Micro-Baseline Stereo

Neel Joshi C. Lawrence Zitnick

Microsoft Research

Figure 1: *Synthetic refocusing using micro-baseline stereo: The depth map computed using micro-baselines can be used to create a synthetic shallow depth-of-field image. We blur the image with a pillbox point-spread function that is scaled as a function of the difference of the relative depth of a particular pixel and a chosen reference depth, which will remain in focus. We show focus for three depths: the foreground, mid-ground, and background.*

Abstract

Tradeoffs exist between the baseline or distance between cameras and the difficulty of matching corresponding points in stereo and structure from motion. Smaller baselines result in reduced disparities reducing the accuracy of depth estimation. Larger baselines increase the range of observed disparities, but also increase the difficulty of finding corresponding points. In this paper, we explore the use of very small baselines, called micro-baselines. Micro-baselines, typically just a few millimeters, provide the advantage that they can be captured using a single camera. That is, a "static" camera that is either hand-held or mounted on a tripod will typically vibrate some small amount while capturing video. We take advantage of the vibrating motion to compute depth information. For hand-held cameras a small amount of motion is generally always present, while many surveillance applications involve cameras mounted outside or on high poles that exhibit this type of motion. Even indoor cameras mounted on tripods move due to human traffic and machine vibrations.

1 Introduction

The baseline or distance between cameras is an important factor in stereo and structure from motion. Smaller baselines reduce the accuracy of computed depths, since the observed disparity of corresponding points is reduced relative to changes in depth. Larger baselines increase the observed disparities, but increase the difficulty of finding corresponding points.

Several works address this issue by using multiple cameras. For instance, multi-baseline stereo uses cameras with large disparities to increase the accuracy of depth estimation, while cameras with smaller baselines are used to disambiguate correct correspondences [Okutomi and Kanade 1991]. When matching images with small baselines, simple window matching costs may be used such as sum of squared distances or normalized correlation [Hirschmller and Scharstein 2007; Tombari et al. 2008]. Many multi-view stereo algorithms also take advantage of these techniques [Seitz et al. 2006]. The use of large baselines [Pritchett and Zisserman 1998] requires more sophisticated matching measures, such as SIFT [Lowe 2004] or MSER [Matas et al. 2002]. Even with these measures, robust correspondence algorithms such as RANSAC are necessary. In structure from motion approaches [Triggs et al. 2000], images with varying baselines can be used to refine correspondences across several images. The robustness of these techniques has been demonstrated in several recent papers using large databases of images [Snavely et al. 2006; Agarwal et al. 2009].

We explore the use of very small baselines, called micro-baselines. Micro-baselines, typically just a few millimeters, provide the advantage that they can be captured using a single camera. That is, a "static" camera that is either hand-held or mounted on a tripod will typically vibrate some small amount while capturing video. We take advantage of the vibrating motion to compute depth information. For hand-held cameras a small amount of motion is generally always present, while many surveillance applications involve cameras mounted outside or on high poles that exhibit this type of motion. Even indoor cameras mounted on tripods move due to human traffic and machine vibrations.

The use of micro-baselines provides three main challenges. First the disparity between images or frames in the video is typically a small number or even a fraction of a pixel. As a result, accurate sub-pixel disparity estimates must be computed. Second, even with accurate sub-pixel estimates, large numbers of images must be obtained to offset inherent noise in the disparity estimation. Finally, the motion of the camera is from random vibrations, so extrinsic calibration information is not known.

2 Previous Work

A broad comparison of stereo vision techniques is the paper by Scharstein and Szeliski [2002]. Several methods have been proposed for sub-pixel correspondence. The work of Takita et. al [2004] use a Phase-Only correlation technique to align two windows. The approach of Psarakis et. al [2005] handles both suppixel alignments as well as photometric distortions. Thevenaz et. al [1998] use a pyramid based approach for sub-pixel registration. Shimizu and Okutomi [2005] analyze sub-pixel estimation error using different functions.

Previous works have also addressed computing depth with a single image [Hoiem et al. 2005].

3 Micro-Baseline Stereo

We will now describe the framework for micro-baseline stereo. Let $\{I_1, ..., I_j, ..., I_n\}$ denote a sequence of n images of an object where each image is a different frame of an input video-sequence. We assume each frame of the sequence is captured from a slightly different viewpoint. For a point ρ, the observed intensity is denoted as $i_j(\rho)$. Given a reference world-space coordinate system, which we choose to be coincident with coordinate system of a reference image I_0, correspondence between reference-coordinate points and coordinates in an arbitrary image is established given the points relative depth, $z(x, y)$, corresponding to the object's surface and the camera-projection matrix P_j at time j. Observations across all n views for all points $\{\rho_1...\rho_j...\rho_m\}$ can be related to the reference coordinate system by:

$$I_0(x, y) = I_j(P_j(x_j, y_j, z(x_j, y_j))). \quad (1)$$

In our work, the central goal in our method is to solve for correspondence and, in turn, the unknown disparity map. The unknowns in this system are the correspondence, disparity map, and camera projection matrices $\{P_1...P_j...P_m\}$. To recover these components we use a structure from motion approach that derives from Tomasi-Kanade factorization [Tomasi and Kanade 1992] and plane+parralax [Criminisi et al. 1998; Vaish et al. 2004].

We model our scene using a plane+parralax framework. Specifically, this frame work models the projection process as:

$$\hat{\rho} = P_j * \hat{\rho}_j = H_j * \hat{\rho}_j + \Delta c_j * \hat{z}, \quad (2)$$

where H_j is a planar homography, $\hat{\rho}$ is a vector of homogeneous 2D points in the reference coordinate system, $\hat{\rho}_j$ are the points in a view j, Δc_j is the relative in-plane displacement of the view relative to the reference view r, and and $\hat{z}$ is a vector of relative depths. The "plane+parralax" nomenclature comes from the process of first aligning points for each view to a reference plane, captured by the homography H_j, followed by computing a rank-q factorization to get the depth-based, i.e. parallax, components Δc_j and $\hat{z}$. Typically, when performing "plane+parralax" one places a planar calibration grid in the scene and uses features on it to compute the planar homographies. Alternatively, one can use a robust homography fitting approach to detected interest points in the scene.

1. **Set The Reference Frame** Pick one frame I_r to be the reference view and set the world-space coordinate system coincident with this view by accordingly transforming all projection matrices relative to P_r.
2. **Compute Dense Per-Pixel Optical Flow** We compute a fine sub-pixel estimate of pairwise correspondence from each image P_j to the reference image P_r using a dense, optical-flow approach, resulting in (x, y) flow vectors F_j for each image
3. **Compute Camera Projection Matrices** Use RANSAC to compute the camera projection matrix P_j for each frame I_j, such that P_j maps I_j to I_r. The RANSAC process will align a single, arbitrary depth plane
4. **Warp Flow Values to Reference Coordinate System** Apply the projection matrices Pj to each view set of flow fields F_j to globally align a depth plane, resulting in flow vectors $\hat{F}_j$
5. **Compute Dense Disparity Map** Find the dense disparity map by computing a rank-1 factorization of the residual flow $F_j - \hat{F}_j$

Figure 2: *Our micro-baseline stereo algorithm.*

Our setup is quite similar to the second approach, but has some subtle differences. Particularly as our baseline is extremely small, on the order of millimeters or less, an in plane rotation and translation alone are a very good approximations to the full planar homography. Under this assumption, the revised projection model is:

$$\hat{p} = R_j * \hat{p}_j + T_j + \Delta c_j * \hat{z}, \quad (3)$$

where R_j is the in-plane rotation and T_j is a global (x, y) translation in the image plane. In this model yaw and pitch rotation and global (non-depth dependent) camera translation are captured by T_j. We note that T_j is a 2D component as the change in depth of the camera relative to the scene is negligible in our setup.

Our goal is to recover the relative depths $\hat{z}$. Once R_j and T_j are computed, $\hat{z}$ can be recovered using an rank-1 factorization [Tomasi and Kanade 1992]. We recover these transformations using a RANSAC process on the results of correspondences computed using an dense optical flow alignment.

There is extensive literature on optical flow [Baker et al. 2007] for tracking pixels that move small amounts from frame to frame. Our case is relatively simple compared to finding general flow since the motion is small with minimal occlusions. We have found both HornSchunck [Horn and Schunck 1980] and patch based SSD methods to work well, if global alignment is performed first.

By running optical flow between each image I_j and the reference image I_r we compute an alternate estimate for Equation 7:

$$\hat{p} = \hat{p}_j + \hat{F}_j, \quad (4)$$

where $\hat{F}_j$ is per-view, vector of per-pixel (x, y) translations chosen to minimized the re-projection error between the reference view and a given frame:

$$F(p_j) = \underset{F(p_j)}{\operatorname{argmin}} \sum_{i \in N(p_j)} (p(I_r(p^i) - I(p_j^i + F(p_j)))^2, \quad (5)$$

for each pixel p_j, where $N(p_j)$ is the set of neighboring pixels – we use 5×5 pixel windows. Sup-pixel accuracy if found by re-sampling the image on a $1/10$ pixel grid using bi-cubic interpolation, and we search by shifting each window by the same $1/10$ pixel stepping size.

By subtracting Equations 7 and 4, we get:

$$F_j = (R_j * \hat{p}_j - \hat{p}_j) + T_j + \Delta c_j * \hat{z}. \quad (6)$$

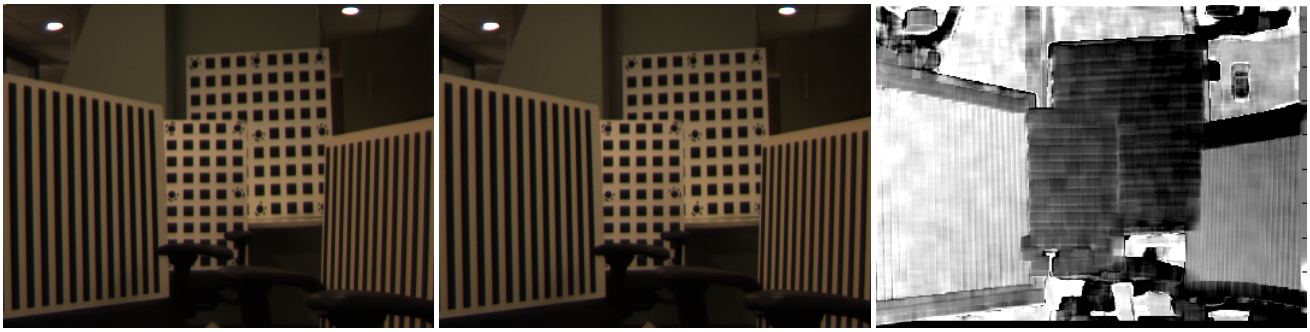

Figure 3: *Test Scene: Two frames from the input video and our relative depth map using 500 images.*

We estimate the rotations and translations for each view using a RANSAC procedure; however, instead of computing this in the direct way, i.e., independently for each view, we use a RANSAC process that uses the same inliers across all views. By using the same set of inliers and thus the same set of corresponding points, we ensure that the same reference plane is fit across all views. With separate inliers, different reference planes could be fit, and this would violate the plane+parralax model.

The final step in our algorithm is to compute the relative depth by factorizing the residual local flow:

$$(F_j - (R_j * \hat{p}_j - \hat{p}_j) + T_j) = \Delta c_j * \hat{z}. \quad (7)$$

The left side is the residual flow, computed using the fit rotations, translations, and per-pixel flows. We stack this system of equations for all images $j = 1...N$, creating a significantly over-constrained system, and compute $\hat{z}$ using a rank-1 factorization using SVD.

4 Results

To validate our method, we present several experimental scenes. Our first experiment is a proof-of concept result in a lab setup. While the remaining results are in real-world, uncontrolled settings.

We filmed datasets with two different setups. For the results in Figure 3, 4, and 5, we filmed approximately 1000 frames at 30 FPS (about 30 seconds of video) from a video camera mounted on an unstable tripod. We imparted small vibrations to the tripod to jitter the camera to acquire micro-baseline views. For the results in Figure 1, we filmed approximately 100 frames at 30 FPS (about 3 seconds of video) using a hand-held smart-phone.

For each method we ran our algorithm to compute relative depth. For our flow-based alignment, we have tried and SSD based approach and Horn-Schunck [Horn and Schunck 1980]. For the SSD approach, we initially compute a global offset using a whole-image SSD search to remove large global translation, and then seed the local flow estimate with this. The local flow computation works on a ± 0.5 pixel window in steps of $1/10$ of a pixel. We found this two level approach to be important for our outdoor scene, where wind causes large global pixel displacements. For Horn-Schunck, we use a pyramid-based implementation with 3 levels and a moderate amount of smoothing.

Our final results use 100-500 images each depending on the dataset. As there is an inherent ambiguity in the sign and magnitude of the relatively depths (an inherent component of the rank-1 factorization), we scale and shift our relative depths to all be positive, with zero relative depth corresponding to the plane at infinity, i.e., the plane with zero disparity.

Figure 3 shows our proof-of concept lab setup. Note that the differing depth planes, as easily visualized using the calibration grid, show up very distinctly in the displayed depth map. There are relatively sharp transitions in the map corresponding to sharp transitions in depth of the grids.

Our next two results in Figure 4 and 5 show more challenging real-world scenes – one filmed indoors and one outside. The indoor scene in Figure 4 is in a small office kitchenette. Note that the sweeping depth of the low textured wall is correctly captured in our result. In addition, the result shows the local structures from objects on the counter. The outdoor scene is very challenging (Figure 4). There are many objects, such as plants and bricks, that contain repeating ambiguous patterns that can cause false matches. These types of ambiguities are extremely hard to handle with a large based-line stereo step, but are handled well with our micro-baseline setup. The small baseline reduces the space of ambiguous matches significantly. This dataset is also challenging as there is a very large depth range, areas that are saturated and textureless, and scene motion due to wind moving the plants, yet our depth map shows the sweeping plane of the building and ground and captures local structures. The result is noisier than the first two to the many challenging aspects in the scene, yet the relative depth map is still reasonable.

Figures 4 and 5, respectively show results where we have computed the relative depth map using 10, 200, and 500 images. As expected, the reconstruction quality increases and stabilizes as more images are added, which removes the inherent noise in the flow estimates.

In Figure 1, we show three more real-world scenes with a range of depths and textures. For these scenes, we use the relative depth map created from 100 input images, to create a synthetic shallow depth-of-field. These results are created by blurring the image with a pillbox (i.e. disk) point-spread function that is scaled as a function of the difference of the relative depth of a particular pixel and a chosen reference depth, i.e., the one that will remain in focus. The shallowness of the depth-of-field is a function of the mapping from depth difference to the blur kernel size. We create a set of focal sweep of images by using a fixed set of reference depths evenly swept through the range of recovered depths. In Figure 1, we show focus for three depths chosen so that the foreground, mid-ground, and background are respectively in focus for each dataset. The synthetic re-focusing operation is computationally fast and could used with a UI, such as a "tap-to-focus", so that a user could focus on a particular object at its corresponding depth.

5 Conclusions and Future Work

We have shown how to recover depth maps from small natural variations that can occur in camera position when recording a short video clip. Our method recovers reasonable relative depth maps

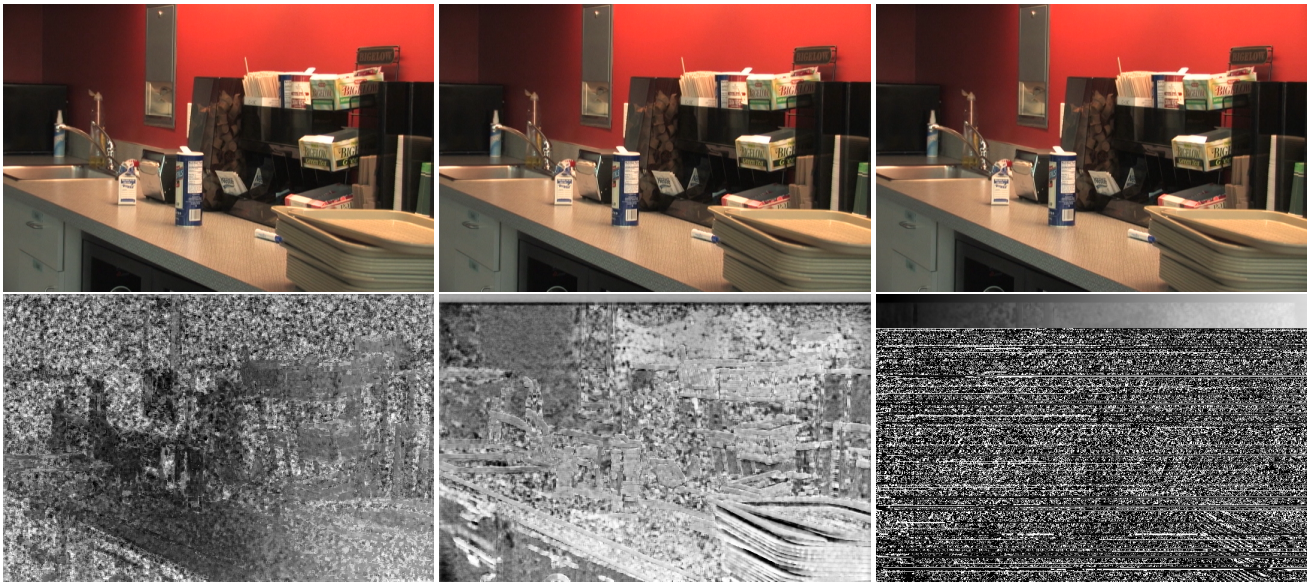

Figure 4: *Indoor Scene: Top row, three images from the input video and bottom row, relative depth maps computed using 10, 200, and 500 images, respectively.*

even for quite changing scenes and does this without any user intervention, external pre-calibration, or complex regularization procedures. The methods are computationally simple and fairly efficient.

Our results suggests several directions for future work. While we intentionally did not explicitly include spatial regularization when computing the depth map; we, nevertheless, have no doubt that our results could be improved by including a regularization model such as Graph Cuts or Belief Propagation. We plan to incorporate one of these methods in the future.

Another extension is deploying this in an automated camera setup. Our method lends well to an iterative update rule, by aligning new input frames to the initial existing coordinate system, and using incremental PCA [Ross et al. 2004] to update the rank-1 factorization. We believe it would be very interesting to have a camera that continually refines its depth model as it records more data.

References

AGARWAL, S., SNAVELY, N., SIMON, I., SEITZ, S. M., AND SZELISKI, R. 2009. Building rome in a day. In *Proc. Int. Conf. on Computer Vision*.

BAKER, S., SCHARSTEIN, D., LEWIS, J., ROTH, S., BLACK, M. J., AND SZELISKI, R. 2007. A database and evaluation methodology for optical flow. In *Proc. Int. Conf. on Computer Vision*.

CRIMINISI, A., REID, I. D., AND ZISSERMAN, A. 1998. Duality, rigidity and planar parallax. In *ECCV '98: Proceedings of the 5th European Conference on Computer Vision-Volume II*, Springer-Verlag, London, UK, 846–861.

HIRSCHMLLER, H., AND SCHARSTEIN, D. 2007. Evaluation of cost functions for stereo matching. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*.

HOIEM, D., EFROS, A., AND HEBERT, M. 2005. Geometric context from a single image. In *Proc. Int. Conf. on Computer Vision*.

HORN, B. K., AND SCHUNCK, B. G. 1980. Determining optical flow. Tech. rep., Cambridge, MA, USA.

LOWE, D. G. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision 60*, 91–110.

MATAS, J., CHUM, O., URBAN, M., AND PAJDLA, T. 2002. Robust wide baseline stereo from maximally stable extremal regions. In *British Machine Vision Conference*.

OKUTOMI, M., AND KANADE, T. 1991. A multi-baseline stereo. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*.

PRITCHETT, P., AND ZISSERMAN, A. 1998. Wide baseline stereo matching. In *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, IEEE Computer Society, Washington, DC, USA, 754.

PSARAKIS, E. Z., AND EVANGELIDIS, G. D. 2005. An enhanced correlation-based method for stereo correspondence with subpixel accuracy. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, IEEE Computer Society, Washington, DC, USA, 907–912.

ROSS, D. A., LIM, J., AND YANG, M.-H. 2004. Adaptive probabilistic visual tracking with incremental subspace update. In *Proc. Eighth European Conference on Computer Vision (ECCV 2004)*, T. Pajdla and J. Matas, Eds., vol. 2. Springer, 470–482.

SCHARSTEIN, D., AND SZELISKI, R. 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision 47*, 1-3, 7–42.

SEITZ, S. M., CURLESS, B., DIEBEL, J., SCHARSTEIN, D., AND SZELISKI, R. 2006. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 519–528.

SHIMIZU, M., AND OKUTOMI, M. 2005. Sub-pixel estimation error cancellation on area-based matching. *Int. J. Comput. Vision 63*, 3, 207–224.

SNAVELY, N., SEITZ, S. M., AND SZELISKI, R. 2006. Photo tourism: exploring photo collections in 3d. In *SIGGRAPH '06:*

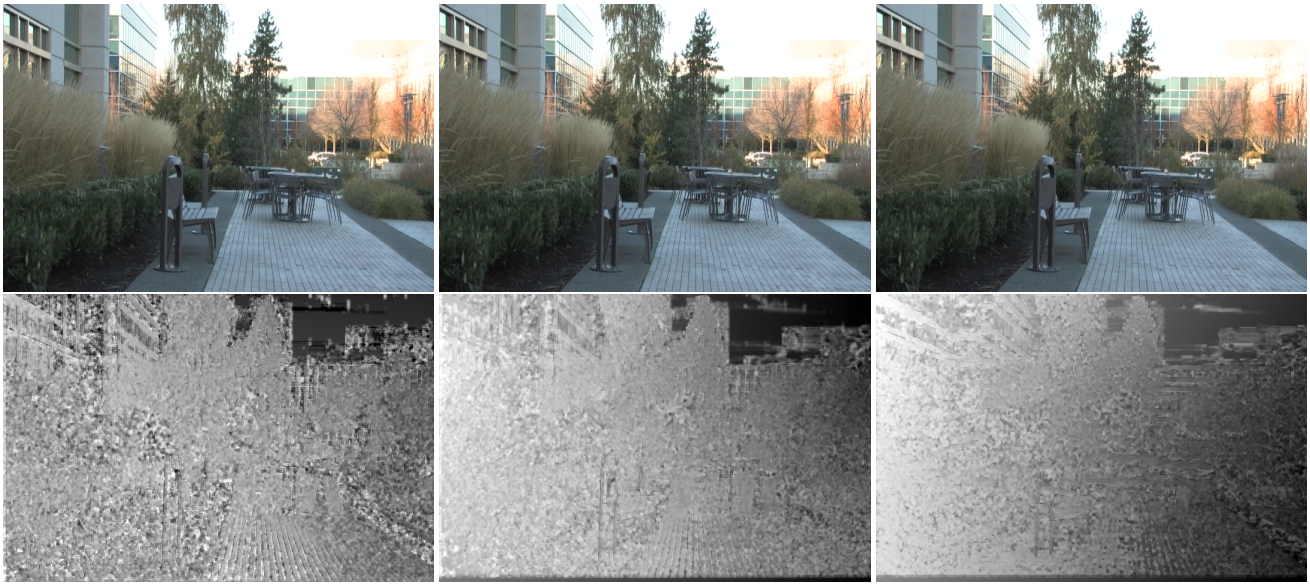

Figure 5: *Outdoor Scene: Top row, three images from the input video and bottom row, relative depth maps computed using 10, 200, and 500 images, respectively.*

ACM SIGGRAPH 2006 Papers, ACM, New York, NY, USA, 835–846.

TAKITA, K., MUQUIT, M., AOKI, T., AND HIGUCHI, T. 2004. A sub-pixel correspondence search technique for computer vision applications. In *IEICE Trans. Fundamentals E87-A*.

THEVENAZ, P., RUTTIMANN, U. E., AND UNSER, M. 1998. A pyramid approach to subpixel registration based on intensity. *Image Processing, IEEE Transactions on 7*, 1, 27–41.

TOMASI, C., AND KANADE, T. 1992. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision 9*, 2 (November), 137–154.

TOMBARI, F., MATTOCCIA, S., STEFANO, L. D., AND ADDIMANDA, E. 2008. Classification and evaluation of cost aggregation methods for stereo correspondence. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*.

TRIGGS, B., MCLAUCHLAN, P. F., HARTLEY, R. I., AND FITZGIBBON, A. W. 2000. *Bundle Adjustment – A Modern Synthesis*, vol. 1883.

VAISH, V., WILBURN, B., JOSHI, N., AND LEVOY, M. 2004. Using plane + parallax for calibrating dense camera arrays. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on 1*, 2–9.