

Panoramic Mesh Model Generation from Multiple Range Data for Indoor Scene Reconstruction^{*}

Wonwoo Lee and Woontack Woo

GIST U-VR Lab.,
Gwangju 500-712, S. Korea
{wlee, woo}@gist.ac.kr

Abstract. In this paper, we propose a panoramic mesh modeling method from multiple range data for indoor scene reconstruction. The input to the proposed method is several sets of point clouds obtained from different viewpoints. An integrated mesh model is generated from the input point clouds. Firstly, we partition the input point cloud to sub-point clouds according to each camera's viewing frustum. Then, we sample the partitioned sub-point clouds adaptively and triangulate the sampled point cloud. Finally, we merge all triangulated models of sub-point clouds to represent the whole indoor scene as one model. Our method considers occlusion between two adjacent views and it filters out invisible part of point cloud without any prior knowledge. While preserving the features of the scene, adaptive sampling reduces the size of resulting mesh model for practical usage. The proposed method is modularized and applicable to the other modeling applications which handle multiple range data.

1 Introduction

Virtual environment (VE) generation is one of major task in virtual reality (VR) applications. The realism of the VE is important since it increase users' immersion to the virtual world. VE is usually created by computer graphic (CG) modeling software, such as Maya and 3DS Max. However, constructing large VE with 3D CG modeling software requires much time and effort. To create realistic VE, we have to design everything of the environment to be modeled before modeling. In this regard, VE generation by modeling the real scene is one of solutions for constructing realistic VE.

With increasing interest in this area, there have been many researches on VE generation from the real scene. One approach is using range scanners to obtain 3D data from the real environment [1][2][3][12][13][14]. The environment is scanned and 3D information of the environment is obtained as point clouds. Textures obtained from the photos of the environment are mapped to the reconstructed model for realism. Range scanners provide accurate data, however they are designed for scanning objects in short distance. Thus, it is inconvenient for modeling large area. In addition, the

^{*} This work was supported in part by MIC through RBRC at GIST, and in part by CTRC at GIST.

range scanners are not affordable in general usage and used in few limited purposes. Another approach is reconstructing environment by extracting 3D information from multi-view images [4]. 3D structure of the environment is generated from relationship among the images. Using cameras for 3D reconstruction contains noises compared with the approach exploiting range scanner. Panoramic images of the environment are also used in modeling [5][6][15]. Panoramic images are taken by omni-directional cameras and 3D scene is reconstructed. There are many different approaches of modeling the real scene, however mesh modeling from 3D data is commonly needed to create surface from the 3D point data.

In this paper, we propose a panoramic mesh model generation method from multiple range data for indoor scene reconstruction. We use 3D vision-based modeling method to create mesh models from multiple noisy range data. The 3D point clouds are obtained from multiple viewpoints and all point clouds are registered in 3D space. The input to the proposed modeling method is the registered point clouds and reconstructed camera matrices of all viewpoints. Firstly, we partition the input point cloud to several sub-point clouds according to each camera's viewing frustum. Then, we sample the partitioned sub-point clouds adaptively and generate a mesh model from each sampled point cloud by triangulation. Finally, we merge all triangulated models of all the partitions to represent the whole indoor scene as one model. The modeling sequence of the proposed method is shown in Fig 1.

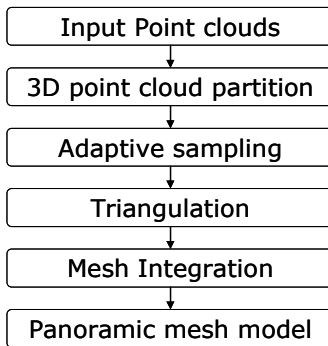


Fig. 1. Panoramic mesh modeling process

Our mesh modeling method considers occlusion between two adjacent views and it filters out invisible part of point cloud without any prior knowledge. While preserving the features of the scene, adaptive sampling reduces the size of resulting mesh model for practical usage. The proposed method is modularized and applicable to the other modeling applications which handle multiple range data.

The rest of this paper is organized as follows. We explain the proposed mesh modeling method in chapter 2. The experimental results and analysis are described in chapter 3. Conclusions and future work are presented in chapter 4.

2 Panoramic Mesh Modeling from Multiple Range Data

2.1 Data Acquisition

In data acquisition step, we obtain 3D data of the indoor environment to be modeled as point clouds using a multi-view camera. The multi-view camera gives a 3D point cloud of a view. To obtain accurate data, we calibrate the multi-view camera and we calculate the intrinsic parameters. The relationship between two adjacent viewpoints is calculated using a co-planar pattern. The data obtained from a viewpoint has its own reference coordinate system. We register all data to locate them in a common reference coordinate system using projection-based registration method [7]. The registered point clouds and the reconstructed camera matrices are input to the modeling process.

2.2 3D Point Cloud Partition

As the first step of the proposed mesh modeling method, we partition input point cloud of the indoor scene into sub-point clouds. Since we obtain 3D points of a view using 3D vision theory, each 3D point of a point cloud corresponds to each pixel of the image captured from a viewpoint. Triangulation of pixels generates naive mesh model of the point cloud. The normal vector of each 3D point is calculated using the triangles around the vertex. We exploit these initial 3D meshes to partition visible points only.

To do partition, we consider each camera's viewing frustum. We discard the points which are outside viewing frustum of the camera. 6 planes surrounding the viewing frustum are calculated and the directions of normal vectors of the planes are set to face inside of the viewing frustum. A 3D point is in the viewing frustum, if the point is upper region of all the planes. This property is evaluated according to equation 1. For a plane L_i , a point p_k is in the lower region of the plane L_i , if γ is negative.

$$\begin{aligned}
 L_i &= a_i x + b_i y + c_i z + d_i & p_k &: (x_k, y_k, z_k) \\
 & a_i x_k + b_i y_k + c_i z_k + d_i = \gamma \\
 \gamma \geq 0 &: p_k \text{ is in the upper region} \\
 \gamma < 0 &: p_k \text{ is in the lower region}
 \end{aligned} \tag{1}$$

Then, we determine if a point in the viewing frustum is visible to the camera or not. For the points in the viewing frustum, visibility confidence is calculated as shown in equation 2.

$$\text{Visibility confidence} = -\frac{\text{Dot}(V_{cam}, N_k)}{d^2} \tag{2}$$

where, V_{cam} is the normalized vector of the camera's viewing direction and N_k is the normalized normal vector of the point. d is the distance between the camera and the point p_k . $\text{Dot}(V_{cam}, N_k)$ is dot product between two vectors.

If the point is visible to camera the confidence value is positive. If not, we assume that the point is not visible from current viewpoint. As shown in Fig 2, the point set S_2 is not visible to the camera C_1 , but is visible to C_2 . Even though it is not visible to C_1 ,

it is in the viewing frustum of C_1 . After the confidence value is calculated, the point set S_2 is culled out. Some part of S_3 is visible to C_1 and C_3 , simultaneously. In this case the point is assigned to the partition where the point has the largest confidence value. As a result of partitioning, we obtain sub-point clouds S_k s and each sub-point cloud S_k is associated to each camera C_k .

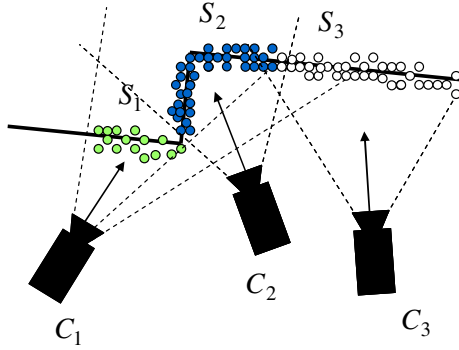


Fig. 2. 3D point cloud partition with viewing frustum

2.3 Adaptive Sampling

There are several points with different coordinates which represent the same point in the real scene in the registered point cloud, since there exist errors in range data and in registration. Thus, we need to remove overlapping points and to sample these points to generate a surface.

For the sub-point cloud S_k , we project it onto the camera's image plane using projection matrix of the camera. We create a grid with resolution of the image used in data acquisition step on the image plane. Each cell of the grid corresponds to a pixel of the image. Then, we search the points which are inside of a k^{th} cell, G_k . Since we know the corresponding 3D coordinates of projected points, the 3D coordinates of the 3D point that corresponds to the cell G_k is calculated from the projected points inside G_k as shown in Fig 3. The (G_{kx}, G_{ky}, G_{kz}) is the 3D point corresponding to the cell G_k .

Median value of each coordinate is calculated and assigned to G_k according to the equation 3.

$$\begin{aligned} G_{kx} &= \text{median}(\text{Seq}(x)) \\ G_{ky} &= \text{median}(\text{Seq}(y)) \\ G_{kz} &= \text{median}(\text{Seq}(z)) \end{aligned} \quad (3)$$

where, $\text{Seq}(x)$, $\text{Seq}(y)$ and $\text{Seq}(z)$ are the sequences of the each coordinates of the 3D points that correspond to the projected points inside the cell G_k .

As the resolution of the image used in data acquisition becomes larger, the number of points of the mesh model increases. However, large size mesh model is not desirable to be used in practical usage, since it requires much hardware resource for rendering and may result in low performance. If the surface the point cloud forms has

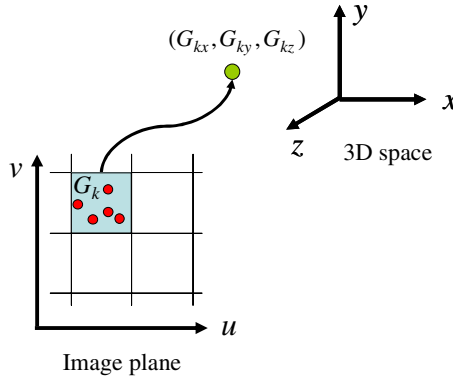


Fig. 3. Calculation of 3D coordinates of a cell

small variation, we can represent the surface with smaller number of points. In this paper, depth information is a key feature to be preserved. Thus, we apply adaptive sampling to simplify the triangulated model while preserving depth information of the scene. To reduce the number of points, we focus on the variation of the z values of the points. We adaptively sample the grid according to the depth variation. The gradient of G_{kz} is calculated with respect to the horizontal and vertical scanline as shown in equation 4. For each scan line, the absolute values of gradient of G_{uz} and G_{vz} are calculated. If δ , the root of squared sum of gradient in horizontal and vertical scanline, is smaller than threshold value, the point is cut out.

$$\frac{\partial G_{uz}}{\partial u} = |G_{(u+1)z} - G_{(u-1)z}| \tag{4}$$

$$\frac{\partial G_{vz}}{\partial v} = |G_{(v+1)z} - G_{(v-1)z}|$$

$$\delta = \sqrt{\left(\frac{\partial G_{uz}}{\partial u}\right)^2 + \left(\frac{\partial G_{vz}}{\partial v}\right)^2} \tag{5}$$

2.4 Triangulation and Mesh Integration

After adaptive sampling we have 2D point cloud and corresponding 3D point cloud in each partition of a view. To triangulate 3D point cloud, we triangulate 2D point cloud using Dealunay triangulation [10]. The 2D triangulation result is applied to corresponding 3D point cloud. Dealunay triangulation result in convex hull of a 2D point cloud, however, the expected result is not convex hull but the triangulation of 2D point cloud preserving its shape of boundary. Thus, we add 4 extreme points of which coordinates are the 4 corners in image coordinate system before triangulation and remove triangles that contain the extreme points after triangulation. For the

realism, we exploit images captured from cameras as textures. For texture mapping the index of a cell in the grid on the image plane is used as texture coordinates of the corresponding 3D point.

After sampled point clouds of all the partitions are triangulated, we merge them to one mesh model to represent whole indoor environment. Since there is no connectivity between two adjacent mesh models, there exist gap between partitions. The points on the boundaries of two adjacent mesh models are triangulated to merge them together. The points on the left and right boundary of a sampled point cloud are the points corresponding to the left-most and right-most cells of the grid on the image plane. For triangulation of two adjacent boundaries, we connect the points which have similar height in 3D space. Then, we triangulate the points the points which have no connection. The mesh integration process is depicted in Fig 4.

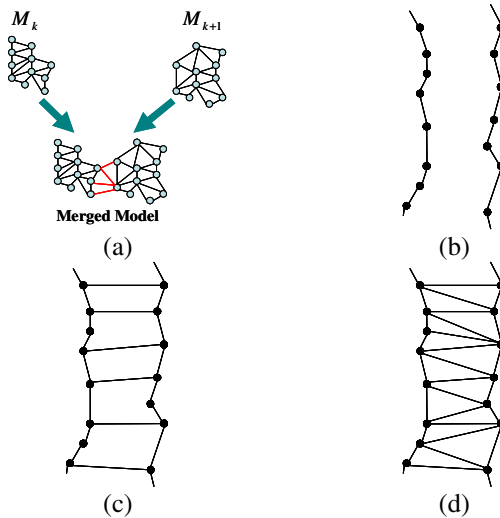


Fig. 4. Mesh integration process (a) Mesh merging of two adjacent models (b) The points on two adjacent boundaries (c) Connection of the points with similar height (d) Triangulation result

3 Experimental Results

To obtain 3D point cloud in each viewpoint, we used Digiclops which is a multi-view camera [9]. In this experiment, we modeled two walls of a room. The point clouds of the room to be modeled were obtained by moving a multi-view camera. There are overlapping areas between cameras' views. When we captured a scene, we used a planar pattern to calibrate the camera.

Fig 5 shows the result of partitioning input point clouds. After the visibility confidence values are calculated, the input point clouds are partitioned to sub-point clouds. Each partition is rendered with different color in Fig 5.

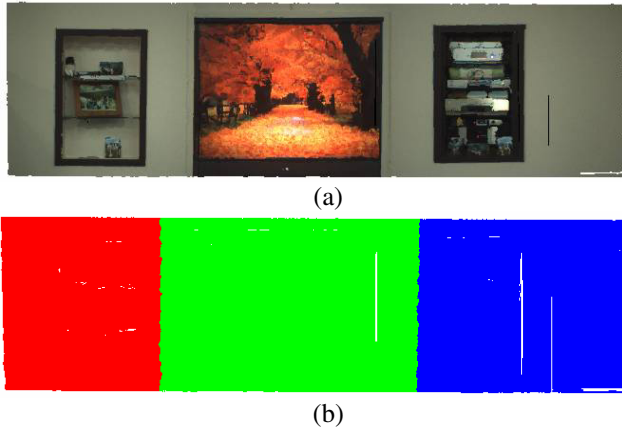


Fig. 5. Partitions of a wall of the room (a) Colored original point cloud (b) Partitioned point cloud

Fig 6 shows a triangulated model to which the adaptive sampling is not applied. The triangulated model is rendered in point and wire-frame in Fig 6(a) and Fig 6(b), respectively. The textured model is shown in Fig 6(c). Since there are many points and the triangles of the mesh model is too small, Fig 6(b) looks like a surface even though it is rendered in wire-frame mode.

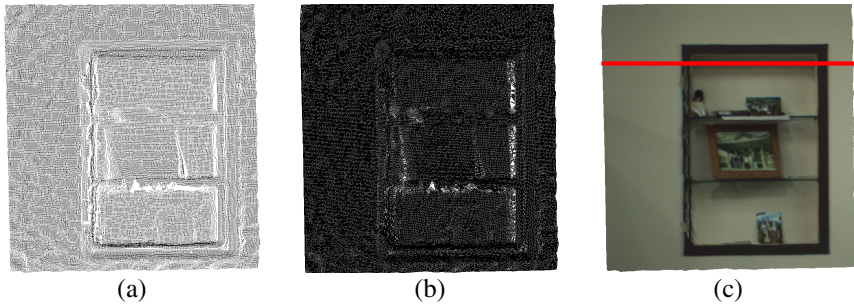


Fig. 6. Triangulated model before adaptive sampling (a) Point model (b) Wire-frame model (c) Textured model

The result of adaptive sampling is shown in Fig 6. Fig 7(a), Fig 7(b) and Fig 7(c) show the point model, wire-frame model and textured model respectively. Note that our key feature, depth information, is preserved and many of points on the smooth area are removed. The adaptive sampling reduces the number of points of the mesh model while preserving the features of the scene. Texture mapping improves the visual quality of the simplified model. Consequently, even though a number of points are removed, the visual quality of the model is not much different from the original model.

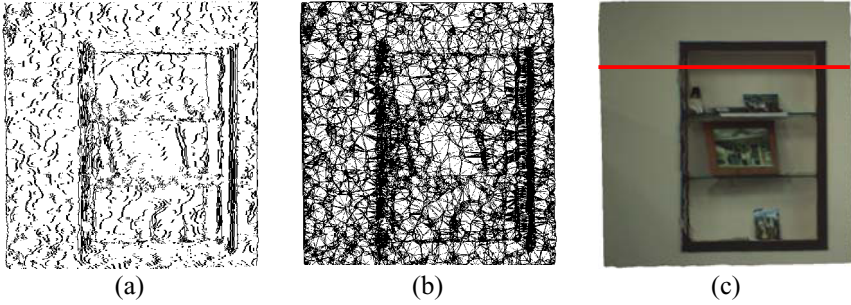


Fig. 7. Triangulated model after adaptive sampling (a) point model (b) wire-frame model (c) textured model

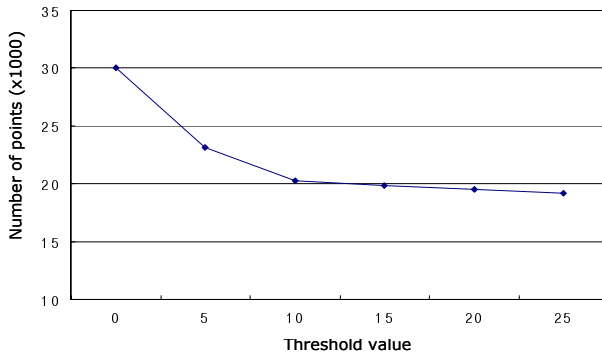


Fig. 8. The number of points of sampled models with different threshold values

Fig 8 shows the number of points with different threshold values. The number of points of the mesh model decreases as the threshold value increases. However, reduction rate decreases as the threshold value increases.

We applied our adaptive sampling algorithm to the red scanline in Fig 6(c) and the result is shown in Fig 9. Fig 9(a) shows the distribution of z axis values of the points on the scanline. Fig 9(b) is the gradient of G_z in horizontal direction. According to the gradient of the z coordinates associated to a grid on an image plane, the points with small gradient values are removed and the points with large variation in z axis are preserved as shown in Fig 9(d). If we select too large threshold value, the precise depth information is lost. We set the threshold value 10cm in this experiment based on our experimental result. In Fig 9, the gradient values are scaled to see the result clearly.

In Fig 10, shows the bookshelf of the scene. Since the reconstructed model has 3D information, we can see that the doll on the bookshelf becomes to be occluded as the rotation angle increases. This is one of major differences from the 2D image-based panorama which is constructed by stitching several 2D images. In 2D image based panorama the scene is static and it is not possible to see different scene according to viewpoint. Thus, users can navigate the reconstructed VE with depth information. This property of 3D panoramic VE provides more realistic feeling to users.

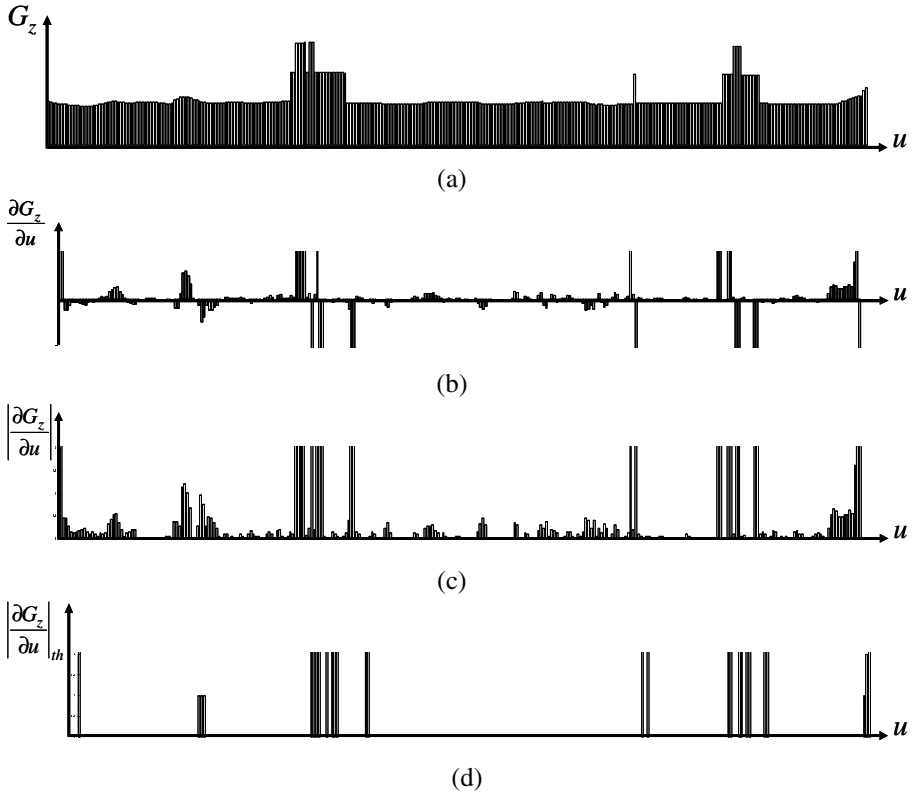


Fig. 9. The result of adaptive sampling of a scan line (a) Raw data of coordinates in z axis (b) Gradient of z coordinates with respect to u (c) Absolute value of gradient (d) Thresholded values

The integrated indoor scene model is shown in Fig 10. The floor is added manually. Fig 11(a) is the bird’s-eye-view and Fig 11(b), Fig 11(c) and Fig 11(d) are magnified view of parts of the panoramic VE model. As shown in the magnified views, the reconstructed indoor scene model is photo-realistic enough to be used in virtual reality applications.

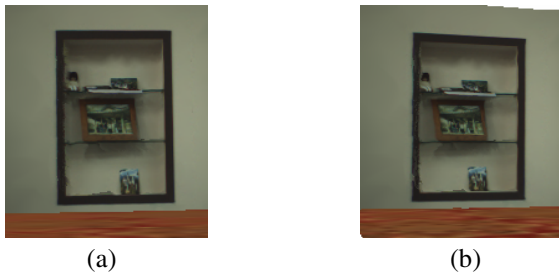




Fig. 10. Depth information of the panoramic mesh model (c) 15° rotation (d) 20° rotation

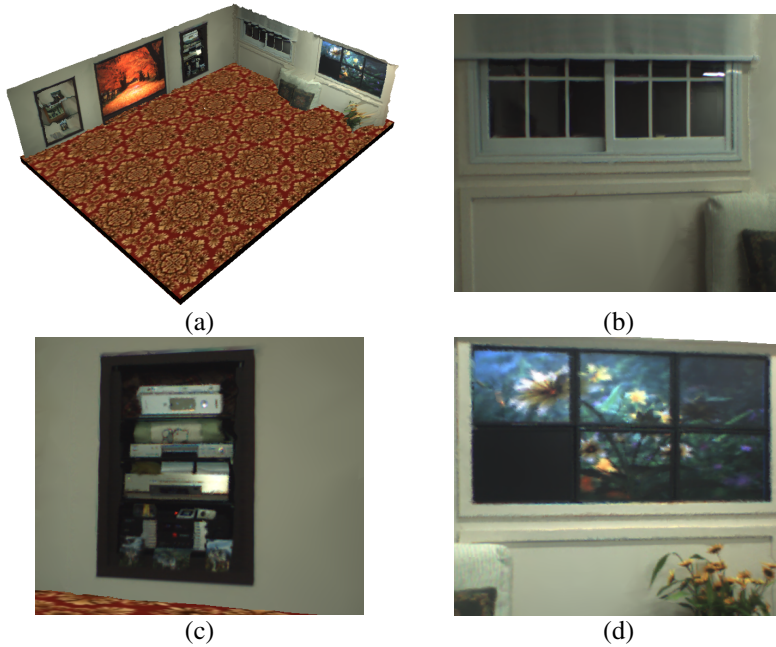


Fig. 11. Modeling result of an indoor scene (a) Bird's-eye-view (b) Window part (c) AV rack part (d) Tiled-display part

4 Conclusions and Future Work

In this paper, we proposed panoramic mesh modeling method from multiple range data for indoor scene reconstruction. The registered 3D point cloud is the input to the algorithm and a panoramic mesh model of indoor scene is generated. The input point cloud is partitioned to sub-point clouds. Each sub-point cloud is sampled and triangulated. After that, we merge all triangulated models of sub-point clouds to represent the whole indoor scene as one model. Our mesh modeling method considers occlusion between two adjacent views and it filters out invisible part of point cloud without any prior knowledge. While preserving the features of the scene, adaptive sampling

reduces the size of resulting mesh model for practical usage. Depth information of the scene is preserved. The proposed method is modularized and applicable to the other modeling applications which handle multiple range data. As future work, we are going to improve our adaptive sampling to create smooth surface from point clouds with smaller number of points. The other work to be done is applying our modeling algorithm to complex scene which contains many objects in our daily life.

References

1. Vitor Sequeira, João Goncalves, M.Isabel Ribeiro, "3D Reconstruction of Indoor Environments", ICIP96, pp.405-408, Lausanne, Switzerland, 1996
2. Y.Sun, J.K.Paik, A.Koschan, and M.A.Abidi, "3D reconstruction of indoor and outdoor scenes using a mobile range scanner", Pattern Recognition, vol.3, pp 653–656, 2002
3. Y. Sun, J. Paik, A. Koschan, and M. Abidi, "Surface modeling using multi-view range and color images," Integrated Computer-Aided Engineering, Vol. 10, No. 1, pp. 37-50, February 2003
4. Johnson, S. Kang, "Registration and Integration of Textured 3-D Data" Tech. report CRL96/4, Digital Equipment Corporation, Cambridge Research Lab, 1996
5. McMillan, L., G. Bishop, "Plenoptic Modeling: An Image-Based Rendering System", Proceedings of SIGGRAPH 95, pp. 39-46, 1995
6. H.Y. Shum, M. Han, and R. Szeliski, "Interactive construction of 3d models from panoramic mosaics", In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'98), pp 427-433, 1998
7. S. Kim and W. Woo, "Projection-based Registration using Multi-view camera for Indoor Scene Reconstruction," 3-D Digital Imaging and Modeling (3DIM), pp. 484-491, 2005
8. P. Fua. From Multiple Stereo Views to Multiple 3-D Surfaces. International Journal of Computer Vision, 24(1), pp. 19-35, August 1997
9. PointGrey Research, <http://www.ptgrey.com>
10. H. Edelsbrunner. "Algorithms in Computational Geometry", Springer-Verlag, New York, 1987
11. R. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision", 2nd Ed. Cambridge University Press(2004)
12. V. Sequeira, K. C. Ng, E. Wolfart, J.G.M Gonçalves, and D.C. Hogg., "Automated 3D reconstruction of interiors with multiple scan-views", Proceedings of SPIE, Electronic Imaging '99, IS&T/SPIE's 11th Annual Symposium, 1999
13. V. Sequeira, J. Goncalves, and M.Isabel Ribeiro, "High-Level Surface Description from Composite Range Image", Proceedings of the 1995 IEEE International Symposium on Computer Vision, Florida, USA, pp.163-168, November 1995
14. S. Elgazzar, R. Liscano, F. Blais, and A. Miles, "3D Data Acquisition for Indoor Environment Modeling Using a Compact Active Range Sensor", Proceedings of IMTC97, 'Sensing, Processing, Networking', Vol. 1, pp 586-592, 1997
15. McMillan, L., and G. Bishop, "Plenoptic Modeling: An Image-Based Rendering System", Proceedings of SIGGRAPH 95, pp. 39-46, 1995