2008－CVIM－165 (4)
2008／11／27

Robust Outdoor Scene Reconstruction by using Wearable Omnidirectional Vision System

Xihao FAN†, Yoshio IWAI†, Masahiko YACHIDA‡
†Graduate School of Engineering Science, Osaka University
1–3 Machikaneyama, Toyonaka, Osaka 560–8531 Japan
‡Faculty of Information Science and Technology, Osaka Institute of Technology
1–79–1 Kitayama, Hirakata, Osaka 573–0196 Japan
jasonfan@yachi-lab.sys.es.osaka-u.ac.jp,
iwai@sys.es.osaka-u.ac.jp,
yachida@is.oit.ac.jp

Abstract— A 3D modeling method for wide outdoor scenes by using a wearable vision system with one omnidirectional sensor is presented in this paper. We propose a multi-baseline stereo vision method that first estimates the pose of camera, and then reconstructs 3D models of scenes. In addition, GPS is employed to optimize the parameters of camera and determined the scale of estimated extrinsic parameters automatically. This paper also reports some experimental results with the proposed method.
Keywords— 3D, omnidirectional, reconstruction, stereo, GPS

1 Introduction

Recently, 3D reconstruction technology has been applied to many fields, such as 3D navigation system, urban landscape models, simulation of road traffic, and so on. At the same time, wearable and portable computers are related to daily life and will be more and more popular. Therefore, we propose a wearable omnidirectional vision system which can recover 3D outdoor scenes and even can be applied to guide a user in real-time by mounting a HMD. This paper will focus on the 3D reconstruction and current experiments are offline.

In order to realize robust 3D models based on omnidirectional images, a number of investigations have been proposed. Sato et al. proposed a method that estimates 3D models by tracking features in an image sequence using multi-cameras system [1]. This method can quickly estimate 3D models, however, as prerequisite it needs six cameras with high resolution and complex merge application. In addition, several landmarks should be known. Kawasaki et al. proposed a method that estimates 3D models by analyzing EPI, which consists of a temporal sequence of images [2]. This method can easily detect feature trajectories without tracking features frame-by-frame. EPI analysis, however, assumes that epipolar lines are always critically on the same plane and the velocity is constant. This assumption is usually invalid, because of motion vibration, especially for our wearable system. Some investigations [3] [4] have also extended this method to estimate the omnidirectional images, where depth estimates can be obtained from two cylindrical panoramic images. However, when depth is estimated only from two images, erroneous estimation unavoidably occurs due to the occlusion and noise of image. Furthermore, since our sequential images are acquired by using wearable camera when walking, they bring accumulative estimated errors when we estimate wide-baseline motion using these original sequential images.

In addition, a hyperboloid omnidirectional vision sensor is frequently used for navigation and 3D modeling applications [5] [6], because it has a wide field of view. This means that the same features are included for a longer interval of the image sequence and a wider baseline can be taken, thus improving the efficient estimation.

In these directions, on the one hand, we describes a method that estimates extrinsic camera parameters based on tracking a number of image features combined with sparse position data from GPS. On the other hand, we propose an omnidirectional multi-baseline stereo method which provides an efficient depth estimation from the rectified images.

In the remainder of this paper, we firstly describe our omnidirectional vision system in section 2. The proposed method that combines with GPS position data for estimation of extrinsic parameters is introduced in Section 3. Section 4 describes our multi-baseline stereo method. Experimental analysis and results obtained using our method are presented in section 5. Finally, discussion and future work are presented in sections 6.

2 Omnidirectional Vision System

Our omnidirectional vision system consists of an omnidirectional vision camera, GPS and PC shown

in Fig.1.

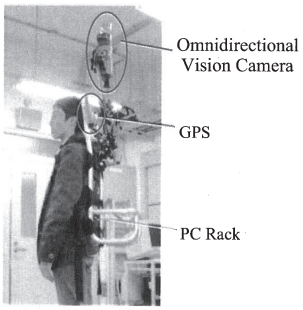

Figure 1: A wearable system with an omnidirectional video GPS and PC.

The omnidirectional vision camera, which is called HyperOmni Vision [7], consists of a hyperboloid mirror mounted in front of a vertically oriented TV camera. The hyperboloid mirror has an attractive geometric property illustrated in Fig.2: an arbitrary point $P(\mathbf{X_w}, \mathbf{Y_w}, \mathbf{Z_w})$ in scenes is mapped onto the hyperboloid mirror $p_{\mathbf{mirr}}(\mathbf{x_m}, \mathbf{y_m}, \mathbf{z_m})$, and then projected into image plane $p_i(x_i, y_i)$ via this point. The position of O_M and O_C are assumed to coincide with the two focal points of the hyperboloid. In this way, sequential omnidirectional images without merge application are acquired. A more detailed description on the design and geometry of the sensor can be found in [7]. The advantage of GPS will be described in section 3.

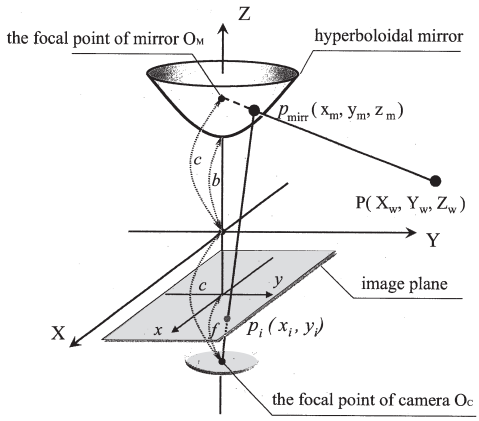

Figure 2: Omnidirectional image formation for a catadioptric vision sensor consisting of a hyperboloid mirror and a Tv camera.

Several considerations have led us to consider panoramic images instead of omnidirectional images. In our application, a virtual panoramic vision sensor is used to construct these panoramas by re-projecting the omnidirectional image onto a virtual cylinder. It is created by specifying a virtual cylinder with unit radius in the mirror coordinates. The cylinder is given by $x^2 + y^2 = 1$. We assume that the cylindrical coordinates coincide with the mirror coordinates. The projection is computed as the intersection of the ray emitting from the origin of the mirror coordinates and passing trough $p_{\mathbf{cyl}}$ with the cylinder surface, as shown in Fig.3.

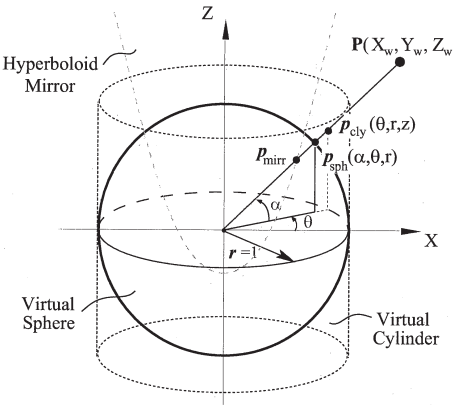

Figure 3: A virtual vision sensor model.

Since vectors are defined to estimate the camera parameters in our coordinates, we introduce function $\mathcal{F}$ to relate vector $\mathbf{x_{cyl}} = [x, y, z]^T$ of a point on the cylindrical surface to its 2D cylindrical coordinates representation, namely panoramic image coordinates, $\mathbf{y} = [\theta, z]^T$:

$$\mathbf{x_{cyl}} = \mathcal{F}(\mathbf{y}) = \begin{bmatrix} \cos\theta \\ \sin\theta \\ z \end{bmatrix}, \qquad (1)$$

where θ is azimuth, $z = r\tan\alpha$ and α is depression angle. The inverse function $\mathcal{F}^{-1}$ is given by:

$$\mathbf{y} = \mathcal{F}^{-1}(\mathbf{x_{cyl}}) = \begin{bmatrix} \tan^{-1}(y/x) \\ z \end{bmatrix}. \qquad (2)$$

A panoramic image $\mathbf{y}$ is then transformed into $\mathbf{x_{cyl}}$ of 3D Cartesian coordinates using Eq.(2). Considering the computability of normal vector when we estimate the camera motion which will be presented in the following section, we prefer transforming vector $\mathbf{x_{cyl}}$ of a virtual unit cylinder to $\mathbf{x_{sph}}$ of a virtual unit sphere (see Fig.3).

3 Camera Parameter Estimation Using Images and GPS

Estimation of the extrinsic camera parameter from sequential images is one of important problems in computer vision, and accurate extrinsic camera parameters are prerequisite for a widely moving camera in an outdoor environment to realize outdoor 3D reconstruction. Many investigations are proposed to improve the errors of camera parameter [8] [9].

As we know, on the one hand, our wearable system leads to more serious vibration coming from walking which causes shift in acquired images. Accumulative error is unavoidable as long as we use only relative constraints among images [10] [11]. However, it is also unpractical to reconstruct high frequency component in motion only by measuring the GPS positions data because the acquisition rate of position information from a general GPS receiver is significantly lower than video rate. On the other hand, the occlusion existing in scene such as moving people and car affect the accuracy of estimation. To avoid these problems, a method based on structure-from-motion (SFM) combined with GPS positions and multi-images feature tracking is proposed.

As precondition, the following two conditions are assumed:

- Camera and GPS are synchronized, namely, images can correspond to every GPS positions.
- Position relation between camera and GPS receiver is always fixed, as well as, the distance between camera and GPS receiver is known.

The proposed method basically consists of feature tracking based on estimation and optimization of parameters. Fig.4 shows the flow diagram of our algorithm. GPS position data and images are synchronously acquired in process (A). Process (B) is performed by the proposed multi-images features tracking method between sequential images. We propose factorization method to initialize parameters in process (C) and then optimize them in process (D). To avoid the accumulative error, GPS position data is introduced to optimize and update camera parameter in the process (E). Furthermore, the scale vector problem of extrinsic parameter can be solved in the process (F).

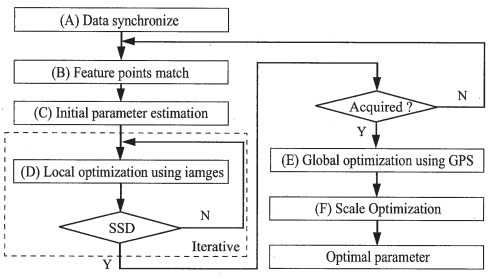

Figure 4: Flow diagram of camera parameter estimation combined with images and GPS.

In the following content of this section, we detail each process.

3.1 Data Synchronize

Firstly, video sequence and GPS data are acquired into database using our wearable system and then picked out at a constant interval. We define the positions of GPS as key-positions, and the corresponding images as key-frames. In this way, we can acquire sequential images between any two key-positions which are labeled with the global position data, and also shorten the baseline of sequential images effectively. In practice, all are offline as preprocesses.

3.2 Feature Points Match

The method of feature points matching is used to automatically convert the image data to point data, and then estimate the correspondence of any two images. A number of investigations were proposed, i.e. RANSAC, KLT, SIFT and so on. In this paper, several hundreds natural feature points are automatically detected by using the Harris corner detector [12] in each panoramic image firstly. Every feature point in the *(i-1)*-th frame is then tentatively matched with the candidate feature points in the *i*-th frame by robust tracking with RANSAC approach. Iteratively, the selected features are re-tracked by deleting outlier and add new candidates, some of them vanish in the following frames. The above approach implemented on all images between neighboring key-positions provides the wide-baseline tracked features and it is efficient to decrease the feature tracking on moving occlusion.

3.3 Initial Parameters Estimation

As the mapping between the Euclidean space and the image space of an omnidirectional image is non-linear, the mapping between the image spaces of an image pair is also non-linear. This means that we cannot directly formulate the point correspondences into a system of linear equations to solve for the mapping function between our image pairs.

In practice, it is a good approximation to assume mosaics based panorama as a cylindrical projection with a single viewpoint [13], if all objects in the scene are relatively far from the projective center.

Based on this assumption, the relationship between image space points can be established as follows.

We first transform each image space points to unit vectors on a unit sphere, S. Let $\mathcal{C}_0$ and $\mathcal{C}_i$ denote relative camera coordinates from which two images are acquired by a central projection onto the spherical surface. Let vector $\mathbf{X}= [X, Y, Z]^T$ denote a scene point P of the world coordinates, which is projected to vector $\mathbf{x_0}$ of the camera coordinates $\mathcal{C}_0$ and vector $\mathbf{x_i}$ of the camera coordinates $\mathcal{C}_i$. We define $\mathcal{C}_0$ as a reference coordinates, i.e. the coordinates system in which vectors are measured. Let $\mathbf{t_i}= [t_x, t_y, t_z]_i^T$ be the translation vector between the reference coordinates and the coordinates $\mathcal{C}_i$ and let $\mathbf{R_i}$ be a rotation matrix between these two coordinates. Then point P can be represented as:

$$r_0\mathbf{x_0} = \mathbf{t_i} + r_i\mathbf{R_i}\mathbf{x_i}, \tag{3}$$

where r_0 and r_i are unknown depths whose values are to be recovered by multi-baseline stereo method in section 4.

Furthermore, the epipolar constraint can be established:

$$\mathbf{x_i^T R_i}(\mathbf{t_i} \times \mathbf{x_0}) = \mathbf{x_i^T RS x_0} = \mathbf{x_i^T E x_0} = 0, \tag{4}$$

where $\times$ denotes the vector product, $\mathbf{S}$ denotes the (3×3) skew symmetric matrix

$$\mathbf{S} = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}, \tag{5}$$

and $\mathbf{E} = \mathbf{RS}$ is known as the essential matrix. Fig.5

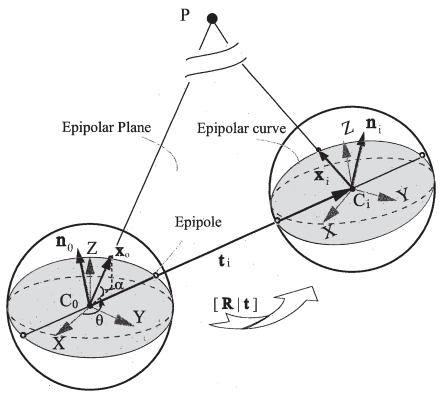

Figure 5: Epipolar geometry for omnidirectional unit sphere

illustrates the principle of the epipolar geometry. An epipolar curve is formed by intersecting the epipolar plane, spanned by $\mathbf{x_0}$ and $\mathbf{t_i}$, with the spherical surface. The points where the line between origin of coordinates $\mathcal{C}_0$ and $\mathcal{C}_i$, namely baseline, intersects with the spherical surface is called the epipole. We can find that there are two epipoles in one unit spherical mode.

Let $\mathbf{x_n^i}= [x_n^i, y_n^i, z_n^i]^T$ be the projective vector at view i and its correspondence of view 0 be $\mathbf{x_n^0}= [x_n^0, y_n^0, z_n^0]^T$, where n is the number of features. From Eq.(4) We can obtain a system of linear equations from a set of correspondences:

$$\begin{bmatrix} x_1^0x_1^i & x_1^0y_1^i & x_1^0z_1^i & y_1^0x_1^i & y_1^0y_1^i & y_1^0z_1^i & z_1^0x_1^i & z_1^0y_1^i & z_1^0z_1^i \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ x_n^0x_n^i & x_n^0y_n^i & x_n^0z_n^i & y_n^0x_n^i & y_n^0y_n^i & y_n^0z_n^i & z_n^0x_n^i & z_n^0y_n^i & z_n^0z_n^i \end{bmatrix} \mathbf{e}$$

$$= \mathbf{De} = \mathbf{0}, \tag{6}$$

where we write the entries of the essential matrix as a vector $\mathbf{e} =[e_1, e_2 \cdots e_9]^T$. A solution for $\mathbf{e}$, i.e. the essential matrix $\mathbf{E}$, thus is found by solving

$$\min_{\mathbf{e}} \|\mathbf{De}\|^2 \qquad \text{subject to} \qquad \|\mathbf{e}\| = 1. \tag{7}$$

The minimum of $\mathbf{e}$ in Eq.(7) is the eigenvector of the moment matrix $\mathbf{M} = \mathbf{D^T D}$ associated with the smallest eigenvalue and can be found by using a singular value decomposition (SVD) of $\mathbf{M}$. This algorithm is known as the 8-point algorithm [14]. An essential matrix has two equal eigenvalues and has rank two [15]. To optimize the estimated matrix, we define the estimated matrix as $\tilde{\mathbf{E}}$ and the optimal essential matrix as $\mathbf{E}$. Let the SVD decomposition of $\tilde{\mathbf{E}}$ be $\mathbf{U\Sigma V^T}$, where $\Sigma = diag(\sigma_1, \sigma_2, \sigma_3)$. The optimal essential matrix $\mathbf{E}$ can be determined as $\mathbf{E} = \mathbf{U\Sigma' V^T}$, where $\Sigma' = diag((\sigma_1 + \sigma_2)/2, (\sigma_1 + \sigma_2)/2, 0)$.

3.4 Local Optimization

Since the 8-point algorithm is very sensitive to noise in the image coordinates, a point normalization method [16] is proposed to decrease the sensitivity. The proposed SVD is just precessed under normalization which is provides by the unit spherical mode. RANSAC [17] algorithm is used to decrease the estimated error of essential parameter $\mathbf{E}$, and then by minimizing SSD (Sum of Squared Differences) to obtain the optimal essential parameter $\mathbf{E}$.

The rotation matrix $\mathbf{R}$ and $\mathbf{S}$ matrix can be determined from the singular value decomposition [18] $\mathbf{E} = \mathbf{U\Sigma' V^T}$ as follows:

$$\mathbf{R} = \mathbf{UYV^T} \text{ or } \mathbf{UY^TV^T}, \tag{8}$$

$$\mathbf{S} = \mathbf{VZV^T} \text{ or } \mathbf{VZ^TV^T}, \tag{9}$$

where

$$\mathbf{Y} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{Z} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \quad (10)$$

There are four possible pairings of $\mathbf{R}$ and $\mathbf{S}$ from initial calculation. Since all images are picked up from video sequences at a constant interval, in practice there are tens of frames in one second, we can assume that every two sequential frames are approximatively stable and all the origins of coordinates between two key-positions approach to co-linear. In other words, the $\mathbf{R}$ is closed to identity matrix is active, and all the normalized translation vectors between two key-positions are equal. Based on these assumptions, the $\mathbf{R}$ and $\mathbf{S}$ can be determined uniquely.

3.5 Global Optimization

However, the assumption in process (D) is untenable when it face to iterative estimation, especially to the estimation of wide-baseline motion, since the errors are accumulated and deviate the practical situation. To avoid the accumulative error, GPS position data is introduced at key-position. In this paper, the mentioned global optimization is focused on the translation estimation.

At each key-position, we compare the estimated error of images with GPS error, and select the parameter with smaller error as the optimal one. The global optimization method is robust to process any wide-baseline motion as short-baseline motion.

3.6 Scale Optimization

A number of investigations about camera parameter estimation only defined up to an arbitrary scale factor [19] [20]. As a consequence, the length of the translation vector relating two images cannot be determined from image information only. In this paper, we employ the GPS to measure the position data, and combine with the GPS position data to estimate the translation vector. Using the assumption in section 3.4, we further assume that the distances of neighboring views are equal. We define $\mathbf{t_i}$ as the translation vector of corresponding views between two key-positions, and $\mathbf{T}$ is the vector of two key-positions, illustrated in Fig.6:

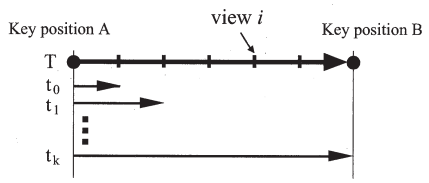

Figure 6: Scale optimization of camera parameters estimation. Vector $\mathbf{T}$ is used to estimate every translation vector $\mathbf{t_i}$ between corresponding views, each magnitude of a translation vector $\|\mathbf{t_i}\| = \|\mathbf{T}\| \times i/k$ approximately and k is the number of the views.

4 Multi-baseline Stereo Vision

In this section we present a multi-baseline stereo vision method for our panoramic images from a set of sequential images. In traditional estimation algorithm, the depth was estimated only using the correspondence of pixels in two images. However, this approach brought large deviation when we estimated the depth. In our multi-baseline stereo algorithm, we estimated the depth among the multi-images. Multi-baseline stereo requires the continuously tracked feature points and the relative camera poses which are solved in the previous section. Subsequently, we rotate these virtual unit spheres based on the reference view such that they all keep the same orientation with reference view. After this rectification, the camera motion can be equivalent to translation only between key points. We define the translation vector of baseline between neighboring key-positions as $\mathbf{T}$, and the translation vector between corresponding views as $\mathbf{t}_i$. Using Eq.(3) and Eq.(2) the correspondence of points in panoramic images can be expressed as:

$$r_0\mathcal{F}(\mathbf{y_0}) = \mathbf{t_i} + r_i\mathcal{F}(\mathbf{y_i}); \quad (11)$$

Given correspondence among $\{\mathbf{y_0}, \mathbf{y_1}, ..., \mathbf{y_i}\}$ in unit spherical model, the known translation vector $\{\mathbf{t_0}, \mathbf{t_1}, ..., \mathbf{t_i}\}$, the depth of a scene point $\{r_1, ..., r_i\}$ from the reference pose can be determined based on the constrain in Fig.7.

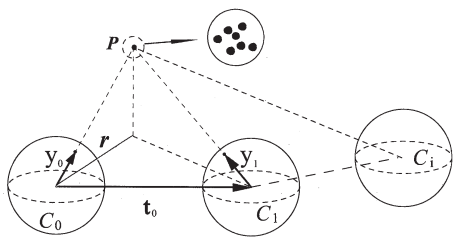

Figure 7: A 3D point P can be inferred from reference view and other views.

Since every view point corresponds to an omni-directional image, every correspondences on $(i-1)$ images between two neighboring key-positions bring $(i-1)$ candidates of depth value r. Then, the depth can be estimated by calculating the geometric centroid of these candidates r.

5 Experiment

In this section, we present the experiments conducted to evaluate our approach. Our real world data was acquired by employing a wearable system equipped with video, GPS and PC. Scenes were acquired in our campus on foot. We extracted the sequential frames at 6 fps from video, namely, the number of the multi-images was defined as 6. The omnidirectional image (720×486 pixels) obtained by the hyperboloid vision sensor is illustrated in Fig.8. Then, they were transformed into panoramic images (628×72 pixels) (see Fig.9).

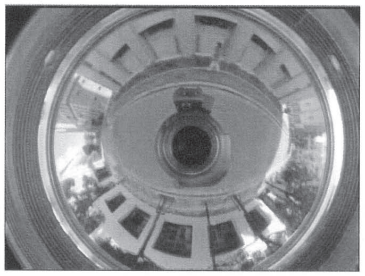

Figure 8: Omnidirectional image is acquired from the hyperboloid vision sensor.

We have carried out three sets of experiments. Two are concerned with the feature tracking and the camera parameters estimation. The other is conducted for estimating depth of feature points in a real outdoor environment.

5.1 For Feature Tracking

In the first experiment, we evaluated our algorithm under partial occlusion. Two kinds of panoramic scenes including a working person and a running car respectively were referred. Then we evaluated the validity of the use of the feature tracking among the multi-images by comparing it with the traditional dual images tracking (see Fig.9). We define the number of feature points in the reference image as 300, the results of tracking are summarized in Table 1. Our approach can effectively eliminate outliers on the moving object, especially, for the slow-moving object.

TABLE 1. STATISTIC ON THE FEATURE TRACKING

Tracked	Dual		Multiple	
Object	*Total*	*Occlusion*	*Total*	*Occlusion*
Person	187	5	118	0
Car	200	8	144	0

5.2 For Camera Parameters Estimation

This experiment was carried out to show the result of camera parameters estimation. By applying SVD to $\mathbf{E}$, rotation and translation could be extracted. Since outliers were decreased effectively, our approach improved the accuracy of estimation. We analyzed the error of estimated essential matrix by comparing our method with traditional method based on the dual images. The iterative was defined as 5 times, 20 times and 50 times respectively when RANSAC algorithm was carried out. From Fig.10 we can find that the error based on multi-images algorithm is distinctly better than traditional algorithm. Furthermore, the proposed approach still improves the average accuracy from 0.84 to 0.46 while decreasing the iterative of RANSAC from 50 times to 20 times. Thus, this approach also reduced the computational cost.

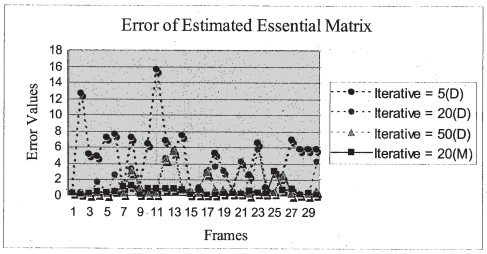

Figure 10: Error of essential matrix based on multi-images tracking (solid line) is compared with that based on dual images tracking (dotted line).

5.3 For Depth Estimation

However, the translation estimation only from images can not locate the global scale when we estimate the 3D points. In this experiment, we already combined the translation vector estimated from images with the translation vector measured from the GPS position data to estimate the real depth, the approach is described in section 3.6 and section 4.

An experiment was performed using image acquired in campus with large depth discontinuities and occlusions. A set of 30 images was extracted in 5 seconds video. The first image in every second was designated as the reference image, illustrated in Fig.11-(a). A rectified image, obtained from the omnidirectional image captured at relative pose $(3.793°, -4.7778°, -5.2962°)$, is shown in Fig.11-(b). Fig.12 represents the depth computed from the reference image and the 5 other images.

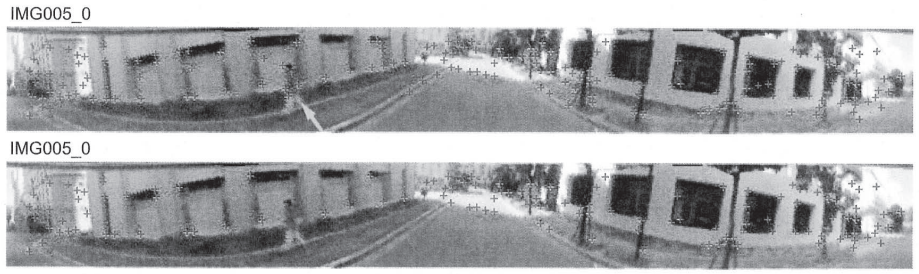

Figure 9: Tracking feature points in the scene with a walking person: (a) shows the result of tracking based on two images and (b) shows the result of tracking based on multi-images.

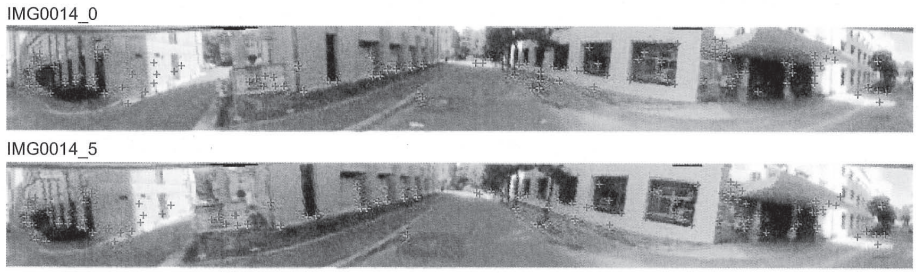

Figure 11: Images acquired in campus with large depth discontinuities and occlusions. (a) is the first image in every second as the reference image and (b) is one of the rectified images.

6 Conclusion

We have proposed a method for recovering 3D structure from sequential omnidirectional images combined with GPS. The proposed method extracts features from the omnidirectional images and estimates the global parameters using GPS position data. Furthermore, multi-baseline stereo method is proposed to reconstruct 3D model.

However, we find the tracked feature points are sparse, that can not satisfy with the true to nature 3D reconstruction. In future research, we intend segmenting graphics based on the current 3D points and estimating the depth by pixel to pixel. As a vivid model, textures which are obtained from our input images should be considered.

References

[1] T. Sato and N. Yokoya: "Multi-baseline Stereo by Maximizing Total Number of Interest Points", Proc. Int. Conf. on Instrumentation, Control and Information Technology, pp. 1471-1477, Sep. 2007.

[2] H. Kawasaki, K. Ikeuchi, and M. Sakauchi: "Spatio-Temporal analysis of omni image", CVPR 2000 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol.2, pp. 577-584, 2000.

[3] J. Gluckman, S. K. Nayar, and K. J. Thoresz: "Real-time omnidirectional and panoramic stereo," Proc. Image Understanding Workshop, 1998.

[4] H. Ishiguro, M. Yamamoto, and S. Tsuji: "Omnidirectional stereo", IEEE Trans. Pattern Anal. Machine Intell., vol. 14, pp. 257-262, Feb. 1992.

[5] K. Yamazawa, Y. Yagi, and M. Yachida: "3D Line Segment Reconstruction by using HyperOmni Vision and Omnidirectional Hough Transforming", Proceedings of IEEE International Conference on Pattern Recognition, vol. 3, pp. 483-486, 2000.

[6] Y. Yagi, K. Shouya, and M. Yachida: "Environmental map generation and egomotion estimation in a dynamic environment for an omnidirectional image sensor", Proceedings of IEEE International Conference on Robotics and Automation, pp. 1487-1492, 2000.

[7] K. Yamazawa, Y. Yagi, and M. Yachida: "Omnidirectional image sensor -HyperOmni Vision-", Proc. 3rd Int. Conf. on Automation Technology, Vol.5, pp.127-132, 1994.

[8] Trung Ngo Thanh, H. Nagahara, R. Sagawa, Y. Mukaigawa, M. Yachida, Y. Yagi: "Robust and Real-time Rotation Estimation of Compound Omnidirectional Sensor", Proceedings of IEEE International Conference on Robotics and Automation, pp. 4226-4231, 2007.

[9] Y. Yokochi, S. Ikeda, T. Sato, and N. Yokoya: "Extrinsic camera parameter estimation based-on fea-

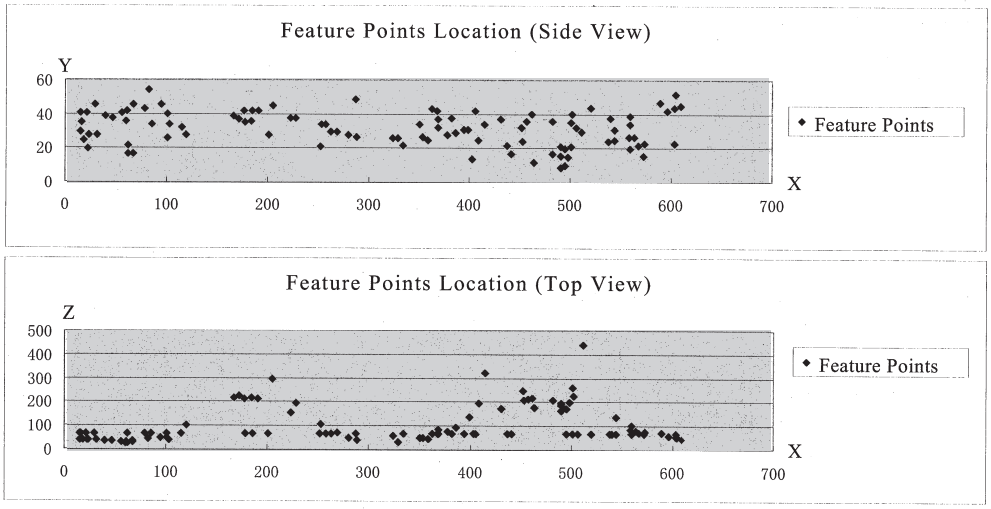

Figure 12: Result of recovering 3D points.

ture tracking and GPS data", Proc. Asian Conf. on Computer Vision, Vol. I, pp. 369-378, Jan. 2006.

[10] A.W. Fitzgibbon, A. Zisserman: "Automatic camera recovery for closed or open image sequences", in: Proc. 5th European Conf. on Computer Vision, Vol. I, pp. 311-326, 1998.

[11] M. Pollefeys, R. Koch, M. Vergauwen, B. Deknuydt, L.V. Gool: "Threedimentional scene reconstruction from images", in Proc. SPIE, Vol. 3958, pp. 215-226, 2000.

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

[13] H. Shum, M. Han, and R. Szeliski: "Interactive Construction of 3D Models from Panoramic Mosaics", Proc. CVPR, pp. 427-433, 1998.

[14] R.I. Hartley: "In defence of the 8-point algorithm", in Proc. IEEE Int. Conf. on Computer Vision (ICCV), pp. 1064-1070, June 1995.

[15] O. Faugeras: "Three-Dimensional Computer Vision: A Geometric Viewpoint", the MIT Press, Cambridge, Massachusetts, 1993.

[16] T. Pajdla, T. Svoboda, and V. Hlaváč: "Epipolar geometry of central panoramic cameras", in Benosman and Kang, chapter 5, pp. 73-102.

[17] M. A. Fischler, R. C. Bolles: "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography", Comm. of the ACM, Vol. 24, pp. 381-395, 1981.

[18] Richard I. Hartley: "Estimation of Relative Camera Positions for Uncalibrated Cameras", Proceedings of the Second European Conference on Computer Vision, pp. 579-587, 1992.

[19] T.J. McKinley, M.M. McWaters, V.K. Jain: "3D reconstruction from a stereo pair without the knowledge ofintrinsic or extrinsic parameters", Proceedings of Second International Workshop on Digital and Computational Video, pp. 148-155, 2001.

[20] H. Kim, Seung-jun Yang, Kwanghoon Sohn: "3D reconstruction of stereo images for interaction between real and virtual worlds", Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality, pp. 169-176, Oct. 2003.