PM-Huber: PatchMatch with Huber Regularization for Stereo Matching

Philipp Heise, Sebastian Klose, Brian Jensen, Alois Knoll
Department of Informatics, Technische Universität München, Germany
{heise,kloses,jensen,knoll}@in.tum.de

Abstract

Most stereo correspondence algorithms match support windows at integer-valued disparities and assume a constant disparity value within the support window. The recently proposed PatchMatch stereo algorithm [7] overcomes this limitation of previous algorithms by directly estimating planes. This work presents a method that integrates the PatchMatch stereo algorithm into a variational smoothing formulation using quadratic relaxation. The resulting algorithm allows the explicit regularization of the disparity and normal gradients using the estimated plane parameters. Evaluation of our method in the Middlebury benchmark shows that our method outperforms the traditional integer-valued disparity strategy as well as the original algorithm and its variants in sub-pixel accurate disparity estimation.

Figure 1: Stereo pair taken from [13] and a point cloud created by using the sub-pixel disparity map generated by our algorithm.

1. Introduction

Most stereo matching algorithms are based on the assumption that the pixels within the matching window share the same disparity value. Further very often only discrete disparity values are considered leading to discrete depth layers. One reason for the widespread use of this simplified model is that the number of likelihood evaluations for more precisely sampled disparities and the inclusion of discretized surface orientations quickly becomes intractable. On the other side sub-pixel accurate depth values are necessary to create plausible and precise meshes or point clouds.

Bleyer et al.[7] showed that the PatchMatch algorithm [4, 5] can be applied for stereo matching using slanted support windows so that instead of just estimating a single disparity value for each pixel a complete disparity plane estimation is made. The PatchMatch algorithm does not try to discretize the space of the likelihood function, but rather relies on randomized sampling and propagation of good estimates. This also results in an implicit smoothing model, when good estimates are propagated in the direct neighbourhood. But the implicit smoothing can also lead to problems when wrong or unreliable estimates are propagated. In the stereo case this problem can occur in homogeneous untextured regions, regions with repeating structures and extreme sampling choices e.g. normals nearly orthogonal to the view direction.

To alleviate these problems an explicit smoothing model based on the combination of PatchMatch and Particle Belief Propagation resulting in the PMBP Algorithm [6] has been recently proposed, leading to improved results compared to the original algorithm. We present an algorithm based on an explicit variational energy formulation combining the PatchMatch stereo algorithm with regularization of the disparity and normal gradients resulting in sub-pixel accurate disparity maps improving the state of the art. Our disparity maps are well suited for the creation of point clouds without discretization or staircasing artifacts as shown in figure 1.

1.1. Contribution

In this paper we show that the projections of scene points belonging to the same planar surface in rectified stereo pairs are fully related by a linear transformation with three degrees of freedom. This has already been shown in [7] for planes in the disparity space and is in the following extended to the real scene space of fully calibrated and rec-

tified stereo cameras.

Our main contribution is an explicit variational smoothness model for the PatchMatch algorithm using quadratic relaxation [12, 17]. In [17, 14] only the first order derivatives of the optical flow vectors and disparity-values have been considered, but the proposed algorithm allows us to control the smoothness of the first-order and second-order derivatives of the disparities. The second-order derivatives of the disparities are implicitly determined by the gradient of the normals estimated by the PatchMatch algorithm. Instead of performing an exhaustive search as in [17, 14] for the evaluation of the data term we employ the PatchMatch algorithm. Evaluation of the proposed method for stereo pairs of the Middlebury benchmark [15] shows its effectiveness in estimating sub-pixel accurate disparity maps. At the time of writing we are currently ranked at position 1 out of about 145 algorithms for the sub-pixel error threshold 0.5.

2. Method

2.1. Slanted support windows

In [7] the authors showed how planes in the disparity space affect the patch neighbourhood. For completeness we repeat their result. Given an image point $\mathbf{p} = [\,x_0\ y_0\ 1\,]^\top$ with the disparity value z_0 and a normal $\mathbf{n} = [\,n_x\ n_y\ n_z\,]^\top$ we can calculate the d parameter of a plane $\pi = \left[\,\mathbf{n}^\top\ d\,\right]^\top$ with $d = -n_x x_0 - n_y y_0 - n_z z_0$. This follows from $\pi^\top [\,x_0\ y_0\ z_0\ 1\,]^\top = 0$, which must hold if the point lies on the plane π. Therefore the disparity value z of any image point $[\,x\ y\,]^\top$ on the plane is given by

$$z = \frac{-n_x x - n_y y + (n_x x_0 + n_y y_0 + n_z z_0)}{n_z}. \tag{1}$$

We can reformulate this as a linear transformation assuming that the point in the second image is given by $\mathbf{p}' = \mathbf{p} - [\,z\ 0\ 0\,]^\top$ with z being the disparity as

$$\mathbf{p}' = \begin{pmatrix} 1 + \frac{n_x}{n_z} & \frac{n_y}{n_z} & -\frac{n_x}{n_z} x_0 - \frac{n_y}{n_z} y_0 - z_0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathbf{p}. \tag{2}$$

For the general case we make use of prior knowledge about the camera projection matrices $P = K[I\,|\,0]$ and $P' = K'[R\,|\,\mathbf{t}]$ with the origin set at the first camera. Then the plane-induced homography from the first to the second camera [10](p. 327) is given by

$$H_\pi = K' \left(R - \frac{\mathbf{t}\,\mathbf{n}^\top}{d} \right) K^{-1} \tag{3}$$

for a plane $\pi = \left[\,\mathbf{n}^\top\ d\,\right]^\top$ with normal $\mathbf{n}$ and distance d to the origin. For a rectified stereo camera setup the rotation is the identity I and the translation between the cameras is given by $[\,b\ 0\ 0\,]^\top$ with b being the baseline between the cameras. Assuming identical intrinsics $K = K'$ due to the rectification process and K being an upper triangular matrix the resulting homography is

$$H_\pi = K \left(I - \frac{1}{d} [\,b\ 0\ 0\,]^\top\ \mathbf{n}^\top \right) K^{-1} \tag{4}$$

$$= I - K \frac{1}{d} \begin{pmatrix} b\,n_x & b\,n_y & b\,n_z \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} K^{-1}. \tag{5}$$

From equation (2) and (5) it follows that in the case of disparity and scene planes the transformation between two rectified images induced by a plane has only three degrees of freedom with a being the scaling, b the shearing and c the translation resulting in the matrix with the following structure

$$\begin{pmatrix} 1+a & b & c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{6}$$

The effects on the support window is shown in figure 2.

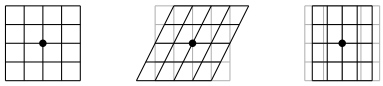

Figure 2: Illustration of the shearing and scaling transformation induced by disparity and scene planes.

To map from the second image to the first image the inverse of the matrix from equation (2) or (5) can be used.

2.2. Model

Given two rectified stereo color images $I_1, I_2 : (\Omega \subset \mathbb{R}^2) \to \mathbb{R}^3$, a disparity map $d : \Omega \to \mathbb{R}$ and a normal map $\mathbf{n} : \Omega \to \{\mathbf{x} \in \mathbb{R}^2 : |\mathbf{x}| \leq 1\}$ our algorithm is based on minimizing an energy of the form

$$E(d, \mathbf{n}) = \lambda E_{\text{data}}(d, \mathbf{n}) + E_{\text{smooth}}(d, \mathbf{n}), \tag{7}$$

consisting of a data term describing the similarity between pointwise matches in the stereo pair and a smoothness term favoring similar disparity and normal values of adjacent pixels. In the following $\mathbf{n}$ refers to the non over-parametrized representation of the normal containing only two components. If needed the normal $\hat{\mathbf{n}}$ with three components can directly be calculated from $\mathbf{n}$ since we only consider the normals from one half of the unit sphere[1]. Our data term is similar to the one used in [7]

$$E_{\text{data}}(d, \mathbf{n}) = \int_\Omega \frac{1}{Z} \sum_{\mathbf{q} \in \mathcal{N}(\mathbf{p})} w(\mathbf{p}, \mathbf{q})\, \rho(\mathbf{q}, T(d, \mathbf{n})\,\mathbf{q})\, d\mathbf{p}. \tag{8}$$

[1] $\hat{\mathbf{n}} = [\,n_x\ n_y\ \sqrt{1 - n_x^2 - n_y^2}\,]^\top$

T is one of the linear transformations parametrized by d and $\mathbf{n}$ given in equation (2) or (5) and ρ measures the pixel similarity between the patches:

$$\begin{aligned}\rho(\mathbf{p},\mathbf{q}) =&(1-\alpha)\min(\|I_1(\mathbf{p})-I_2(\mathbf{q})\|_1,\tau_{\text{col}})\\ &+\alpha\min(\|\nabla I_1(\mathbf{p})-\nabla I_2(\mathbf{q})\|_1,\tau_{\text{grad}}).\end{aligned}\tag{9}$$

$I_1(\mathbf{p})$ and $I_2(\mathbf{q})$ in the previous equation are the linearly interpolated pixel color-values in the respective stereo images and ∇I is an four channel image containing the image derivatives calculated by the horizontal and vertical Sobel operator and diagonal gradients calculated using central differences. The derivatives are calculated from grayscale versions of the stereo images. The function w in equation (8) computes a weighting mask based on the color similarity between the center pixel $\mathbf{p}$ and the other pixels $\mathbf{q}$ inside the patch

$$w(\mathbf{p},\mathbf{q}) = e^{-\gamma(\mathbf{p},\mathbf{q})||I_1(\mathbf{p})-I_1(\mathbf{q})||_1}.\tag{10}$$

In our formulation of w the γ value changes with distance to the center

$$\gamma(\mathbf{p},\mathbf{q}) = \gamma_{\min}+\gamma_{\text{radius}}\ \text{smoothstep}(0,r_{\max},|\mathbf{q}-\mathbf{p}|).\tag{11}$$

The reasoning behind the varying γ is that pixels close to the center belong more likely to the same plane and that pixels far away have to be very similar in terms of their color-distance to get the same consideration. This formulation is different to a decreasing weighting factor with increasing distance. Z is an normalization constant with

$$Z=\sum_{\mathbf{q}\in\mathcal{N}(\mathbf{p})} w(\mathbf{p},\mathbf{q}).\tag{12}$$

Our regularization term E_{smooth} imposes spatial smoothness on the disparity-values d and the normals $\mathbf{n}$ resulting in

$$E_{\text{smooth}}(d,\mathbf{n}) = \int_\Omega g(\mathbf{p})\,|\nabla d|_{\epsilon_d} + g(\mathbf{p})\,|\nabla\mathbf{n}|_{\epsilon_n}\,d\mathbf{p},\tag{13}$$

with $|\,.\,|_\epsilon$ being the robust Huber norm

$$|\mathbf{x}|_\epsilon = \begin{cases}\frac{|\mathbf{x}|^2}{2\epsilon} & \text{if } |\mathbf{x}|\le\epsilon\,,\\ |\mathbf{x}|-\frac{\epsilon}{2} & \text{else}\end{cases}.\tag{14}$$

As depth and normal discontinuities often occur at strong image gradients we introduce the per-pixel weighting function $g(\mathbf{p})$ with

$$g(\mathbf{p}) = e^{-\zeta|\nabla I_1(\mathbf{p})|^\eta}.\tag{15}$$

2.3. Solution

Following [3, 12, 17, 14] we use quadratic relaxation to decouple our data and regularization term. Introducing an auxiliary vector field allows us to perform two alternating minimizations approximating the original minimization problem. This results in the following auxiliary energy formulation

$$\begin{aligned}E_{\text{aux}}(d_u,\mathbf{n}_u,d_v,\mathbf{n}_v) = \int_\Omega &\lambda\,E_{\text{data}}(d_u,\mathbf{n}_u)\\ &+\frac{\theta}{2}(\Pi_v-\Pi_u)^\top\,\Sigma\,(\Pi_v-\Pi_u)\\ &+E_{\text{smooth}}(d_v,\mathbf{n}_v)\,d\mathbf{p},\end{aligned}\tag{16}$$

with

$$\Pi_w = [\,\mathbf{n}_w\ d_w\,]^\top\tag{17}$$

and $\Sigma=\text{diag}(\sigma_n,\sigma_n,\sigma_d)$ being a diagonal matrix weighting the squared distances of the normals and the disparity values. Forcing θ to infinity drives the variables Π_u and Π_v together and results in $\lim_{\theta\to\infty} E_{\text{aux}}\approx E$. We split the optimization of the E_{aux} into two sub-problems, namely one optimization problem involving Π_u with fixed Π_v and another one with Π_v and fixed Π_u. We collect all fixed terms independent of argument minimizing variable in a constant c.

Fixed Π_u, solve for Π_v

For optimization of the energy E_{aux} we make use of a primal-dual formulation of the Huber-ROF model as described by Chambolle et al. [8]. The Legendre-Fenchel transformation of the weighted Huber norm $g\,|\mathbf{x}|_\epsilon$ using $a\,h(x)\Rightarrow a\,h^*(\frac{p}{a})$ $(a>0)$ is given by

$$\begin{aligned}(g\,|\mathbf{x}|_\epsilon)^*(p) =& g\sup_{\mathbf{x}}\left\{\frac{1}{g}\mathbf{x}^\top\mathbf{p}-|\mathbf{x}|_\epsilon\right\}\\ =&\frac{\epsilon}{2\,g}\mathbf{p}^\top\mathbf{p}+\delta\left(\frac{1}{g}\mathbf{p}\right),\end{aligned}\tag{18}$$

where δ is the indicator function. With the previous result the minimization problem of E_{aux} with respect to d_v can be written as

$$\arg\min_{d_v} E_{\text{aux}} = \arg\min_{d_v}\sup_{\mathbf{p}_d} E(d_v,\mathbf{p}_d)\tag{19}$$

$$\begin{aligned}=\arg\min_{d_v}\sup_{\mathbf{p}_d}\Bigg\{&\int_\Omega g(\mathbf{p})\,\langle\nabla d_v,\mathbf{p}_d\rangle\\ &-\frac{\epsilon_d}{2\,g(\mathbf{p})}\mathbf{p}_d^\top\mathbf{p}_d-\delta\left(\frac{1}{g(\mathbf{p})}\mathbf{p}_d\right)\\ &+\frac{\theta\sigma_d}{2}\,|d_v-d_u|^2\,d\mathbf{p}+c\Bigg\}.\end{aligned}\tag{20}$$

Figure 3: *From left to right:* One image of the portal stereo pair from [2], our disparity map after initialisation, disparity map after the 1st iteration, the final disparity map and two images with different views of a point-cloud generated using the final disparity map.

We take the derivative of $E(d_v, \mathbf{p}_d)$ with respect to d_v and $\mathbf{p}$ and using the divergence theorem we get

$$\frac{\partial E(d_v, \mathbf{p}_d)}{\partial d_v} = g(\mathbf{p}) \operatorname{div} \mathbf{p}_d + \theta \sigma_d (d_v - d_u) \tag{21}$$

$$\frac{\partial E(d_v, \mathbf{p}_d)}{\partial \mathbf{p}_d} = g(\mathbf{p}) \nabla d_v - \frac{\epsilon_d}{g(\mathbf{p})} \mathbf{p}_d. \tag{22}$$

The formulation of the E_{aux} minimization with respect to $\mathbf{n}_v$ is analogous and leads to the following derivatives

$$\frac{\partial E(\mathbf{n}_v, \mathbf{p}_n)}{\partial \mathbf{n}_v} = g(\mathbf{p}) \operatorname{div} \mathbf{p}_\mathbf{n} + \theta \sigma_n (\mathbf{n}_v - \mathbf{n}_u) \tag{23}$$

$$\frac{\partial E(\mathbf{n}_v, \mathbf{p}_n)}{\partial \mathbf{p}_n} = g(\mathbf{p}) \nabla \mathbf{n}_v - \frac{\epsilon_n}{g(\mathbf{p})} \mathbf{p}_n \tag{24}$$

with $\mathbf{p}_n$ being the dual variable. To solve the energy minimization with respect to Π_v we use gradient descent and ascent as in [14]

$$\frac{\mathbf{p}_d^{t+1} - p_d^t}{\beta_d} = g(\mathbf{p}) \nabla d_v^t - \frac{\epsilon_d}{g(\mathbf{p})} \mathbf{p}_d^{t+1} \tag{25}$$

$$\frac{d_v^{t+1} - d_v^t}{\nu_d} = -\, g(\mathbf{p}) \operatorname{div} \mathbf{p}_d^{t+1} - \theta \sigma_d (d_v^{t+1} - d_u) \tag{26}$$

$$\frac{\mathbf{p}_n^{t+1} - p_n^t}{\beta_n} = g(\mathbf{p}) \nabla \mathbf{n}_v^t - \frac{\epsilon_n}{g(\mathbf{p})} \mathbf{p}_n^{t+1} \tag{27}$$

$$\frac{\mathbf{n}_v^{t+1} - \mathbf{n}_v^t}{\nu_d} = -\, g(\mathbf{p}) \operatorname{div} \mathbf{p}_n^t - \theta \sigma_n (\mathbf{n}_v^{t+1} - \mathbf{n}_u) \tag{28}$$

and perform several inner iterations using the following update rules

$$p_d^{t+1} = \operatorname{proj}\left(\frac{p_d^t + \beta_d\, g(\mathbf{p})\, \nabla d_v^t}{1 + \beta_d \epsilon_d\, g(\mathbf{p})^{-1}} \right) \tag{29}$$

$$d_v^{t+1} = \frac{d_v^t + \nu_d (\theta \sigma_d d_u - g(\mathbf{p}) \operatorname{div} \mathbf{p}_d^{t+1})}{1 + \nu_d \theta \sigma_d} \tag{30}$$

$$p_n^{t+1} = \operatorname{proj}\left(\frac{p_n^t + \beta_n\, g(\mathbf{p})\, \nabla \mathbf{n}_v^t}{1 + \beta_n \epsilon_n\, g(\mathbf{p})^{-1}} \right) \tag{31}$$

$$\mathbf{n}_v^{t+1} = \frac{\mathbf{n}_v^t + \nu_n (\theta \sigma_n \mathbf{n}_u - g(\mathbf{p}) \operatorname{div} \mathbf{p}_n^{t+1})}{1 + \nu_n \theta \sigma_n} \tag{32}$$

where proj projects back onto the unit sphere

$$\operatorname{proj}(\mathbf{x}) = \frac{\mathbf{x}}{\max(1, |\mathbf{x}|)}. \tag{33}$$

The projection fulfills the constraint of the dual variable $|p| \leq 1$. The super-script denotes here the iteration number. For the step sizes β_d, ν_d, β_n and ν_n we use the values of ALG3 reported by Chambolle et al. [8]. Handa et al. [9] also give a good introduction and further details to the Legendre-Fenchel transform and its applications.

Fixed Π_v, solve for Π_u

Instead of performing an exhaustive search as done in [17, 14] we employ a variant of the PatchMatch stereo algorithm. Given a set of samples $\mathcal{S}(\mathbf{p})$ for each point $\mathbf{p}$, the best sample

$$s^\star = \underset{\Pi_u \in \mathcal{S}(\mathbf{p})}{\arg\min}\, \lambda\, E_{\text{data}}(\Pi_u) + \frac{\theta}{2} (\Pi_u - \Pi_v)^\top \Sigma\, (\Pi_u - \Pi_v) \tag{34}$$

is stored at $\Pi_u^{t+1}(\mathbf{p})$ after each iteration. We do not follow the sequential pixel processing scheme from [7], but use a completely parallel approach. Our set $\mathcal{S}(\mathbf{p})$ is defined as

$$\begin{aligned} \mathcal{S}(\mathbf{p}) = &\mathcal{S}_\mathcal{N}(\mathbf{p}) \cup \{\Pi_v(\mathbf{p})\} \cup \mathcal{S}_{\text{rnd}\,\mathcal{N}}(\mathbf{p}) \cup \mathcal{S}_{\text{rnd}}(\mathbf{p}) \\ &\cup \mathcal{S}_{\text{view}}(\mathbf{p}) \cup \mathcal{S}_{\text{rnd}\,\star}(\mathbf{p}). \end{aligned} \tag{35}$$

$\mathcal{S}_\mathcal{N}(\mathbf{p})$ contains the 3×3 patch of samples centered around $\mathbf{p}$ from the previous iteration. The set $\mathcal{S}_{\text{rnd}\,\mathcal{N}}(\mathbf{p})$ contains only one particle from Π_u^t randomly chosen from the 7×7 neighbourhood around $\mathbf{p}$. $\mathcal{S}_{\text{rnd}}(\mathbf{p})$ is one completely randomly chosen sample. The set $\mathcal{S}_{\text{view}}(\mathbf{p})$ contains the view propagated particles. Each position $\mathbf{p}$ has storage for a few view particles and particles from the other view are propagated if storage is still available. $\mathcal{S}_{\text{rnd}\,\star}(\mathbf{p})$ is an slightly randomly perturbed particle based on the best particle from $\mathcal{S}(\mathbf{p}) \backslash \mathcal{S}_{\text{rnd}\,\star}(\mathbf{p})$.

In figure 3 different stages of our algorithm are shown for a stereo pair and the corresponding final disparity map together with a generated point cloud. The randomized sampling after the initialisation is clearly visible in the image, but already after one iteration the first samples have been successfully propagated in the neighbourhood.

2.4. Implementation Details

We perform the depth and normal map estimation in both images of the stereo pair. This allows us to perform the view propagation of samples and also left-right consistency checking. The left-right consistency checking plays an important role in our algorithm, because it allows the removal of inconsistent results. Especially in the occluded areas arbitrary particles with inconsistent disparity and normal values are very often persistent. Therefore after each PatchMatch iteration - before we apply the Huber-ROF smoothing - we fill the occluded areas with the next non-occluded plane-particle from the same scanline with the more distant depth value at the occluded position as illustrated in figure 4. This is similar to the post-processing proposed in [7] but without the weighted median filtering step. Our occlusion checking not only uses the depth values but also the plane normals and allows only disparity differences up to 0.5 and normal deviations of $5°$. For lookup of the plane parameters in the second image we do not use linear interpolation but nearest neighbour sampling. The occlusion-filling is also done for the final result and is the only post-processing step we perform. For the initialisation we found it beneficial

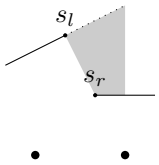

Figure 4: The occluded gray area in the first view is filled using the plane parameters from position s_l. Resulting in disparity values as indicated by the dotted line. s_r although also visible in both views is not chosen, because its plane would result in closer disparity values.

to draw normal samples more restrictively and the normals of the first PatchMatch iteration are within the 0.5 radius $|[\, n_x \; n_y \,]^\top| \leq 0.5$. To allow propagation and refinement of the particles in the first iterations of the algorithm we perform a few iterations with $\theta = 0$. We control the values of θ during the iterations using the $\mathrm{smoothstep}$ function. Each iteration consists of one PatchMatch iteration followed by several inner iterations for smoothing using the weighted Huber-ROF model.

2.5. Runtime

Our algorithm has been designed to be executed on massively parallel architectures. Our PatchMatch sampling strategy is completely parallel in contrast to the original PatchMatch stereo algorithm. Also the Huber-ROF subproblem can be solved very efficiently on parallel architectures. The runtime of our algorithm highly varies with the parameter settings and number of iterations. For the high-quality settings as used for the Middlebury benchmark evaluation our algorithm has a runtime of about 2 minutes. For the PatchMatch stereo algorithm the authors reported a runtime of about 1 minute for an average Middlebury pair [7]. Different settings for our algorithm allow the estimation of disparity maps in a few seconds. Our current GPU implementation is completely unoptimized and several obvious performance enhancements have not been exploited yet.

2.6. Method Parameters

In the following we assume that the values of the stereo image channels are in the range $[0, 1]$. The size of the patch considered in the data term is 41×41 pixels centered around the pixel $\mathbf{p}$. For setting the $\alpha, \tau_{\mathrm{col}}$ and τ_{grad} parameters we mainly follow [7] and set them to $\{\alpha, \tau_{\mathrm{col}}, \tau_{\mathrm{grad}}\} = \{0.05, 0.04, 0.01\}$. The new parameters $\gamma_{\min}, \gamma_{\mathrm{radius}}$ are set to 5 and 39 and $r_{\max}$ to $\left\lfloor \sqrt{2 \cdot 20^2} \right\rfloor$. The parameters of the weighting function g are set to $\{\zeta, \eta\} = \{3, 0.8\}$. ϵ_n and ϵ_d of the robust Huber norm were both set to 0.001. The value of $\theta \cdot \sigma_n$ starts at 0 and goes up to 50 with an additional offset of 5 for the weighted Huber-ROF smoothing of the normals. For the intermediate disparity maps we use a range from 0 to 1, therefore $\theta \cdot \sigma_d$ takes values between 0 and $\frac{50}{d_{\max}}$ again with an special offset of $\frac{5}{d_{\max}}$. $d_{\max}$ is the maximum allowed disparity value. For the computation of the data term we set $\lambda = 50$.

3. Evaluation

For the evaluation of our algorithm we use the Middlebury stereo benchmark [15, 1]. Our results for the Middlebury stereo benchmark were made using constant parameters as described in the previous section. The maximum allowed disparity was fixed to 60 and used for all four pairs. This shows that our algorithm does not necessarily need to know the disparity range in advance. Our Middlebury benchmark results for the error threshold 0.5 are shown in table 1. At the time of writing we are currently ranked at position 1 out of about 145 algorithms for the sub-pixel error threshold 0.5. We achieve results comparable or better than the original PatchMatch stereo implementation [7] and the PMBP method [6] that also has an explicit smoothing model. The final disparity maps and also the error maps for the 0.5 error threshold are shown in figure 5. For the error-threshold 1 our algorithm has rank 25. As mentioned

	Avg.	Tsukuba			Venus			Teddy			Cones		
	Rank	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
1. Our method	5.3	7.12 $_{9}$	7.80 $_{8}$	13.7 $_{7}$	1.00 $_{8}$	1.40 $_{9}$	7.80$_{13}$	5.53$_{2}$	9.36$_{2}$	15.9$_{3}$	**2.70**$_{1}$	**7.90**$_{1}$	**7.77**$_{1}$
2. SubPixSearch	5.8	5.60 $_{2}$	6.23 $_{2}$	9.46 $_{3}$	1.07$_{10}$	1.64$_{10}$	7.36 $_{9}$	6.71$_{5}$	11.0$_{4}$	16.9$_{5}$	4.02$_{7}$	9.76$_{5}$	10.3$_{7}$
3. PMF	8.8	11.0$_{30}$	11.4$_{27}$	16.0$_{25}$	0.72 $_{4}$	0.92 $_{3}$	5.27 $_{4}$	**4.45**$_{1}$	9.44$_{3}$	**13.7**$_{1}$	2.89$_{2}$	8.31$_{3}$	8.22$_{2}$
⋮			⋮			⋮			⋮			⋮	
5. PMBP	12.9	11.9$_{39}$	12.3$_{35}$	17.8$_{42}$	0.85 $_{6}$	1.10 $_{4}$	6.45 $_{7}$	5.60$_{3}$	12.0$_{6}$	15.5$_{2}$	3.48$_{3}$	8.88$_{4}$	9.41$_{4}$
⋮			⋮			⋮			⋮			⋮	
10. PatchMatch	20.1	15.0$_{57}$	15.4$_{56}$	20.3$_{69}$	1.00 $_{9}$	1.34 $_{8}$	7.75$_{12}$	5.66$_{4}$	11.8$_{5}$	16.5$_{4}$	3.80$_{5}$	10.2$_{6}$	10.2$_{6}$

Table 1: First three entries from the Middlebury stereo benchmark [15] and additionally the results from PMBP [6] and the original PatchMatch-Stereo [7] algorithm. Our algorithm is currently ranked at position 1 out of about 145 algorithms for the error-threshold 0.5. Subscripts denote rankings in the table.

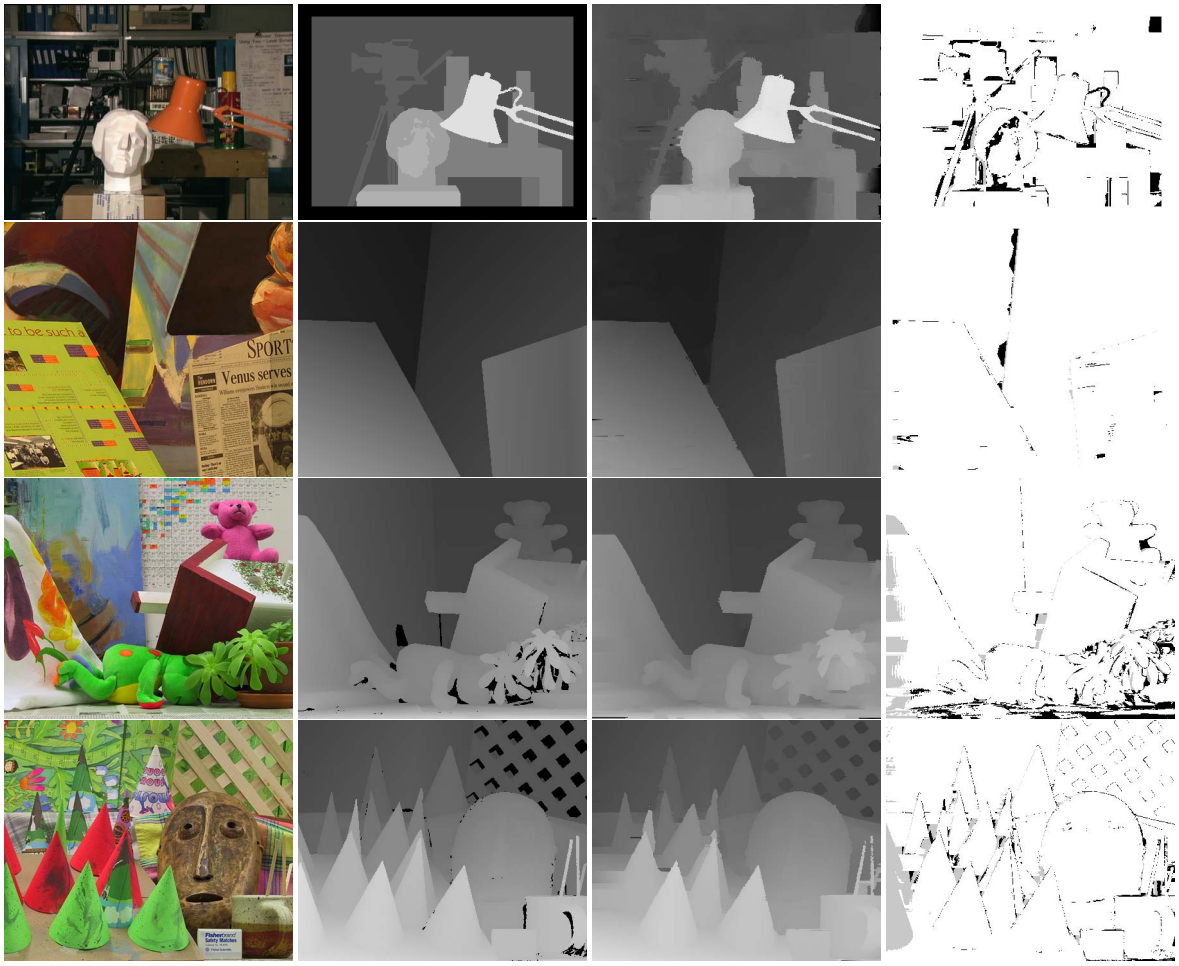

Figure 5: *From left to right*: one of the input images, ground-truth disparity map, our result and the disparity errors > 0.5. *From top to bottom*: Middlebury stereo pairs [15] Tsukuba, Venus, Teddy and Cones.

before, we did not perform the weighted median filtering step of the original algorithm and we assume that this leads to the slightly worse results for the 1 threshold. To emphasize the sub-pixel accuracy of our algorithm we also created point clouds of some Middlebury datasets that contain planar and curved surfaces as depicted in figure 6. The head

and the ground-plane of the Art scene are well reproduced by the point cloud. Also the curved surface of the platform in the Baby1 scene is very smooth and does not exhibit staircasing or discrete depth layer effects. For the Phong shaded point clouds the normals estimated by our algorithm have been used instead of estimating them using neighbouring vertices. Videos of the point clouds can be found the supplementary material.

In order to show that our algorithm also works for more realistic data we tested it using two rectified and downscaled images from Strecha et al. [18]. The resulting point cloud is shown in figure 7. Another point cloud created from our disparity maps is shown in figure 1.

Figure 6: Colored and Phong shaded point clouds of the Middlebury datasets Art, Baby1, Cones and Cloth3 [16, 11].

Figure 7: A point cloud created from a rectified stereo pair. Images provided by Strecha et al. [18].

4. Conclusion

We presented an new approach to combine the randomized sampling of the PatchMatch algorithm with an explicit variational smoothing method that gives control of the disparity and normal gradients. Our evaluation shows that we achieve very good sub-pixel results in the Middlebury benchmark that make our algorithm well suited for the generation of point clouds or meshes. In the future we would like to extend our algorithm to multi-view, which probably can be done using equation (3). The estimated normals are also maybe useful for depthmap merging and multi-view reconstruction. Additionally we would like to optimize our current GPU OpenCL implementation towards real-time frame-rates. Also a modified version for the estimation of optical flow is already planned.

References

[1] Middlebury stereo benchmark. http://vision.middlebury.edu/stereo/. 5

[2] Portal stereo scene. http://cmp.felk.cvut.cz/~cechj/GCS/stereo-images/. 4

[3] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition—modeling, algorithms, and parameter selection. *Int. J. Comput. Vision*, 67(1):111–136, 2006. 3

[4] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman. PatchMatch: a randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (TOG)*, 28(3):24, 2009. 1

[5] C. Barnes, E. Shechtman, D. Goldman, and A. Finkelstein. The generalized patchmatch correspondence al-

gorithm. *Computer Vision–ECCV 2010*, pages 29–43, 2010. 1

[6] F. Besse, C. Rother, A. Fitzgibbon, and J. Kautz. PMBP: PatchMatch Belief Propagation for Correspondence Field Estimation. In *Proceedings of the British Machine Vision Conference*, pages 132.1–132.11. BMVA Press, 2012. 1, 5, 6

[7] M. Bleyer, C. Rhemann, and C. Rother. PatchMatch Stereo - Stereo Matching with Slanted Support Windows. *Proc. BMVC*, pages 1–11, July 2011. 1, 2, 4, 5, 6

[8] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011. 3, 4

[9] A. Handa, R. A. Newcombe, A. Angeli, and A. J. Davison. Applications of legendre-fenchel transformation to computer vision problems. Technical Report DTR11-7, Imperial College - Department of Computing, September 2011. 4

[10] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 2

[11] H. Hirschmuller and D. Scharstein. Evaluation of Cost Functions for Stereo Matching. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8, 2007. 7

[12] Y. Huang, M. K. Ng, and Y.-W. Wen. A Fast Total Variation Minimization Method for Image Restoration. *Multiscale Modeling & Simulation*, 7(2):774–795, Jan. 2008. 2, 3

[13] P. Monasse. Quasi-Euclidean Epipolar Rectification. *Image Processing On Line*, 2011, 2011. 1

[14] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. DTAM: Dense Tracking and Mapping in Real-Time. *ICCV '11: Proceedings of the 2011 International Conference on Computer Vision*, pages 1–8, Aug. 2011. 2, 3, 4

[15] D. Scharstein and R. Szeliski. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. *Int. J. Comput. Vision*, 47(1-3):7–42, 2002. 2, 5, 6

[16] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, 2003. 7

[17] F. Steinbrücker, T. Pock, and D. Cremers. Large displacement optical flow computation without warping. *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1609–1614, 2009. 2, 3, 4

[18] C. Strecha, R. Fransens, and L. Van Gool. Combined depth and outlier estimation in multi-view stereo. *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, 2:2394–2401, 2006. 7