DENSE AND CONTINUOUS DEPTH ESTIMATION USING A SLIDING CAMERA

Kailin Ge, Student Member, IEEE, Jianjiang Feng, Member, IEEE, and Jie Zhou, Senior Member, IEEE

Department of Automation, Tsinghua University, Beijing, China
gkl05@mails.tsinghua.edu.cn; {jfeng, jzhou}@tsinghua.edu.cn

ABSTRACT

3D information of real-world scenes provides important clues for many computer vision tasks. We present a simple but effective sliding camera system as well as a corresponding stereo reconstruction framework to retrieve 3D information of static scenes. By fusing geometric properties of the sliding camera system, our reconstruction algorithm achieves higher accuracy than conventional methods in quantitative experiments. Besides, the practicality of our system is validated on real world scenes.

Index Terms— multiple baseline stereo, sliding camera, constrained bundle adjustment, variational depth estimation

1. INTRODUCTION

Image-based multi-view stereo reconstruction is an important research area of computer vision. According to the different kinds of multi-view source, there are roughly three categories: 1) images are captured by specialized camera arrays (e.g, Yang et al. [1]); 2) all images are unordered and discontinuous (e.g, Snavely et al. [2]); 3) images are captured by one or several moving camera (e.g, Pollefeys et al. [3], Zhang et al. [4], Pradeep et al. [5]). For the first category, since there is no need to estimate camera poses on the fly, camera arrays are able to adapt complex scenes, but are often expensive and cumbersome. For the second category, it is often hard to find out correct camera poses which is required by later stereo algorithms, especially in complex scenes. The third category is a trade-off, where cameras are partially constrained by continuity that facilitates pose estimation, and this property make it much more practical. However, in some complex circumstances (e.g., with a lot of noises or with few feature points), the constraint of continuity is not sufficient.

In this paper, a sliding camera system (Fig. 1) is used to overcome the shortcomings of freely moving setups. We propose a dense stereo reconstruction framework (Fig. 2) which utilize not only the continuity of camera but also geometric properties of the slider. Unlike freely moving setups, the slider provides a strong constraint which greatly facilitates camera pose estimation. It should be noticed that sliding camera has also been used in [6, 7], but for target tracking and synthetic aperture imaging; dense stereo reconstruction is not involved in these works. The contribution of this paper is twofold:

1. We analyze geometric properties of a sliding camera, and propose a constrained camera model for camera pose estimation;
2. By merging cross-ratio property, we propose a variational framework for depth estimation.

The remainder of this paper is organized as follows: Section 2 reviews related work; Section 3 describes our constrained camera pose estimation algorithm, while Section 4 describes our variational depth estimation algorithm. Experiments and evaluations are shown in Section 5, and Section 6 is summary and prospect of this paper.

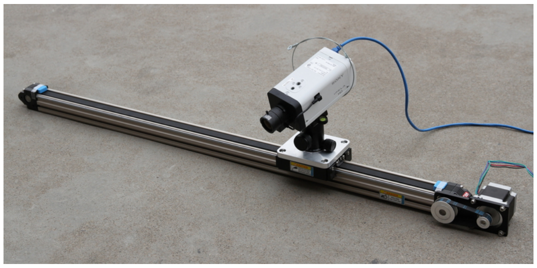


Fig. 1. The sliding camera system, in which a single camera on the track is controlled by a stepping motor.

2. RELATED WORK

As described in Section 1, moving camera based algorithms for 3D reconstruction is more practical than the other two categories. In recent decade, a series of methods were proposed [8, 9, 4] for restoring depth maps from image sequences captured by a freely moving camera. Especially, the state-of-the-art method [4] restores high quality depth maps using a hybrid framework, in which loopy belief propagation (LBP) algorithm [10] for solving Markov random field (MRF) is used. But without the geometric constraints provided by the slider, their precision is heavily degraded by camera pose estimation errors.

Besides, sliding camera setup is also used in [6, 7], but for different purpose: Nakabo et al. [6] installs two PTZ cameras on the same slider track, and localize a dynamic target in real-time by scheduling two cameras. Zhang et al. [7] proposes a sliding camera based algorithm for synthetic aperture imaging. These works do not involve dense stereo reconstruction.

3. CAMERA POSE ESTIMATION

As described in Section 1, in our system the camera is mounted to a straight track, which guarantees the camera will move straightly and parallelly. When moving along the track, the camera captures a series of images $\{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_n\}$, and our task is to calculate one (or more) view dependent depth map $\mathcal{D}_k,\ k \in \{1, 2, \ldots, n\}$ from these images. For a static scene, this setup is equivalent to a linear array with a large number of cameras, so we formulate our system as a linear camera array in this paper.

Trajectories of feature points are first extracted using KLT tracker [11], which is essential for camera pose estimation. Different trajectories represent different potential 3D points and their observations. Because the cameras are co-linear, we could find a coordinate system so that optical centers of all cameras are on X-axis, and the one of the first camera is the origin (Fig. 3). Thus, the optical centers of cameras could be denoted as $O_1 = (c_1, 0, 0)$, $O_2 =$

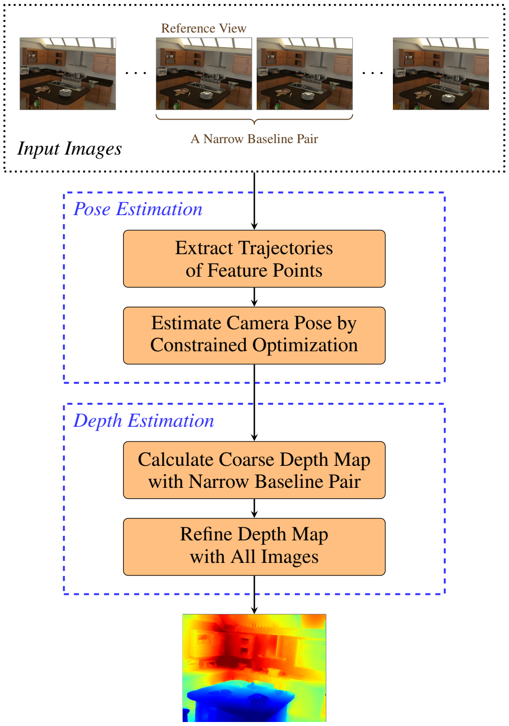

Reference View
...
...
A Narrow Baseline Pair
Input Images
Pose Estimation
Extract Trajectories of Feature Points
Estimate Camera Pose by Constrained Optimization
Depth Estimation
Calculate Coarse Depth Map with Narrow Baseline Pair
Refine Depth Map with All Images

Fig. 2. Flowchart of the proposed depth estimation system: 1) Camera poses are first estimated from feature point trajectories under the constraint of linear translation. 2) Coarse depth map for the reference view is calculated from a narrow-baseline pair; then refined depth map is obtained using information from all views.

$(c_2, 0, 0)$, $\ldots$, $O_n = (c_n, 0, 0)$, where $c_1 = 0$. On the other hand, the cameras share the same rotation matrix $\mathbf{R}$ since they are parallel. Therefore, we propose the following constrained bundle adjustment model to solve camera poses:

$$\min_{\substack{\mathbf{R}, c_1, \ldots, c_n \\ X^{(1)}, \ldots, X^{(m)}}} \left(\sum_{\substack{1 \leq k \leq n \\ 1 \leq l \leq m}} \delta_k^{(l)} \left\| x_k^{(l)} - P_k\left(X^{(l)}\right) \right\|_2^2 \right), \tag{1}$$

where P_k projects a 3D point $X^{(l)}$ to a 2D image coordinate on camera k using projection matrix $\mathbf{P}_k = \mathbf{KR}\left[\, \mathbf{I} \,\middle|\, -O_k \,\right]$, and $\delta_k^{(l)}$ is visibilities of corresponding projections.

4. DEPTH ESTIMATION

In this section, we describe how we embed a specific geometric property into a variational optimization for depth estimation. First of all, we introduce the cross-ratio property of the sliding camera system.

4.1. Cross-Ratio Property

As illustrated in Fig. 4, by stacking all images in Fig. 3 together, the projections of the same 3D point are co-linear, and all projection lines intersect at the same place e. Actually, e is the motion vanishing point of all feature points, and its homogeneous coordinate is $e = \mathbf{KR}[1,\ 0,\ 0]^{\mathrm{T}}$ (the projection of X-axis). Using cross-ratio invariant property in perspective geometry [12, 13], the following equation holds true:

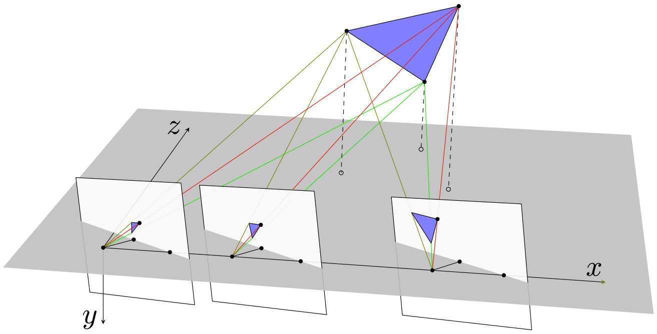


Fig. 3. Geometry relationship between cameras which are constrained by the slider.

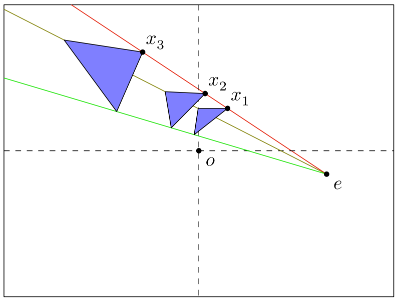


Fig. 4. After stacking all captured images together, the projections of the same 3D point are co-linear with e.

$$\frac{\left(\overrightarrow{ex_j} \cdot \overrightarrow{d(x_k)}\right)\left(\overrightarrow{x_i x_k} \cdot \overrightarrow{d(x_k)}\right)}{\left(\overrightarrow{ex_i} \cdot \overrightarrow{d(x_k)}\right)\left(\overrightarrow{x_j x_k} \cdot \overrightarrow{d(x_k)}\right)} = \frac{\overrightarrow{O_i O_k} \cdot \vec{D}}{\overrightarrow{O_j O_k} \cdot \vec{D}}, \tag{2}$$

where x_i, x_j, x_k are the projections of the same 3D point on image $\mathcal{I}_i, \mathcal{I}_j, \mathcal{I}_k$, and

$$\overrightarrow{d(x_k)} = \frac{\overrightarrow{ex_k}}{\left|ex_k\right|}.$$

Since $\overrightarrow{O_i O_k} \cdot \vec{D} = c_k - c_i$, $\overrightarrow{O_j O_k} \cdot \vec{D} = c_k - c_j$, $\overrightarrow{ex_j} = \overrightarrow{ex_k} + \overrightarrow{x_k x_j}$ and $\overrightarrow{ex_k} \cdot \overrightarrow{d(x_k)} = \left|ex_k\right|$, the equation above can be rewritten as

$$\overrightarrow{x_k x_j} \cdot \overrightarrow{d(x_k)} = \frac{\left(c_j - c_k\right)\left|ex_k\right|\left(\overrightarrow{x_k x_i} \cdot \overrightarrow{d(x_k)}\right)}{\left(c_i - c_k\right)\left|ex_k\right| + \left(c_i - c_j\right)\left(\overrightarrow{x_k x_i} \cdot \overrightarrow{d(x_k)}\right)}. \tag{3}$$

It is worth to figure out that Equation (3) always holds true with or without intrinsic calibration, or even when e is at infinity. So it can be conveniently applied to image coordinate transformations.

This property provides an important geometric rule for finding correspond 2D coordinates, which is essential for the later multi-view stereo.

4.2. Depth Initialization

Thanks to the continuity of the slider, we can find a view i which is near to the reference view k. They constitute a narrow-baseline pair,

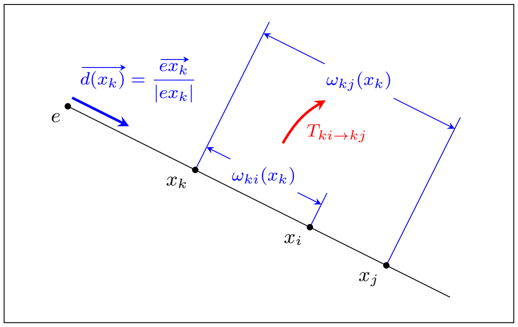


Fig. 5. $T_{ki\to kj}$ encodes the cross-ratio property of our system, which transforms coordinates between different view pairs. $\omega_{ki}(x_k)$ is the disparity of x_k between view k and view i, so as $\omega_{kj}(x_k)$.

between which it is much easier to find out pixel correspondences. This makes it feasible to solve our task by continuous variational methods [14, 15], which is much more accurate than MRF methods [16, 10].

It is a two-view stereo problem for view k and view i, and we use the following variational optimization model to get a initial coarse result:

$$\min_{\omega_{ki}} \; E_D(\omega_{ki}) + \gamma E_S(\omega_{ki}). \tag{4}$$

In this optimization, ω_{ki} is a function mapping a coordinate $x_k \in \Omega$ to the tangential disparity along the epipolar line (as illustrated in Fig. 5) between view k and view i. E_D is the data term for measuring pixel variance, which is defined as

$$E_D(\omega_{ki}) = \int_\Omega c(x_k)\Psi\Big(\big\|\mathcal{I}_k(x_k) - \mathcal{I}_i\big(x_k + \omega_{ki}(x_k)\cdot\overrightarrow{d(x_k)}\big)\big\|_2^2\Big)\mathrm{d}x_k, \tag{5}$$

where $\mathcal{I}_k$ and $\mathcal{I}_i$ are 3-channel RGB images, and $\overrightarrow{d(x_k)}$ (as illustrated in Fig. 5) is previously defined in Equation (2). $\Psi(x^2) = \sqrt{x^2+\varepsilon^2}$ is a robust energy function, and a confidence factor is defined as

$$c(x_k) = 1 - \frac{\sigma_c^2}{\big\|\nabla\mathcal{I}_k(x_k)\big\|_2^2 + \sigma_c^2}.$$

E_S is the smoothing term, defined as

$$E_S(\omega_{ki}) = \int_\Omega \xi_k(x_k)\Psi\Big(\big\|\nabla\omega_{ki}(x_k)\big\|_2^2\Big)\mathrm{d}x_k, \tag{6}$$

where ξ_k is the edge prior to preserve discontinuities:

$$\xi_k(x_k) = \begin{cases} 0.1 & \text{if } \big\|\nabla\mathcal{I}_k(x_k)\big\|_2 > \sigma_c, \\ 1 & \text{otherwise.} \end{cases}$$

Optimization (4) can be solved by the methods in [14, 15]: With an all-zero initial solution, the result converges gradually along a fine pyramid of step 0.9, and each step is solved by the fixed point iteration. After this initialization phase, a coarse depth map is available for further processing.

4.3. Depth Refinement

To exploit informations from all views, we adjust Optimization (4) to

$$\min_{\omega_{ki}} \; E_D^{\text{all}}(\omega_{ki}) + \gamma E_S(\omega_{ki}), \tag{7}$$

where E_D^{all} is a new data term defined as

$$E_D^{\text{all}}(\omega_{ki}) = \int_\Omega c(x_k)\sum_{j\neq k}\Psi\Big(\big\|\mathcal{I}_k(x_k) - \mathcal{I}_j\big(x_k + T_{ki\to kj}(\omega_{ki})(x_k)\cdot\overrightarrow{d(x_k)}\big)\big\|_2^2\Big)\mathrm{d}x_k. \tag{8}$$

Here $T_{ki\to kj}$ is a functional constructed according to Equation (3):

$$\begin{gathered} T_{ki\to kj}: \begin{array}{ccc} (\Omega\to\mathbb{R}) & \to & (\Omega\to\mathbb{R}) \\ \omega_{ki} & \mapsto & \omega_{kj} \end{array}, \\ \omega_{kj}(x_k) = \frac{(c_j - c_k)\big|ex_k\big|\omega_{ki}(x_k)}{(c_i - c_k)\big|ex_k\big| + (c_i - c_j)\omega_{ki}(x_k)}. \end{gathered} \tag{9}$$

As illustrated in Fig. 5, $T_{ki\to kj}$ encodes the cross-ratio property which conveniently transforms coordinates between different views. With the help of $T_{ki\to kj}$, Optimization (7) gracefully integrates information from all available images captured by our system. It can also be solved by fixed-point iteration, and converges from the initial solution.

5. EXPERIMENTS

To quantitatively evaluate the accuracy of our algorithm, we built a data set with Blender[1], a free and open-source software which renders 3D scene models into 2D images with depth maps. The data set contains three image sequences – the "Shelf", the "Kitchen" and the "Lobby". The "Shelf" is an ill-posed case and the most difficult one. The "Kitchen" contains lots of depth discontinuities. The "Lobby" contains large proportion of plane areas. Using the ground-truth depth maps generated by Blender, we get quantitative errors of reconstruction results, and comparison results is shown in Fig. 6 (a) and Fig. 7. Our algorithm achieves the highest accuracy in all three sequences.

To further verify the practicality of our algorithm, we tested it with real scenes captured by our sliding camera system, and the results are shown in Fig. 6 (b). Although there are no ground truth for quantitative evaluation, the fidelity of the rendered point clouds is satisfactory.

6. CONCLUSIONS

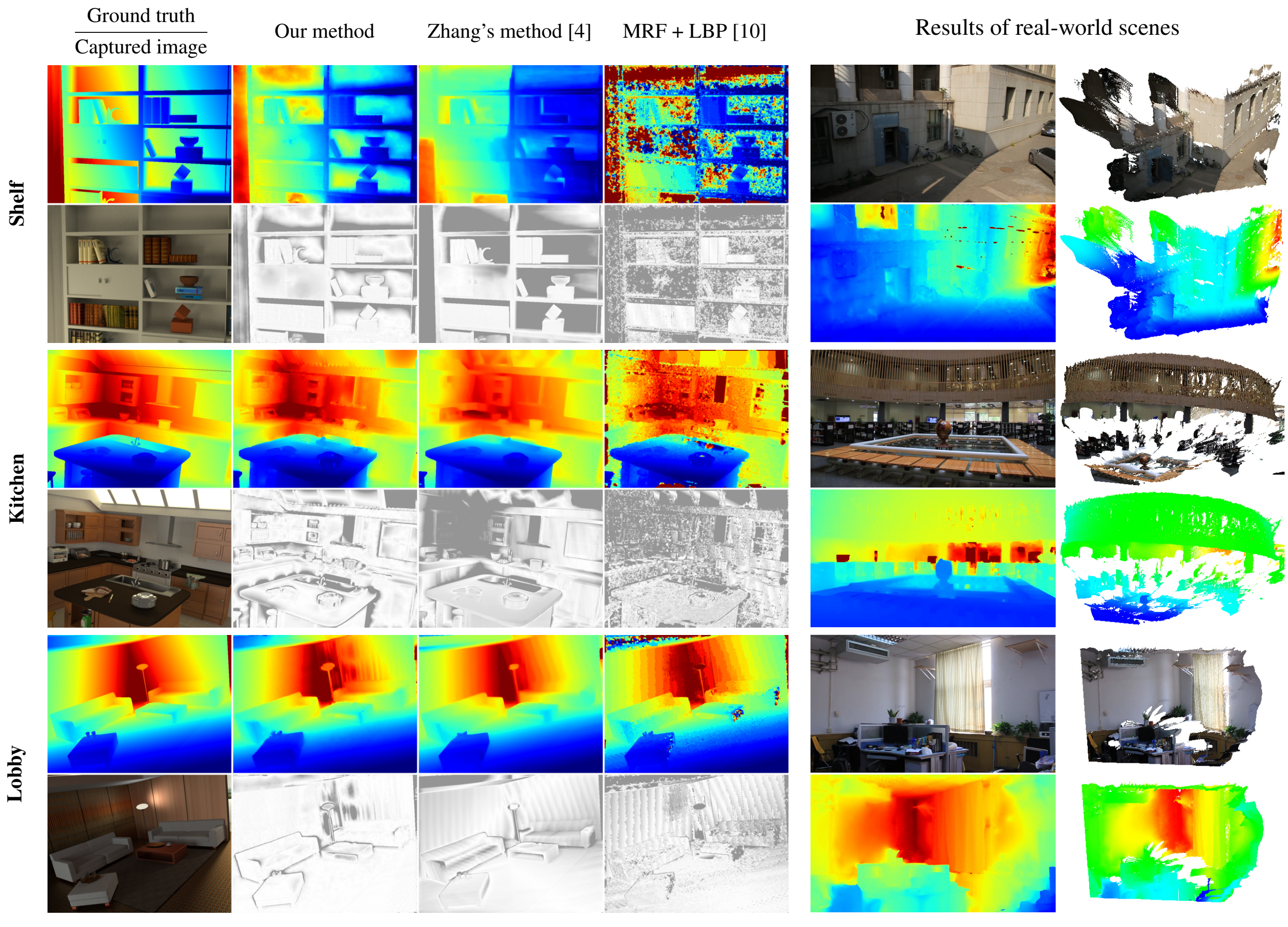
In this paper, we propose a dense stereo reconstruction framework for a sliding camera system. By encoding the inherent geometric properties of the sliding camera system into both camera pose estimation and variational depth optimization, our algorithm produces more accurate results than conventional algorithms.

Right now our system and algorithm works for static scenes only. We will work on extending our algorithm to handle simple dynamic scenes in the future.

7. ACKNOWLEDGEMENT

This work is supported by the National Natural Science Foundation of China under Grants 61225008, 61373074 and 61373090, the National Basic Research Program of China under Grant 2014CB349304, the Ministry of Education of China under Grant 20120002110033, and the Tsinghua University Initiative Scientific Research Program.

[1] `http://www.blender.org/`



(a) (b)

Fig. 6. (a) Results on the synthetic data set. Odd rows show the ground truth depth maps and those obtained by three different methods, while even rows show corresponding captured images and error maps of three different methods with respect to the ground truth. Darker areas in error maps mean larger errors. (b) Results of real-world scenes. Captured image, restored depth map and rendered point clouds (with two different coloring setups) are shown here.

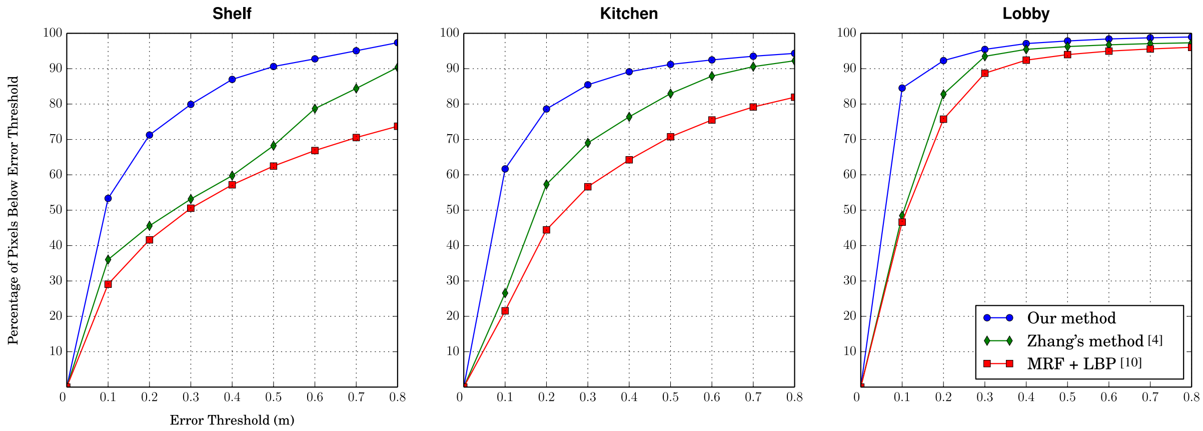


Fig. 7. Statistical comparison between different depth estimation methods on the synthetic data set. Our algorithm achieves higher accuracy than conventional methods.

8. REFERENCES

[1] Wenzhuo Yang, Guofeng Zhang, Hujun Bao, Jiwon Kim, and Ho Young Lee, "Consistent depth maps recovery from a trinocular video sequence," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 1466–1473.

[2] Noah Snavely, Steven M. Seitz, and Richard Szeliski, "Photo tourism: exploring photo collections in 3d," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 835–846, 2006.

[3] Marc Pollefeys, Luc Van Gool, Maarten Vergauwen, Frank Verbiest, Kurt Cornelis, Jan Tops, and Reinhard Koch, "Visual modeling with a hand-held camera," *International Journal of Computer Vision*, vol. 59, no. 3, pp. 207–232, 2004.

[4] Guofeng Zhang, Jiaya Jia, Tien-Tsin Wong, and Hujun Bao, "Consistent depth maps recovery from a video sequence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 6, pp. 974–988, June 2009.

[5] V. Pradeep, C. Rhemann, S. Izadi, C. Zach, M. Bleyer, and S. Bathiche, "Monofusion: Real-time 3d reconstruction of small scenes with a single web camera," in *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, Oct 2013, pp. 83–88.

[6] Y. Nakabo, T. Mukai, Y. Hattori, Y. Takeuchi, and N. Ohnishi, "Variable baseline stereo tracking vision system using high-speed linear slider," in *Proceedings of the IEEE International Conference on Robotics and Automation*, April 2005, pp. 1567–1572.

[7] Xiaoqiang Zhang, Yanning Zhang, Tao Yang, and Zhengxi Song, "Calibrate a moving camera on a linear translating stage using virtual plane + parallax," in *Proceedings of the Intelligent Science and Intelligent Data Engineering*, vol. 7751, pp. 48–55. 2013.

[8] SingBing Kang and Richard Szeliski, "Extracting view-dependent depth maps from a collection of images," *International Journal of Computer Vision*, vol. 58, no. 2, pp. 139–163, 2004.

[9] P. Gargallo and P. Sturm, "Bayesian 3d modeling from images using multiple depth maps," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2005, vol. 2, pp. 885–891.

[10] Pedro F. Felzenszwalb and Daniel P. Huttenlocher, "Efficient belief propagation for early vision," *International Journal of Computer Vision*, vol. 70, no. 1, pp. 41–54, 2006.

[11] Jean-Yves Bouguet, "Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm," *Intel Corporation*, vol. 2, pp. 3, 2001.

[12] T. Moons, L. Van Gool, M. Proesmans, and E. Pauwels, "Affine reconstruction from perspective image pairs with a relative object-camera translation in between," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 1, pp. 77–83, Jan 1996.

[13] Richard Hartley and Andrew Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2003.

[14] Thomas Brox, Andrs Bruhn, Nils Papenberg, and Joachim Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Proceedings of the European Conference on Computer Vision*, vol. 3024, pp. 25–36. 2004.

[15] Yebin Liu, Xun Cao, Qionghai Dai, and Wenli Xu, "Continuous depth estimation for multi-view stereo," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 2121–2128.

[16] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *Proceedings of the IEEE International Conference on Computer Vision*, 2001, vol. 2, pp. 508–515.