# Real-Time Acquisition and Representation of 3D Environmental Data

Se-Ho Lee, Seong-Gyun Jeong, Tae-Young Chung, and Chang-Su Kim
School of Electrical Engineering, Korea University, Seoul, Korea
E-mail: coolhandluke@korea.ac.kr, sg_jeong@korea.ac.kr, tychung@mcl.korea.ac.kr, changsukim@korea.ac.kr

***Abstract*—We develop a mobile system to acquire and represent 3D environmental data for modeling indoor spaces. The system is composed of a laser range finder (LRF) and an omni-directional camera. Multiple 3D point clouds from different viewpoints are acquired as the geometric information by scanning a scene with the LRF, while an omni-directional texture image is acquired with the omni-directional camera. We merge those multiple 3D point clouds into a single point cloud. We then combine the point cloud and the texture image into a complete 3D mesh model in three steps. First, we downsample the point cloud based on a voxel grid and estimate the normal vector of each point. Second, using the normal vectors, we reconstruct a 3D mesh based on the Poisson surface reconstruction. Third, to assign texture information to the mesh surface, we estimate the matching region in the omni-directional image that corresponds to each face of the mesh. Simulation results demonstrate that the proposed system can reconstruct indoor spaces effectively.**

## I. Introduction

Representing indoor spaces as 3D models is necessary in various robotics applications. Whereas many researches on the 3D representation of objects have been carried out [1], [2], there have been only limited researches on the 3D representation of large indoor spaces, since it is difficult to acquire and register a huge amount of 3D data.

Various approaches to 3D mapping have been proposed in robotics. In [3]-[7], 3D map generation techniques have been proposed. However, they assume that the pose information of sensors is given. In [8], a system, composed of a range sensor and eight CCD cameras, was developed to acquire 3D data. The calibration of the eight cameras was performed to register data. Note that 3D maps can be represented by flat surfaces [9] or 3D meshes [10], [11]. In [9], 3D mapping was done through the data clustering and the plane fitting. This approach has low complexity, but is limited to planar structures only. On the other hand, using 3D meshes demands high complexity, but it can represent non-planar structures more faithfully. Thus, we represent indoor spaces as 3D meshes in this work.

Our system is composed of a laser range finder (LRF) and an omni-directional camera. Using the system, we can acquire 3D environmental data, which represent the geometric information as 3D point clouds and the texture information as an omni-directional image, respectively. The point clouds are acquired by scanning a scene with the LRF, while the omni-directional image is captured by the omni-directional camera. After acquiring 3D environmental data, we represent them in a 3D mesh. First, we downsample the point clouds using a voxel grid to remove outliers and regularize the points. Second, we compute the normal vectors of the downsampled points and reconstruct a 3D mesh surface using the Poisson reconstruction technique. Finally, we assign texture information to the mesh surface using the omni-directional image. Fig. 1 is the flow chart of the proposed algorithm.

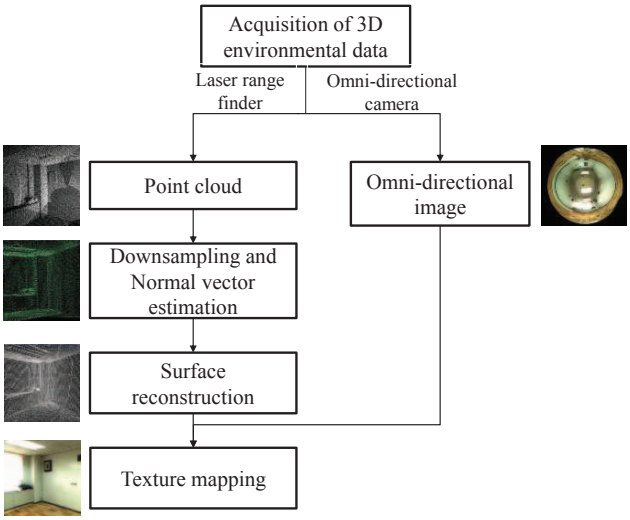



Fig. 1. Flow chart of the proposed 3D mapping system.

The rest of this paper is organized as follows. Section II explains the acquisition of 3D environmental data. Section III proposes the mesh representation scheme for 3D environmental data. Section IV provides experimental results. Finally, Section V gives concluding remarks.

## II. Acquisition of 3D Environmental Data

Fig. 2 shows the proposed system, composed of a LRF and an omni-directional camera. By forming a laser beam, the LRF records the distances from the sensor to objects as a point cloud. The omni-directional camera provides texture information of the whole space. Note that the proposed system

This work was supported partly by the Global Frontier R&D Program on <Human-centered Interaction for Coexistence> funded by the National Research Foundation of Korea grant funded by the Korean Government(MEST) (NRF-M1AXA003-2011-0031648), and partly by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST) (No. 2012-011031).

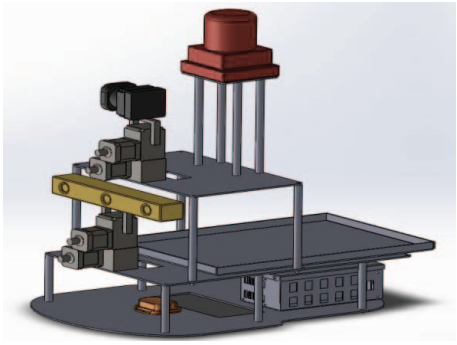


Fig. 2. The data acquisition system.

is similar to [8]. However, the proposed system makes it easy to calibrate the geometric data and the texture data, since only the calibration between the LRF and the omni-directional camera is required.

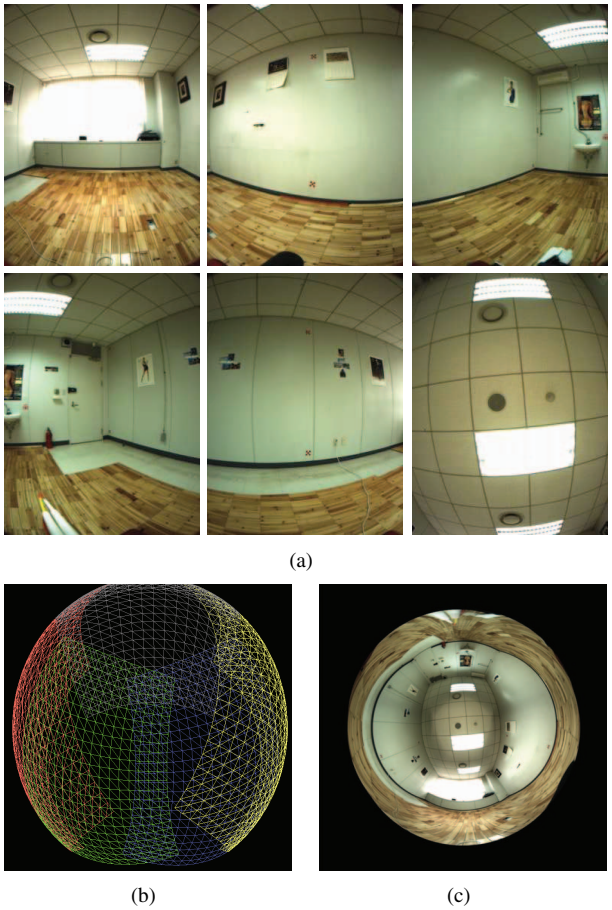
### A. 3D Point Clouds

The LRF can acquire a 3D point cloud for a limited area only because of the narrow field of view. It is necessary to acquire 3D point clouds several times in order to cover the entire indoor space. For example, by acquiring a 3D point cloud eight times, while rotating the LRF 45 degrees after each repetition, we can capture the entire 3D geometric information.

We merge these 3D point clouds into a single 3D point cloud for compact representation. To merge the clouds, the extrinsic information of the LRF, which includes the rotation and translation matrices from the reference coordinate system, should be estimated. We adopt the patch-based pose estimation algorithm in [12] to estimate the rotation and translation matrices. Then, by rotating and translating each point cloud, we acquire the merged 3D point cloud set.

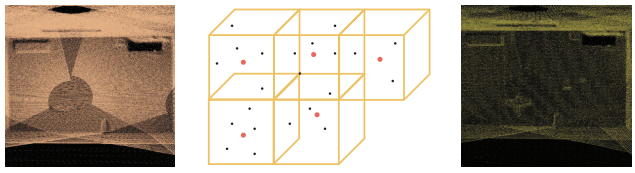
### B. Texture Maps

The omni-directional camera consists of multiple cameras, which capture images at different directions, covering 360 degrees. These images should be stitched and blended. We adopt the method in [13], [14] to acquire the omni-directional image. For example, Fig. 3(a) shows multiple images captured by the omni-directional camera. Each image is projected onto a 3D mesh in Fig. 3(b). The polygon colors in Fig. 3(b) denote the source images in Fig. 3(a). The overlapped regions between the images are blended and smoothed to construct the omni-directional image in Fig. 3(c). This omni-directional image is used to reconstruct 3D mesh texture.



(a) (b) (c)

Fig. 3. An example of the omni-directional image acquisition: (a) images from individual cameras, (b) the projection of those images onto a dome-shaped 3D mesh, in which the colors of polygons denote the source images, and (c) the stitched omni-directional image.



(a) (b) (c)

Fig. 4. Downsampling a 3D point cloud: (a) an original 3D point cloud, (b) a voxel grid for the downsampling, in which red points depict the centroids of the voxels, and (c) the downsampled 3D point cloud.

## III. 3D Mesh Representation

### A. Point Cloud Downsampling

Fig. 4(a) shows a merged 3D point cloud. The distribution of points is non-uniform due to the overlapping of the multiple source point clouds. The non-uniform distribution can lead to inaccurate normal vector estimation [15], which is necessary to reconstruct the structure information from unorganized 3D points. Moreover, too many points result in high complexity for the normal vector estimation. Therefore, we downsample and regularize the 3D point cloud.

We adopt the voxel-based downsampling [16]. First, we divide the point cloud using a voxel grid. Then, we calculate the centroid of the 3D points within each voxel. We use the centroid as the representative geometric information, instead of the original 3D points. During the downsampling, we also remove outliers in the 3D point cloud. Fig. 4 illustrates the downsampling process.

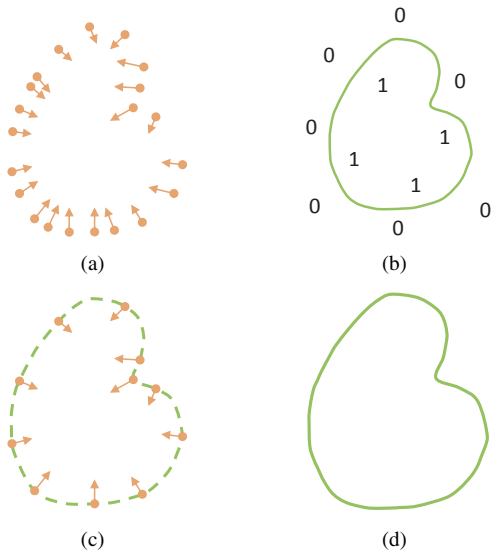



Fig. 5. Poisson surface reconstruction: (a) a set of normal vectors $\mathbf{N}$, (b) an indicator function $\zeta$, (c) the gradient of the indicator function $\nabla\zeta$, and (d) the reconstructed surface.

### B. Normal Vector Estimation

Assuming that a point and its neighboring points, $\mathbf{x}_i, i = 1, 2, ..., k$, lie on the same plane, we can estimate the normal vector of each point using the least-squares plane fitting [17]. Each plane can be represented with a point $\mathbf{c}_x$ and a normal vector $\mathbf{n}_x$, which can be estimated by minimizing the sum of the squared distances from the points to the plane. The distance from a point $\mathbf{x}_i$ to the plane is given by $(\mathbf{x}_i - \mathbf{c}_x) \cdot \mathbf{n}_x$. In the least-squares fitting, $\mathbf{c}_x$ is the centroid of the points, given by

$$\mathbf{c}_x = \overline{\mathbf{x}} = \frac{1}{k}\sum_{i=1}^{k}\mathbf{x}_i. \tag{1}$$

Then, $\mathbf{n}_x$ is computed from the eigenvalue decomposition

$$\Sigma\mathbf{u} = \lambda_j\mathbf{u}, \quad j \in 0, 1, 2 \tag{2}$$

where

$$\Sigma = \frac{1}{k}\cdot\sum_{i=1}^{k}(\mathbf{x}_i - \mathbf{c}_x)\cdot(\mathbf{x}_i - \mathbf{c}_x)^T \tag{3}$$

denotes the covariance matrix of $\mathbf{x}_i$, and $\mathbf{u}$ and $\lambda$ denote the eigenvector and the eigenvalue, respectively. Then, $\mathbf{n}_x$ is given by the eigenvector $\mathbf{u}_0$ corresponding to the smallest eigenvalue $\lambda_0$. From Eqs. (2) and (3), we can estimate the normal vectors of all points. These normal vectors are used to reconstruct a 3D surface.

### C. Surface Reconstruction

Based on the Poisson surface reconstruction algorithm [17], we extract a 3D surface from points and their normal vectors. Suppose that points and normal vectors are given as in Fig. 5(a). Then, we define an indicator function $\zeta$, which assigns value 1 to the inside of the surface, and value 0 to the outside, in Fig. 5(b). If we assume that the points lie near the surface, the gradient of the indicator function should be similar to the normal vectors of the points [17], as shown in Fig. 5(c). Therefore, we estimate the indicator function by

$$\min_{\zeta}\|\nabla\zeta - \mathbf{N}\| \tag{4}$$

where $\mathbf{N}$ denote the set of normal vectors. The perfect solution, which makes the norm in Eq. (4) zero, may not exist, since $\mathbf{N}$ is often not integrable. Instead, we use the divergence operator and obtain an approximation indicator function by solving

$$\triangle\zeta = \nabla\cdot\mathbf{N} \tag{5}$$

Then, we select an appropriate isovalue by averaging the indicator function. Finally, we extract the corresponding iso-surface solution and represent it as a triangle mesh using the marching cubes algorithm [18]. Fig. 5(d) shows the reconstructed surface. However, the Poisson surface reconstruction may generate erroneous triangle meshes. To overcome this issue, we discard a triangle mesh, if it is far from the sampled point cloud.

### D. Texture Mapping

For texture mapping, we estimate correspondences between the reconstructed 3D mesh surface and the omni-directional image. Let $(u, v)$ and $(X, Y, Z)$ denote the pixel location at the omni-directional image and its corresponding position in the 3D domain, respectively. Fig. 6(a) illustrates the relationship between $(u, v)$ and $(X, Y, Z)$. According to [19], $(u, v)$ can be expressed as

$$u = \phi\frac{X}{r}, \quad v = \phi\frac{Y}{r}, \tag{6}$$

where $R = \sqrt{X^2+Y^2+Z^2}$, $r = \sqrt{X^2+Y^2}$, and $\phi = \arccos\frac{Z}{R}$. From Eq. (6), we extract the texture information for each triangular face on the mesh surface. Then, by mapping the texture to the face, we finally represent the 3D environmental data, as illustrated in Fig. 6(b).

## IV. Experimental Results

We evaluate the performance of the proposed system by acquiring 3D environmental data for an indoor room and representing them with a 3D mesh. We rotate the LRF about 40 degree after each scanning step to obtain overlapping point clouds. Simultaneously, we capture multiple images and acquire the omni-directional image by stitching these images.

Fig. 7 shows 3D point clouds at nine different directions in the room. Each 3D point cloud has 133,760 points. We merge them into a single point cloud, which is shown in Fig. 8. The merged point cloud consists of 1,203,840 points. We use the omni-directional camera 'Ladybug 2,' which captures six images at different directions. Fig. 9 shows the omni-directional image of the room.

Fig. 10(a) shows the downsampled point cloud obtained by the voxel-based downsampling method. It is composed of 163,293 points, which correspond to 5.98% of the original data. We can see that the points are uniformly distributed. Fig. 10(b) shows the reconstructed surfaces. The mesh contains

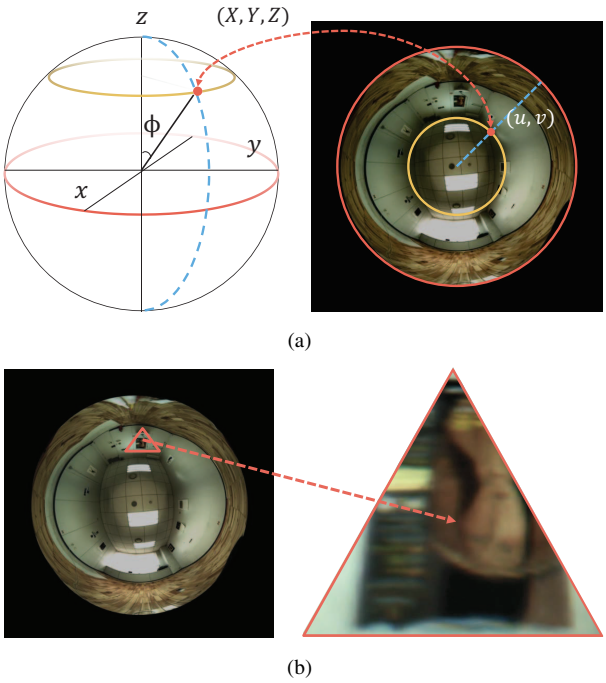



Fig. 6. Texture mapping: (a) the relationship between a 3D point $(X, Y, Z)$ and its corresponding point $(u, v)$ in the omni-directional image and (b) the texture mapping to a triangular face.

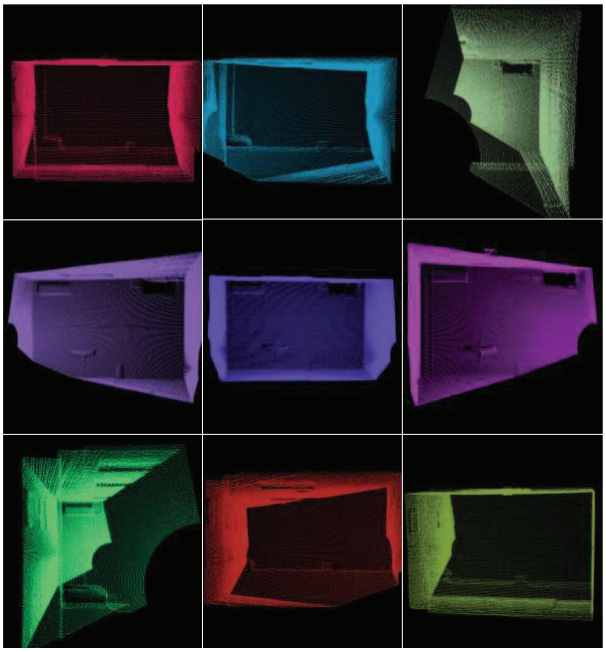


Fig. 7. Acquired 3D point clouds at nine different directions in a room.

72,040 and 142,559 triangles. Fig. 10(c) shows the texture-mapped surfaces, which represent the indoor environment faithfully.

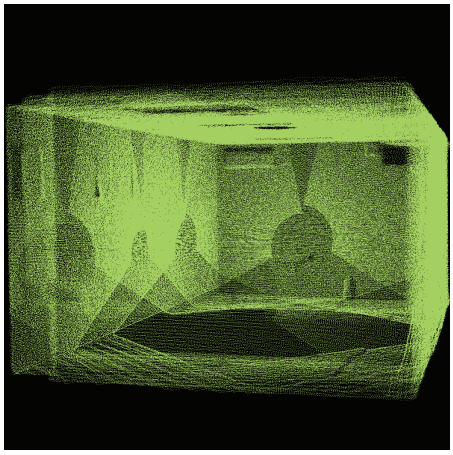


Fig. 8. The merged point cloud.

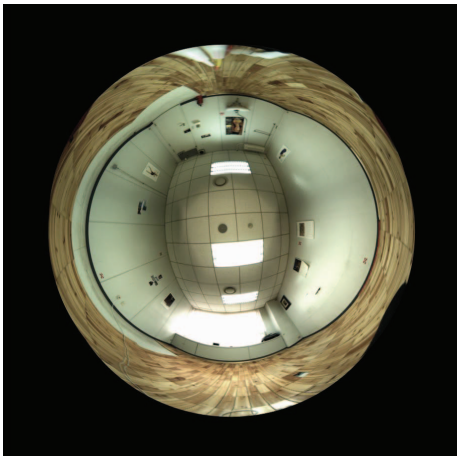


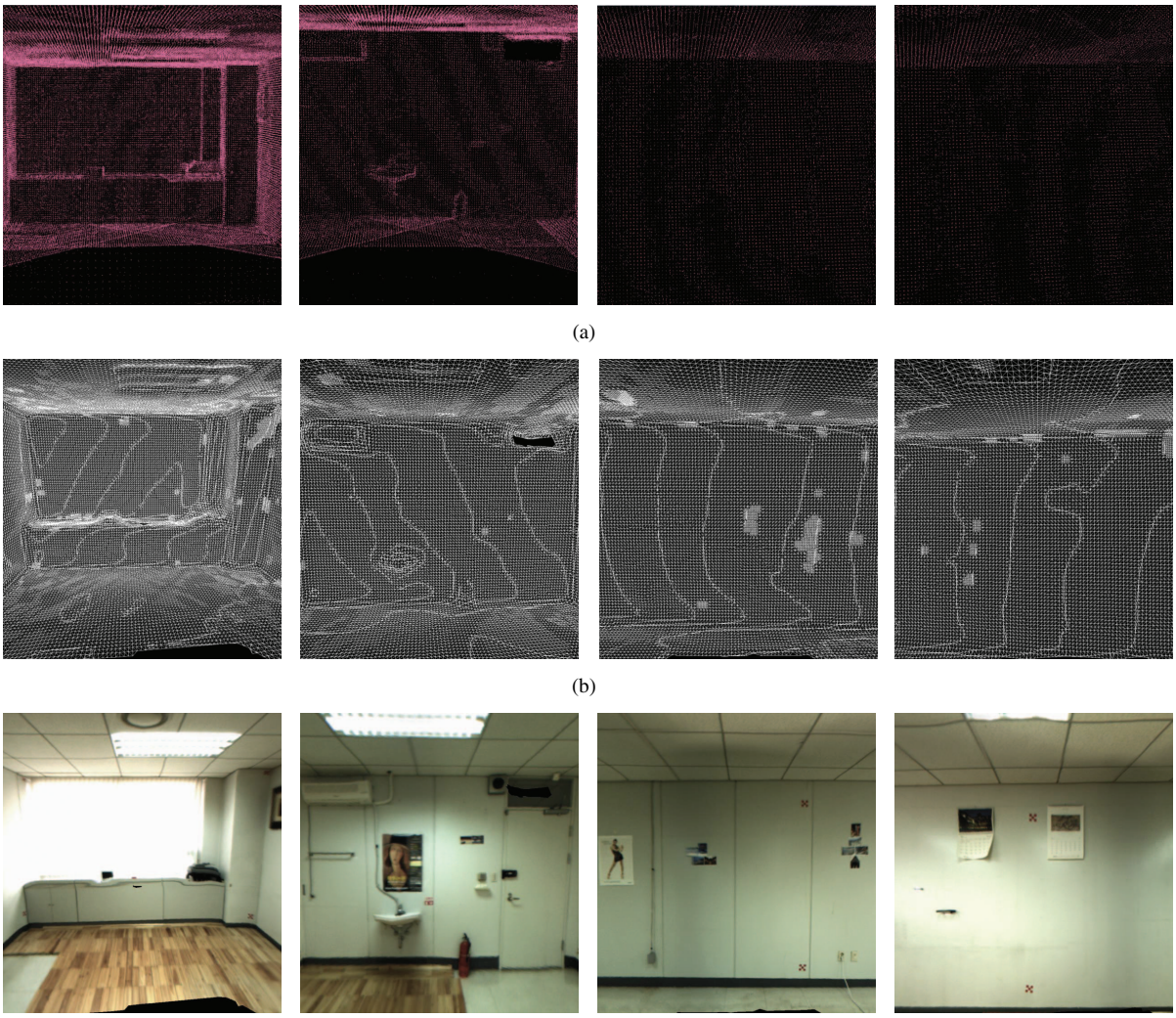
Fig. 9. The omni-directional image of the room.

## V. Conclusions

We proposed a framework to acquire and represent 3D environmental data. We developed a system equipped with an LRF and an omni-directional camera. The proposed algorithm reconstructs a mesh surface from LRF point clouds and assign the texture information to the surface, by employing the color information from the omni-directional image. For the surface reconstruction, the proposed algorithm first decimates 3D point clouds using a voxel grid and estimates the normal vector of each point. Then, by solving the Poisson equation, we reconstruct the surface. For the texture mapping, the proposed algorithm uses the geometric relationship between the 3D point cloud and the corresponding pixel positions in the omni-directional image. Experimental results demonstrated that the system can acquire indoor scenes and represent them faithfully.

## References

[1] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. "Reconstruction and representation of 3D objects with radial basis functions," in *Proc. Computer Graphics and Interactive Techniques,* pp. 67-76, 2001.



(a) (b) (c)

Fig. 10. (a) Downsampled point clouds, (b) reconstructed surfaces, and (c) texture-mapped surfaces.


[2] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. "Surface reconstruction from unorganized points," *Computer Graphics,* vol. 26, no. 2, pp. 71-78, Jul. 1992.

[3] P. Allen, and I. Stamos, "Integration of range and image sensing for photorealistic 3D modeling," in *Proc. IEEE Int. Conf. Robotics and Automation,* pp. 1435-1440, 2000.

[4] R. Bajcsy, G. Kamberova, and L. Nocera, "3D reconstruction of environments for virtual reconstruction," *IEEE Workshop on Applications of Computer Vision,* 2000.

[5] S. Becker, M. Bove, "Semiautomatic 3-D model extraction from uncalibrated 2-D camera views," *SPIE Symp. Electronic Imaging,* 1995.

[6] P. Debevec, C. Taylor, J. Malik, "Modeling and rendering architecture from photographs," *Int. Conf. Computer Graphics and Interactive Techniques,* 1996.

[7] H. Shum, M. Han, R. Szeliski, "Interactive construction of 3d models from panoramic mosaics," *in Proc. Int. Conf. Computer Vision and Pattern Recognition,* 1998.

[8] S. Hakim, P. Boulanger, and F. Blais, "A mobile system for indoors 3-d mapping and positioning," *Fourth Conf. Optical 3-D Measurement Techniques,* Sept. 1997.

[9] J. Chen, and B. Chen "Architectural modeling from sparsely scanned range data," *International Journal of Computer Vision,* vol. 78, no. 2-3, pp. 223-236, 2008.

[10] S. Thrun, "A probabilistic online mapping algorithm for teams ofmobile robots," *Int. J. Robot. Res.,* vol. 20, no. 5, pp. 335-363, 2001.

[11] M. Montemerlo, D. Hähnel, D. Ferguson, R. Triebel, W. Burgard, S. Thayer, W. Whittaker, and S. Thrun, "A system for three-dimensional robotic mapping of underground mines," Comput. Sci. Dept., Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep. CMU-CS-02-185, 2002.

[12] K. Pathak, A. Birk, N. Vaškevičius, and J. Poppinga, "Fast registration based on noisy planes with unknown correspondences for 3-D mapping," *IEEE Trans. Robotics* vol. 26, no. 3, pp. 424-441, June 2010.

[13] Shree K. Nayar, "Catadioptirc omnidirectional camera," in *Proc. Int. Conf. Computer Vision and Pattern Recognition,* Jun. 1997.

[14] http://www.ptgrey.com/support/kb/printable.asp?a=4&q=250&ST=

[15] R. Rusu, "Semantic 3D object maps for everyday manipulation in human living environments," Ph.D. dissertation, Computer Science department, Technische Universität München, Germany, October 2009.

[16] S. Azernikov, and A. Fischer. "Surface reconstruction of freeform objects based on hierarchical space decomposition," *International Journal of Shape Modeling,* vol. 9, no. 2, pp. 177-190, 2003.

[17] M. Kazhdan, M. Bolitho, and H. Hoppe. "Poisson surface reconstruction," in *Proc. Eurographics Symposium on Geometry Processing,* Sept. 2006.

[18] W. Lorensen, and H. Cline, "Marching cubes: a high resolution 3D surface construction algorithm," *Computer Graphics,* vol. 21, no. 4, pp. 163-169, 1987.

[19] http://www.ptgrey.com/