}$ for three pixels in the Örebro Castle dataset.

Photo consistency based measurements such as this one generally results in noisy and ambiguous functions. Figure 1 shows $E_{\boldsymbol{p}}$ for three typical pixels in the Örebro Castle dataset. To handle this the smoothness term $E_{\boldsymbol{pq}}$ is added to the energy. (By the subscript $\boldsymbol{pq}$ we mean that it only depends on the assignments at $\boldsymbol{p}$ and $\boldsymbol{q}$.) This term is meant to penalize unrealistic assignments such as surfaces with high depth variation. The simplest possible interaction is the so called Potts model, which adds a penalty if neighboring depth assignments are different [1]. In this work we follow [6] which uses a slightly more complicated interaction that penalizes second order smoothness, see Figure 2. In addition to assigning a depth $z(\boldsymbol{p})$ we also assign a depth gradient $\nabla z(\boldsymbol{p})$ to pixel $\boldsymbol{p}$, which allows us to determine the tangent of z at $\boldsymbol{p}$. To enforce smoothness we extend the tangent at $\boldsymbol{p}$ to $\boldsymbol{q}$ and measure the difference $V_{\boldsymbol{pq}}$ to the depth assignment at $\boldsymbol{q}$. Intuitively, if the function is smooth then the tangent should be a good approximation, and therefore $V_{\boldsymbol{pq}}$ should be small. Note in particular that planar depth functions do not incur any penalty.

B. Fusion Moves

Optimizing energies, such as (1), is difficult since $E_{\boldsymbol{p}}$ is often non-convex and even non-differentiable. An approach that has been effective are the so called fusion moves [4]. Given two assignments z^0 and z^1 we fuse them into a new one with lower energy by solving

$$\min_{x \in \{0,1\}^n} E\big(x \cdot z^0 + (\mathbf{1} - x) \cdot z^1\big), \tag{3}$$

where $\cdot$ is element-wise multiplication. These moves are very effective since they allow changes of a large number of pixels at the same time. In addition they are not local and can therefore escape bad local minimas. In the case of pairwise energies, such as (1), the fusion move can be solved using standard methods [9], [11].

The candidate solutions are typically selected by randomly sampling functions from some low dimensional class, for example [8], [6] use planes and piecewise planes. The motivation for using such a subclass of functions is that sampling can be made very efficient and the resulting solution typically approximates the optimal solution well. However to achieve truly smooth estimations we need to go beyond a fixed class of functions and generate more general surfaces. In this work we show that for the stereo energies that we consider this can be done efficiently using local optimization. Our strategy will be to mix the usage of random sampling of planes with local refinement to generate proposals which are then fused with the current solution.

III. Parametrization of the Interaction

In order to do local refinement we need to find a parametrization of the distance $V_{\boldsymbol{pq}}$. In [6] the tangent plane at $\boldsymbol{p}$ is parametrized using a unit normal $\boldsymbol{n_p}$ and a scalar $d_{\boldsymbol{p}}$ via the affine plane equation $\boldsymbol{n_p}^T x - d_{\boldsymbol{p}} = 0$. Such a parametrization does however introduce a non-linear constraint ($\|\boldsymbol{n_p}\|^2 = 1$). In addition this parametrization introduces a singularity. If the tangent plane is selected such that it contains the entire viewing ray, then no unique depth can be determined for that pixel.

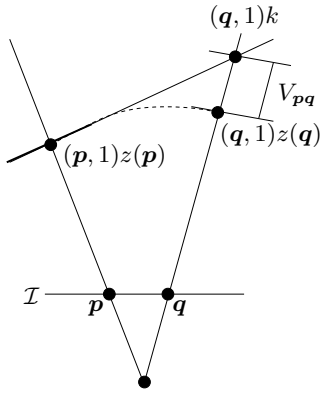


Fig. 2: The regularization interaction $V_{\boldsymbol{pq}}$ measures the deviation from the neighboring tangent along the viewing ray. Note that interactions are modeled in depth space, working in disparity space would require a modified model.

In the following, we instead derive an expression for the distance $V_{\boldsymbol{pq}}$ in terms of the assigned depths z and gradients

∇z at the points $\boldsymbol{p}$ and $\boldsymbol{q}$. The point $(\boldsymbol{q},1)k$ is the intersection between the tangent plane and the viewing ray at $\boldsymbol{q}$. The line between $(\boldsymbol{p},1)z(\boldsymbol{p})$ and $(\boldsymbol{q},1)k$ is contained in the tangent plane and can therefore be found by linearizing the curve $(\boldsymbol{p}+t\boldsymbol{v},1)z(\boldsymbol{p}+t\boldsymbol{v})$, where $\boldsymbol{v}$ is a unit vector such that $\boldsymbol{q}=\boldsymbol{p}+s\boldsymbol{v}$ for some s. We get

$$\boldsymbol{l}(t) = \begin{pmatrix} \boldsymbol{p} \\ 1 \end{pmatrix} z(\boldsymbol{p}) + t\left(\begin{pmatrix} \boldsymbol{p} \\ 1 \end{pmatrix} z'_{\boldsymbol{v}}(\boldsymbol{p}) + \begin{pmatrix} \boldsymbol{v} \\ 0 \end{pmatrix} z(\boldsymbol{p}) \right). \tag{4}$$

At the intersection point $(\boldsymbol{q},1)k$, we have

$$\boldsymbol{l}(t) = k\begin{pmatrix} \boldsymbol{q} \\ 1 \end{pmatrix} = k\left(\begin{pmatrix} \boldsymbol{p} \\ 1 \end{pmatrix} + s\begin{pmatrix} \boldsymbol{v} \\ 0 \end{pmatrix} \right). \tag{5}$$

Identification of the coefficients yields

$$k = z(\boldsymbol{p}) + z'_{\boldsymbol{v}}(\boldsymbol{p})t, \tag{6}$$

$$s = \frac{z(\boldsymbol{p})t}{z(\boldsymbol{p}) + z'_{\boldsymbol{v}}(\boldsymbol{p})t} \Leftrightarrow t = \frac{sz(\boldsymbol{p})}{z(\boldsymbol{p}) - sz'_{\boldsymbol{v}}(\boldsymbol{p})}. \tag{7}$$

Therefore we get the residuals

$$k - z(\boldsymbol{q}) = \frac{z(\boldsymbol{p})^2}{z(\boldsymbol{p}) - sz'_{\boldsymbol{v}}(\boldsymbol{p})} - z(\boldsymbol{q}). \tag{8}$$

IV. Energy and Optimization

Next we formulate the optimization problem. We denote the regularization terms

$$V_{\boldsymbol{pq}}(z) = \frac{z(\boldsymbol{p})^2}{z(\boldsymbol{p}) - sz'_{\boldsymbol{v}}(\boldsymbol{p})} - z(\boldsymbol{q}). \tag{9}$$

The energy consists of a regularization term and a data term

$$\sum_{\boldsymbol{p}} \sum_{\boldsymbol{q}\in\mathcal{N}(\boldsymbol{p})} h\left(V_{\boldsymbol{pq}}(z)\right) + \lambda \sum_{\boldsymbol{p}} D_{\boldsymbol{p}}(z). \tag{10}$$

Here $D_{\boldsymbol{p}}$ is a data term that evaluates the cost of the assigned depth for pixel $\boldsymbol{p}$. We will assume that this function is densely sampled such that its global minima can be found by simply searching all the sample points. The one dimensional function h is typically of the form $h(x) = \min(|x|^p, \tau)$ where τ is some threshold level and $p \in \{1,2\}$. The only assumption we make is that it is piecewise differentiable so that the minimum can be found by computing stationary points.

Optimizing energy (10) is typically very challenging since the data term is often non-differentiable with lots of local minima. In addition, the smoothness term is a sum of non-convex functions. To handle these problems we decouple the terms by introducing two new sets of variables; $x_{\boldsymbol{pq}}$ and $y_{\boldsymbol{p}}$.

We constrain these variables to be $x_{\boldsymbol{pq}} = V_{\boldsymbol{pq}}(z)$ and $y_{\boldsymbol{p}} = z(\boldsymbol{p})$. The Augmented Lagrangian [10] is now

$$\begin{aligned} L(x,y,z,\lambda) &= \sum_{\boldsymbol{p}} \sum_{\boldsymbol{q}\in\mathcal{N}(\boldsymbol{p})} h(x_{\boldsymbol{pq}}) \\ &+ \sum_{\boldsymbol{p}} \sum_{\boldsymbol{q}\in\mathcal{N}(\boldsymbol{p})} \lambda_{\boldsymbol{pq}}(x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z)) \\ &+ \sigma \sum_{\boldsymbol{p}} \sum_{\boldsymbol{q}\in\mathcal{N}(\boldsymbol{p})} (x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z))^2 \\ &+ \sum_{\boldsymbol{p}} \left(\lambda_{\boldsymbol{p}}(y_{\boldsymbol{p}} - z(\boldsymbol{p})) + \sigma(y_{\boldsymbol{p}} - z(\boldsymbol{p}))^2\right) \\ &+ \sum_{\boldsymbol{p}} D_{\boldsymbol{p}}(y_{\boldsymbol{p}}). \end{aligned} \tag{11}$$

When applying ADMM we get the subproblems

$$\min_{x_{\boldsymbol{pq}}} h(x_{\boldsymbol{pq}}) + \lambda_{\boldsymbol{pq}}(x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z)) + \sigma(x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z))^2 \tag{12}$$

$$\begin{aligned} \min_z \sum_{\boldsymbol{p}} \Bigg(& \sum_{\boldsymbol{q}\in\mathcal{N}(\boldsymbol{p})} \lambda_{\boldsymbol{pq}}(x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z)) + \sigma(x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z))^2 \\ & + \left(\lambda_{\boldsymbol{p}}(y_{\boldsymbol{p}} - z(\boldsymbol{p})) + \sigma(y_{\boldsymbol{p}} - z(\boldsymbol{p}))^2\right) \Bigg), \end{aligned} \tag{13}$$

$$\min_{y_{\boldsymbol{p}}} \lambda_{\boldsymbol{p}}(y_{\boldsymbol{p}} - z(\boldsymbol{p})) + \sigma(y_{\boldsymbol{p}} - z(\boldsymbol{p}))^2 + D_{\boldsymbol{p}}(y_{\boldsymbol{p}}) \tag{14}$$

In addition we obtain the dual update rules (see [10])

$$\lambda_{\boldsymbol{pq}}^{k+1} = \lambda_{\boldsymbol{pq}}^{k} + \sigma(x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z)), \tag{15}$$

$$\lambda_{\boldsymbol{p}}^{k+1} = \lambda_{\boldsymbol{p}}^{k} + \sigma(y_{\boldsymbol{p}} - z(\boldsymbol{q})). \tag{16}$$

This decoupling of terms has the following positive effects: The terms $h(x_{\boldsymbol{pq}})$ and $D_{\boldsymbol{p}}(y_{\boldsymbol{p}})$, that are non-smooth and difficult to approximate locally, end up in two different sub-problems both of which are separable and where optimization can be carried out over individual pixels separately, greatly reducing the search space. The coupling between pixels appears in problem (13) where the involved functions are smooth and can be optimized locally using standard descent methods. In the following subsections we outline our solution strategies for the individual problems.

A. Problem (12)

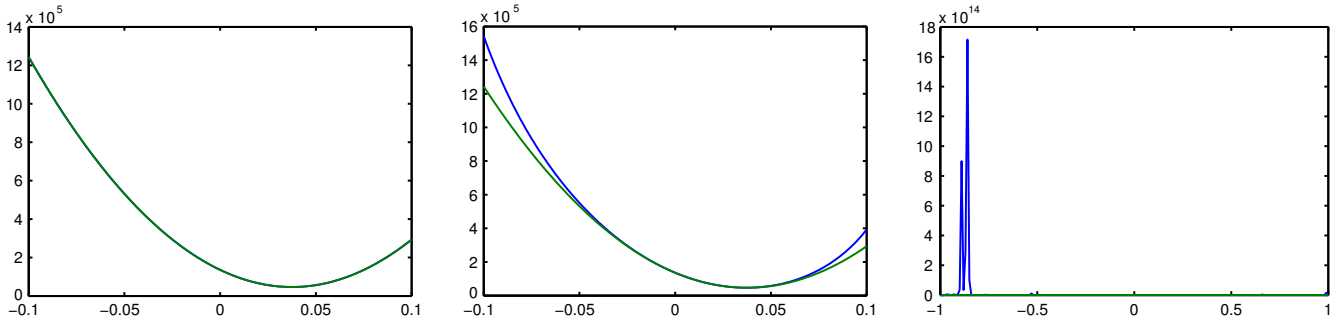
To solve (12) we first note that the optimum must be in either a stationary point or in a transition between differentiable segments of the function h. Since we will be using $h(x_{\boldsymbol{pq}}) = \min(|x_{\boldsymbol{pq}}|, \tau)$ in the experiments we illustrate the process using this choice. We get four cases:

1) $|x_{\boldsymbol{pq}}| > \tau$. Taking derivatives of (12) gives
$$\lambda_{\boldsymbol{pq}} + 2\sigma(x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z)) = 0. \tag{17}$$
Solving for $x_{\boldsymbol{pq}}$ gives the stationary point. Note that the solution may violate $|x_{\boldsymbol{pq}}| > \tau$. In this case the solution is false and there is no stationary point in the interval. However since we compare the energies of all "candidate" minimizers we do not have to test for this. We are guaranteed that one of the candidates is the global minimizer of (12).
2) $-\tau < x_{\boldsymbol{pq}} < 0$. In this case the stationary point is given by
$$-1 + \lambda_{\boldsymbol{pq}} + 2\sigma(x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z)) = 0. \tag{18}$$
3) $0 < x_{\boldsymbol{pq}} < \tau$. Here the stationary point is given by
$$1 + \lambda_{\boldsymbol{pq}} + 2\sigma(x_{\boldsymbol{pq}} - V_{\boldsymbol{pq}}(z)) = 0. \tag{19}$$
4) In addition we need to test the two transition points $x_{\boldsymbol{pq}} = \pm\tau$ and $x_{\boldsymbol{pq}} = 0$.

B. Problem (13)

The objective function in (13) is similar to non-linear least squares problem. We will apply a Levenberg-Marquart approach to solve it. We linearize the residual

$$x_{\boldsymbol{pq}} - \left(\frac{z(\boldsymbol{p})^2}{z(\boldsymbol{p}) - sz'_{\boldsymbol{v}}(\boldsymbol{p})} - z(\boldsymbol{q})\right). \tag{20}$$



(a) Our parametrization. (b) Using unit normals. (c) Using unit normals at a larger scale.

Fig. 3: The effects of parameterizing the problem using the parametrization with unit normals form [6] instead of our parametrization. Green curve shows our approximation and the blue curve the exact error function. For our parametrization the approximation cannot be distinguished from the exact function. Using unit normals at a larger scale gives visible singularities.

Note that $y_{\boldsymbol{p}} - z(\boldsymbol{q})$ is already linear in terms of the unknowns ($z(\boldsymbol{p})$,$z(\boldsymbol{q})$ and $z'_{\boldsymbol{v}}(\boldsymbol{p})$), and does therefore not require modification. The approximation that we make using linearization is in fact in most cases very accurate. This can be heuristically explained by looking at the equivalent expression

$$x_{\boldsymbol{pq}} - \left(z(\boldsymbol{p}) + s z'_{\boldsymbol{v}}(\boldsymbol{p}) + \frac{(s z'_{\boldsymbol{v}}(\boldsymbol{p}))^2}{z(\boldsymbol{p}) - s z'_{\boldsymbol{v}}(\boldsymbol{p})} - z(\boldsymbol{q}) \right). \quad (21)$$

Assuming that s is small (recall that this is the distance between pixels) the nonlinear term is likely to be neglectable for reasonable values of the derivative $z'_{\boldsymbol{v}}(\boldsymbol{p})$.

Figure 3 shows an example plot of the exact error function and its least squares approximation, that is, residuals are linearized into $A\delta - b$ and the non-linear least squares objective is approximated with $||A\delta - b||^2$. Here δ represents the increment in the assignments $z, \nabla z$. We plot the functions in the direction $-2A^T b$ which is the gradient of the approximation. (The data for this figure was taken from the first iteration of local refinement of Örebro Castle, see Figure 5.) For comparison we also plot the approximation obtained when parameterizing with unit normals as in [6]. Note that the two parameterizations are the same in some (possibly very small) local neighborhood. Therefore their linear approximations will be the same. However, it can be seen from Figure 3 that the unit normal parametrization deviates faster from its approximation. In addition it has singularities.

C. *Problem* (14)

Since the function $D_{\boldsymbol{p}}$ is one dimensional and sampled densely it is easy to optimize it by simply searching the sample values. To solve (14) we simply recompute the samples of $D_{\boldsymbol{p}}$ by adding

$$\lambda_{\boldsymbol{p}}(y_{\boldsymbol{p}} - z(\boldsymbol{p})) + \sigma(y_{\boldsymbol{p}} - z(\boldsymbol{p}))^2 \quad (22)$$

to $D_{\boldsymbol{p}}(y_{\boldsymbol{p}})$ and chose the optimum as the best new sample.

V. Implementation and Results

In this section we test our proposed approach. We compare two approaches; Fusion moves with sampled planar proposals without local refinement vs. with local refinement. In all of the experiments we use normalized cross correlation (with a minus sign to get a minimization) with patch size 3×3 as data term. As regularization term we use the truncated L_1 term $h(V_{\boldsymbol{pq}}) = \min(|V_{\boldsymbol{pq}}|, \tau)$. For the specific choices of λ and τ see Figure 5. For local refinement we use the ADMM approach. Since the problem is not convex convergence is not guaranteed for fixed σ. Therefore we start with σ at a low value (0.1 in our implementation) and slowly increase it each iteration to a high value (10 seems to be enough for convergence). The specific update rule is

$$\sigma^{k+1} = \eta \sigma^k, \quad (23)$$

where η is determined by ensuring σ^k is 10 in the last iteration.

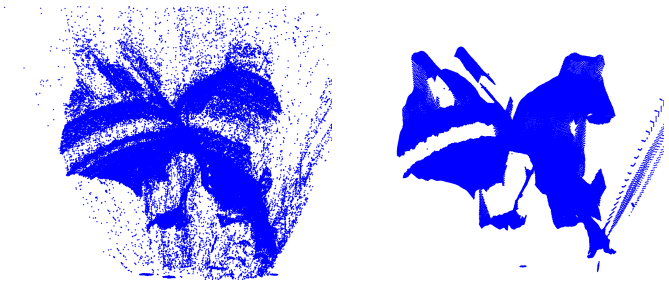


Fig. 4: Comparison of 3D-point positions in unregularized and regularized solutions for the Eglise dataset. Left: unregularized, Right: Regularized. The surface reconstruction (viewed from a different viewpoint) of the regularized solution can also be seen in Figure 5.

In Figure 4 we show the difference between an unregularized solution and one obtained with regularization. We only show the 3D-points since the unregularized solution does not have any normal estimation. (A surface reconstruction of the regularized solution is displayed in the results in Figure 5).

In Figure 5 we show the results of applying the two competing approaches to 4 datasets of varying difficulty; From smooth highly textured surfaces (Église du Dôme) to non-smooth untextured surfaces (Nijo Castle). Note that, in addition to ambiguous texture, the Nijo Castle data set contains people that have walked around between images making the data term incorrect at these positions. We do not handle this in any special way other than applying more regularization. In all of these cases, the fusion moves provide solutions that approximate the underlying surfaces well. However, the planar nature of the proposals gives the appearance of piecewise planarity. In contrast, with local refinement the resulting surfaces have a much smoother appearance and at the same time capture fine details better. In addition the local refinement also repairs some defects, most likely caused by insufficient sampling, such as the hole visible on the roof of the Nijo castle gate.

VI. Acknowledgments

This work has been funded by the Swedish Research Council (grants no. 2012-4213 and 2012-4215), the Crafoord Foundation and the Australian Research Councils Discovery Early Career Researcher Award project (DE130101775).

References

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

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

[3] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient belief propagation for early vision," *Int. J. Comput. Vision*, vol. 70, no. 1, pp. 41–54, Oct. 2006.

[4] V. S. Lempitsky, C. Rother, S. Roth, and A. Blake, "Fusion moves for markov random field optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 8, pp. 1392–1405, 2010.

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

[6] C. Olsson, J. Ulén, and Y. Boykov, "In defense of 3d-label stereo," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[7] G. Li and S. Zucker, "Differential geometric inference in surface stereo," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 1, pp. 72–86, 2010.

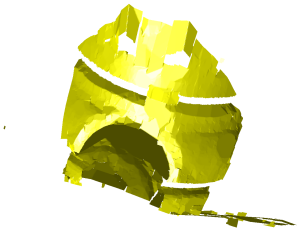
[8] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon, "Global stereo reconstruction under second order smoothness priors," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

[9] 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.

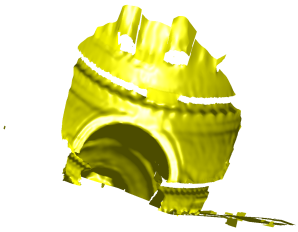
[10] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[11] V. Kolmogorov, "Convergent tree-reweighted message passing for energy minimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1568–1583, 2006.

Image



No local refinement (E: 14028).

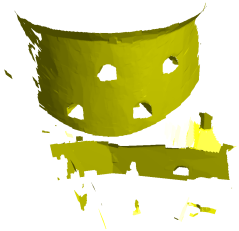


With local refinement (E: 13411).

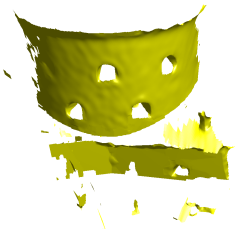
Local refinement and texture.

(a) Église du Dôme, Paris. ($\lambda = 0.5$, $\tau = 0.18$)

Image



No local refinement (E: 7661).

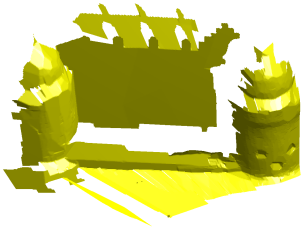


With local refinement (E: 7424).

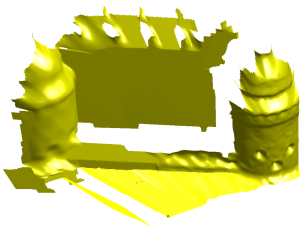
Local refinement and texture.

(b) Skansen Lejonet, Gothenburg. ($\lambda = 0.5$, $\tau = 0.2$)

Image



No local refinement (E: 25823).

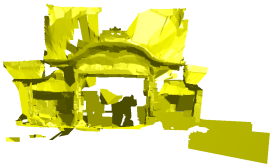


With local refinement (E: 23559).

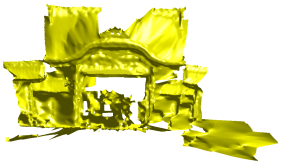
Local refinement and texture.

(c) Örebro Castle, Örebro. ($\lambda = 0.5$, $\tau = 0.16$)

Image



No local refinement (E: 5471).



With local refinement (E: 5254).

Local refinement and texture.

(d) Nijo Castle, Kyoto. ($\lambda = 0.5$, $\tau = 0.67$)

Fig. 5: Reconstructed stereo surfaces, the energy of each solution is given inside parenthesis. After local refinement the energies are lower and the surfaces are more detailed.