Dense 3-D Reconstruction of an Outdoor Scene by Hundreds-baseline Stereo Using a Hand-held Video Camera

Tomokazu Sato†, Masayuki Kanbara†, Naokazu Yokoya† and Haruo Takemura‡

†Graduate School of Information Science, Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0101, Japan
{tomoka-s, masay-ka, yokoya}@is.aist-nara.ac.jp
‡Cybermedia Center, Osaka University

Abstract

Three-dimensional (3-D) models of outdoor scenes are widely used for object recognition, navigation, mixed reality, and so on. Because such models are often made manually with high costs, automatic 3-D reconstruction has been widely investigated. In related work, a dense 3-D model is generated by using a stereo method. However, such approaches cannot use several hundreds images together for dense depth estimation because it is difficult to accurately calibrate a large number of cameras. In this paper, we propose a dense 3-D reconstruction method that first estimates extrinsic camera parameters of a hand-held video camera, and then reconstructs a dense 3-D model of a scene. In the first process, extrinsic camera parameters are estimated by tracking a small number of predefined markers of known 3-D positions and natural features automatically. Then, several hundreds dense depth maps obtained by multi-baseline stereo are combined together in a voxel space. So, we can acquire a dense 3-D model of the outdoor scene accurately by using several hundreds input images captured by a hand-held video camera.

1. Introduction

Three-dimensional (3-D) models of outdoor scenes are widely used for object recognition, navigation, mixed reality, and so on. Because such models are often made manually with high costs, automatic and dense 3-D reconstruction is desired. In the field of computer vision, there are many researches that reconstruct 3-D models from multiple images [1].

One of the major approaches to 3-D reconstruction is to use static stereo [2, 3, 4]. However, conventional methods cannot use a large number of images because it is difficult to calibrate a large number of cameras accurately. Therefore, these methods become sensitive to noise. Although many researchers often use a constraint of surface continuity to reduce noises, such an approach limits a target scene and may sometimes reduce accuracy of reconstruction.

One of other approaches is to use an image sequence that is called shape-from-motion [5, 6, 7, 8, 9, 10, 11]. The method can automatically recover camera parameters and 3-D positions of natural features by tracking 2-D positions of natural features in captured images. Factorization algorithm [5, 6, 7] is one of the well known shape from motion methods that can estimate a rough 3-D scene stably and efficiently by assuming an affine camera model. However, when the 3-D scene is not suitable for the affine camera model, estimated camera parameters are not reliable. Therefore, this method is not suitable for reconstructing a dense 3-D model by stereo method.

Some other methods of shape-from-motion are based on projective reconstruction method [8, 9, 10, 11]. Most of the methods reconstruct only a limited scene from a small number of images and are not designed to obtain a dense model. A method [11] which recovers camera parameters and a dense scene can reconstruct only a simple outdoor scene without occlusion from a small number of images. The method seems to be difficult to reconstruct a complex outdoor scene because it uses the constraint of surface continuity in dense depth estimation.

In order to reconstruct a complex outdoor scene densely and stably, we propose a new 3-D reconstruction method that first recovers extrinsic camera parameters of an input image sequence that consists of several hundreds images, and then reconstructs a dense model of a scene by combining several hundreds depth maps. In the first process, we use a camera parameter estimation method [12]. This method uses a small number of predefined markers of known 3-D positions and many natural features for stable and efficient estimation of extrinsic camera parameters. The first frame of the input image sequence must contain six or more markers, because generally extrinsic camera parameters can be determined by using a linear least-squares minimization method from at least six points whose 3-D positions in real

world and 2-D positions in the image are known. These predefined markers are not necessary to be visible throughout an input sequence because 3-D positions of natural features are detected in the process of estimation. It should be noted that we assume a perspective camera model and intrinsic camera parameters (focal length, pixel size, center of image, radial distortion factor coefficient) must be estimated in advance. Next, dense depth maps are computed by using an extended multi-baseline stereo method from hundreds images. Finally, several hundreds depth maps are combined together in a voxel space. The proposed method can reconstruct a complex outdoor scene densely and accurately by using several hundreds images of a long sequence without the constraint of surface continuity.

This paper is structured as follows. Section 2 describes a method of estimating extrinsic camera parameters of a hand-held video camera by tracking markers and natural features. In Section 3, we describe a method of dense depth estimation and integration of these dense data in a voxel space. Then, we demonstrate two experimental results of 3-D reconstruction from real outdoor image sequences to show the feasibility of the proposed method in Section 4. Finally, Section 5 describes conclusion and future work.

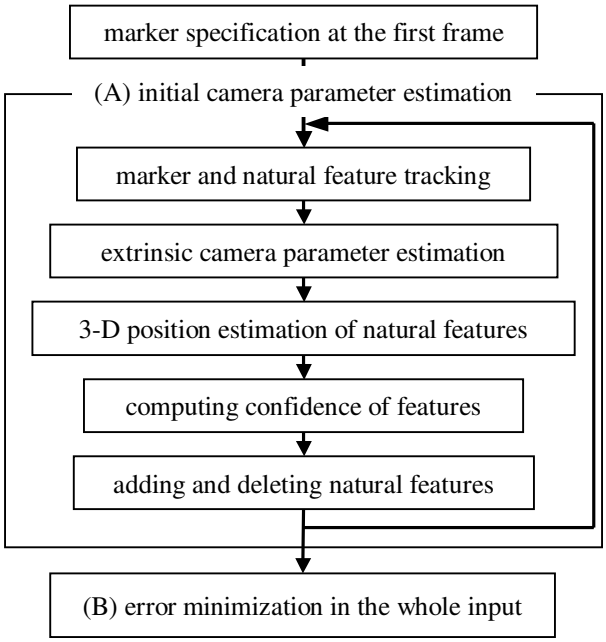


Figure 1: Flow diagram of camera parameter estimation.

2. Camera parameter estimation by tracking features

This section describes an extrinsic camera parameter estimation method which is based on tracking features (markers and natural features). Figure 1 shows the flow diagram of our algorithm. First, we must specify the positions of six or more markers in the first frame of input sequence, and extrinsic camera parameters in the first frame are estimated. Then extrinsic camera parameters in all the frames are determined by iterating the processes at each frame (A). Finally, extrinsic camera parameters are refined by minimizing the accumulation of estimation errors over the whole input (B). Using this approach, we can estimate extrinsic camera parameters efficiently and accurately by automatically adding and deleting features regardless of the visibility of initial markers.

2.1 Initial camera parameter estimation in each frame

By iterating the following processes from the first frame to the last frame, initial extrinsic camera parameters and 3-D positions of natural features are determined.

2.1.1 Marker and natural feature tracking

Tracking natural features usually suffers from two problems: One is that a center of tracked natural feature drifts because of accumulation of tracking error (a), the other is that a natural feature is tracked incorrectly when a similar image pattern exists near by the feature (b). To solve the problem (a), we employ Harris's interest operator [13, 14] to detect corners or cross-points of edges in the input images. Local maxima of this operator are used as candidate positions of tracking features. For the problem (b), tentative camera parameters computed by robust estimation are used to limit a searching region for natural feature tracking.

The feature tracking procedure consists of the following five steps for the f-th frame ($f \geq 2$) of an image sequence.

(1) The markers used in the $(f-1)$-th frame are searched in the f-th frame by using predefined color and shape information.

(2) Every feature in the $(f-1)$-th frame is tentatively matched with candidate feature points in the f-th frame which exist inside of a searching window placed around the feature position in the $(f-1)$-th frame by using a standard template matching. These candidate feature points are determined by selecting the local maxima of the measures computed by Harris's interest operator.

(3) The robust estimation is started. At the i-th iteration, first, n features $\mathbf{P}_i = \{p_{i1}, p_{i2}, \ldots, p_{in}\}$ are randomly sampled from the tentatively tracked natural features in Step (2), and temporary camera parameter $\hat{\mathbf{M}}_i$ is estimated using $\mathbf{P}_i$. Then, the median RM_i of re-projection

errors R_{ifp} is computed for estimated temporary camera parameter $\hat{\mathbf{M}}_i$. The re-projection error of feature p is defined as the squared distance between the tracked position $\mathbf{x}_{fp}$ and the projected position $\hat{\mathbf{x}}_{fp}$ of 3-D position $\mathbf{S}_p$ that is already estimated until the previous frame. The re-projection error R_{ifp} and the median RM_i of R_{ifp} are defined by the following equations.

$$R_{ifp} = |\mathbf{x}_{fp} - \hat{\mathbf{x}}_{fp}|^2. \quad (1)$$

$$RM_i = median(R_{if1}, R_{if2}, \ldots, R_{ifm}). \quad (2)$$

where m is a number of natural features in the f-th frame. The algorithm of estimating the camera parameter $\hat{\mathbf{M}}_i$ from tracked features $\mathbf{P}_i$ is described in Section 2.1.2.

(4) After g times iteration of Step (3), the tentative camera parameter $\bar{\mathbf{M}}_f$ for limiting a searching region is selected from temporary camera parameters ($\hat{\mathbf{M}}_1, \cdots, \hat{\mathbf{M}}_g$) by minimizing the following LMedS criterion.

$$LMedS = min(RM_1, RM_2, \ldots, RM_g). \quad (3)$$

(5) The features at the $(f-1)$-th frame are bound to the candidate positions in the f-th frame by searching the limited searching window. The center of the limited searching window is a projected position of $\mathbf{S}_p$ by camera parameter $\bar{\mathbf{M}}_f$. Note that the size of this searching window should be smaller than the size of window used in Step (2).

2.1.2 Extrinsic camera parameter estimation

In this section, extrinsic camera parameters are estimated by using 2-D positions of features in the image and 3-D positions of features in real world. In the proposed method, the re-projection error defined in Eq. (1) is used as a measure for estimation error. The camera parameter $\mathbf{M}_f$ at the f-th frame is estimated by minimizing the estimation error E_f defined as follows:

$$E_f = \sum_p W_{fp} R_{fp}, \quad (4)$$

where W_{fp} is a weighting coefficient for the feature p at the f-th frame and is computed by considering the confidence that is described in Section 2.3. In this paper, we assume that a camera parameter has six degrees of freedom (camera posture and position) and its coordinate system is an orthogonal coordinate system.

Since estimating camera parameters is a non-linear minimization problem, there exist problems concerning local minima and calculation cost. To avoid these problems, in the first step, an initial camera parameter $\tilde{\mathbf{M}}_f$ is estimated by a linear least-squares minimization method. Note that $\tilde{\mathbf{M}}_f$ has twelve degrees of freedom. Next, the estimated camera parameter $\tilde{\mathbf{M}}_f$ is linearly adjusted to reduce the degree of freedom to six by assuming that the direction of optical axis is correctly estimated. Finally, $\mathbf{M}_f$ is determined so as to minimize E_f by using a gradient descent method from the adjusted camera parameter. Because the initial camera parameter is expected to be close to the true camera parameter, the estimation error E_f could be globally minimized.

2.1.3 3-D Position estimation of natural features

The 3-D position $\mathbf{S}_p$ of the natural feature p in real world is estimated by using multiple frames from the tracking started frame of the feature p to the current frame. The position $\mathbf{S}_p$ is computed by minimizing a sum of squared distances between $\mathbf{S}_p$ and straight lines in 3-D that connect the centers of projection and positions $\mathbf{x}_{fp}$ of feature p in used frames f as shown in Figure 2.

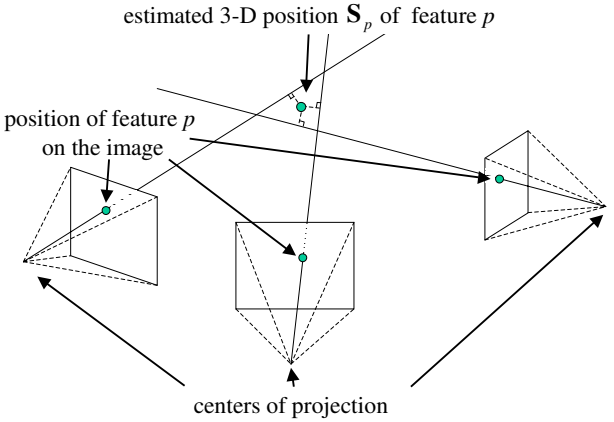


Figure 2: Estimating 3-D position of natural feature in real world.

2.1.4 Computing confidences of features

The confidences of features computed in the current frame are used for a weighting coefficient of camera parameter estimation in the next frame and for a measure of deleting natural features in the current frame. The tracked position $\mathbf{x}_{fp}$ of feature p in the f-th frame does not perfectly correspond to the re-projected position $\hat{\mathbf{x}}_{fp}$ because of tracking error. We assume that the distribution of the tracking errors can be approximated by a Gaussian probability density function. The probability that $\mathbf{x}_{fp}$ corresponds to the true position is represented as follows:

$$p(\mathbf{x}_{fp}) = \frac{1}{2\pi\sigma_p^2} exp(-\frac{|\mathbf{x}_{fp} - \hat{\mathbf{x}}_{fp}|^2}{2\sigma_p^2}). \quad (5)$$

The total probability P_f for all the features in the f-th frame is given by the following equation.

$$P_f = \prod_p p(\mathbf{x}_{fp}). \tag{6}$$

The camera parameter $\mathbf{M}_f$ that maximizes the above P_f is obtained by minimizing

$$EM_f = \sum_p \frac{|\mathbf{x}_{fp} - \hat{\mathbf{x}}_{fp}|^2}{2\sigma_p^2}, \tag{7}$$

where σ_p^2 is computed by re-projection errors up to the $(f-1)$-th frame. Then, the confidence W_{fp} of feature p that is tracked from the $(f-k)$-th to the $(f-1)$-th frames is defined by comparing Eqs. (4) and (7) as follows:

$$W_{fp} = \frac{1}{2\sigma_p^2} = \frac{k}{2} \left\{ \sum_{i=f-k}^{f-1} |\mathbf{x}_{ip} - \hat{\mathbf{x}}_{ip}|^2 \right\}^{-1}. \tag{8}$$

2.1.5 Addition and deletion of natural features

Feature candidates that satisfy all the following conditions are added to the set of natural features at every frame.

- The confidence is over a given threshold.
- The matching error is less than a given threshold.
- The output value of Harris's operator is more than a given threshold.
- The maximum angle between lines that connect the estimated 3-D position of the feature candidate and center of projections from the tracking started frame to the current frame is more than a given threshold.

On the other hand, natural features which satisfy at least one of the following conditions are considered to be unreliable and are deleted from the set of natural features at every frame.

- The confidence is under a given threshold.
- The matching error is more than a given threshold.

2.2 Error minimization in the whole input

By using the method described above, the camera parameters and the 3-D positions of natural features can be estimated over the whole frames. However, the accumulation of estimation error occurs. Therefore, in the final step, the accumulation of estimation error is minimized over the whole input. The accumulated estimation error E is given by the sum of re-projection errors as in Eq. (9) and is minimized with respect to the camera parameters $\mathbf{M}_f$ and the 3-D positions $\mathbf{S}_p$ of natural features over the whole input.

$$E = \sum_f \sum_p W_p |\mathbf{x}_{fp} - \hat{\mathbf{x}}_{fp}|^2. \tag{9}$$

The camera parameters and feature positions that are already estimated by earlier processes for each frame are used for initial values. W_p is a weighting coefficient for the feature p in the final frame of the image sequence. Note that, when the feature p is deleted in the f-th frame, $W_{(f-C)p}$ is used instead of W_p, and the positions of feature p from the $(f-C)$-th frame to the f-th frame are not used for this optimization, where C is a constant, since the features during the period are considered to be unreliable. Because the initial values of parameters are considered to be close to the true values, the error E is expected to be globally minimized efficiently by a standard gradient descent method.

3. Dense 3-D reconstruction by hundreds-baseline stereo

In this section, we describe a dense 3-D reconstruction method using camera parameters estimated by the method described in Section 2. First, a dense depth map for each image is computed by using a multi-baseline stereo method, then a 3-D model is reconstructed by combining obtained dense depth maps in a voxel space.

3.1. Dense depth estimation by multi-baseline stereo

A depth map is computed for each frame by using a multi-baseline stereo technique [15]. Depth value z of pixel (x, y) in the f-th frame is computed by using the k-th to the l-th frames $(k \le f \le l)$ around the f-th frame. In the following expression, we assume the focal length as 1 for simplicity. As shown in Figure 3, the 3-D position of the pixel (x, y) can be expressed by (xz, yz, z), and we can define the projected position $(\hat{x}_j, \hat{y}_j)$ of the 3-D position (xz, yz, z) onto the j-th frame $(k \le j \le l)$ as follows:

$$\begin{pmatrix} a\hat{x}_j \\ a\hat{y}_j \\ a \\ 1 \end{pmatrix} = \mathbf{M}_j \mathbf{M}_f^{-1} \begin{pmatrix} xz \\ yz \\ z \\ 1 \end{pmatrix}, \tag{10}$$

where a is a parameter. As shown in Figure 3, the point $(\hat{x}_j, \hat{y}_j)$ is constrained on the projected line of the 3-D line connecting the position (xz, yz, z) and the center of projection in the f-th frame. In the multi-baseline method, SSD (Sum of Squared Differences) is employed as an error function, that is computed as the sum of squared differences between the window W in the f-th frame centered at (x, y) and that in the j-th frame centered at $(\hat{x}_j, \hat{y}_j)$. We define the SSD

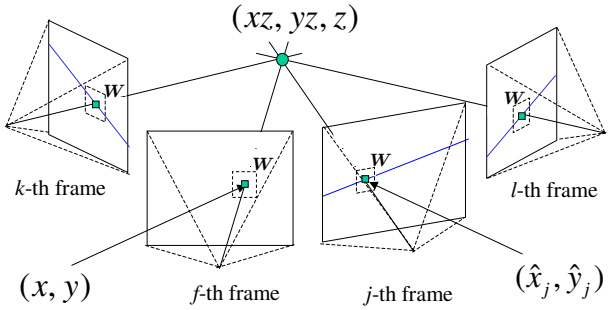


Figure 3: A 3-D position of a pixel (x, y) and its projection onto successive image planes.

function for the j-th frame in Eq. (11) using RGB components (I_R, I_G, I_B).

$$SSD_{fj}(x, y, o_x, o_y) = \sum_{(u-o_x, v-o_y) \subseteq W} \{(I_{Rf}(x+u, y+v) - I_{Rj}(\hat{x}_j + u, \hat{y}_j + v))^2 + (I_{Gf}(x+u, y+v) - I_{Gj}(\hat{x}_j + u, \hat{y}_j + v))^2 + (I_{Bf}(x+u, y+v) - I_{Bj}(\hat{x}_j + u, \hat{y}_j + v))^2\}, \quad (11)$$

where o_x and o_y are offsets of the window W for x and y axes, respectively.

In the multi-baseline stereo method, the depth z of (x, y) is determined so as to minimize the SSSD (Sum of SSD) from the k-th frame to the l-th frame. We define a modified SSSD in Eq. (12) using the median of SSD because the template of window W in the f-th frame may be occluded in other frames.

$$SSSD_f(x, y, o_x, o_y) = \sum_{j=k}^{l} \begin{cases} SSD_{fj}(x, y, o_x, o_y); \\ \quad SSD_{fj}(x, y, o_x, o_y) \leq T \text{ and } |j - f| > D, \\ 0; \quad \text{otherwise.} \end{cases} \quad (12)$$

where,

$$T = median(SSD_{fk}(x, y, o_x, o_y), \cdots, SSD_{f(f-D-1)}(x, y, o_x, o_y), SSD_{f(f+D+1)}(x, y, o_x, o_y) \cdots, SSD_{fl}(x, y, o_x, o_y)). \quad (13)$$

Note that images from the $(f-D)$-th frame to the $(f+D)$-th frame are not used for computing SSSD, because baselines in these frames are not long enough to estimate depth stably. Multiple centered window approach [16] is also used to reduce estimation errors around occlusion boundaries. Then SSSD is extended to SSSDM as follows:

$$SSSDM_f(x, y) = \min_{(u,v) \subseteq W} (SSSD_f(x, y, u, v)). \quad (14)$$

We can estimate the depth value $z(x, y)$ correctly by minimizing SSSDM unless the pixel (x, y) is occluded in more than $(l - k - 2D)/2$ frames. Additionally, we avoid a local minimum problem and achieve stable depth estimation using a multiscale approach [4]. Note that we use the linear interpolation to compute the depth value z in the regions without informative textures because the confidence of estimated z is low in such regions.

3.2. 3-D model reconstruction in a voxel space

In this paper, a 3-D model is reconstructed in a voxel space by combining several hundreds dense depth maps. In the voxel space, each voxel has two values A and B which are voted by already estimated depth values and camera parameters. As shown in Figure 4, both A and B are voted when the voxel is projected onto a pixel (x, y) of an image. Value A is voted if depth of the voxel in camera coordinate system is equal to z of (x, y). On the other hand, value B is voted when depth of the voxel is equal to or less than z of (x, y). We use the ratio A/B as a normalized voting value. A 3-D model is then reconstructed by selecting the voxel whose A/B is more than a given threshold. Note that the color of the voxel is decided by computing a mean color of pixels that have been voted to the value A of the voxel.

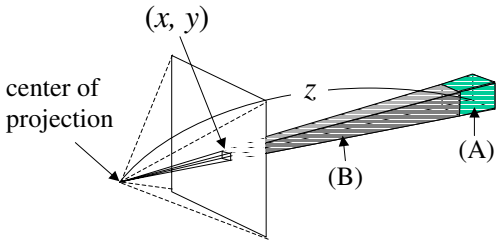


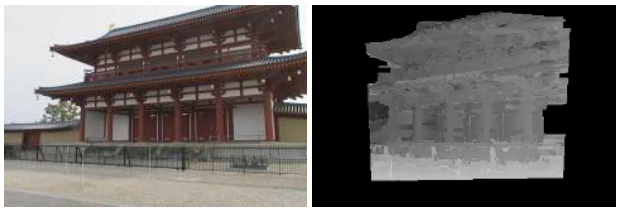
Figure 4: Voxel voting by a pixel (x, y) whose depth value is z. (A) is a region whose values A and B are voted. (B) is a region whose value B is voted.

4. Experiments

We have conducted two experiments: One is a model reconstruction of a single building and the other is a reconstruction of a street scenery. Both scenes are complex and have many occlusions. In both experiments, we use a hand-held CCD camera (Sony VCL-HG0758) with a wide conversion lens (Sony VCL-HG0758). The intrinsic camera parameters are estimated by Tsai's method [17] in advance.

4.1. Reconstruction of building

In this experiment, we captured a single building by walking around the building like an arc viewing it at the center of image. Figure 5(a) shows a sampled sequence of images that contain physically reconstructed Suzaku-mon Gate, whose original construction was made approximately 1300 years ago in Nara, the ancient capital of Japan. This image sequence lasts 40 seconds and has 599 frames (720×480 pix-



100th frame

180th frame

260th frame

340th frame

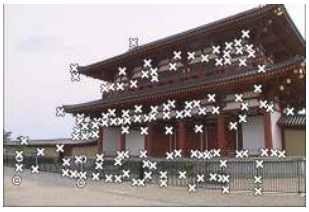
420th frame

499th frame

(a) Input images (b) Dense depth maps

Figure 5: Input images and estimated dense depth maps (Suzaku-mon).

first frame



200th frame

400th frame

599th frame

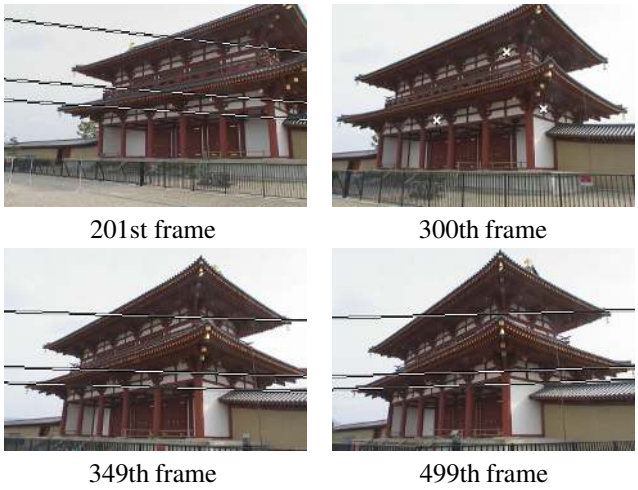
Figure 6: Results of feature tracking (Suzaku-mon).

els, progressive scan). Figure 6 shows results of feature tracking. White circles and crosses represent the tracked markers and natural features, respectively. Colored markers are attached to the top and bottom of three poles, and these poles are stood in the front of the target building. We specified these six markers in the first frame image and the markers are tracked automatically using predefined shape and color information. As shown in Figure 6, natural features are successfully detected at corners and cross-points of edges. We also confirmed that most of natural features are tracked stably.

In this experiment, a dense depth map of the f-th frame is obtained by using every two frames from the $(f-100)$-th to the $(f+100)$-th frames excluding the $(f-15)$-th to the $(f+15)$-th frames. Figure 7 shows projected lines of the 3-D lines connecting three white crosses and the center of projection in the 300-th frame. We can observe that camera parameters are correctly estimated, because the cross marked points of the building are correctly projected onto the lines as shown in Figure 7.

Figure 5(b) shows computed dense depth maps in which depth values are coded in intensity. It is confirmed that correct depth values are obtained for most part of the images. However there exist some incorrect depth values between a column and a wall of the building because there are no textures around the wall of the building. The linear interpolation is used for determining depth values in these areas.

Figure 8 shows a 3-D model with textures obtained by combining 399 dense depth maps together in the way of voxel voting that is described in Section 3.2. In Figure 8, the estimated camera path and posture are superimposed as curved lines and quadrilateral pyramids, respectively. In this experiment, the voxel space is constructed of 10cm cube voxels. A wall behind a column of the building is reconstructed even if the wall is occluded from time to time.



201st frame 300th frame

349th frame 499th frame

Figure 7: Projected lines of specified points in multi-baseline stereo method (Suzaku-mon).

We also observe that some positions are holed because these pixels are not visible enough for sufficient precision in the image sequence.

4.2. Reconstruction of street scenery

In this experiment, a street is captured as shown in Figure 9(a) by the CCD camera put on a slowly moving car. This image sequence lasts 19 seconds and has 284 frames (720×480 pixels, progressive scan). Figure 10 shows results of feature tracking. White circles and crosses represent the tracked markers and natural features, respectively. The 3-D positions of the markers are measured by a total station (Leica TCR307JS), and seven markers are tracked manually in the images in advance. The curved lines in Figure 11 indicate the camera path and the quadrilateral pyramids indicate the camera postures drawn at every 30 frames.

A dense depth map of the f-th frame is obtained by using 30 frames from the $(f + 6)$-th to the $(f + 35)$-th frames. As shown in Figure 9(b), it is confirmed that correct depth values are obtained for most part of the images even around the occlusion edges. However, there exist some incorrect depth values at the right of the trees because these pixels are occluded by the trees during over 15 frames. The depth values are also incorrect around the right edge of the images because the disparities of these regions are too small to estimate the depth.

Figure 11 shows a 3-D model with textures obtained by combining 249 dense depth maps together. In this experiment, the voxel space is constructed of 10cm cube voxels. Note that many parts of walls are holed around the windows of the buildings. We confirmed that it is difficult to reconstruct the reflective objects.

5. Conclusion

In this paper, a dense 3-D reconstruction method from a monocular image sequence captured by a hand-held video camera is proposed. In this method, first, extrinsic camera parameters are estimated over the whole input sequence by tracking both markers and natural features. Then, at each frame, a dense depth map is computed by the multi-baseline stereo using already estimated camera parameters. Finally, a 3-D model is reconstructed by combining hundreds dense depth maps in a voxel space.

In experiments, the dense 3-D scene reconstruction is accomplished for long image sequences captured in complex outdoor environments successfully with stable camera parameter estimation and dense depth estimation. However, we observe that some parts of the reconstructed model are holed. In future work, more accurate model reconstruction will be explored by using the confidence of depth value. Integration of 3-D models from multiple image sequences should further be investigated for obtaining a complete surface model.

Acknowledgments
This work was supported in part by Internet Systems Research Laboratories, NEC Corporation.

References

[1] N. Yokoya, T. Shakunaga and M. Kanbara: "Passive Range Sensing Techniques: Depth from Images," IEICE Trans. Inf. and Syst., Vol. E82-D, No. 3, pp. 523–533, 1999.

[2] S. T. Barnard and M. A. Fischler: "Computational Stereo," ACM Computing Survays, Vol. 14, No. 4, pp. 553–572, 1982.

[3] Y. Ohta and T. Kanade: "Stereo by Intra- and Inter- Scanline Search Using Dynamic Programming," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. PAMI-7, No. 2, pp. 139–154, 1985.

[4] N. Yokoya: "Surface Reconstruction Directly from Binocular Stereo Images by Multiscale-multistage Regularization," Proc. 11th Int. Conf. on Pattern Recognition, Vol. I, pp. 642–646, 1992.

[5] C. Tomasi and T. Kanade: "Shape and Motion from Image Streams under Orthography: A Factorization Method," Int. Journal of Computer Vision, Vol. 9, No. 2, pp. 137–154, 1992.

[6] J. Poleman and T. Kanade: "A Paraperspective Factorization Method for Shape and Motion Recovery," Tech. Rep. CMU-CS-93-219, Carnegie-Mellon Univ., 1993.

[7] D. D. Morris and T. Kanade: "A Unified Factorization Algorithm for Points, Lines Segments and Planes with Uncertainty Models," Proc. 6th Int. Conf. on Computer Vision, pp. 696–702, 1998.

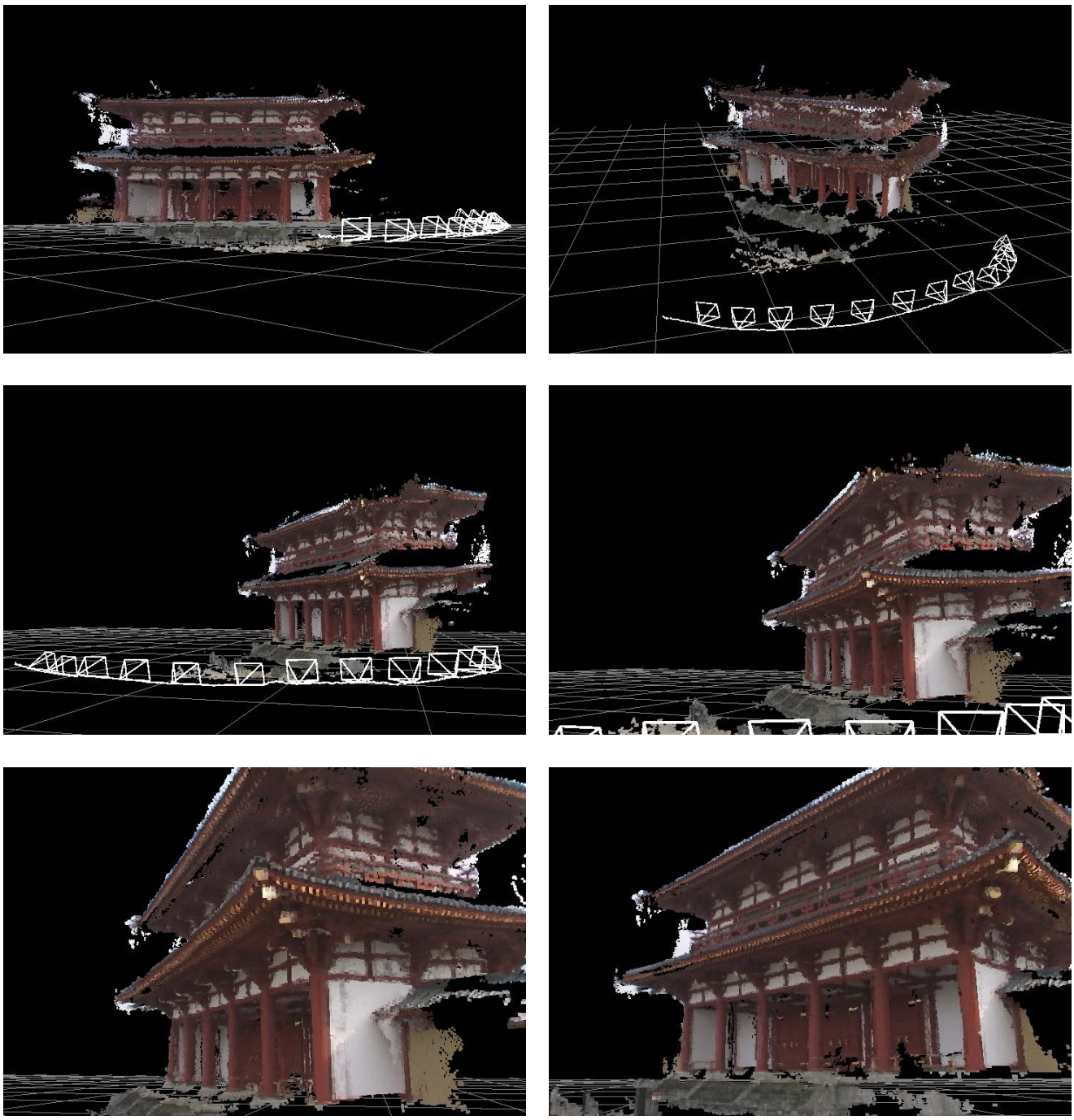
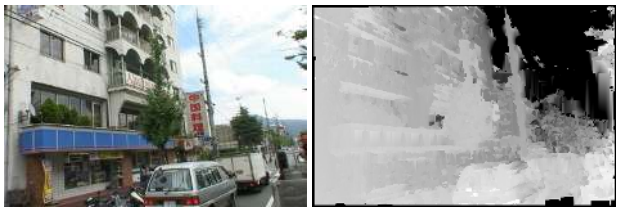


Figure 8: Results of dense outdoor scene recovery as well as estimated camera positions and postures (Suzaku-mon).



first frame

50th frame

100th frame

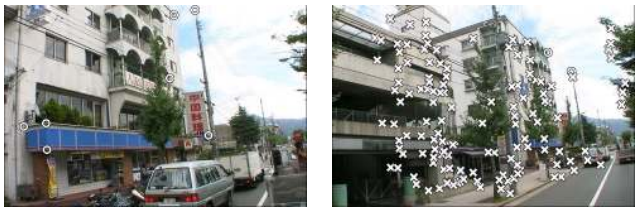
150th frame

200th frame

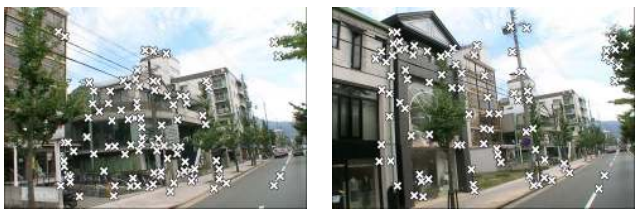
249th frame

(a) Input images (b) Dense depth maps

Figure 9: Input images and estimated dense depth maps (street scenery).



first frame 95th frame



190th frame 284th frame

Figure 10: Results of feature tracking (street scenery).

[8] P. Beardsley, A. Zisserman and D. Murray: "Sequential Updating of Projective and Affine Structure from Motion," Int. Jour. of Computer Vision, Vol. 23, No. 3, pp. 235–259, 1997.

[9] H. S. Sawhney, Y. Guo, J. Asmuth and R. Kumar: "Multi-view 3D Estimation and Application to Match Move," Proc. IEEE Workshop on Multi-view Modeling and Analysis of Visual Scenes, pp. 21–28, 1999.

[10] G. Roth and A. Whitehead: "Using Projective Vision to Find Camera Positions in an Image Sequence," Proc. 13th Int. Conf. on Vision Interface, pp. 87–94, 2000.

[11] M. Pollefeys, R. Koch, M. Vergauwen, A. A. Deknuydt and L. J. V. Gool: "Three-dimentional Scene Reconstruction from Images," Proc. SPIE, Vol. 3958, pp. 215–226, 2000.

[12] T. Sato, M. Kanbara, H. Takemura and N. Yokoya: "3-D Reconstruction from a Monocular Image Sequence by Tracking Markers and Natural Features," Proc. 14th Int. Conf. on Vision Interface, pp. 157–164, 2001.

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

[14] C. Schmid, R. Mohr and C. Bauckhage: "Comparing and Evaluating Interest Points," Proc. 6th Int. Conf. on Computer Vision, pp. 230–235, 1998.

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

[16] R. Kumar, H. S. Sawhney, Y. Guo, S. Hsu and S. Samarasekera: "3D Manipulation of Motion Imagery," Proc. Int. Conf. on Image Processing, pp. 17–20, 2000.

[17] R. Y. Tsai: "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision," Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 364–374, 1986.

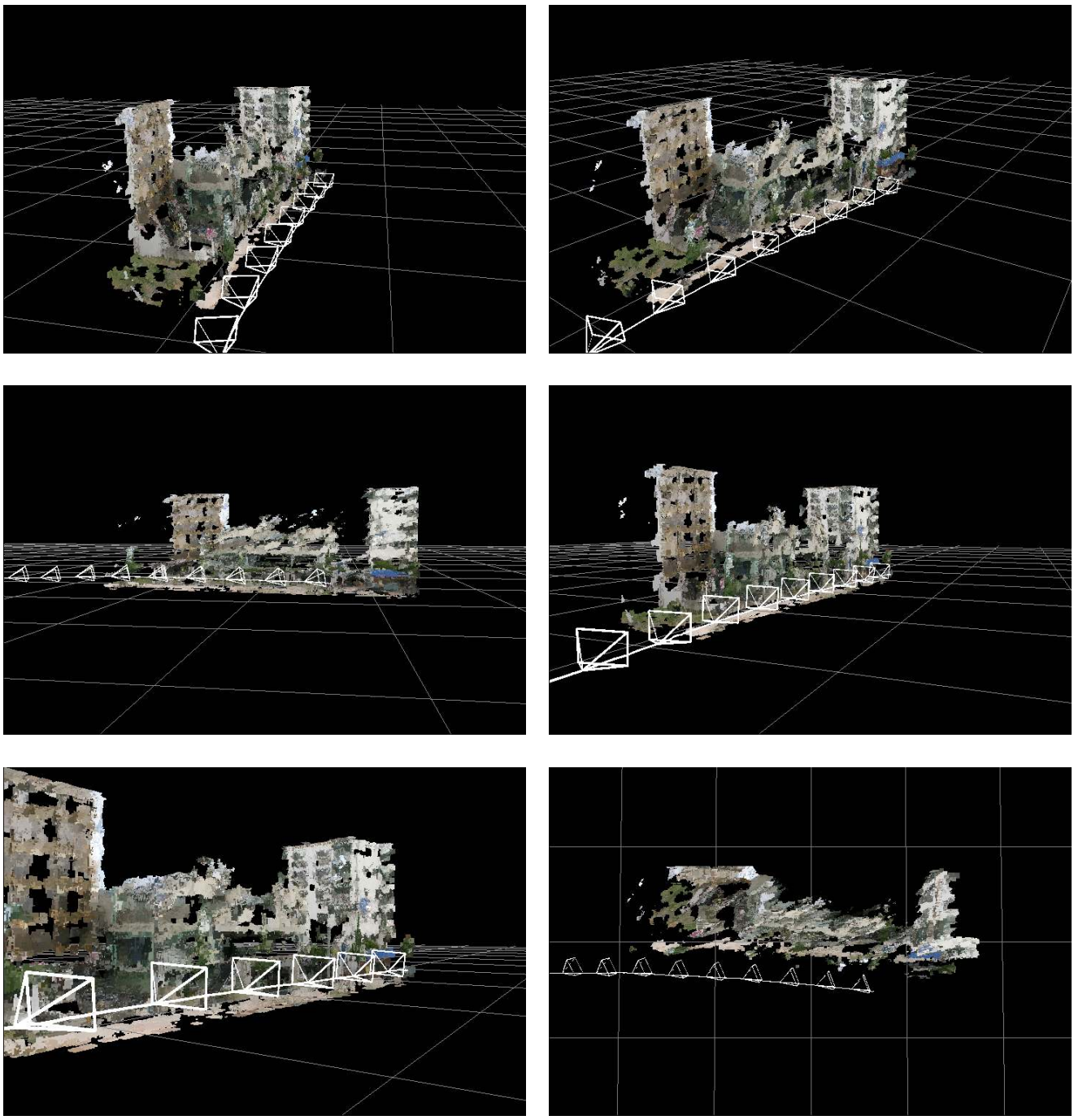


Figure 11: Results of dense outdoor scene recovery as well as estimated camera positions and postures (street scenery).