

Projection-based Registration Using a Multi-view Camera for Indoor Scene Reconstruction*

Sehwan Kim and Woontack Woo
GIST U-VR Lab.
Gwangju 500-712, S.Korea
{skim, wwoo}@gist.ac.kr

Abstract

A registration method is proposed for 3D reconstruction of an indoor environment using a multi-view camera. In general, previous methods have a high computational complexity and are not robust for 3D point cloud with low precision. Thus, a projection-based registration is presented. First, depth are refined based on temporal property by excluding 3D points with a large variation, and spatial property by filling holes referring neighboring 3D points. Second, 3D point clouds acquired at two views are projected onto the same image plane, and two-step integer mapping enables the modified KLT to find correspondences. Then, fine registration is carried out by minimizing distance errors. Finally, a final color is evaluated using colors of corresponding points and an indoor environment is reconstructed by applying the above procedure to consecutive scenes. The proposed method reduces computational complexity by searching for correspondences within an image plane. It not only enables an effective registration even for 3D point cloud with low precision, but also need only a few views. The generated model can be adopted for interaction with as well as navigation in a virtual environment.

1. Introduction

Image-based 3D reconstruction of a real environment plays a key role in providing visual realism for navigation in and interaction with a virtual environment (VE). Conventional modeling of a real environment with 3D modeling tools is time-consuming and the generated models are not realistic. Though the reconstruction method using active range sensors, combined with a camera, generates an accurate and realistic model, they require expensive sensing equipments

and time-intensive reconstruction processing. Furthermore, alignment of 3D point cloud with a texture map is also required. On the other hand, image-based 3D reconstruction methods not only preserve realism but also provide a rather simple modeling procedure. Especially, off-the-shelf multi-view cameras, providing color as well as depth images in real-time, enables to generate image-based models more easily. Therefore, a delicate registration is required to register 3D point clouds, acquired from the multi-view camera in a few directions, for 3D reconstruction of a real environment.

Various registration methods were proposed. Besl et al. proposed ICP (Iterative Closest Point) algorithm [1], and Johnson presented Color ICP to reconstruct an indoor environment [2][3]. Blais et al. exploited a simulated annealing to minimize a cost metric based on total distance between all matches [4]. On the other hand, Nishino presented an optimization method based on M-estimator to implement a robust registration for several range images [5]. Especially, Pulli developed a data acquisition device and adopted a projective registration employing planar perspective warping [6]. Sharp defined invariant features to improve ICP [7], and Fisher applied projective ICP to Augmented Reality applications [8]. But, most methods depend on expensive equipments and require much time to generate 3D models [4][5][7]. In addition, if 3D point clouds have large error bound it is hard to guarantee the accuracy of the registration results [1][2][3]. Furthermore, stereo cameras are usually used for object modeling, and not for an indoor scene reconstruction [6].

To address these problems, a novel projection-based registration method is proposed. First, depth image is refined based on spatio-temporal property of 3D point cloud using adaptive uncertainty region. Second, 3D point clouds acquired at two adjacent views are projected onto the same image plane, and corre-

* This research was supported by the MIC, Korea, under the ITRC support program supervised by the IITA

spondences are searched for within an overlapping area through the modified KLT (Kanade-Lucas-Tomasi) feature tracker. Then, two sets of 3D point cloud are registered by minimizing Euclidean distance errors with Levenberg-Marquardt algorithm. Finally, each triangulated point is evaluated referring corresponding points, and a new color is assigned to each point in the overlapping area. Consequently, we reconstruct an indoor environment by applying the above procedure to consecutive scenes.

The proposed 3D reconstruction method for a real environment has the following characteristics. First of all, although a multi-view camera provides 3D point cloud which has low precision, its pervasiveness provides a chance to simplify 3D reconstruction of a real environment [9][10]. For this purpose, the proposed method estimates camera pose using projected 2D correspondences instead of 3D coordinates. Thus, registration is carried out effectively even if the precision of acquired 3D point cloud is relatively low compared to that of precise optical sensor. Besides, computational complexity is reduced as it searches for the correspondences in a 2D image plane. This also enables a more convenient and fast reconstruction, combined with the multi-view camera which provides 3D information in real-time. In addition, the proposed method simplifies 3D reconstruction by placing a multi-view camera only at a few positions in an indoor environment.

The paper is organized as follows. In Chapter 2, we explain the depth image refinement. Fine registration is mentioned in Chapter 3. Then, color selection for the overlapping area and extension to the consecutive scenes are described in Chapter 4. After experimental results are analyzed in chapter 5, conclusions and future work are presented in chapter 6.

2. Depth Image Refinement

2.1. Erroneous 3D Points

Unlike optical sensor-based methods which use active range sensing technique, passive techniques use the images generated by light reflected by objects. However, disparity estimation results in inherent stereo mismatching errors, usually at depth discontinuity and on homogeneous areas. The errors cause poor registration results. Thus, unreliable regions should be eliminated before registration. In this regard, a depth image is refined by employing its spatio-temporal property.

In the first step, erroneous points are removed using the temporal property that erroneous depth values change dramatically in 3D space with time. In the second step, holes are filled on the basis of the spatial

property that there is a spatial correlation between neighboring pixels. Figure 1 is a flow diagram for 3D reconstruction of two scenes in a real environment.

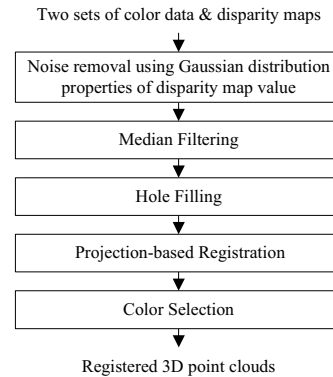


Figure 1. Flow diagram for 3D reconstruction

To analyze the characteristics of 3D point cloud, mean and standard deviation are calculated for each pixel after acquiring N_f depth images for the same scene. However, depth values of some parts have large variations even in a static scene. The reason is that although the materials of objects are assumed to obey the properties of Lambertian surface, some parts do not satisfy the postulation. In general, depth values drift with time, camera position, illumination conditions of a scene, etc. These factors induce large variations in depth values, especially at depth discontinuity areas as well as on homogeneous areas, when disparity is estimated. Thus, these unstable parts should be removed.

In the depth image of a static scene, depth variation of each pixel is modeled as a Gaussian distribution. After investigating depth value of each pixel, we get rid of the pixels whose depth variation is larger than the threshold value for the i^{th} pixel as follows.

$$\sigma_i > \alpha Th_i \quad (1)$$

where σ_i represents a standard deviation for depth variation of the i^{th} pixel. α and Th_i denote a scale factor and a threshold value for the i^{th} pixel, respectively.

However, note that the threshold values depend on 3D coordinates of a scene with respect to the optical center of the multi-view camera. This is because the disparity estimation error increases as an object moves away from the camera. Therefore, Th_i must be re-expressed as a function of 3D coordinates with respect to the optical center of the camera.

2.2. Adaptive Uncertainty Region

To decide Th_i in terms of 3D coordinates for a scene, error bounds of each axis should be calculated for

every ray, which starts from the optical center and passes through corresponding pixel on the image plane. A 3D point within the estimated error bound cannot be distinguished from the error due to the inherent disparity estimation errors. The above description derives us to conclude that an adaptively changing shape appears to be ellipsoidal in 3D space, and we call it *Adaptive Uncertainty Region*.

The multi-view camera has correlation errors in disparity estimation and calibration errors. For a multi-view camera with a constant Field Of View (FOV), we assume that Gaussian noise distribution linearly increases with the distance along x and y axes; and Gaussian noise distribution increases monotonically with the distance along z axis as shown in Figure 2 [11]. Based on 3D coordinates for x , y and z axes, we determine tolerances for each axis.

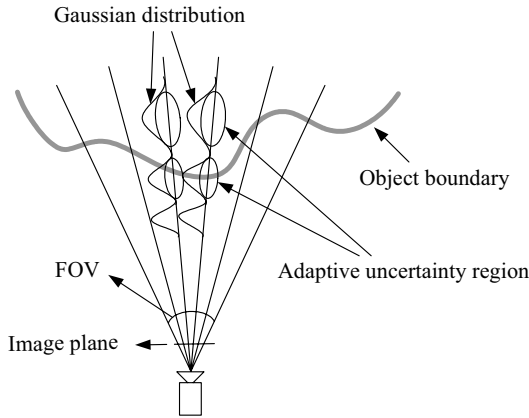


Figure 2. Adaptive uncertainty region

Each 3D point, calculated from corresponding 2D pixel, generates an ellipsoidal uncertainty region in 3D space. Thus, threshold, Th_i , is determined referring the adaptive uncertainty region that changes according to 3D coordinates of a scene, and is described as follows.

$$\frac{(x - x_c)^2}{(\Delta x)^2} + \frac{(y - y_c)^2}{(\Delta y)^2} + \frac{(z - z_c)^2}{(\Delta z)^2} = 1 \quad (2)$$

where (x_c, y_c, z_c) denotes a translation vector from optical center of the camera to center of the uncertainty region. Δx , Δy and Δz represent uncertainty distances along each axis. However, the ellipsoid should be rotated with respect to the optical center reflecting the direction of the ray. Therefore, after the ellipsoid of Eq. (2) is rotated with respect to the optical center using Eq. (4), it moves to (x_c, y_c, z_c) .

$$\frac{x^2}{(\Delta x)^2} + \frac{y^2}{(\Delta y)^2} + \frac{z^2}{(\Delta z)^2} = 1 \quad (3)$$

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R_1 R_2 \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} \quad (4)$$

$$R_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & z_c/d & y_c/d \\ 0 & -y_c/d & z_c/d \end{pmatrix}$$

$$R_2 = \begin{pmatrix} d & 0 & x_c \\ 0 & 1 & 0 \\ -x_c & 0 & d \end{pmatrix}, \quad d = \sqrt{y_c^2 + z_c^2}$$

where (x', y', z') is a final uncertainty region in terms of 3D coordinates of a scene with respect to the optical center. Therefore, Eq. (1) is reexpressed reflecting 3D coordinates, (x_c, y_c, z_c) , with respect to optical center of the camera as well as direction of ray as follows.

$$\sigma_i > \alpha Th_i(x_c, y_c, z_c) \quad (5)$$

Through the above step, some parts of a scene, which do not meet the assumption of Lambertian surface, are removed. The boundary of an object and homogeneous areas are also excluded. Then, Median filter is applied to remove spot noises.

However, hole filling is required on the holes, generated during the above step, and homogeneous areas where disparity may not be estimated. Thus, spatial property for a current point, i.e. spatial correlation among 3D points of four neighborhood pixels, is exploited [11]. However, this step may generate errors if depth difference between adjacent pixels is large. Therefore, this step is applied only when the depth discontinuity is less than a threshold, Th_{dd} . After investigating the 3D coordinates of each of four directions, we apply this step only to the holes whose size is so small that we can consider each of them as a plane.

3. Registration using Correspondences

The depth image refinement removes inherent stereo mismatching errors, and reduces error bound of 3D point cloud. However, precision of 3D point cloud is still low for registration. The registration exploiting the conventional ICP, which employs the shortest distance, is inappropriate since error bound of 3D point cloud is relatively large. Thus, a projection-based registration method is proposed to carry out a pairing process that searches for correspondences between 3D point clouds of destination and source views. Figure 3 shows the projection-based registration of Figure 1 in detail.

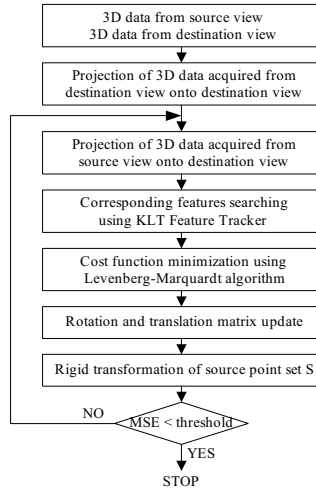


Figure 3. Flow diagram for projection-based registration

3.1. Initial Registration

A multi-view camera is attached to a mobile robot and its pose is estimated by detecting and tracking features of a scene. First, a coplanar calibration pattern and structural constraints of the multi-view camera are used for its calibration [12][13]. Then, intrinsic parameters and camera pose are estimated with respect to a reference position in a real environment. Finally, a rigid-body transformation is applied to 3D coordinates of feature points at each camera view to estimate the poses of the camera at each view. Therefore, partial 3D point clouds, acquired at each position, are initially registered. Even though the initial registration provides a combined 3D point cloud for a scene, it still requires fine registration that provides a correct camera pose.

3.2. Fine Registration

Correct pairing plays an essential role in accurate registration. Figure 4(a) and Figure 4(b) are the projection results of 3D point clouds, which are acquired at the destination and source views, onto the destination view. Mid-luminance value is assigned to unprojected area to differentiate it from projected one. It should be noted that the projection of 3D point cloud of source view onto the destination view creates self-occlusion. This is eliminated based on the rays that originate at the camera center and pass through each pixel on the image plane. Theoretically, Figure 4(b) should exactly overlap with Figure 4(a). However, discrepancies occur due to the errors in disparity estimation, camera calibration, etc. Therefore, accurate geometric relationship between two views is found by minimizing

Euclidean distance errors between correspondences within the overlapping area in terms of projection matrix of a source view. Accordingly, fine registration should be accomplished to compensate the errors induced by disparity estimation, camera calibration, and so forth.

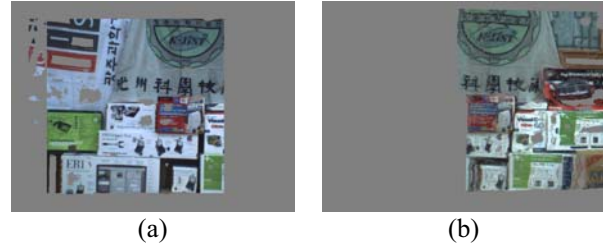


Figure 4. Projection of 3D point cloud onto image plane (a) projection of 3D point cloud of destination view onto its own view (b) projection of 3D point cloud of source view onto the destination view

In general, projection of 3D point cloud of the source view onto the destination view generates floating-point numbers. Thus, some pixels do not have any value as shown in Figure 6(a). These unprojected pixels can generate false alarms when corresponding features are searched for through a modified KLT feature tracker. Therefore, to preserve an original image as well as to remove unprojected pixels, a special care should be taken.

A two-step integer mapping is presented to meet these requirements. In Figure 5, grid points are on the lines. White circles represent grid points, and black circles denote projected pixels of 3D point cloud of the source view onto the destination view.

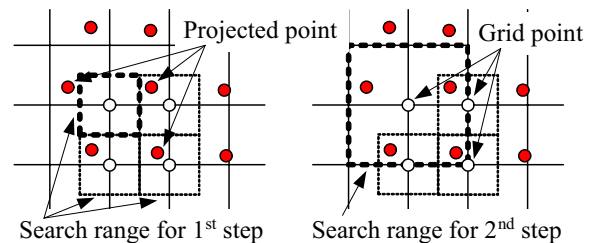


Figure 5. Two-step integer mapping

At the first step, a search range is set to $-0.5 \sim +0.5$ along x and y axes, respectively, for each grid point. The color of each grid point is evaluated by considering weights, which are decided by relative distances with all pixels within the search range. In most cases, only one projected point is included in the first range.

However, there exist grid points which do not include any projected point at the first step. At the second step, the search range is expanded to $-1.0 \sim +1.0$

and a similar procedure is accomplished. Figure 6 shows the results. Figure 6(a) is an enlarged part of Figure 4(b), and Figure 6(b) is the result after the two-step integer mapping is applied. The presented method improves the modified KLT feature tracking performance by removing unmapped grid points and preserving an original image as it is, at the same time.



Figure 6. Two-step integer mapping (a) before (b) after

We search for corresponding feature points, and use Euclidean distance between them to define a total cost function within the overlapping area. To guarantee the correct pairing, RANSAC is also applied. By minimizing the following cost function, a final pose of the source view is estimated. That is, we can estimate the pose of source view $\{R_{Src}, T_{Src}\}$, with respect to the pose of destination view $\{R_{Dst}, T_{Dst}\}$ through minimizing the distance errors on N_{feat} feature points as follows.

$$\begin{aligned} & \text{Given two sets of corresponding points,} \\ & \text{Find } \{R_{Src}, T_{Src}\} \text{ w.r.t } \{R_{Dst}, T_{Dst}\} \\ & \text{such that } \underset{\{R_{Src}, T_{Src}\}}{\text{arg min}} E \end{aligned} \quad (6)$$

where

$$E = \sum_{i=0}^{N_{feat}-1} \sqrt{(x_{Dst,i} - x_{Src,i})^2 + (y_{Dst,i} - y_{Src,i})^2}$$

where $(x_{Dst,i}, y_{Dst,i})$ and $(x_{Src,i}, y_{Src,i})$ represent points on projected 2D image planes of 3D point cloud of the destination and source views, respectively. E denotes Euclidean distance between them. The total distance error is minimized by Levenberg-Marquardt algorithm.

4. Surface reconstruction of 3D point cloud

After the estimation of the camera parameters of source view with respect to destination view, trimming and color selection, for duplicate 3D point clouds within the overlapping area, are carried out. After registration of two views, each grid point of the overlapping area in the destination view has its own correspondence in the source view. This means that we can obtain the corresponding features in 3D point cloud of the original source view. Thus, final 3D coordinates

are calculated by a linear triangulation method for the overlapping area [14].

Then, color adjustment is required to consider changes in lighting conditions depending on camera position. We suppose that all materials within a captured scene satisfy the properties of Lambertian surface.

$$\begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} = \left(v \times \begin{bmatrix} R_{Dst} \\ G_{Dst} \\ B_{Dst} \end{bmatrix} + u \times \begin{bmatrix} R_{Src} \\ G_{Src} \\ B_{Src} \end{bmatrix} \right) / (u + v) \quad (7)$$

where u and v are distances from left and right borders of the overlapping area to the current pixel, respectively. R_{Dst} (or G_{Dst} , B_{Dst}) and R_{Src} (or G_{Src} , B_{Src}) are red (or green, blue) component of a current pixel for both images. On the other hand, R' (or G' , B') means final colors for the overlapping area.

To reconstruct a final surface, after placing a multi-view camera at several positions successively and acquiring images and 3D point clouds, we apply the above procedure to them. Figure 7 shows a conceptual diagram describing an indoor scene reconstruction.

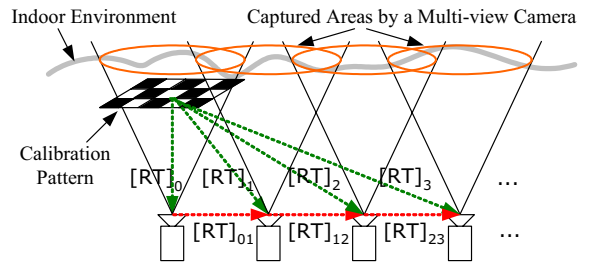


Figure 7. Indoor scene reconstruction

As shown in Figure 7, we first place a calibration pattern whose relative position from an indoor environment is already known. Then, based on the pattern, we estimate intrinsic and extrinsic parameters of a multi-view camera with respect to the origin of the world coordinate system. We calculate $[R T]_0$ by exploiting Intra-/Inter-calibration which employs Tsai's algorithm and structural information of the camera [11]. Through the above-mentioned procedure, we estimate the pose of the second view with respect to the first view, $[R T]_{01}$. That is, we can calculate the pose of the second view from the reference point, $[R T]_1$. Thus, we can get the camera pose of the i^{th} view from the reference point, $[R T]_{i-1}$, and carry out 3D surface reconstruction. By using the proposed method, we do not need a dense depth estimation process employing the multi-baseline stereo technique. Thus, we can reduce computational complexity.

5. Experimental Results and Analysis

The experiments were carried out under a normal illumination condition of a general indoor environment. We used Digiclops, an IEEE 1394 multi-view camera for color image and 3D point cloud acquisition, and a Xeon 2.8 GHz CPU computer [9]. We employed a coplanar pattern with 7×5 grid points to get an initial camera pose from the origin of the world coordinate system. Distance between two consecutive points of the pattern is 10.6 cm.

Figure 8 shows the results of depth image refinement. Figure 8(a) and Figure 8(b) depict original and corresponding disparity images. 3D point cloud acquired from a camera and the results of depth refinement and hole filling are demonstrated in Figure 8(c) and Figure 8(d), respectively. In this regard, we set N_f as 30 and Th_{dd} as 0.15. Initial value of N_{feat} is set as 200. We can observe from the results that invalid areas, such as object boundary, homogeneous areas and non-Lambertian surface, are effectively removed. Holes, whose depth difference is very small, are also filled.

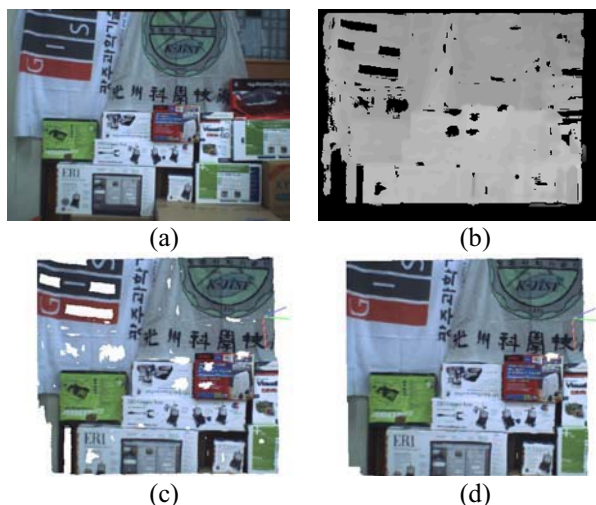


Figure 8. Depth refinement (a) original image (b) depth image (c) 3D point cloud before depth refinement (d) 3D point cloud after depth refinement

Figure 9 demonstrates the results of minimizing the Euclidean distances between corresponding features. That is, we applied the modified KLT feature tracker to Figure 4(a) and Figure 4(b), and minimized the distances between them. Figure 9(a) and Figure 9(b) are projected images of 3D point clouds, which are acquired at the destination and source views, onto the destination view. Corresponding features are also marked. Enlarged areas are also shown in Figure 9(c) and Figure 9(d) at the initial and final steps. Red and white markers represent corresponding features of

source and destination views, respectively. We can see that the distances between correspondences are effectively minimized.

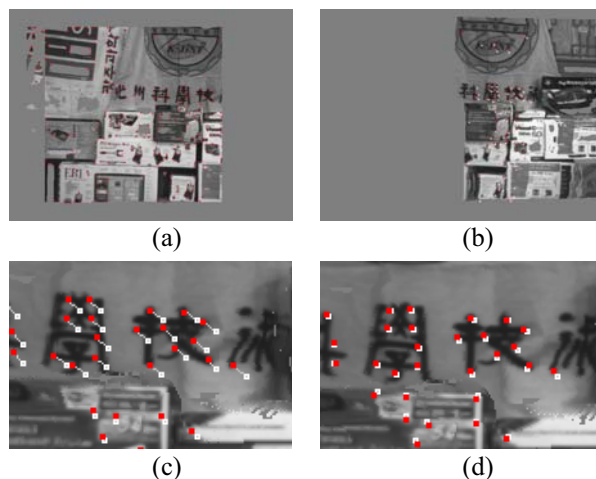


Figure 9. Corresponding features (a) projection of 3D point cloud of destination view onto its own view (b) projection of 3D point cloud of source view onto the destination view (c) enlarged area of (b) before error minimization (d) enlarged area of (b) after error minimization

Figure 10 illustrates the registration results. Figure 10(a) and Figure 10(b) show a combined 3D point cloud of both views and a registered 3D point cloud after applying the proposed method, respectively. On the other hand, Figure 10(c) and Figure 10(d) are enlarged areas for the corresponding scenes. By observing the boundary of a circle, Chinese and English characters, we can see that the registration works well.

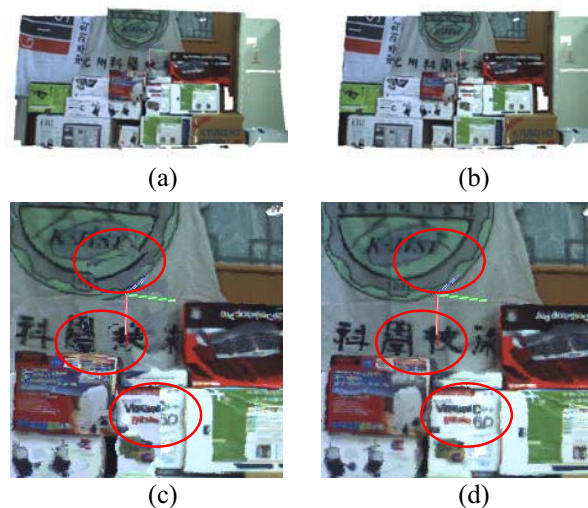
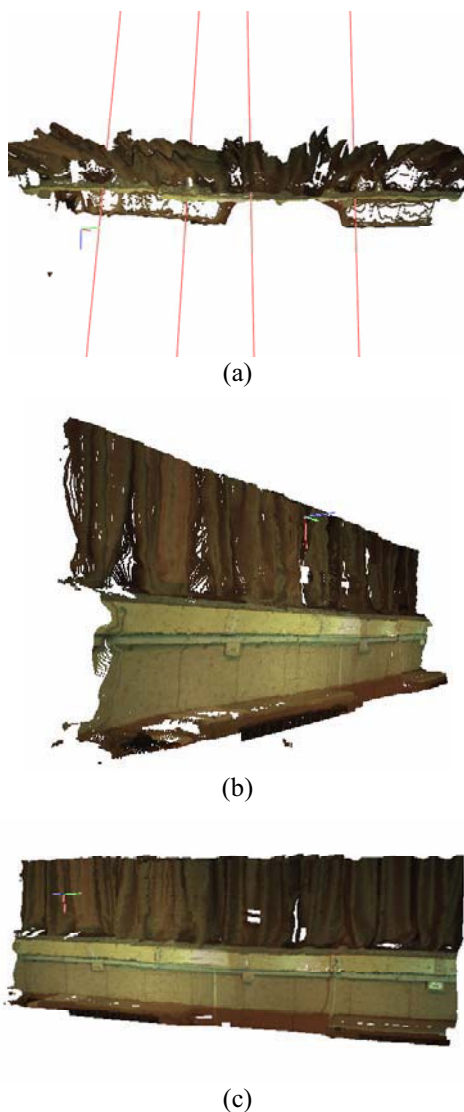
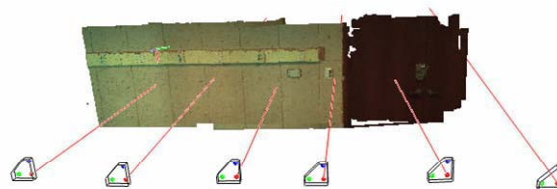


Figure 10. Registration results (a) combined 3D point cloud (b) registered 3D point cloud (c) enlarged area of (a) (d) enlarged area of (b)

The registration results for one wall of an indoor environment are shown in Figure 11. We let the multi-view camera located around a wall, and acquired color images and 3D point clouds. As shown in Figure 11(a), we moved the camera four times around the wall and registered partial 3D point clouds to generate a depth-preserved 3D wall. Figure 11(b) and Figure 11(c) are scenes seen from left and front sides. The right part of Figure 11(c) is enlarged in Figure 11(d). On the other hand, the front view of the registration results for the views acquired at six positions is illustrated in Figure 11(e) and a zoomed-in scene is shown in Figure 11(f). As demonstrated, by applying the proposed registration method to a few sets of 3D point cloud, we can achieve dense 3D reconstruction for an indoor environment.



(d)



(e)



(f)

Figure 11. Indoor scene reconstruction (a) top view of 4 registered views (b) left view (c) front view (d) enlarged area of (c) (e) front view of 6 registered views (f) zoomed-in scene

We compared the proposed method with conventional ICP and color ICP on the basis of PSNR (Peak Signal to Noise Ratio), which represents the visual quality of registered results, in Figure 12. From the results, we can observe that ICP and color ICP are difficult in being applied to the 3D point cloud, which has large disparity estimation errors. The reason is that those methods are just based on the closest distance and do not consider neighborhood pixels. Projection-based ICP just projects 3D point cloud of source view onto destination view and searches for paired 3D point. However, it is difficult to guarantee correct corresponding features in case of the large error bound.

On the other hand, the proposed method tracks the corresponding features by exploiting surrounding information for each block in a 2D image plane. Then, it minimizes the distance between the corresponding features. Thus, we can see that the proposed method provides results that are more reliable. We took advantage of the following measure for comparing performance in the overlapping area.

$$PSNR = 20 \log_{10} \frac{255}{\sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (Y_{Src,i} - Y_{Dst,i})^2}} \quad (dB) \quad (8)$$

where N is the number of pixels, which are valid for both images; and $Y_{Src,i}$ and $Y_{Dst,i}$ denote the luminance value of the i^{th} point on each projected 3D point cloud of source and destination views, respectively.

The visual quality of the proposed method is superior to those of conventional ICP and color ICP. Furthermore, the convergence of the proposed method is faster than that of [11] ($\alpha = 7.0$, $N_B = 192$) since the proposed method employs the corresponding features. Besides, visual quality is also better than that of [11]. Although it seems that the conventional ICP and color ICP converge faster, they take longer to search for correspondences than does the proposed method. The reason is that our method is based on a 2D image plane while the conventional ICP and color ICP search for correspondences in 3D space.

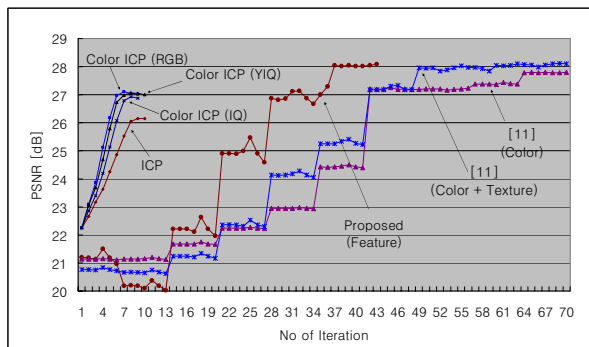


Figure 12. Performance comparison

6. Conclusions and Future Work

We proposed a novel registration method exploiting color and depth information acquired from a multi-view camera to carry out 3D reconstruction for an indoor environment. We proved that even though the error of depth information is relatively large compared to that of laser-scanned data, 3D point clouds are effectively registered between two views. We also showed that an effective reconstruction is possible using a few views of the real environment instead of many 2D images. There are still several remaining challenges. First, we have to reduce convergence rate for registration. Global registration should be optimized to do 3D reconstruction for the entire indoor environment. Natural augmentation of virtual objects into the reconstructed room environment requires light source estimation and analysis to match illumination condition of the VE. Up to now, we limit the proposed method to only indoor

scenes due to the disparity estimation errors. However, if the precision is enough high, it can be applied to outdoor scenes.

7. References

- [1] P. J. Besl and N. D. McKay, "A Method for Registration of 3-D Shapes," *IEEE Trans. on Pattern Recognition and Machine Intelligence*, vol. 14, no. 2, pp. 239-256, 1992.
- [2] A. Johnson and S. Kang, "Registration and Integration of Textured 3-D Data," Tech. report CRL96/4, Digital Equipment Corporation, Cambridge Research Lab, Oct., 1996.
- [3] S. Kang and R. Szeliski, "3-D scene data recovery using omnidirectional multibaseline stereo," Tech. report CRL-95-6 Oct 1995.
- [4] G. Blais and M. D. Levine, "Registering multiview range data to create 3-D computer objects," *IEEE Trans. PAMI*, vol. 17, no. 8, pp. 820-824, 1995.
- [5] K. Nishino and K. Ikeuchi, "Robust Simultaneous Registration of Multiple Range Images Comprising A Large Number of Points," ACCV2002, 2002.
- [6] K. Pulli, *Surface Reconstruction and Display from Range and Color Data*, Ph.D. dissertation, University of Washington, 1997.
- [7] G. C. Sharp, S. W. Lee and D. K. Wehe, "Invariant Features and the Registration of Rigid Bodies," *IEEE Int'l Conf., on Robotics and Automation*, pp. 932-937, 1999.
- [8] R. Fisher, "Projective ICP and Stabilizing Architectural Augmented Reality Overlays," *Int. Symp. on Virtual and Augmented Architecture (VAA01)*, pp 69-80, 2001.
- [9] Point Grey Research Inc., <http://www.ptgrey.com>, 2002.
- [10] VIDERE DESIGN, <http://www.videredesign.com/>, 2004.
- [11] S. Kim, K. Kim and W. Woo, "Projection-based Registration using Color and Texture Information for Virtual Environment Generation," *Lecture Notes in Computer Science 3331*, pp. 434-443, 2004.
- [12] R. Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 323-344, 1987.
- [13] K. Kim and W. Woo, "Multi-view Camera Tracking for Modeling of Indoor Environment," *Lecture Notes in Computer Science 3331*, pp. 288-297, 2004.
- [14] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, March 2004.