# An Efficient RGB-D Camera Based Point Cloud Registration Algorithm

Chih-Hung Huang, Shang-Jhih Jhang and Chi-Yi Tsai*
Department of Electrical and Computer Engineering,
Tamkang University,
New Taipei City 251, Taiwan R.O.C.
E-mail: *chiyi_tsai@mail.tku.edu.tw

**Abstract—**This paper proposes an indoor scene three-dimensional (3D) reconstruction system using pan-tilt platform and RGB-D camera. The proposed system can automatically reconstruct 3D indoor scenes on a fixed position. An efficient point cloud registration algorithm is proposed to align point clouds based on extrinsic parameters of the RGB-D camera from every presetted pan-tilt control points. Then, a local registration method is performed to refine the alignment result. Experimental results verify the quality and efficiency of the proposed point cloud alignment algorithm by comparing with a state-of-the-art method.

***Keywords—Image registration; scene reconstruction; point cloud alignment; multi-view calibration; RGB-D cameras***

## I. INTRODUCTION

3D scene reconstruction is a very important topic in many applications of robot vision, such as environment mapping [1], scene recognition [2], augmented reality [3,4] and SLAM [5,6]. In the topic of 3D scene reconstruction, the existing methods usually utilize numerous sensors such as stereo camera, RGB-D camera, Time-of-flight (TOF) camera, laser scanner and Lidar, etc. RGB-D cameras use structured light technique to obtain 3D point clouds, each of them obtained from different time step has its individual coordinate system. Here, we refer it as camera's coordination $C_i$ at time step $i$. In the 3D reconstruction system, we aim to transform all the $C_i$ into a common world coordinate $W$. Therefore, to define a world coordinate as the mapping target for all coordinates of $C_i$, we also need transformation matrices that map each $C_i$ onto the coordinate $W$. In our method, the transformation matrices are obtained using the extrinsic parameters of the camera to produce a coarse reconstruction result, then we use one of the existing point cloud local registration methods to refine the global 3D reconstruction result.

There are many local registration methods proposed in literature. Some local registration and global registration methods were introduced in [7,8]. There are four kinds of local registration methods, and we use iterative closest point (ICP) method to carry out our research. ICP is a method proposed by Beal et al. [9], which presents a three-dimensional shape that can be used for free-form curves and surfaces for efficient registration. Later, there have been many extension of ICP methods. Rusinkiewicz et al. [10] proposed an alternative to ICP that uses a method based on uniform normal spatial sampling variables. This method has a better convergence for smaller scenes and sparse features. Low et al. [11] proposed an approximate nonlinear least-squares optimization method to solve the solution of point-to-plane ICP, which improve the efficiency for finding the best solution. Segal et al. [12] proposed a new plane-to-plane ICP method that combines the original ICP and point-to-plane ICP methods with better experimental results than the previous two. Serafin and Grisetti [13] proposed a new optimization cost function, not only includes point-to-point distance, but also the surface normal vector or surface tangent, the results not only shows faster speed of convergence, it also has better robustness. Microsoft has made great contributions in terms of application. [14] proposed a system called Kinect Fusion, the system is based on GPU acceleration ICP method.

In this paper, an indoor scene 3D reconstruction method is proposed. The proposed method includes an online and an offline operation. In the rest of this paper, Section 2 presents system architecture of the proposed method. Then the implementation detail of the system is introduced in Section 3. The experimental results are presented in Section 4. The performance of the proposed method is validated by comparing with a state-of-the-art method. The contribution of this work is summarized in Section 5.

## II. SYSTEM ARCHITECTURE

In this section, we introduce the system architecture of our proposed method. The purpose of point cloud registration is to find the 3D transformation between two point clouds. The transformation can be expressed by a rotation $\mathbf{R}$ and a translation $\mathbf{t}$. To obtain the rotation $\mathbf{R}$ and translate $\mathbf{t}$, there are two different point registration methods: global registration and local registration. To illustrate these two methods, assuming that there are two point clouds $\mathbf{p}_i \in \mathbf{P}$ and $\mathbf{q}_j \in \mathbf{Q}$, where $\mathbf{p}_i = [x_i, y_i, z_i]^T$ is $i$-th data point in point cloud $\mathbf{P}$, $\mathbf{q}_j = [x_j, y_j, z_j]^T$ is $j$-th data point in point cloud $\mathbf{Q}$. Then we define an Euclidean distance between two data points $\|\mathbf{p}_i - \mathbf{q}_j\|$. To reduce this distance, a cost function can be defined with respect to the rotation $\mathbf{R}$ and translation $\mathbf{t}$ such that

$$E = \sum_{k=1}^{N} \left\|(\mathbf{R}\mathbf{p}_k + \mathbf{t}) - \mathbf{q}_k\right\|^2 , \tag{1}$$

 *ISBN 978-1-5090-4897-7*

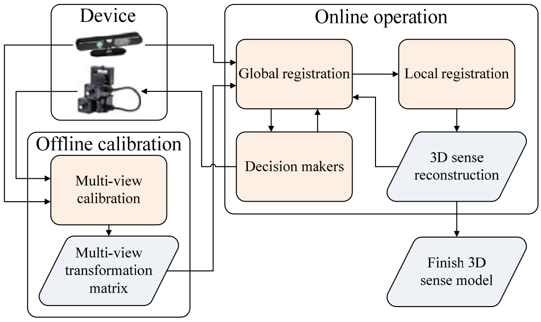


Figure 1. The proposed system architecture, which includes device, offline calibration, and online operation.

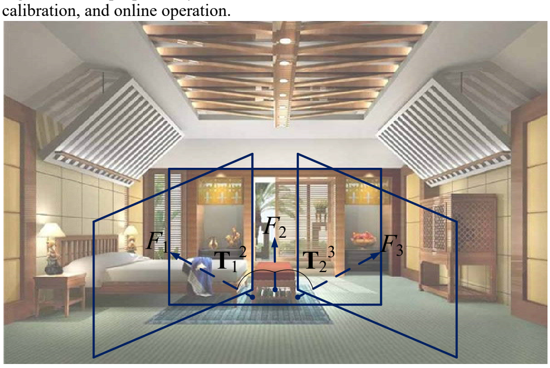

Figure 2. Illustration for the shooting scene; the blue rectangle is the image plane.

where $E$ is sum of errors over all $N$ points.

Local registration requires two initial point clouds **P** and **Q** to be close enough, so that it can find the rigid body transformation of **P** and **Q**. The traditional ICP method has been popular in recent years, it defines the cost function $E_{local}$ as follows

$$E_{local} = \sum_{k=1}^{N} \left\| \left( \mathbf{R}\mathbf{p}_k^l + \mathbf{t} \right) - \mathbf{q}_k^l \right\|^2 , \tag{2}$$

where $\mathbf{p}_k^l$ and $\mathbf{q}_k^l$ are corresponding points that are close enough, then it could be minimized using optimization. The advantage of the ICP method is that it can find the optimal transformation with higher precision, but it require closer initial distance between two point clouds.

Global registration can start from two point clouds with any posture, regardless of the initial distance. There have been many methods that use global registration to find corresponding points between point clouds or use geometric method to find the local corresponding points (the 4PCS [15]). After finding the corresponding points between **P** and **Q**, the cost function $E_{global}$ is then defined as

$$E_{global} = \sum_{k=1}^{N} \left\| \left( \mathbf{R}\mathbf{p}_k^g + \mathbf{t} \right) - \mathbf{q}_k^g \right\|^2 , \tag{3}$$

where $\mathbf{p}_k^g$ and $\mathbf{q}_k^g$ are corresponding points, than it can be minimized using optimization. The advantage of global registration is that it is possible to perform point cloud alignment from any location, but its computation complexity is usually higher than the local registration.

Figure 1 shows the proposed 3D scene reconstruction system architecture. First, the point cloud data for scene reconstruction need to be obtained from the device sensor. We rotate the RGB-D camera through pan-tilt platform to capture a portion of the scene. As shown in Figure 2, a set of frames $F_1$, $F_2$, $F_3$ was taken when rotating the camera, where $\mathbf{T}_1^2$ is the angle transformation matrix from $F_1$ to $F_2$. $\mathbf{T}_2^3$ is the transformation matrix from $F_2$ to $F_3$, $F_i$ includes RGB and depth information, which is then converted into a color point cloud $\mathbf{P}_i=\{\{\mathbf{p}_1,\mathbf{c}_1\},\{\mathbf{p}_2,\mathbf{c}_2\},\ldots\}$, where $\mathbf{p}_j=[x_j,y_j,z_j]^T$ is the $j$-th point position in the point cloud (in the $F_i$ coordinate), $\mathbf{c}_j=[r_j,g_j,b_j]^T$ is the color of the $j$-th point in the point cloud. Since each $F_i$ has its own coordinate system, it is necessary to convert them into the same coordinate system using transformation matrix $\mathbf{T}_i$.

In the online operation, the establishment of the 3D scene reconstruction model is started. The system is divided into motor control part and point cloud alignment part. The decision makers distribute the process. Next, the 3D scene reconstruction is performed by the following steps:

- The initial frame $F_0$ is extracted, and the point cloud $\mathbf{P}_0$ is generated. Its coordinates are defined as the world coordinates **W**.
- Control the motor to turn to the next fixed point and return a signal after it finish the process.
- Capture frame $F_i$ and generate point cloud, and perform coarse registration.
- Find the transformation matrix $\mathbf{T}_i^0$ that converts $\mathbf{P}_i$ to $\mathbf{P}_0$.
- Convert $\mathbf{P}_i$ to the world coordinates system to complete the global registration.
- Using the existing ICP or ICP-variant method to perform fine registration.
- If completed, the final 3D scene reconstruction model is generated; otherwise, return to the second step.

Moreover, we perform outlier removal based on some existing methods [16] after completing the 3D scene reconstruction. The detailed action of the point cloud registration is described in $\mathbf{T}_1^2$ and $\mathbf{T}_2^3$, which are transformation matrices between two image planes.

## III. PROPOSED METHOD

This paper proposes a 3D scene construction method which combine motor control and camera calibration. The system performs camera calibration using detected feature points on a chessboard, and obtains relative transformation matrices between fixed postures through offline calibration, the transformation matrices will also be used in global registration. Next, we introduce the offline calibration and global registration methods as well as how to combine with local registration.

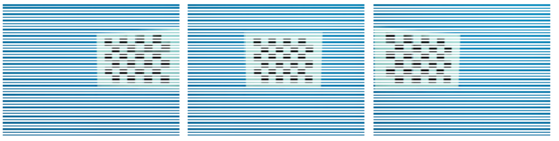

Figure 3. Illustration for camera's viewing angles from left to right.

## A. *Offline calibration*

This section focuses on the proposed method, which use multi-view camera calibration to generate multi-view transformation matrices. The intrinsic parameter matrix $\mathbf{K}$ can be obtained by fixed extrinsic parameters $[\mathbf{R} \mid \mathbf{t}] = [\mathbf{I} \mid 0_{3\times1}]$. Next, the purpose of our method is to obtain all extrinsic parameter matrices under multiple viewing angles. Figure 3 shows the illustration of different viewing angles. In each image, a target (chessboard) is captured, and the corresponding extrinsic parameter matrices can be obtain using camera calibration. The following formula can be obtained from the estimated extrinsic parameters

$$\begin{bmatrix} \mathbf{P}_i \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_r^i & \mathbf{t}_r^i \\ 0_{1\times3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{P}_r \\ 1 \end{bmatrix} = \mathbf{T}_r^i \begin{bmatrix} \mathbf{P}_r \\ 1 \end{bmatrix}, \quad (4)$$

where $\mathbf{P}_r$ are coordinates of the key points on chessboard, and they use grids on chessboard as coordinate system, $\mathbf{T}_r^i$ is the transformation matrix that project $\mathbf{P}_r$ to $\mathbf{P}_i$. Using camera calibration, we can obtain $\mathbf{T}_r^i$ in equation (4). Then, we aim to obtain a relative transformation matrix between different angles of view. Assume that $F_0$ is in the world coordinates $\mathbf{W}$. Then, all key points $\mathbf{P}_i$ should be converted to $F_0$, which lead to the following result

$$(\mathbf{R}_r^i)^T \mathbf{P}_i - (\mathbf{R}_r^i)^T \mathbf{t}_r^i = (\mathbf{R}_r^0)^T \mathbf{P}_0 - (\mathbf{R}_r^0)^T \mathbf{t}_r^0 . \quad (5)$$

After unrolling and rearranging the formula, we can obtain

$$\mathbf{P}_0 = \mathbf{R}_r^0 (\mathbf{R}_r^i)^T \mathbf{P}_i - \mathbf{R}_r^0 (\mathbf{R}_r^i)^T \mathbf{t}_r^i + \mathbf{t}_r^0 = \mathbf{R}_i^0 \mathbf{P}_i + \mathbf{t}_i^0 , \quad (6)$$

where we define the rotation matrices $\mathbf{R}_i^0 = \mathbf{R}_r^0 (\mathbf{R}_r^i)^T$ and translation vector $\mathbf{t}_i^0 = -\mathbf{R}_r^0 (\mathbf{R}_r^i)^T \mathbf{t}_r^i + \mathbf{t}_r^0$ that transform $\mathbf{P}_i$ to $\mathbf{P}_0$. Finally, we can obtain the transformation matrices between all viewing angles using formula (4) such that

$$\mathbf{T}_i^0 = \begin{bmatrix} \mathbf{R}_i^0 & \mathbf{t}_i^0 \\ 0_{1\times3} & 1 \end{bmatrix}, \quad (7)$$

which is saved for online operations.

## B. *Online operation*

This section explains how to combine the multi-view transformation matrix to perform global registration and local registration. Since we already have transformation matrices, the global registration can be done very easily, the initial world coordinates $\mathbf{W}$ is set as $F_0$, and the transformation matrices are also mapped to $F_0$. First of all, we take $F_1$ as an example if we want point cloud $\mathbf{P}_1$ on $F_1$ to be mapped to $F_0$, we can use the formula

$$\mathbf{P}_1^0 = \mathbf{T}_1^0 \mathbf{P}_1 . \quad (8)$$

where $\mathbf{P}_1^0$ is the transformed point cloud $\mathbf{P}_1$ on $F_0$ coordinate system, but the point cloud alignment is not over yet. Then, we must use the local registration method to make $\mathbf{P}_0$ and $\mathbf{P}_1$ point clouds align better. Thus, we use some of the existing local registration methods, such as ICP. After the transformation computation, the new point cloud is given by

$$(\mathbf{P}_1^0)' = (\mathbf{T}_1^0)' \mathbf{P}_1^0 , \quad (9)$$

where $(\mathbf{P}_1^0)'$ is the point cloud from $\mathbf{P}_1$ to coordinate $F_0$ after local registration. $(\mathbf{T}_1^0)'$ is a transformation matrix produced after global registration and local registration of $\mathbf{P}_1^0$. If we transform point clouds for all viewing angles using (8) and (9) onto same coordinate system, we can obtain the reconstructed point cloud $\mathbf{P}_w = \{\mathbf{P}_0, (\mathbf{P}_1^0)', (\mathbf{P}_2^0)', \cdots, (\mathbf{P}_N^0)'\}$, where $N$ represents the number of viewing angles.

# IV. Experimental Result

In this paper, we propose a combination of motor control and camera correction to perform point cloud alignment. Due to the need for offline calibration, we did not use the existing public database. The hardware devices used in the experiment includes a notebook computer, a RGB-D camera and a pan-tilt unit. We used Xtion PRO LIVE RGB-D camera developed by ASUS [17] and fixed the RGB-D camera to the pan-tilt unit, which is D46-17 manufactured by FLIR [18]. The pan-tilt unit was used to drive the RGB-D camera for point clouds capture. All experiments were done on the notebook computer with Intel Core i7-3530M CPU and 8GB memory.

Figure 4 shows the experimental results, in which the pictures listed in left column are the original point clouds represented in the camera coordinate system, respectively. The middle column of the pictures presents the registration results of the Super4PCS method [15]. If only using Super4PCS, there is an initial aligned result (upper middle), which can be coupled with the ICP method to make point clouds closer (lower middle). The pictures listed in right column are the results of the proposed method. In the result of the proposed method, an initial registration result (top right) can be obtained efficiently. Although in overlap regions the proposed method is slightly worse than the Super4PCS, but the similar result can be obtained after ICP operation.

Moreover, the proposed method requires much lower processing time than the Super4PCS method. Table 1 shows the average RMS errors of both methods. In the case of average RMS, the proposed method is slightly worse than Super4PCS. However, after the local registration, the RMS error is significantly reduced, and both RMS error results are very close. Table 2 shows the average processing time, our method averaged at 5.8781ms, and Super4PCS averaged at 32.833s, an improvement on processing speed of nearly 56000 times. And after adding the local registration, the difference average time is about 20 times. Therefore, the proposed method can greatly reduce the processing time of both local and global registration processes.

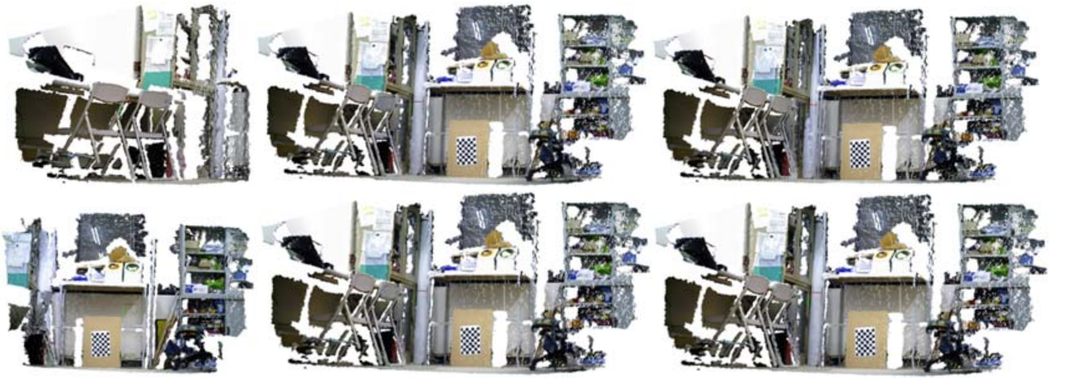

Figure 4. Experiment results. The left column is the original point cloud data, the upper middle is the Super4PCS experimental results, the lower middle is the Super4PCS + ICP experimental results. upper right is the experimental results for our method, and the lower right is the result of our method + ICP.

Table 1. Average RMS Results

| | Proposed method | Super4PCS | Proposed method + ICP | Super4PCS + ICP |
|---|---|---|---|---|
| Average RMS | 26.4172 | 26.2111 | 20.4790 | 20.6596 |

Table 2. Average Processing Time

| | Proposed method | Super4PCS | Proposed method + ICP | Super4PCS + ICP |
|---|---|---|---|---|
| Average time(ms) | 5.8781 | 32833 | 17037 | 34411 |

## V. Conclusion

In this paper, we propose a method to align the scattered point clouds based on camera calibration. This method can quickly converge and only need to perform point cloud transformations once, without the need for any iterative method. Combining the proposed method with local registration can further improve the result. Our global registration method is able to process in real time. In the future, this technology will be further extended with different moving devices on a variety of different 3D model reconstruction, such as object model reconstruction.

## Acknowledgment

This work was supported by the Ministry of Science and Technology of Taiwan, ROC under grant MOST 105-2221-E-032-024 and was supported in part by NUWA Robotics Co., Ltd..

## References

[1] A.A.S. Souza, R. Maia and L.M.G. Goncalves, “3D Probabilistic Occupancy Grid to Robotic Mapping with Stereo Vision,” *book edited by Asim Bhatti*, isbn 978-953-51-0660-9, DOI: 10.5772/49050, 2012.

[2] A. Saxena, S. H. Chung and A. Y. Ng, “3-D Depth Reconstruction from a Single Still Image,” *Int. J. of Computer Vision*, vol. 76, issue 1, pp 53-69, Jan. 2008.

[3] A. Kahn, “Reducing the Gap Between Augmented Reality and 3D Modeling with Real-Time Depth Imaging,” *Int. J. of Virtual Reality*, vol. 17, issue 2, pp 111-123, Jun. 2013.

[4] S. Yang, D. Maturana and S. Scherer, “Real-time 3D Scene Layout from a Single Image Using Convolutional Neural Networks,” *in Proc. IEEE International Conf. on Robotics and Automation*, Stockholm, Sweden, pp. 2183-2189, 2016.

[5] C. Choi, A. J. B. Trevor and H. I. Christensen, “RGB-D Edge Detection and Edge-based Registration,” *in Proc. IEEE Int. Conf. on Intelligent Robots and Systems*, Tokyo, Japan, pp. 1568-1575, 2013.

[6] F. Liu, Q. Lv, H.Lin, Y. Zhang and K. Qi, “An image registration algorithm based on FREAK-FAST for visual SLAM,” in *Proc. Chinese Control Chonference*, 2016.

[7] C. Matabosch, J. Salvi, D. Fofi and F. Meriaudeau, “Range image registration for industrial inspection,” in *Proc. of the Int. Society for Optical Engineering*,San Jose, CA, vol. 5679, pp. 216-227, 2005.

[8] J. Salvi, C. Matabosch, D. Fofi and J. Forest, “A review of recent range image registration methods with accuracy evaluation,” *Int. J. Image and Vision Computing*, vol. 25, issue 5, pp. 578-596, May 2007.

[9] P. J. Besl and N. D. McKay, “A Method for Registration of 3-D Shapes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, issue 2, pp. 239-256, Feb. 1992.

[10] S. Rusinkiewicz and M. Levoy, “Efficient Variants of the ICP Algorithm,” in *Proc. IEEE Int. Conf. on 3-D Digital Imaging and Modeling*, Stanford Univ., CA, USA, pp. 145-152, 2001.

[11] K.-L. Low, “Linear Least-Squares Optimization for Point-to-Plane ICP Surface Registration,” T*echnical Report TR04-004 Department of Computer Science University of North Carolina at Chapel Hill*, 2004.

[12] A. Segal, D. Hahnel and S. Thrun, “Generalized-ICP,” in *Proc. Int. Conf. on Robotics Science and Systems V*, University of Washington, Seattle, USA, vol. 2, no. 4, 2009.

[13] J. Serafin and G. Grisetti, “NICP: Dense Normal Based Point Cloud Registration,” in *Proc. IEEE Int. Conf. on Intelligent Robots and Systems*, Hamburg, Germany, pp. 742-749, 2015.

[14] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison and A. Fitzgibbon, “KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera,” in *Proc. of the 24th annual ACM symposium on User interface software and technology*, Santa Barbara, California, USA, pp 559-568, 2011.

[15] D. Aiger, N. J. Mitra and D. Cohen-Or, “4-Points Congruent Sets for Robust Pairwise Surface Registration,” in *Proc. of ACM SIGGRAPH*, Suntec Singapore, vol. 10, no. 3, pp. 1-10, 2008.

[16] K. Wolff, C. Kim, H. Zimmer, C. Schroers, M. Botsch, O. Sorkine-Hornung and A. Sorkine-Hornung, “Point Cloud Noise and Outlier Removal for Image-Based 3D Reconstruction,” in *Proc. IEEE Int. Conf. on 3D Vision*, Stanford Univeristy, California, USA, 2016.

[17] ASUS Xtion Pro [Online]. Available: https://www.asus.com/3D-Sensor/Xtion_PRO/

[18] Pan-tilt Unit [Online]. Available: http://www.flir.com/mcs/view/?id=53707