Real-Time Dense Geometry from a Handheld Camera

Jan Stühmer[1,2], Stefan Gumhold[2], and Daniel Cremers[1]

[1] Department of Computer Science, TU München
[2] Department of Computer Science, TU Dresden

Abstract. We present a novel variational approach to estimate dense depth maps from multiple images in real-time. By using robust penalizers for both data term and regularizer, our method preserves discontinuities in the depth map. We demonstrate that the integration of multiple images substantially increases the robustness of estimated depth maps to noise in the input images. The integration of our method into recently published algorithms for camera tracking allows dense geometry reconstruction in real-time using a single handheld camera. We demonstrate the performance of our algorithm with real-world data.

1 Introduction

Reconstructing the geometry of the environment from a hand-held camera is among the classical topics in computer vision. While sparse reconstructions of a finite number of tracked points can easily be done in real-time [1,2], the fast computation of dense reconstructions from a moving camera remains an open challenge.

Traditionally there are two complementary approaches to estimating dense geometry, namely the reconstruction of depth maps (often called 2.5d reconstructions) from stereo image pairs and the reconstruction of full $3D$ structure from multiple images. While we have observed substantial advances in dense $3D$ reconstruction from multiple images, many of these approaches are to date not real-time capable [3,4]. Moreover, they typically require a larger number of around 30 calibrated images making them unsuited for live scene reconstructions from a single moving camera. On the other hand, there exist many approaches to reconstructing dense depth maps from pairs of images [5,6]. While these approaches were shown to provide excellent results on dense depth estimation, they are typically too computationally intense for real-time applications, moreover, they are rather noise sensitive since they only exploit two images.

In this paper, we propose a variational approach for computing dense depth maps from multiple images with real-time performance. The key idea is to adopt recently developed high-accuracy optic flow algorithms [7] to the problem of depth map estimation from multiple images. Depth maps are computed by sequential convex optimization by means of a primal-dual algorithm. In particular, we prove that the primal variables can be efficiently computed using a sophisticated thresholding scheme. To obtain optimal performance, the dense depth

M. Goesele et al. (Eds.): DAGM 2010, LNCS 6376, pp. 11–20, 2010.

maps are computed in coarse-to-fine-manner on the GPU while the camera coordinates are simultaneously computed on the CPU using recently developed algorithms. Our experiments demonstrate that the algorithm allows to compute dense high-quality depth maps from a moving camera in real-time. Moreover, our quantitative evaluation confirms that using multiple images substantially improves the noise-robustness of estimated depth maps.

After submission of this manuscript we became aware that the problem of reconstructing depth maps from a handheld camera was independently addressed in the recent work of Newcombe and Davisson [8]. In the latter work, the authors first estimate an optical flow field from consecutive images and subsequently use this flow field to update a depth map. In contrast, we propose a variational approach which directly provides a depth field. This seems more appropriate to us: Why estimate a 2D motion vector for each pixel, if - apart from the camera motion - the considered scene is static? One consequence of the proposed solution to directly determine the depth field is that our algorithm is real-time capable on a single graphics card whereas the approach of Newcombe and Davison needs several seconds per frame on two GPUs.

2 Robust Estimation of Depth Maps from Images

In Section 2.1 we introduce our mathematical framework for computing dense depth maps for the simpler case of two input images. In Section 2.2 we extend this formulation and introduce a novel variational approach for estimating depth maps from multiple images. In Section 2.3 we propose a primal-dual algorithm which substantially generalizes the one of Zach et al and which allows to efficiently minimize the proposed functional.

First we give an introduction to our notation. Let us assume a given set of gray value images $\{I_i : \Omega_i \to \mathbb{R}\}$ with $i \in \{0, \dots, N\}$ that were taken from different viewpoints with the same camera. Let us further assume, that the corresponding camera poses (location and orientation of the camera) and the projection $\pi : \mathbb{R}^3 \to \mathbb{R}^2$ that projects from homogeneous coordinates to pixel coordinates are known. The depth map h, that should be estimated, is a scalar field which is defined with respect to the coordinate frame of one of the images. Let us denote this camera image without loss of generality as I_0 such that $h : \Omega_0 \to \mathbb{R}$ assigns a depth value to every pixel of I_0. By using homogeneous 2D coordinates $\mathbf{x} = (x_1, x_2, 1)^T \in \Omega_0$ we can express the position of each $3D$ surface point $\mathbf{X}$ of the depth map by multiplying the homogeneous $2D$ vector by the depth value: $\mathbf{X}(\mathbf{x}, h) := h(x_1, x_2) \cdot \mathbf{x}$.

Note that the above position vector is relative to the coordinate frame of I_0. The projection of such a $3D$ point $\mathbf{X}$ onto another image plane Ω_i can be achieved by $\pi(\exp(\hat{\xi}_i) \cdot \mathbf{X})$, where ξ_i is the camera pose for each image relative to the coordinate frame of I_0. The camera poses are given in so called *twist coordinates* $\xi \in \mathbb{R}^6$. The *hat-operator* transforms ξ_i such that the *twist* $\hat{\xi}_i \in se(3)$ gives the exponential coordinates of the rigid-body motion that transforms the coordinate frame of I_0 into the coordinate frame of I_i.

2.1 Stereo Estimation Using Two Images

Let us introduce our mathematical framework for the simplest case, when two images are provided. To estimate a heightmap h from these two images we propose the following variational formulation consisting of an L_1 data penalty term and an L_1 total variation (TV) regularization of the depth map

$$E(h) = \lambda \int_{\Omega_0} \Big| I_1\big(\pi\big(\exp(\hat{\xi_1})\,\mathbf{X}(\mathbf{x},h)\big)\big) - I_0\big(\pi(\mathbf{x})\big)\Big| \; d^2\mathbf{x} + \int_{\Omega_0} |\nabla h| \; d^2\mathbf{x}, \qquad (1)$$

where the data term $I_1\big(\pi\big(\exp(\hat{\xi_1})\,\mathbf{X}(\mathbf{x},h)\big)\big) - I_0\big(\pi(\mathbf{x})\big)$ measures the difference of the image intensities of I_0 and the image intensities that are observed at the projected coordinates in I_1. Above data term is motivated by the *Lambertian* assumption, that the observed intensity is independent of the viewpoint as long as the same surface point is observed in both views. The TV-norm regularizer allows to preserve discontinuities in the depth map, e.g. at object boundaries, while the robust data term lowers the sensitivity towards outliers in cases where objects are invisible by occlusion or when the input images are affected with noise. In the following we will use the simplified notation $I_1(\mathbf{x},h)$ for $I_1\big(\pi\big(\exp(\hat{\xi_1})\,\mathbf{X}(\mathbf{x},h)\big)\big)$.

We begin with a linearization of $I_1(\mathbf{x},h)$ by using the first order Taylor expansion, i.e.

$$I_1(\mathbf{x},h) = I_1(\mathbf{x},h_0) + (h - h_0)\,\frac{d}{dh} I_1(\mathbf{x},h)\Big|_{h_0} \qquad (2)$$

where h_0 is a given depth map. The derivative $\frac{d}{dh} I_1(\mathbf{x},h)$ can be considered as a directional derivative in direction of a differential vector on the image plane that results from a variation of h It can be expressed as the scalar product of the gradient of $I_1(\mathbf{x},h)$ with this differential vector, i.e.

$$\frac{d}{dh} I_1(\mathbf{x},h) = \nabla I_1(\mathbf{x},h) \cdot \frac{d}{dh} \pi\big(\exp(\hat{\xi})\,\mathbf{X}(\mathbf{x},h)\big). \qquad (3)$$

The differential vector mentioned above needs to be calculated with respect to the chosen camera model.

Using the linear approximation for $I_1(\mathbf{x},h)$ and by reordering the integrals the energy functional (Eq. 1) now reads

$$E(h) = \int_{\Omega_0} \Big\{ \lambda \underbrace{\Big| I_1(\mathbf{x},h_0) + (h - h_0)\,\frac{d}{dh} I_1(\mathbf{x},h)\Big|_{h_0} - I_0(\mathbf{x})\Big|}_{\rho_1(\mathbf{x},h_0,h)} + \big|\nabla h\big| \Big\} \, d^2\mathbf{x}. \qquad (4)$$

Though this energy functional is much simpler than the original functional (Eq. 1), the task of minimizing it is still difficult, because both the regularization term and the data term are not continuously differentiable.

We introduce an auxiliary function u that decouples the data term and the regularizer, leading to the following convex approximation of Eq. 4:

$$E_\theta = \int_{\Omega} \Big\{ |\nabla u| + \frac{1}{2\theta}(u - h)^2 + \lambda\,|\rho_1(h)| \Big\} \; d^2\mathbf{x}, \qquad (5)$$

where θ is a small constant and $\rho_1(h)$ denotes the current residual of the data term (by omitting the dependency on h_0 and $\mathbf{x}$). It is immediate to see that for $\theta \to 0$ the minimization of the above functional results in both h and u being a close approximation of each other.

This minimization problem can be solved efficiently in real-time by minimizing the data term with a simple thresholding scheme and using a primal dual algorithm for the minimization of the ROF energy [9].

2.2 Extension to Multiple Images

Let us now consider the case when multiple input images are given. In the previous section we formulate our energy model for the classical stereo task in case of two images. Compared to previous approaches that employ the epipolar constraint by using the fundamental matrix the main difference is that here we formulate the data term relative to the coordinate system of one specific view and use the perspective projection to map this coordinate system to the second camera frame. This makes it easy to incorporate the information from other views by simply adding up their data terms. We propose the following energy functional to robustly estimate a depth map from multiple images

$$E(h) = \lambda \int_{\Omega} \sum_{i \in \mathcal{I}(\mathbf{x})} |\rho_i(\mathbf{x}, h)| \; d^2\mathbf{x} + \int_{\Omega} |\nabla h| \; d^2\mathbf{x} \tag{6}$$

where $\mathcal{I}(\mathbf{x})$ contains the indices of all images for which the perspective projection $\pi(\exp(\hat{\xi}_i) \cdot \mathbf{X}(\mathbf{x}, h))$ is inside the image boundaries. With $\rho_i(\mathbf{x}, h)$ we denote the residual of the linearized data term for image I_i

$$\rho_i(\mathbf{x}, h) = I_i(\mathbf{x}, h_0) + (h - h_0)\, I_i^h(\mathbf{x}) - I_0(\mathbf{x}), \tag{7}$$

where $I_i^h(\mathbf{x})$ is a simplified notation for the derivative $\frac{d}{dh} I_i(\mathbf{x}, h)\Big|_{h_0}$.

By using the above functional we should expect two benefits. First of all algorithms using only two images are not able to estimate disparity information in regions that are occluded in the other view or simply outside of its image borders. The use of images from several different views should help in these cases because information from images where the object is not occluded can be used. The use of the L_1-norm in the data terms allows an increased robustness towards outliers in cases where objects are occluded. The second benefit of using multiple images is the increased signal to noise ratio that provides much better results when the input images are affected by noise, which is a typical property of image sequences acquired by webcams or consumer market camcorders.

This functional is more complicate to solve because the data term consists of the sum of absolute values of linear functions, that cannot be minimized using the simple thresholding scheme proposed in [7]. In [4] the authors extend the thresholding scheme to data terms of the form $\sum_i |x - b_i|$, with a set of constants $\{b_i \in \mathbb{R}\}$. Unfortunately the data term in the proposed functional is not of such form. Nevertheless, we will show in the next section that the thresholding concept can be generalized to a substantially larger class of functionals.

2.3 Generalized Thresholding Scheme

In this section we provide a substantial generalization of the thresholding scheme which also applies to multiple images and more sophisticated data terms.

We decouple the smoothness and data term by introducing an auxiliary function u and get the following convex approximation of Eq. 6:

$$E_\theta = \int_\Omega \left\{ |\nabla u| + \frac{1}{2\theta}(u-h)^2 + \lambda \sum_{i \in \mathcal{I}(\mathbf{x})} |\rho_i(\mathbf{x}, h)| \right\} d^2\mathbf{x}, \tag{8}$$

The above functional is convex so an alternating descent scheme can be applied to find the minimizer of E_θ:

1. For h being fixed, solve

$$\min_u \int_\Omega \left\{ |\nabla u| + \frac{1}{2\theta}(u-h)^2 \right\} d^2\mathbf{x} \tag{9}$$

   This is the ROF energy for image denoising [10,9].

2. For u being fixed, solve

$$\min_h \int_\Omega \left\{ \frac{1}{2\theta}(u-h)^2 + \lambda \sum_{i \in \mathcal{I}(\mathbf{x})} |\rho_i(\mathbf{x}, h)| \right\} d^2\mathbf{x} \tag{10}$$

   This minimization problem can be solved point-wise.

A solution for the minimization of the the ROF energy, the first step in our alternating scheme, was proposed in [9], that uses a dual formulation of Eq. 9. For the convenience of the reader we reproduce the main results from [9].

Remark 1. The solution of Eq. 9 is given by

$$u = h - \theta\, \mathbf{div}\, \mathbf{p}, \tag{11}$$

where $\mathbf{p} = (p_1, p_2)$ is a vector field and fulfills $\nabla(\theta\, \mathbf{div}\, \mathbf{p} - h) = |\nabla \theta\, \mathbf{div}\, \mathbf{p} - h|\mathbf{p}$, which can be solved by the following iterative fixed-point scheme:

$$\mathbf{p}^{k+1} = \frac{\mathbf{p}^k + \tau \nabla(\, \mathbf{div}\, \mathbf{p}^k - h/\theta)}{1 + \tau |\nabla(\, \mathbf{div}\, \mathbf{p}^k - h/\theta)|}\,, \tag{12}$$

where $\mathbf{p}^0 = \mathbf{0}$ and the time step $\tau \leq 1/8$.

The second step of the alternation scheme, Eq. 10, can be solved point-wise, but shows some difficulties as it is not continuously differentiable. Nevertheless we provide a closed-form solution by generalizing the thresholding concept to data terms of the form $\sum\limits_i |a_i\, x - b_i|$.

By taking a look at Eq. 7 we see, that for fixed h_0 and $\mathbf{x}$ the residuals of the linearized data terms ρ_i can be expressed in the general form of linear functions, $\rho_i(\mathbf{x}, h) = a_i\, h + b_i$, with $a_i := I_i^h(\mathbf{x})$ and $b_i := I_i(\mathbf{x}, h_0) - h_0\, I_i^h(\mathbf{x}) - I_0(\mathbf{x})$. The absolute valued functions $|\rho_i(h)|$ are differentiable with respect to h except at their critical points, where a function equals zero and changes its sign. Let us denote those critical points as

$$t_i := -\frac{b_i}{a_i} = -\frac{I_i(\mathbf{x}, h_0) - h_0\, I_i^h(\mathbf{x}) - I_0(\mathbf{x})}{I_i^h(\mathbf{x})}\,, \tag{13}$$

where $i \in \mathcal{I}(\mathbf{x})$.

At these points Eq. 9 is not differentiable, as the corresponding ρ_i changes its sign. Without loss of generality we can assume that $t_i \leq t_{i+1}$, i.e. we obtain a sorted sequence of $\{\rho_i : i \in \mathcal{I}(\mathbf{x})\}$, that is sorted by the values of their critical points. In order to avoid special cases we add $t_0 = -\infty$ and $t_{|\mathcal{I}(\mathbf{x})|+1} = +\infty$ to this sequence.

Proposition 1. *The minimizer of Eq. 10 can be found using the following strategy: If the stationary point*

$$h_1 := u - \lambda\theta \left(\sum_{i \in \mathcal{I}(\mathbf{x}): i \leq k} I_i^h(\mathbf{x}) - \sum_{j \in \mathcal{I}(\mathbf{x}): j > k} I_j^h(\mathbf{x}) \right) \tag{14}$$

lies in the interior of (t_k, t_{k+1}) for some $k \in \mathcal{I}(\mathbf{x})$, then $h = h_1$. Else the minimizer of Eq. 10 can be found among the set of critical points:

$$h = \arg \min_{h_2 \in \{t_i\}} \left(\frac{1}{2\theta}(u-h)^2 + \lambda \sum_{i \in \mathcal{I}(\mathbf{x})} |\rho_i(\mathbf{x}, h_2)| \right) . \tag{15}$$

Proof. Eq. 10 is differentiable with respect to h in the interior of intervals (t_k, t_{k+1}). Let us assume that the stationary point

$$h_1 := u - \lambda\theta \sum_{i \in \mathcal{I}(\mathbf{x})} \Big(\operatorname{sgn}\left(\rho_i(\mathbf{x}, h_1)\right)\, I_i^h(\mathbf{x}) \Big) \tag{16}$$

exists and lies in the interior of the interval (t_k, t_{k+1}), then

$$\sum_{i \in \mathcal{I}(\mathbf{x})} \Big(\operatorname{sgn}\left(\rho_i(\mathbf{x}, h_1)\right)\, I_i^h(\mathbf{x}) \Big) = \sum_{i \in \mathcal{I}(\mathbf{x}): t_i < h_1} I_i^h(\mathbf{x}) - \sum_{j \in \mathcal{I}(\mathbf{x}): t_j > h_1} I_j^h(\mathbf{x}) \tag{17}$$

$$= \sum_{i \in \mathcal{I}(\mathbf{x}): i \leq k} I_i^h(\mathbf{x}) - \sum_{j \in \mathcal{I}(\mathbf{x}): j > k} I_j^h(\mathbf{x})\,. \tag{18}$$

This stationary point exists, iff it stays in the interior of (t_k, t_{k+1}) for some k. If none of the proposed stationary points stays in the interior of its corresponding interval, the minimizer of Eq. 10 resides on the boundary of one of the intervals, i.e. it can be found among the set of critical points $\{t_i\}$. □

3 Implementation

Because the linearization of the data term (Eq. 7) only holds for small displacements of the projected coordinates, the overall innovation of the depth map is limited. To overcome this, the energy minimization scheme is embedded into a coarse-to-fine approach: Beginning on the coarsest scale a solution h is computed. This solution is used as new point h_0 for the linearization on the next finer scale. By using this scheme we not only employ an iterative linearization, but also utilize the multi-scale approach to avoid convergence into local minima. When processing a consecutive sequence of input images, an initialization of the coarsest scale can be achieved by transforming the depth map computed in the preceding frame to the current camera pose, thus utilizing the sequential property of the input data.

We embedded our method into a recently published camera tracking approach, that allows tracking of a handheld camera in real-time [11]. An integral part of this camera tracker is the storage of *keyframes*. While the pose for the current camera image needs to be estimated in real-time, and thus contains a significant amount of noise in the pose estimation, the camera pose associated to each keyframe can be refined iteratively, leading to very accurate estimates for the keyframes. Instead of using subsequent images with noisy real-time pose estimates, our approach enables to estimate a depth map in a similar fashion to the strategy employed in the camera tracker, by estimating the depth map using the current camera image and the N closest keyframes to the current pose. By using the much better camera pose estimates of the keyframes, the amount of noise in the camera poses is minimized.

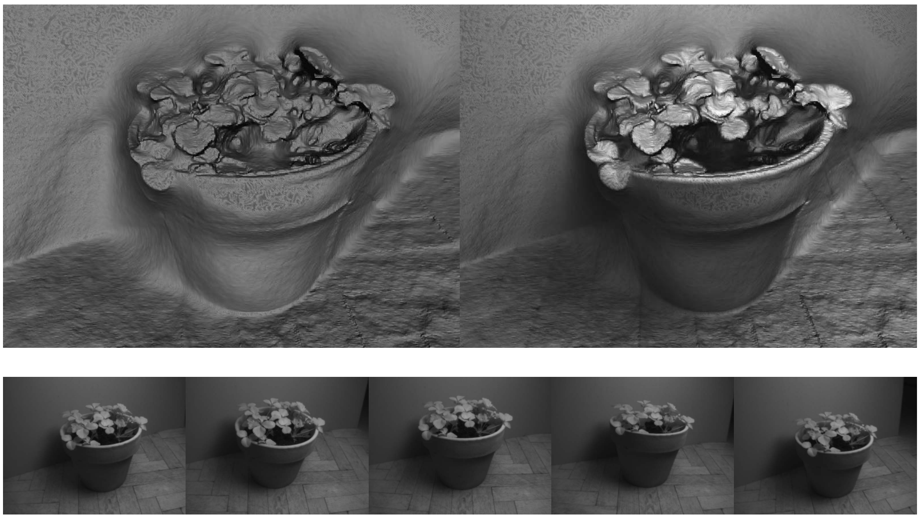


Fig. 1. Dense depth maps computed from images of a hand-held camera

4 Experimental Results

High-accuracy dense depth maps from a hand-held camera: The proposed algorithm allows to compute dense depth maps from a moving camera. Figure 1 shows the reconstruction result from 5 input images. In contrast to the commonly used structure-and-motion algorithms [1,2], the proposed method computes a dense geometry rather than the location of sparse feature points. Another example is given in Figure 2 that shows the reconstruction result of an office scene. Note the accurate reconstruction of small-scale details like the network cable.

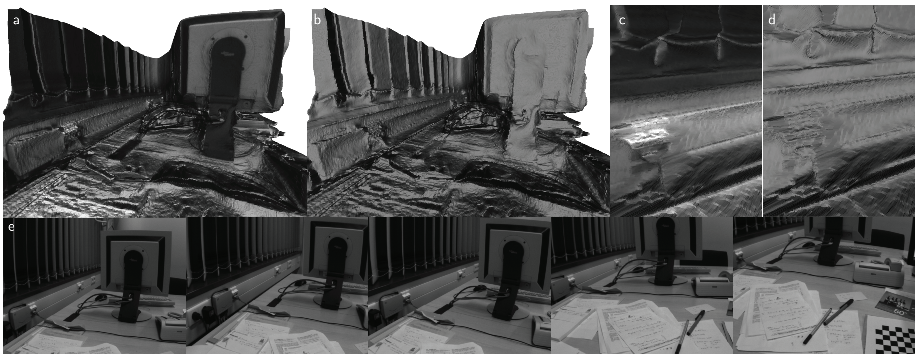
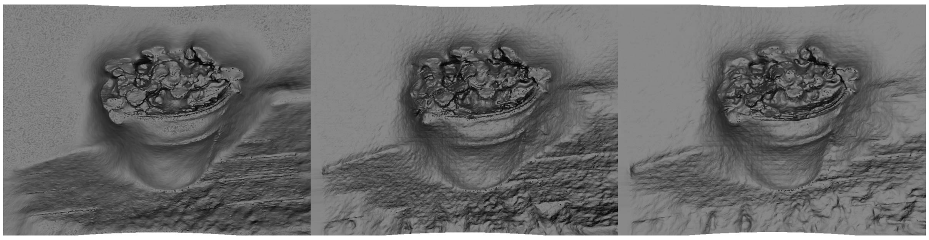


Fig. 2. Textured (a,c) and untextured geometry (b,d). Note the accurate reconstruction of small-scale details like the network socket and cords. (e) Images.

Realtime geometry reconstruction: The proposed primal-dual scheme can be efficiently parallelized on the GPU. The joint estimation of camera motion on the CPU allows for live dense reconstructions of the scene. Clearly there is a tradeoff between speed and accuracy of the reconstructed geometry. Figure 3 shows reconstruction results from 5 input images with different parameter settings and for different resolutions of the resulting depth map. For evaluation we used a standard personal computer equipped with a NVidia GTX 480 graphics card and implemented our method using the CUDA framework. With high quality parameter settings, an accurate reconstruction of the scene can be computed at 1.8 frames per second (fps). A slightly less accurate reconstruction can be obtained at 11.3 fps. In both cases, the input images and reconstructed depth map have a resolution of 640×480 pixels. By reducing the resolution of the computed depth map, even realtime performance can be reached with 24 fps at a depth map resolution of 480×360. In the two latter cases, a slightly different numerical scheme is used: a number of 4 internal iterations is performed before the data is exchanged with other blocks of the parallelized implementation, resulting in small blocking artifacts visible in the reconstruction.

Table 1. Parameter settings for different frame rates

Quality Setting	High	Medium	Low
Pyramid Levels	24	10	7
Pyramid Scale-Factor	0.94	0.8	0.7
Iterations per Level	120	70	70
Internal Iterations	1	4	4
Frames per Second	1.8	11.3	24



(a) 1.8 fps **(b)** 11.3 fps **(c)** 24 fps

Fig. 3. Trade-off between speed and accuracy

Quantitative evaluation of the noise robustness: In contrast to traditional stereo approaches, the proposed framework makes use of multiple images in order to increase the robustness of the reconstruction. Figure 4 shows the reconstruction error $\epsilon = \frac{\int_\Omega (h_\sigma - h_{\sigma=0})^2 \, d\mathbf{x}}{\int_\Omega h_\sigma^2 \, d\mathbf{x} + \int_\Omega h_{\sigma=0}^2 \, d\mathbf{x}}$ as a function of the noise level σ. In contrast to the two-frame formulation, the integration of multiple frames is substantially more robust to noise.

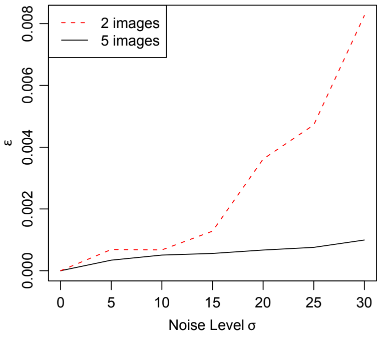


Fig. 4. Reconstruction error ϵ as a function of the noise level σ. The integration of multiple images is significantly more robust to noise.

5 Conclusion

We proposed a variational method to compute robust dense depth maps from a handheld camera in real-time. The variational approach combines a robust regularizer with a data term that integrates multiple frames rather than merely two. Experimental results confirm that the integration of multiple images substantially improves the noise robustness of estimated depth maps. The nonlinear and non-convex functional is minimized by sequential convex optimization. To this end, we adapt a primal-dual algorithm originally proposed for optical flow to the problem of depth map estimation, and show that the primal update can be solved in closed form by means of a sophisticated thresholding scheme. While the camera motion is determined on the CPU, the depth map is estimated on the GPU in a coarse-to-fine manner, leading to dense depth maps at a speed of 24 frames per second.

References

1. Jin, H., Favaro, P., Soatto, S.: Real-time 3-d motion and structure of point-features: A front-end for vision-based control and interaction. In: Int. Conf. on Computer Vision and Pattern Recognition, pp. 2778–2779 (2000)
2. Nister, D.: Preemptive ransac for live structure and motion estimation. In: IEEE Int. Conf. on Computer Vision, pp. 199–206 (2003)
3. Kolev, K., Cremers, D.: Continuous ratio optimization via convex relaxation with applications to multiview 3d reconstruction. In: CVPR, pp. 1858–1864. IEEE, Los Alamitos (2009)
4. Zach, C., Pock, T., Bischof, H.: A globally optimal algorithm for robust TV-L^1 range image integration. In: IEEE Int. Conf. on Computer Vision, Rio de Janeiro, Brazil. LNCS. IEEE, Los Alamitos (2007)
5. Pock, T., Schoenemann, T., Graber, G., Bischof, H., Cremers, D.: A convex formulation of continuous multi-label problems. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 792–805. Springer, Heidelberg (2008)
6. Slesareva, N., Bruhn, A., Weickert, J.: Optic flow goes stereo: A variational method for estimating discontinuity-preserving dense disparity maps. In: Kropatsch, W.G., Sablatnig, R., Hanbury, A. (eds.) DAGM 2005. LNCS, vol. 3663, pp. 33–40. Springer, Heidelberg (2005)
7. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime TV-L1 optical flow. In: Hamprecht, F.A., Schnörr, C., Jähne, B. (eds.) DAGM 2007. LNCS, vol. 4713, pp. 214–223. Springer, Heidelberg (2007)
8. Newcombe, R.A., Davison, A.J.: Live dense reconstruction with a single moving camera. In: Int. Conf. on Computer Vision and Pattern Recognition (2010)
9. Chambolle, A.: An algorithm for total variation minimization and applications. J. Math. Imaging Vis. 20(1-2), 89–97 (2004)
10. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Phys. D 60(1-4), 259–268 (1992)
11. Klein, G., Murray, D.: Parallel tracking and mapping for small AR workspaces. In: Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2007), Nara, Japan (November 2007)