Omni-directional Multi-baseline Stereo without Similarity Measures

Tomokazu Sato and Naokazu Yokoya
Graduate School of Information Science, Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0192, Japan
E-mail:{tomoka-s, yokoya}@is.naist.jp

Abstract

This paper proposes a novel method for estimating depth from long-baseline image sequences captured by a pre-calibrated moving omni-directional multi-camera system (OMS). Our idea for estimating an omni-directional depth map is very simple; only counting interest points in images is integrated with the framework of conventional multi-baseline stereo. Even by a simple algorithm, depth can be determined without computing similarity measures such as SSD and NCC that have been used for traditional stereo matching. The proposed method realizes robust depth estimation against image distortions and occlusions with lower computational cost than traditional multi-baseline stereo method. These advantages of our method are fit for characteristics of omni-directional cameras. In experiments, validity and feasibility of our algorithm are shown for both a synthetic and real outdoor scene.

1. Introduction

Depth map estimation from omni-directional images is one of very important problems in omni-directional computer vision, because depth information is used in a number of different applications such as 3-D reconstruction, surveillance, and new view synthesize. In this paper, we focus on the framework of multi-baseline stereo for moving camera system that is one of standard for depth estimation for a large number of image input.

The original multi-baseline stereo method was proposed by Okutomi and Kanade [1] for multiple image input. The multi-baseline stereo has such a very good feature that an arbitrary number of images can be simultaneously used for depth estimation. This increases the accuracy of depth estimation and decreases the ambiguity in stereo matching. After their work, several researchers have utilized and also extended the original multi-baseline stereo also for an omni-directional camera system [2, 3, 4, 5, 6].

Recently, some researchers have employed the multi-baseline stereo framework for a freely moving camera [3, 4, 5]. A freely moving video camera including a moving omni-directional camera is suitable for 3-D modeling of a large scale environment because it easily makes a long distance baseline between cameras. However, there exist some problems in multi-baseline stereo method for a moving video camera as follows:

(1) Changing patterns in video images: Free motion of video camera causes significant changes of looks for corresponding points in video frames. Traditional similarity measure such as SSD (Sum of Squared Differences), which has been used for the multi-baseline stereo, is weak for this effect.

(2) Occlusions: When a point on an object where depth should be estimated is occluded by other objects in a part of input video, the occluder gives a negative score to the score function of the multi-baseline stereo: SSSD (Sum of SSD). This negative score prevents the algorithm from obtaining correct estimation of depth map around occlusions.

(3) Computational cost: A large number of images consume the large amount of memory and computational resources. That is especially hard problem for an OMS that can take high-resolutional multiple images for each position of the system. Some patches for the problems (1) and (2) may also increase computational time.

To avoid these problems, we use interest points such as corners and cross points of edges in video images as shown in Figure 1. The framework of our depth estimation is basically the same as the original multi-baseline stereo except for a newly employed score function: TNIP (Total Number of Interest Points). The idea is based on the assumption that the corners of objects and cross points of texture edges in the 3-D space (3-D interest points) will appear in video images as 2-D interest points at the projected positions of the 3-D interest points. By searching a depth that maximizes the total number of 2-D interest points under epipolar constraint, the depth can be determined as a position of a 3-D interest point. A flow diagram of our algorithm is shown in

Figure 1: Examples of interest points.

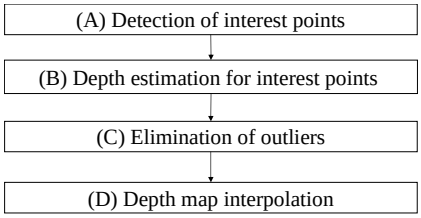


Figure 2: Flow diagram of dense depth map estimation.

Figure 2. First, interest points are detected for all the video images (A). Next, depths are estimated by multi-baseline stereo using TNIP (B). Outliers of estimated depths are rejected based on their confidences (C). Finally, dense depth maps are generated by interpolating sparse depth data (D).

By our method, the problems pointed out earlier can be solved; (1) interest points are strong for the effect of changing distortion, (2) the new score function TNIP is not significantly affected by occluders, (3) computational cost is drastically decreased because depth can be determined by only counting interest points. However there exists one restriction that we cannot estimate depths for non-interest points by using TNIP, that is not a critical problem for 3-D modeling and some other applications because usually 3-D interest points contain corners of the 3-D model. In most cases, depth interpolation is sufficient.

Note that TNIP based multi-baseline stereo approach may be felt similar to the feature based stereo methods for binocular [7, 8] and trinocular [9, 10] stereo matching that use edge and corner features as candidates of corresponding points. However, there is a notable difference that the feature based stereo methods need to use similarity measures to determine depths. In these methods, feature points are only used as candidates of corresponding points. Moreover, conventional feature based approaches are not designed to treat a large number of images simultaneously.

The rest of this paper is structured as follows. First, the original multi-baseline stereo method for a moving video camera is briefly described in Section 2. In Section 3, the new score function TNIP for multi-baseline stereo is proposed. Stages (A) to (D) in Figure 2 for estimating a dense depth map are detailed in Section 4. Experimental results with simulation and a real scene then show the validity and feasibility of the proposed method in Section 5. Finally, Section 6 describes conclusion and future work.

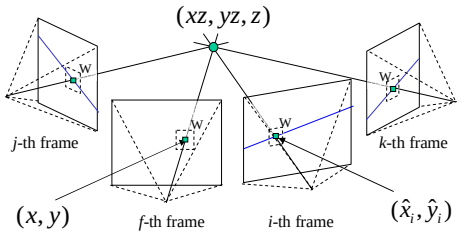


Figure 3: 3-D position of (x, y) with depth z and its projected line to each frame.

2. Multi-baseline stereo by using SSSD

In this section, firstly, coordinate systems of a general moving camera are defined. The principle of the multi-baseline stereo [1] using SSSD is then briefly summarized.

2.1. Definition of coordinate systems for moving camera

In the multi-baseline stereo, as shown in Figure 3, a depth z of a pixel (x, y) in the f-th frame is estimated by using images from the j-th to k-th frame $(j \leq f \leq k)$. In the following, for simplicity, we assume that the focal length is 1 and lens distortion effect has already been corrected by known intrinsic parameters. In this case, a 3-D position of (x, y) with depth z is represented as (xz, yz, z) in the camera coordinate system of the f-th frame. The 3-D position (xz, yz, z) is projected to the position $(\hat{x}_i, \hat{y}_i)$ in the image of the i-th frame by the following expression.

$$\begin{pmatrix} a\hat{x}_i \\ a\hat{y}_i \\ a \\ 1 \end{pmatrix} = \mathbf{M}_{fi} \begin{pmatrix} xz \\ yz \\ z \\ 1 \end{pmatrix}, \tag{1}$$

where a is a parameter, $\mathbf{M}_{fi}$ denotes a 4×4 transformation matrix from the camera coordinate system of the f-th frame to the camera coordinate system of the i-th frame. In the multi-baseline stereo, as shown in Figure 3, the point $(\hat{x}_i, \hat{y}_i)$ is constrained on the epipolar line, which is the projection of the 3-D line connecting the position (xz, yz, z) and the center of projection in the f-th frame onto the i-th frame.

2.2. Depth estimation using SSSD

In the traditional multi-baseline stereo, depth z of pixel (x, y) is determined by using the similarity measure SSD.

The SSD is computed as the sum of squared differences between two image patterns that have a certain size W. The SSD for (x, y) in the f-th frame and $(\hat{x_i}, \hat{y_i})$ in the i-th frame is defined using image intensity I as follows.

$$SSD_{fixy}(z) = \sum_{(u,v)\subseteq W} \left\{ I_f(x+u, y+v) - I_i(\hat{x_i}+u, \hat{y_i}+v) \right\}^2. \quad (2)$$

To evaluate the error of the depth z for all the input images, the SSD is summed up as follows:

$$SSSD_{fxy}(z) = \sum_{i=j}^{k} SSD_{fixy}(z). \quad (3)$$

The depth z is determined for each frame so as to minimize the SSSD function. Generally, to find a global minimum of the SSSD, depth z should be searched for all the depth range along a 3-D line from a reference pixel (x, y).

If the pixel (x, y) in the f-th frame is occluded by other objects in the i-th frame, $SSSD_{fxy}(z)$ for the true depth z is increased by the occluder because $SSD_{fixy}(z)$ gives a large error. Thus, to obtain a correct depth at such an occluded part, some other computationally expensive extensions should be added to the original multi-baseline stereo. For example, a modified SSSD can be computed by summing up only lower halves of SSDs [5, 11]. However, there still remains the computational cost problem and the image distortion problem.

3. Multi-baseline stereo by counting interest points

In this section, a new score function TNIP is defined to estimate depth z of pixel (x, y) using the multi-baseline stereo framework. Generally, feature points in a 3-D space, such as corners of objects and cross points of texture edges, appear as 2-D feature points in images at projected positions of the 3-D feature points. Such a 2-D feature points can be easily detected by interest operators such as Harris's [12] and Moravec's [13] operators.

In this study, depth z is determined so as to maximize the TNIP score function that is defined as follows.

$$TNIP_{fxy}(z) = \sum_{i=j}^{k} \sum_{(u,v)\subseteq W} H_i(\hat{x_i}+u, \hat{y_i}+v). \quad (4)$$

$$H_i(u, v) = \begin{cases} 1 \text{ ; interest point exists at} \\ \quad (u, v) \text{ in } i\text{-th frame}. \\ 0 \text{ ; otherwise} \end{cases} \quad (5)$$

The TNIP score represents the total number of interest points that exist in $(\hat{x}_i, \hat{y}_i)$ centered windows W for all the frames. Note that the size of W should be appropriately small because interest points are not detected at positions far from projected positions of (xz, yz, z) when there exists a feature point in 3-D space. It should be noted that the new score function TNIP is originated assuming the use of a large number of images. It shows its real ability when used for a long image sequence.

By using the TNIP instead of the SSSD function in the multi-baseline stereo, computational time can be drastically decreased because the time consuming process of comparing intensity patterns can be removed from the depth estimation. Moreover, maximizing the TNIP can obtain a correct depth even if projected image patterns of 3-D objects are much distorted by the motion of camera because positions of interest points are not affected by such a geometric transformation. The TNIP has another good feature that the TNIP is not significantly influenced by occluders because it counts only positive scores. These claims will be justified by experiments later.

4. Dense depth map estimation from an image sequence

This section details each stage of dense depth map estimation shown in Figure 2. In our method, first, interest points in all the input images are detected by Harris interest operator (A). Next, depth of each detected interest point is determined by the multi-baseline stereo framework with the TNIP score function (B). Outliers of estimated depths are then eliminated by using their confidences defined by considering the consistency among the results in multiple frames (C). Finally, dense depth maps are generated by interpolating sparse depth data (D).

4.1. Detection of interest points

In the first stage (A), interest points such as corners and cross points of edges are detected in input images by Harris interest operator [12]. Harris operator has been evaluated by Schmid as one of the best operators with respect to the repeatability rate of detected position under different geometric transformation [14][1].

Input images are first smoothed by a Gaussian operator. After that, deviations I_x and I_y of pixel intensity are computed for all the pixels in x and y directions, respectively. Feature quantity $F(\mathbf{x})$ of the pixel $\mathbf{x} = (x, y)$ is then computed as a minimum eigenvalue of the matrix $\mathbf{A}$ that is defined below.

$$\mathbf{A} = \sum_{\mathbf{x}\in W} \begin{pmatrix} I_x(\mathbf{x})^2 & I_x(\mathbf{x})I_y(\mathbf{x}) \\ I_x(\mathbf{x})I_y(\mathbf{x}) & I_y(\mathbf{x})^2 \end{pmatrix}. \quad (6)$$

[1] Although SIFT(Scale Invariant Feature Transform)[15] is one of good operator, detected positions by SIFT are usually far from edges. In this research, SIFT does not fit for our purpose.

$$F(\mathbf{x}) = \min(\lambda_1, \lambda_2), \tag{7}$$

where λ_1 and λ_2 are eigenvalues of the matrix $\mathbf{A}$. After computing the feature quantity $F(\mathbf{x})$ for all the pixels, local maxima of $F(\mathbf{x})$ are detected as interest points. In this stage, interest points are detected for all the frames in an image sequence.

4.2. Depth estimation for interest points

In the second stage (B), depths of all the interest points detected in the stage (A) are computed by maximizing the TNIP score function defined in Section 3. The depth z is searched to find a maximum TNIP in a given range of depth along a 3-D line from each reference pixel. By repeating the estimation of depth z for all the interest points in the input image sequence, sparse depth data can be acquired.

Note that, in this stage, any intensity images are not needed for depth estimation. Only 2-D positions of interest points and camera parameters should be stored to compute TNIP. It means that our method needs only 1/8 memory space to compute depth by compared with SSSD, if 8 bit grayscale images are assumed to be used for SSSD.

4.3. Elimination of outliers

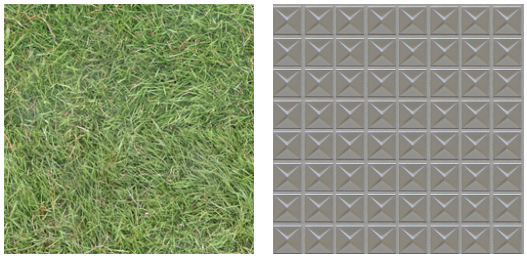
In the third stage (C), unreliable depths are rejected by cross validation approach for multiple image input. As shown in Figure 3, the depth z estimated by using TNIP relates interest points at projected positions of the 3-D feature point (xz, yz, z) in neighboring frames with one another. In this paper, the confidence C_p of an interest point p is defined by using this relation as follows.

$$C_p = \frac{\sum_{i \in \mathbf{L}_p} \{0; p \notin \mathbf{L}_i, 1; p \in \mathbf{L}_i\}}{|\mathbf{L}_p|}, \tag{8}$$

where $\mathbf{L}_p$ denotes a set of interest points in neighboring frames related by the depth z of the interest point p. C_p means the consistency among depth data obtained in the neighboring frames, and is computed as a ratio that interest points $i (i \in \mathbf{L}_p)$ are mutually related with the interest point p. If all the depths are correctly estimated, $\mathbf{L}_i (\forall i \in \mathbf{L}_p)$ should equal to $\mathbf{L}_p$. Only in this case, C_p becomes maximum value 1. In this stage, depth of the interest point p whose confidence C_p is lower than a given threshold is regarded as an outlier, and is deleted.

4.4. Depth map interpolation

Depth interpolation is necessary to generate dense depth maps because the TNIP-based multi-baseline stereo cannot estimate a depth at non-interest points. In this paper, we simply assume that the space among interest points is expected to be planar because interest points are corners of 3-D objects. In the final stage (D), input images are divided into triangle regions using interest points with Delaunay's triangulation method [16]. Depth data are then linearly interpolated for these triangles.



(a) plane 1 (b) plane 2

Figure 4: Textures of planes.

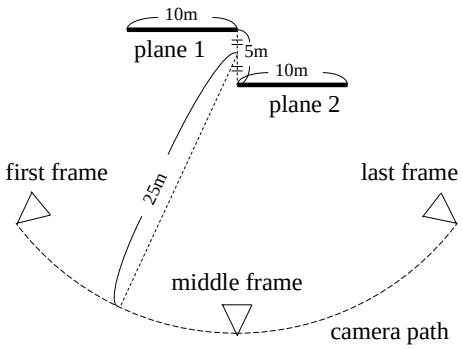
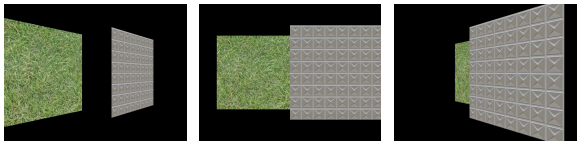


Figure 5: Layout of planes and camera path in simulation.



(a) first frame (b) middle frame (c) last frame

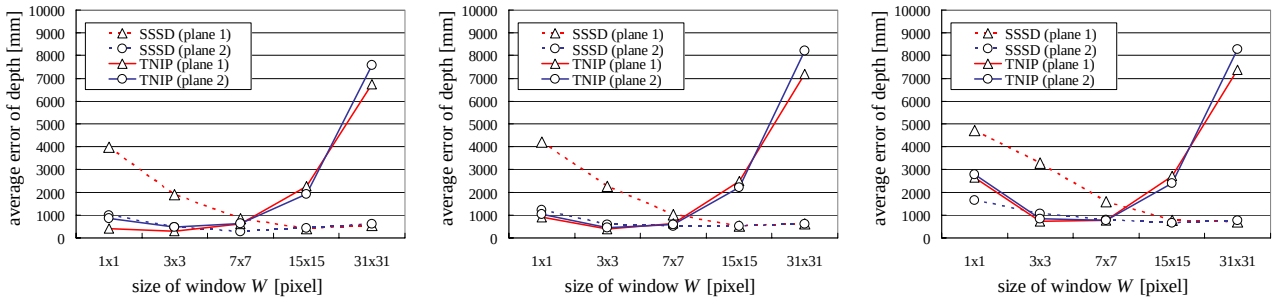
Figure 6: Sampled frames from 91 input images.

5. Experiments

We have carried out two kinds of experiments. One is concerned with the accuracy comparison between SSSD and TNIP in computer simulation. The other is conducted for estimating a dense depth map for a real outdoor environment.

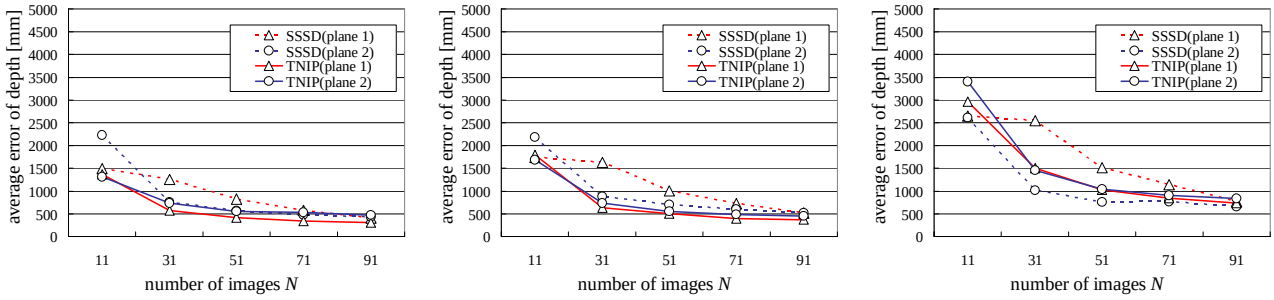
5.1. Comparison between SSSD and TNIP in computer simulation

This experiment is carried out to show the validity of the use of the TNIP score function for feature based multi-baseline stereo by comparing it with the traditional score function SSSD.



(a) σ=0.0 pixel (b) σ=1.0 pixel (c) σ=2.0 pixels

Figure 7: Average errors in depth estimation (varying window size and noise level).



(a) σ=0.0 pixel (b) σ=1.0 pixel (c) σ=2.0 pixels

Figure 8: Average errors in depth estimation (varying number of input images and noise level).

(1) Setup of simulation: In the experiment, two textured planes are located in a virtual environment, and a virtual camera takes an image sequence by moving the camera around these planes. To observe characteristics of TNIP and SSSD, two kinds of texture patterns are used for the planes, as shown in Figure 4. The layout of the planes and the motion path of virtual camera are illustrated in Figure 5. Totally 91 input images, some of which are shown in Figure 6, are taken by the moving camera whose motion draws a quarter circle as illustrated in Figure 5. By the motion of the camera, the plane 1 is occluded by the plane 2 in after half of the input images, and both textures are apparently distorted by the camera motion. This is one of very severe input for traditional multi-baseline stereo. To observe raw characteristic of both score functions, outliers are not deleted and depth data is not interpolated. To take into account camera calibration errors about intrinsic and extrinsic camera parameters, the Gaussian noise with standard deviation σ is added to projected positions of 3-D points. The searching range of depth for TNIP and SSSD is set as 3,000mm (near) to 35,000mm (far) in this simulation.

(2) Evaluation for varying window size and noise level: Firstly, depth of each interest point is estimated and evaluated for varying two parameters; the size of window W in Eqs. (2) and (4), and the noise level σ for projected position of 3-D point. In this experiment, all of the input images are used to compute TNIP and SSSD. Figure 7 denotes average errors of estimated depth values with respect to varying window sizes and noise levels. This experiment has given us an conclusion that 3×3 window is the best for TNIP and larger size windows have never give good scores because too large size windows cannot distinguish interest points in an image. On the other hand, in the case of SSSD, 15×15 window gives totally better results than other window sizes. Note that TNIP's results for plane 1 and 2 have approximately the same accuracy regardless of existence of occluded part in plane 1 in contrast with SSSD's. This means that depths by TNIP are not affected by occlusions and the difference of texture patterns.

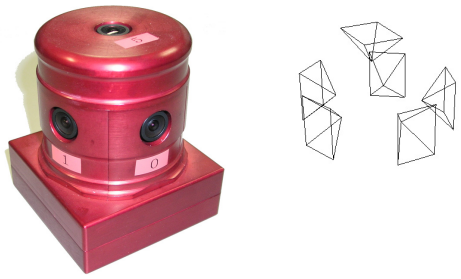
(3) Evaluation for varying number of input images and noise level: Next, to observe the behavior of each score function for the varying number of input images, 91 input images that are used to compute the score functions are thinned out to N images. According to the previous results, in this case, window sizes for TNIP and SSSD are fixed to 3×3 and 15×15, respectively. Figure 8 denotes average errors of estimated depths with respect to varying noise levels and number of input images. In all the results, average errors are monotonously decreased with respect to the number of input images. In this figure, we can confirm that TNIP and SSSD's behavior and accuracy for the varying number of input images are almost the same except that the accu-

racy for the SSSD's result on the occluded plane 1 is a little worse than the other's.

(4) Computational time: Table 1 indicates the average time to estimate a depth of a single pixel using all the 91 input images with respect to different window sizes. The computation time is measured by using a PC (CPU: Pentium-4 Xeon 3.20GHz dual, Memory: 2GB). Now, we can conclude that the computational cost of TNIP (3×3 window) is about 40 times cheaper than SSSD's (15×15 window) to achieve almost the same accuracy.

Table 1: Average computational time for estimating a depth of a single pixel [milli-seconds].

window size	1×1	3×3	7×7	15×15	31×31
SSSD	10.4	23.1	85.6	356.4	1529
TNIP	7.2	8.2	9.2	10.8	19.0



(a) appearance (b) view volume

Figure 9: Omni-directional multi-camera system: Ladybug.

5.2. Dense depth map estimation in an outdoor environment

In this experiment, an outdoor environment is captured by an Omni-directional Multi-camera System (OMS): Ladybug [17]. Figure 9 shows an appearance and view volume of Ladybug. This camera system has six radially located camera units and takes synchronized six image sequences at 15fps (resolution of each camera: 768×1024 pixels).

First, the outdoor environment was captured by the OMS as 3,000 images (500 frames). Figure 10 shows a sampled frame of six input image sequences. Intrinsic camera parameters including geometric relations among fixed camera units are calibrated in advance by using a marker board and a 3-D laser measure [18]. Extrinsic camera parameters of the input image sequences are estimated using bundle adjustment by tracking both a small number of feature landmarks of known 3-D positions and a large number of natural features of unknown 3-D positions in input images across adjacent camera units[19]. Figure 11 illustrates the recovered camera path that is used as an input for depth estimation. The curved line and pyramids denote the motion

Figure 10: Sampled frame of input image sequences.

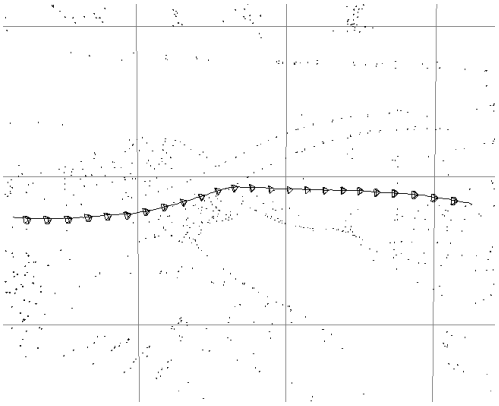
path of a camera unit and its posture at every 20 frames, respectively. The length of the camera path is approximately 29m. The accuracy of estimated camera path is evaluated as 50mm about camera position and 0.07degree about camera posture[19].

Next, omni-directional depth maps are actually estimated by the method described in Section 4. In the stage (A), interest points are detected in all the frames of six input image sequences by Harris operator. In this experiment, 1,750 interest points are detected on average in a single input image (10,500 points per frame).

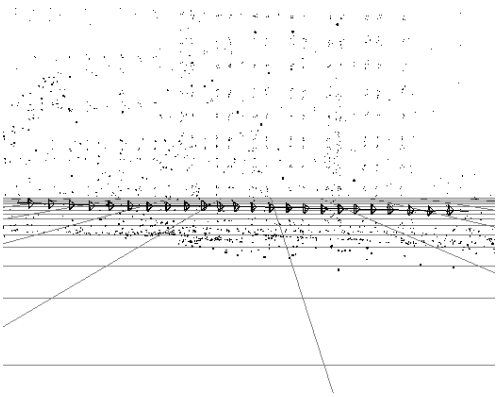
In the stage (B), depths of all the interest points detected in the stage (A) are estimated using the TNIP score function. In this stage, interest points in the $(f-100)$-th to the $(f+100)$-th frames at every 2 frames (606 images,101 frames) are used to estimate depth data of the f-th frame. The size of window W was set as 3×3 pixels according to the result of the computer simulation described in the previous section. The searching range to find a maximum TNIP in this stage is 1,000mm (near) to 80,000mm (far).

In the stage (C), low confidence depths are eliminated. The threshold for C_p defined in Eq. (8) was set as 0.5 in this experiment. Figure 12 shows the results of depth estimation for the images in Figure 10. In this figure, depth values are corded in intensity. Figure 13 indicates TNIP values of randomly selected six interest points in Figure 12. We can confirm from Figure 13 that each TNIP plot has a single apparent peak at a certain depth value and there are no other comparable peaks. This clearly shows that depth estimation can be easily achieved for these interest points.

Finally, omni-directional dense depth maps are generated in the stage (D). Figure 14 shows a panoramic image that is generated from six input images shown in Figure 10. Figure 15 shows the corresponding dense depth map. By



(a) top view



(b) side view

Figure 11: Camera path of OMS used for input (29m).

comparing these figures, we can confirm that a dense depth map is correctly computed for most parts of the input image. However, some incorrect depths are also observed around the boundaries between the buildings and the sky. These incorrect results are caused by depth interpolation over different objects. To improve the result, region information in input images should be considered in the triangulation stage (D).

6. Conclusion

In this paper, a novel multi-baseline stereo for a moving camera has been proposed, where depth can be determined by only counting the number of interest points. The proposed method shows its real ability when used for long-time and long-baseline image sequences and is congenial with omni-directional images. Our method has be proven to be robust against occlusions and image distortion problem that are caused by camera motion. Additionally, the computational cost for the proposed method is also cheaper than the method based on the traditional SSSD function. In experiments, these claims have been justified by using both

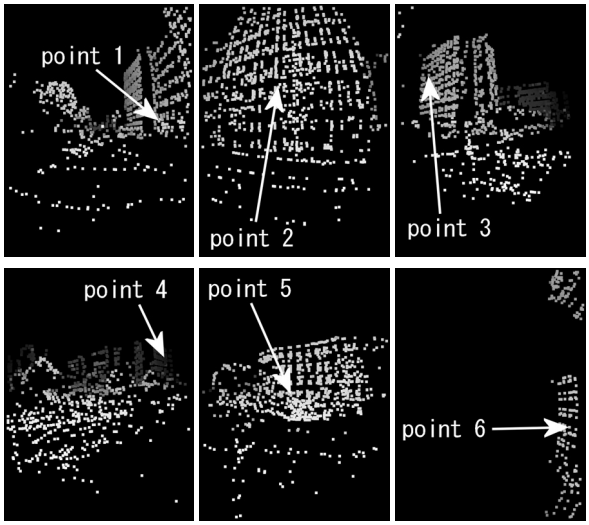
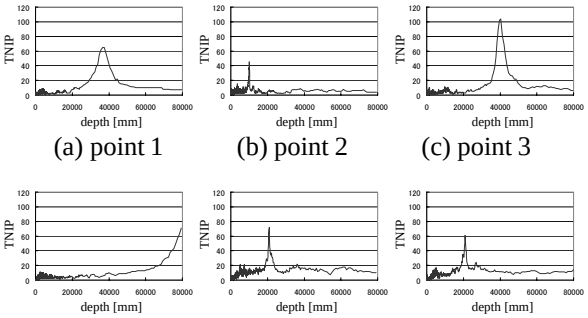


Figure 12: Result of depth estimation for interest points in Figure 10.



(a) point 1 (b) point 2 (c) point 3

(d) point 4 (e) point 5 (f) point 6

Figure 13: TNIP scores for searching depth.

synthetic and real image sequences. In future work, estimated depth maps will be integrated to reconstruct a 3-D model of a large outdoor environment.

References

[1] M. Okutomi and T. Kanade: "A Multiple-baseline Stereo," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 15, No. 4, pp. 353–363, 1993.

[2] S. B. Kang, J. A. Webb, C. Zitnick and T. Kanade: "A Multi-baseline Stereo System with Active Illumination and Real-time Image Acquisition," Proc. Int. Conf. on Computer Vision, pp. 88–93, 1995.

[3] S. B. Kang and R. Szeliski: "3-D Scene Data Recovery using Omnidirectional Multibaseline Stereo," Int. Journal of Computer Vision, Vol. 25, No. 2, pp. 167–183, 1997.

[4] W. Zheng, Y. Kanatsugu, Y. Shishikui and Y. Tanaka: "Robust Depth-map Estimation from Image Sequences with Pre-

Figure 14: Panoramic image generated from six images acquired by OMS.

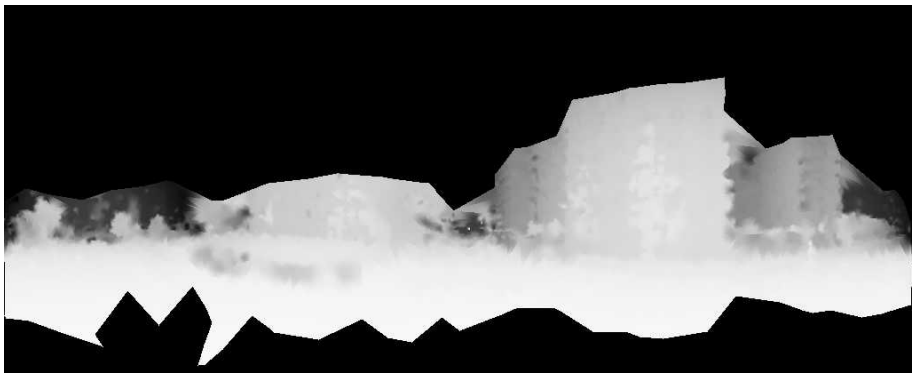


Figure 15: Generated dense depth map.

cise Camera Operation Parameters," Proc. Int. Conf. on Image Processing, Vol. II, pp. 764–767, 2000.

[5] T. Sato, M. Kanbara, N. Yokoya and H. Takemura: "Dense 3-D Reconstruction of an Outdoor Scene by Hundreds-baseline Stereo Using a Hand-held Video Camera," Int. Journal of Computer Vision, Vol. 47, No. 1-3, pp. 119–129, 2002.

[6] M. Okutomi, Y. Katayama and S. Oka: "A Simple Stereo Algorithm to Recover Precise Object Boundaries and Smooth Surface," Int. Journal of Computer Vision, Vol. 47, No. 1-3, pp. 261–273, 2002.

[7] H. H. Baker: "Edge Based Stereo Correlation," Proc. Image Understanding Workshop, pp. 168–175, 1980.

[8] W. E. L. Grimson: "Computational Experiments with a Feature-based Stereo Algorithm," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 7, No. 1, pp. 17–34, 1985.

[9] S. Pollard, M. Pilu, S. Hayes and A. Lorusso: "View Synthesis by Trinocular Edge Matching and Transfer," Image and Vision Computing, Vol. 18, No. 9, pp. 739–748, 2000.

[10] M. Agrawal and L. S. Davis: "Trinocular Stereo Using Shortest Paths and the Ordering Constraint," Int. Journal of Computer Vision, Vol. 47, No. 1-3, pp. 43–50, 2002.

[11] S. B. Kang, R. Szeliski and J. Chai: "Handling Occlusions in Dense Multi-view Stereo," IEEE Conf. on Computer Vision and Pattern Recognition, Vol.1, pp. 103–110, 2001.

[12] C. Harris and M. Stephens: "A Combined Corner and Edge Detector," Proc. Alvey Vision Conf., pp. 147–151, 1988.

[13] H. Moravec: "Towards Automatic Visual Obstacle Avoidance," Proc. Int. Joint Conf. on Artificial Intelligence, p. 584, 1977.

[14] C. Schmid, R. Mohr and C. Bauckhage: "Evaluation of Interest Point Detectors," Int. Journal of Computer Vision, Vol. 37, No. 2, pp. 151–172, 2000.

[15] D. G. Lowe: "Distinctive Image Features from Scale Invariant Keypoints," Int. Journal of Computer Vision, Vol. 60, No. 2, pp. 91–110, 2004.

[16] P. Heckbert Ed.: Graphics Gems IV, pp. 47–59, Academic Press, 1994.

[17] Point Grey Research Inc.: "Ladybug," http://www.ptgrey.com/.

[18] S. Ikeda, T. Sato and N. Yokoya: "High-resolution Panoramic Movie Generation from Video Streams Acquired by an Omnidirectional Multi-camera System," Proc. IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent System, pp. 155–160, 2003.

[19] T. Sato, S. Ikeda and N. Yokoya: "Extrinsic Camera Parameter Recovery from Multiple Image Sequences Captured by an Omni-directional Multi-camera System," Proc. European Conf. on Computer Vision, Vol. 2, pp. 326–340, 2004.