International Journal of Smart Computing and Artificial Intelligence
International Institute of Applied Informatics
2017, Vol.1, No.2, P.1 - 20

Spatial Surface Reconstruction for Complex Environment Using Color-Depth Sensors

Yung-Cheng Huang [*], Feng-Li Lian [†]

Abstract

In this paper, the structure-entropy-based features are used to describe the energy of the complex environment and then the entropy energy is further used to extract the region of the spatial structural change. Moreover, by finding the maximum entropy energy of the overlapping area, the relative pose between two consecutive frames can be estimated. In the final step, the iterative closest point (ICP) is utilized to determine the rigid transformation matrix for the remaining region. Extensive experimental results show that the proposed method generates more accurate result than that by using the traditional ICP algorithm.

Keywords: Point cloud, red-green-blue-depth (RGB-D) sensor, rigid transformation, structural entropy, three-dimensional (3D) reconstruction.

1 Introduction

With the development of three-dimensional scanners [1] and the success of two-dimensional simultaneous localization and mapping (2D-SLAM) [2], three-dimensional surface reconstruction has been intensely studied recently. The reconstruction technique can be divided into two parts, namely, environment reconstruction and object reconstruction. Environment reconstruction is a straightforward extension of object reconstruction, which aims to reconstruct everything in a scene.

In comparison with a 2D map, the 3D environment reconstruction provides more complete detail about the scene, including rich visual information and 3D geometric shapes [3]. 3D geometric shapes provide the structural information of the environment, and rich visual information helps observer to recognize objects in the environment. In various computer vision fields, a rich 3D map has been used in numerous applications, such as robot vision [4], virtual and augmented reality [5], and entertainment [6].

[*] National Taiwan University, Taipei, Taiwan
[†] National Taiwan University, Taipei, Taiwan

Practically speaking, most 3D environment reconstruction approaches have relied on expensive 3D scanners, such as laser range finders, suffering from the lack of visual information. On the other hand, monocular vision devices are difficult to generate precise and robust depth information. However, the appearance of low-cost and reliable 3D scanners, such as Microsoft Kinect or ASUS Xtion PRO, provides complementary frames to capture both visual and geometric information of environment simultaneously. The complementary frames are called RGB-D frames which store RGB and depth variables for each pixel. In addition, the RGB-D frame can be transformed into RGB-XYZ point cloud data format. Each 3D scanner has different characteristics. For instance, the frames captured by stereo camera have more invalid values or noise in depth channel compared to Kinect, especially in environments with sparsely textured areas. On the other hand, the resolution of the measured depth estimated by laser range finder is more accurate compared to the other 3D scanners. Additionally, the field size of one view affects the quality of 3D map. Each scan can only capture partial area of the environment. In order to combine each scan to build a global 3D map, many approaches have been developed.

Most of the 3D environment reconstruction approaches rely on data registration. By doing so, three-dimensional dataset scanned in different viewpoints can be transformed into the same coordinate system by aligning overlapping components of these datasets. With providing both color and spatial information, humans or robots can easily perceive their environments, such as minimal invasive surgery. In general, the task of three-dimension environment reconstruction can be divided into three stages: feature descriptor estimation, outlier rejection, and transformation estimation. First, feature descriptor estimation is used to find some distinct features with their special characteristics. Second, feature outlier removal can remove the incorrect corresponding pairs between two consecutive frames. Third, transformation estimation uses the correct corresponding pairs to find the transformation matrix which can transfer different viewpoint frames into global coordinate.

In this paper, the proposed method utilizes the structure-entropy-based features to describe the energy of the environment. The regions of the spatial structural change are extracted by analyzing the structural entropy energy. Then, a new method, called entropy image matching, is presented to remove outliers. With finding maximum entropy energy of the overlapping area, the relative pose between two consecutive frames can be estimated, which can serve as a good initial guess for transformation estimation. In the final step, a rigid transformation matrix is determined by implementing the iterative closest point (ICP) method on the remaining regions. With the transformation matrix, all of the frames can be transformed into global coordinate and a 3D virtual map with point cloud format can be identified. Experimental result shows that the accuracy of the proposed method is better than traditional ICP algorithm, and the proposed method is unaffected by color or even in complete darkness compared to RGB-based feature mapping method.

This paper is organized as follows. The introduction is provided in Section 1 and the related literature is surveyed in Section 2. Section 3 presents the key ingredients of the proposed three-dimensional environment construction. Sections 4 and 5 provide detailed analysis and comparison of the proposed approach on different testing scenarios and Section 6 concludes this paper.

2 Related Work and Literature Survey

This section reviews the main category of 3D surface reconstruction of indoor environment and surveys related literatures on feature descriptor estimation, outlier rejection, and transformation estimation.

2.1 Three-Dimensional Environment Reconstruction

In the last few years, many approaches have been focused on 3D virtual environment reconstruction using RGB-D scanner. Generally speaking, these approaches can be divided into two distinct parts, namely, dense points matching, and sparse feature matching. Several approaches align data based on sparse feature points matching, using a number of distinct points or features to represent the whole data, and finding the correspondence. Moreover, the algorithms can be differentiated based on the types of input data, such as color information, or spatial geometric. For example, the authors in [7] and [8] use ‘pure vision’ to achieve real-time structure from motion. A state-of-the-art of sparse feature matching approach is presented in [9], which improves the two-stage localization method by fusing the SIFT features and points, and the tracking with the RANSAC and ICP algorithms. The joint optimization for both appearance and shape can generate a precise localization result. Recently, some approaches replace the SIFT features with other robust features, such as the FAST-feature-based version [3], [10] and the SURF-feature-based version [11]. More recently, some researchers notice that the feature matching algorithms fail in the repetitive patterns environment or no texture area. In order to overcome this problem, some researchers introduce geometric information into the registration scheme, such as the planar-surface patches extracted from data [12].

On the other hand, several approaches use dense points matching method to estimate the relative position between two datasets. The concept of dense points matching is different from the concept of sparse feature matching, where all points in each dataset contribute to the scene alignment. The work of dense points matching is proposed in [13], [14]. A mixture of Gaussians is used to fit the distribution of the geometric data, and an approximation of the correlation measure is used for a mixture of Gaussians [15]. Furthermore, the screened Poisson surface reconstruction is proposed in [16] and used to explicitly incorporate the data points as interpolation constraints. By the improvement, the time complexity of the solver is reduced and it enables faster, higher-quality surface reconstructions.

2.2 Feature Descriptor Estimation

Estimating features is an essential step to registration. Local features are referred to as interest points or salient points which efficiently reduce the amount of data to be analyzed. Feature can be a value, or a vector which is formed by these neighboring points. In addition, features can be used to describe the characteristics of color information, geometric shape, or even both of color and geometric information. For example, a common color feature using intensity gradient of pixel, called scale-invariant feature transform (SIFT) [17], and a modified version of SIFT, namely, speeded up robust features (SURF) [18], are used to reduce the computation time. Also, color binary features, such as BRIEF [19], ORB [20], BRISK [21], or FREAK [22], aim to reduce more computation cost. However, the detection of these RGB features is affected by illumination or area without texture.

On the other hand, geometric features provide specific parts of a structure model. The features are directly operated on the 3D coordinate rather than on the 2D coordinate, such as plane estimation [23]. Spin images (SI) in [24] encode surface properties in a local object-oriented system. The 3D shape context in [25] is a 3D extension of the 2D shape context descriptor. Signature of histograms of orientations (SHOT) in [26] relies on the definition of a repeatable local Reference Frame (RF) to define a signature structure. Fast point feature histograms (FPFH) in [27] are based on the combination of certain geometrical relationships. In addition, a feature, combining both RGB and geometric information in [28], called geometric and photometric local feature (GPLF), uses both the geometric and photometric information of 3D point clouds from RGB-D camera and is integrated into a descriptor.

2.3 Outliers Rejection

After finding features in the data, the features may be matched incorrectly. That is, possible outliers might lead to inaccurate results or poor performance, In order to solve this problem, outliers must be removed. One of the commonly used methods is random sample consensus (RANSAC) [29]. Instead of greedy search for feature matching, RANSAC fits the parameters of the feature pair model that are valid for inlier points. It iteratively selects samples to compute a 6-DOF transform matrix with minimal error metric [27]. Binary features can be efficiently matched with locality sensitive hashing (LSH) [30].

2.4 Transformation Estimation

Data registration may consider rigid transformation or non-rigid transformation. Rigid transformation estimates relative pose with a 6-DOF transformation matrix. In order to estimate a 6-DOF rigid transformation matrix between two consecutive frames, many approaches have been proposed. These approaches can be divided into two main parts, namely, correspondence-based, and non-correspondence-based. Correspondence-based matching needs pairwise points as inputs, such as singular value decomposition (SVD) [31]. The core idea is similar to the least-squares method with more than eight points, leading to an over-determined system. On the other hand, non-correspondence-based algorithms estimate pose without prior knowledge of association of two consecutive frames. The most commonly used matching algorithm is Iterative Closest Point (ICP) algorithm in [32]. ICP finds an optimal rigid transformation matrix by associating the closest points between two datasets. However, due to aligning two datasets without correspondence, poor initial position or noise affect the correctness of the result. In order to improve the robustness of ICP, many related modified approaches are proposed. For examples, point-to-plane (P-L) version of ICP [33] modifies its cost function which minimizes error along local normals. One of the other alternative ICP versions is the 3D normal distribution transform (3D-NDT) in [34], which uses normal distribution to represent input data and increases more robustness to outliers.

3 Three-Dimensional Environment Construction

In this section, the proposed 3D environment reconstruction method called Entropy-based feature ICP (EnICP) is presented. Two of the RGB-D frames are captured from Kinect at time *t-1*, and time *t*, respectively, to estimate the pose between these two consecutive frames. Thus, the first step of the proposed registration method is to estimate the normal vectors for input points, and then calculate the entropy of image points. The points with entropy energy are extracted for the match task in the next step. According to the output data format of Kinect, a set of ordered 4D points are divided into two images, namely, RGB image, and depth image. In this paper, an 'Entropy Image' is proposed and used to remove the outliers, which is used to find the initial pose between two consecutive frames. Finally, the remaining points with entropy energy are defined as the target cloud at the previous time *t-1*, and the remaining points with entropy energy are defined as the source cloud at the current time *t*. Then, the target cloud and the source cloud can be used to construct a rigid transformation, estimated by iterative closest point (ICP), and is applied to the source cloud to align it onto the target cloud. The results after registration are presented as the 3D map.

3.1 Camera Calibration

The pin-hole model projects a 2D image position $\mathrm{f}(u,v)$ into the 3D real-world coordinate system $(\mathrm{x},\mathrm{y},\mathrm{z})$ by the intrinsic matrix. In order to estimate intrinsic matric of the Kinect device, the camera calibration is used in this section. A 3D point in the world coordinate system is denoted by $X=(\mathrm{x},\mathrm{y},\mathrm{z})$ and its corresponding 2D projection in the image plane is denoted by $m=(u,v,1)$. Then, the relationship between X and m is modeled as $sm^T = AX$. The detail of the formulation are shown in Eq. (1) – Eq. (3), where A is the intrinsic matrix of the Kinect device, and s is a scale factor. In this case, $s=z$. The general camera calibration procedure requires the orientation of different checkerboard poses. First, the four extreme corners on the rectangular checkerboard pattern in the RGB image are selected manually. Based on the two similar triangles which are presented in Eq. (1) and Eq. (2), the intrinsic matrix of the Kinect device can then be estimated.

$$u=\frac{f}{px}\cdot\frac{x}{z}+u_0=f_x\frac{x}{z}+u_0 \tag{1}$$

$$v=\frac{f}{py}\cdot\frac{y}{z}+v_0=f_y\frac{y}{z}+v_0 \tag{2}$$

$$s\begin{bmatrix}u\\v\\1\end{bmatrix}=\begin{bmatrix}f_x&0&u_0\\0&f_y&v_0\\0&0&1\end{bmatrix}\begin{bmatrix}x\\y\\z\end{bmatrix}=AX \tag{3}$$

$$\text{where } A=\begin{bmatrix}f_x&0&u_0\\0&f_y&v_0\\0&0&1\end{bmatrix},\quad X=\begin{bmatrix}x\\y\\z\end{bmatrix}$$

3.2 Calculation of Feature Descriptor

In order to filter the data of input point cloud P, a subset of points P_f, which are sampled from the points based on the distinct and persist characteristics. Two consecutive clouds could be replaced by their subset clouds, and it is more efficient to find the transformation between two clouds. However, a distinct and persist characteristics, called feature, is difficult to find. Currently, many reliable features have been developed. In general, the descriptor of these features contains information, coming from its surrounding neighbors. Single value is not robust, and is not expressive enough.

Point feature based approaches can be categorized in two different types, color space and spatial space. In color space, one widely used color feature detector and descriptor is the scale invariant feature transformation (SIFT). Each SIFT feature point provides the position in the image coordinate and the 1×128 descriptor is extracted from its neighbors. The SIFT algorithm takes an intensity image as input, the intensity value is calculated from the triplet of value

(R, G, B). Also, a number of more recent color features, such as SURF and FAST, provide good performance. On the other hand, in spatial space, the feature descripts the geometric change, and is extracted from the neighbors. The selection is determined by an absolute distance, such as jump edges, and generated by the depth discontinuities or point feature histograms (PFH) descriptors. The mean curvature around the points is generated by using a multiple-dimensional histogram based on the relationships between the points and their normal vector.

As the normal vectors alone do not provide enough information about the geometry, the entropy energy, estimated by surrounding points, is used to describe the disorder of normal vectors. The core idea is to extract the points in the region of geometric change, such as the edge between wall and ceiling, or the edge of the box. It is almost like edge detector in 3D version. By doing this, the extracted points represent the cloud, so that the complexity of the registration problem is reduced. However, the region of geometric change is affected by taking different value of selecting radius, including the radius of normal vectors estimation and the radius of entropy estimation r_E . The step by step of the entropy-based feature estimation is listed in Algorithm 1.

Algorithm 1: Entropy-Based Feature Estimation

```
Inputs: A point cloud P with n points
Outputs: Entropy Image $I_E$ with u×v pixels
Part I. Surface Normal Vector Calculation
1.  Select a point $x_i$ in Point cloud $P$,where $i = 1, 2, 3, \dots, n$
2.  Select neighboring points $P_j$ of $x_i$ in a sphere with radius $r_N$, where $j = 1, 2, 3, \dots, m$
3.  $C_i = \frac{1}{m}\sum_{j=1}^{m}(p_j - x_i)(p_j - x_i)^T$ , a covariance matrix $C_i$ is defined
4.  The eigenvector $v$ of $C_i$ corresponding to the smallest eigenvalue $\lambda$ is the approximation of normal vector of $x_i$, denoted $n_i$
5.  Repeat Step 1 until $i = n$
Part II. Entropy Estimation
6.  Select neighboring points $P_k$ of $x_i$ in a sphere with radius $r_E$, where $k = 1, 2, 3, \dots, M$
7.  The inner product of normal vectors of $x_i$ and $P_k$ are stored in $q_{i,k}$
8.  The histogram of quantized intensity values $q_{i,k}$ with bin size Bis generated as follows: $p(q_{i,k}) = \frac{1}{Q}\mathrm{hist}(\langle n_i, n_k \rangle)$, where $Q = q_{i,1} + q_{i,2} + \dots + q_{i,M}$
9.  $Entropy(x_i) = -\sum_{q_{i,k}} p(q_{i,k}) \cdot \log p(q_{i,k})$, the entropy of $x_i$ can be estimated.
10. Repeat Step 1 until $i = n$
    When $Entropy(x_i) \neq 0$, then $x_i \in P_f$, because of the characteristics of Kinect data format, the $Entropy(x_i)$ can be reshape in u×vdimension, where $u \times v = n$, then, a entropy image $I_E$ is generated.
```

3.3 Entropy Image Matching

In this section, the entropy image matching process for two consecutive frames is used to reject the remaining entropy outliers and guess an initial position for the ICP algorithm. Although the frames can be down-sampled by extracted entropy points in the region of geometric change, the problem of outliers still exists. In order to remove the outliers between two consecutive frames, the entropy image matching method is proposed to solve the problem of outliers.

In the first step, two of the consecutive entropy images are selected. Then, the selected entropy images are named as target entropy image and source entropy image. The target entropy image is extracted at the previous position, and, the source entropy image is extracted at the current position. The problem of translation between two frames can be solved by finding the best overlapping area with the largest entropy energy, and is defined in Eq. (4).

$$F(x) = \sum_{i,j\in R} \alpha \cdot f_E(\mathrm{u}+x_i, \mathrm{v}+x_j, \mathrm{t}) \cdot f_E(\mathrm{u}, \mathrm{v}, \mathrm{t}-1) \tag{4}$$

$$x^* = \arg\max_x F(\mathrm{x}) \tag{5}$$

where $x = (\mathrm{x}_i, \mathrm{x}_j)$ is a shift motion vector between two consecutive frames $\mathrm{X}_i, \mathrm{X}_j$ in the pixel space. In order to provide a roughly initial guess for ICP, $\mathrm{X}_i, \mathrm{X}_j$ in pixel space must be transformed into the real world coordinate by the intrinsic matrix. α is the weight factor, and usually $\alpha = 1$.

3.4 Geometric Relationship Estimation

After finishing the outlier rejection and initial position guessing, the geometric relationship between the remaining points with entropy energy, defined as the target cloud at the previous time *t-1*, and the remaining points with entropy energy, defined as the source cloud at the current time *t* is estimated by the iterative closest point (ICP) algorithm. Geometric relationship estimation is also called data registration, which brings datasets acquired from different arbitrary poses into the same coordinate system. A consistent global 3D map is generated by finding a proper rotation matrix and translation vector between two consecutive frames. Figure 1 shows the process of registration between two individual point cloud data acquired from different views. Figure 1(a) shows that an individual scan is captured in different view, red arrow represents camera center at time *t-1*, and blue arrow represents camera center at time *t*. Figure 1(b) shows that an individual point cloud have transformed into the same coordinate. In this case, a point cloud captured at time *t* is transformed into the coordinate of time *t-1*. The 3×3 rotation matrix R and the 3×1 translation vector T are estimated by the ICP algorithm. Every source cloud and every target cloud are assumed to be a rigid body, the relationship between source cloud and target cloud is formulated as follows:

$$P_t = R \times P_s + T \tag{6}$$

where P_s is defined as the source cloud, and P_t as the target cloud. Then, the transformation matrix can be written as a 4×4 matrix as follows:

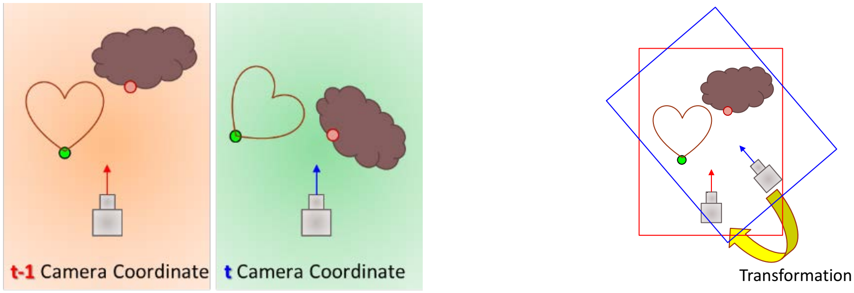
$$M_{t-1,t} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_1 \\ R_{21} & R_{22} & R_{23} & T_2 \\ R_{31} & R_{32} & R_{33} & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

where $M_{t-1,t}$ is the transformation matrix used to transform a data acquired at time t into time *t-1* coordinate system. However, a global consistent point cloud map consists of different arbitrary poses data, the data should be transformed into global coordinate system. Then, the transformation matrix from Eq. (7) can be modified as follows.

$$C_t = M_{0,1}M_{1,2}\ldots M_{t-2,t-1}M_{t-1,t} = \prod_{i=1}^{t} M_{i-1,i} \tag{8}$$

where C_t , a global transformation matrix, transforms a data acquired at time t into the global coordinate system. Therefore, a global consistent 3D point cloud map is built by each individual transformed point cloud, and can be formulated as follows:

$$\begin{bmatrix} X_t^{global} \\ Y_t^{global} \\ Z_t^{global} \\ 1 \end{bmatrix} = C_t \begin{bmatrix} x_t^{local} \\ y_t^{local} \\ z_t^{local} \\ 1 \end{bmatrix} \Rightarrow P_t^{global} = C_t P_t^{local} \tag{9}$$



(a) (b)

Figure 1: The process of registration between two individual point cloud data acquired from different views.
(a) Captured an individual scan in different view, red arrow represents camera center at time $t-1$, and blue arrow represents camera center at time t. (b) An individual point cloud have transformed into the same coordinate. In this case, a point cloud captured at time t is transformed into the coordinate at time $t-1$.

4 Experimental Results and Analysis

In order to evaluate the performance of the 3D environment reconstruction algorithm, a test sequence of overlapping data, captured at regular laboratory area, is built. The experimental scene is constructed with a size of 3m×3m×2m as shown in Figure 2. To evaluate the performance, the Kinect sensor is mounted with a laser scanner (Hokuyo URG-04LX-UG01) as shown in Figure 3(a). The checkerboard is set behind the Kinect sensor as shown in Figure 3(b). The Kinect sensor is moved by the given commands according to the grid sheet with the resolution 1cm×1cm on the ground, shown in Figure 3(c) and 3(d). In addition, the pan angle is given by the angle gage of the cradle head of BENRO BH0 tripod, as shown in Figure 3(e). The coordinate of experimental dataset is defined as follows: the X axis is parallel to the ground and the white wall, and the Y axis is perpendicular to the ground, and the Z axis is parallel to the ground and perpendicular to the white wall. The camera trajectory is composed by five paths as shown in Figure 2.

The set of experimental data consists of 24 camera poses, including Kinect RGB-D data, laser data, and calibration image:

1. In Path 1, Camera moves in the X axis by -0.05m, -0.1m, -0.15m at each step.

2. In Path 2, Camera moves in the Z axis by +0.05m, +0.1m, -0.05m, -0.1m at each step.
3. In Path 3, Camera moves in the X axis by 0.1m and rotates 7.5° until it reaches 15°.
4. In Path 4, Camera moves in the X axis by 0.1m and rotates 7.5° until it reaches 30°.
5. In Path 5, Camera moves in the X axis by 0.1m and rotates 7.5° until it reaches 90°.

The bottom picture in Figure 2 illustrates the above-mentioned camera poses.

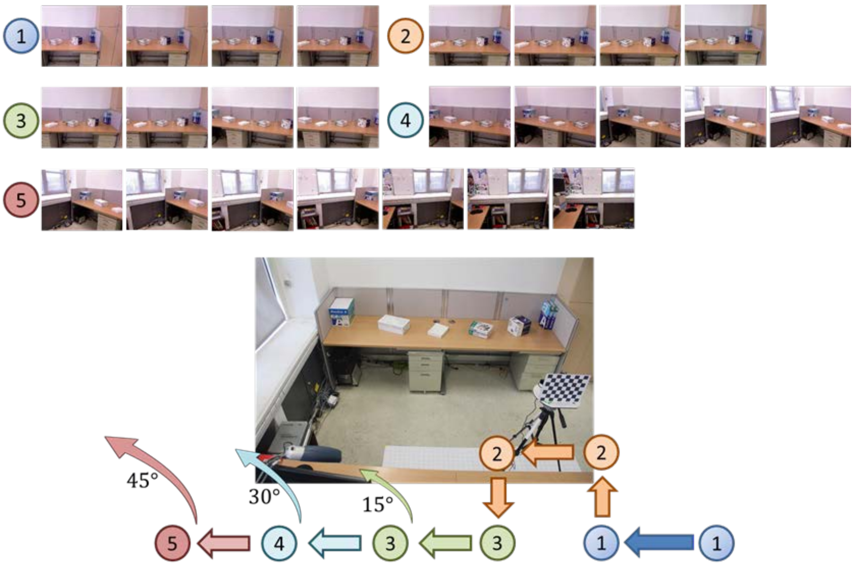
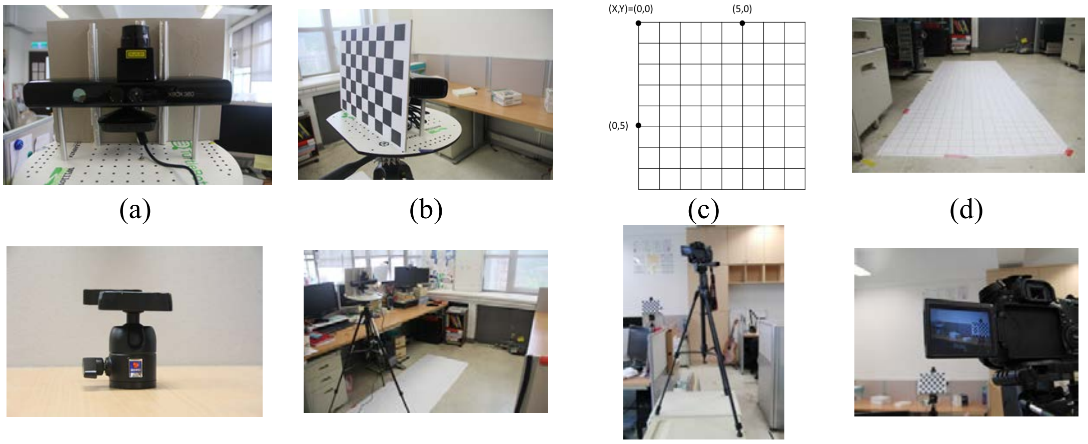


Figure 2: Experimental scene.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3: Experiment platform and accessories.
(a) Kinect sensor with HOKUYO laser scanner. (b) Kinect sensor with checkerboard. (c) The grid sheet with resolution 1cm X 1cm. (d) The grid sheet in the experimental scene. (e) The cradle head of BENRO BH0 tripod. (f) All apparatus in the experimental scene. (g) DSLR camera as the supervisor. (h) The viewpoint of DSLR camera with checkerboard.

The implementation of the proposed EnICP method is divided into five stages. The first stage is data acquisition. That is, the frames captured from the environment to be mapped. The captured frames must contain overlapping areas between two consecutive frames. At the second stage, the normal vector calculation estimates the normal vector for each point. The result of normal vector calculation is shown in Figure 4(b). At the third stage, the entropy estimation extracts the structure region with entropy energy. According to Algorithm 1, the set of neighbors in a sphere of radius r_N is defined as 0.05m. Also, the entropy searching radius r_E is defined as 0.025m. The result of entropy estimation is shown in Figure 4(c). In order to provide a good visualization, the regions with the top 25% entropy energy are labeled with red, those of the 25% - 50% entropy energy are labeled with orange, those of the 50% - 75% entropy energy are labeled with yellow, those of the final 75% - 100% entropy energy are labeled with green, and the regions without labeling color denote no entropy energy. The fourth stage is entropy image matching. Because of the Kinect data format, a 3D point cloud can be projected into one 2D image plane. The process of projection is shown in Figure 5. Figure 6 shows two of the entropy images taken at different locations for r_N=0.05m and r_E=0.025m. The estimated energy of entropy is shown and the extracted points are exactly at the edge of spatial change.

The process of entropy image matching can be divided into two stages. The first stage is searching with large range, that is, 10 pixels at each iteration. The second stage is the searching with precise range, that is, 1 pixel at each iteration. The result of motion vector is regarded as the initial position of the ICP algorithm. Figure 7(a) shows the process of finding maximum entropy energy from Figure 6. The red area represents source entropy image, and the green area represents target entropy image. The white area is the overlapping area. The matching result is shown in Figure 7(b). The result shows the translation between target entropy image and source entropy is 75 pixels in horizontal and 1 pixel in vertical. The red area represents source entropy image, and the green area represents target entropy image. The white area is the overlapping area between two consecutive frames. The points out of the overlapping area are regarded as outliers. However, the ICP algorithm can converge to local minima when the given initial position is bad. In order to avoid the local minima, the result of shift path can serve as a good initial guess for the ICP algorithm.

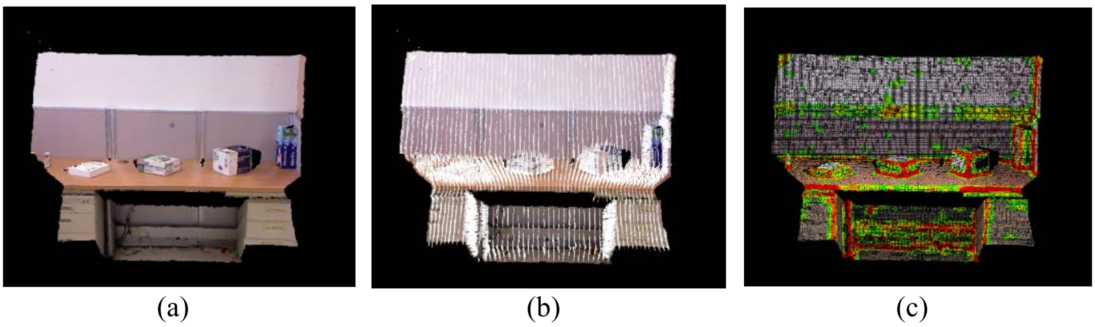


Figure 4: Sample result of feature descriptor calculation.
(a) The original scene. (b) The original scene with normal vectors. (c) The structure region with entropy energy. To provide a good visualization, the regions with top 25% entropy energy are labeled with red, those of the 25% - 50% entropy energy are labeled with orange, those of the 50% - 75% entropy energy are labeled with yellow, those of the 75% - 100% entropy energy are labeled with green, and the regions without labeling color mean no entropy energy.

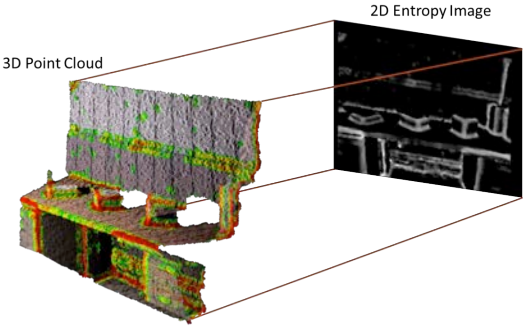
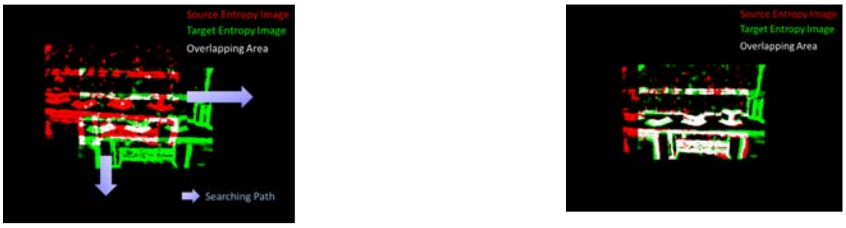


Figure 5: The illustration of projection of Kinect data.

(a) (b)

Figure 6: These two parameters r_N and r_E of Entropy Image.
The radii of neighborhood selection are assigned as follows: $r_N = 0.05m$ and $r_E = 0.025m$.
The extracted points are exactly distributed over the edge of geometric change.



(a) (b)

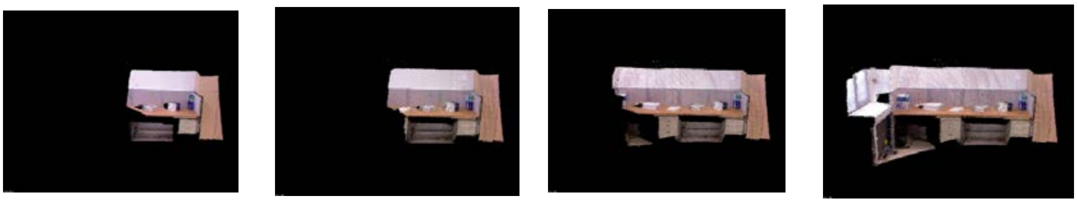
Figure 7: (a) The process of finding maximum entropy energy matching. (b) The matching result between two consecutive frames.

Furthermore, the Kinect RGB-D data can be used to reconstruct a 3D map. The registration of individual point clouds estimated with different 3D environment reconstruction methods are plotted onto the world coordinate with point cloud format using Point Cloud Library [35]. Figure 8 shows the mapping result of the ICP algorithm. It is obvious that the 3D map of ICP does not align together. The result of ICP fails to consistently combine the sequential poses, due to the effect of the outliers. The same sequence estimated by the proposed EnICP method is shown in Figure 9. Figure 10(a) shows the experimental scene, Figure 10(b) shows the 3D map reconstructed by EnICP, and Figure 10(c) shows the local view of 3D map reconstructed by EnICP. The result by EnICP has better alignment than that by ICP. Figure 11 shows the reconstruction result by the RGB-RANSAC algorithm. Figure 11(b) shows the misaligned place in the red circle. Figure 11(c) shows the local view of RGB-RANSAC algorithm. Figure 12 shows the result given by the camera localization. The mapping of camera localization in the first 10 data without rotation pose is shown in Figure 12(a). It provides good result, but the data are misaligned in last 14 frames. Figure 12(b) shows the result of camera localization. Figure 13 shows the top view of all the 3D environment reconstruction methods in this thesis. Figure 13(b) shows the result of RGB-RANSAC algorithm in the Z component has better performance.

The qualitative and quantitative results in this section demonstrate various aspects of the 3D environment reconstruction performance, including the dense point-to-point ICP (ICP), the Entropy feature-based point-to-plane ICP (EnICP), the RGB feature-based RANSAC mapping algorithm (RGB-RANSAC), the camera localization (Camera), and the ground truth laser data. The result of ICP is worse in this experiment compared to the other localization approach, especially when data have rotation pose. That is because the ICP algorithm needs a good initial pose between two consecutive frames, and does not have the ability of rejection outliers. The performance of the proposed EnICP method is close to the result of the RGB-RANSAC algorithm. In order to generate a better result, the RGB-RANSAC needs an environment filled with rich visual information. On the other hand, the EnICP needs a structure environment to estimate the entropy energy. In this experiment, the EnICP method helps improve performance when the ICP method does not have sufficient initial pose information. Another benefit of using EnICP has an ability to reject outliers by entropy image matching. The RGB-RANSAC result has better performance in X-axis and Z-axis, and worse performance in Y-axis. The result of camera localization shows that the method only has a good performance when the Kinect moves in one dimension. The localization by given command has a better performance in this experiment.

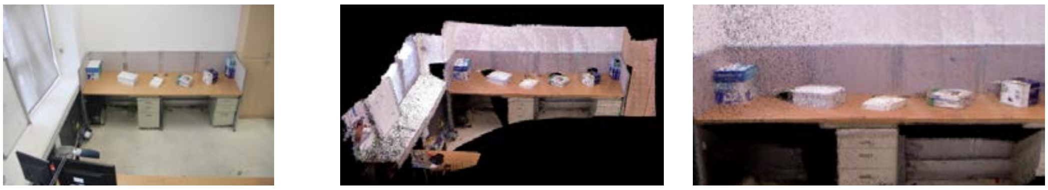
(a) (b)

Figure 8: The misaligned 3D map reconstructed by ICP algorithm in different viewpoint. (a) The misaligned result of ICP. (b) The top view result of ICP.



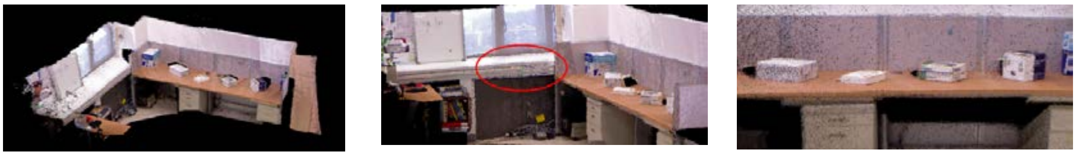
(a) Step 0 (b) Step 6 (c) Step 12 (d) Step 18

Figure 9: The mapping result of the proposed EnICP method.



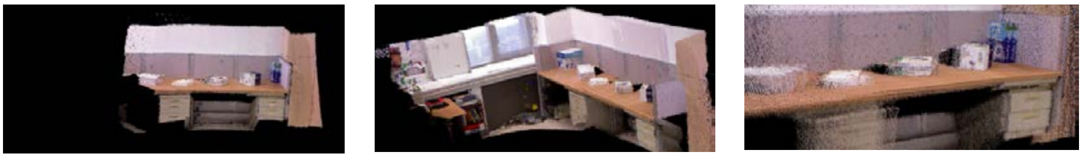
(a) (b) (c)

Figure 10: The 3D environment map by the proposed EnICP method.
(a) The experimental scene. (b) The 3D map reconstructed by the proposed EnICP method. (c) The local view.



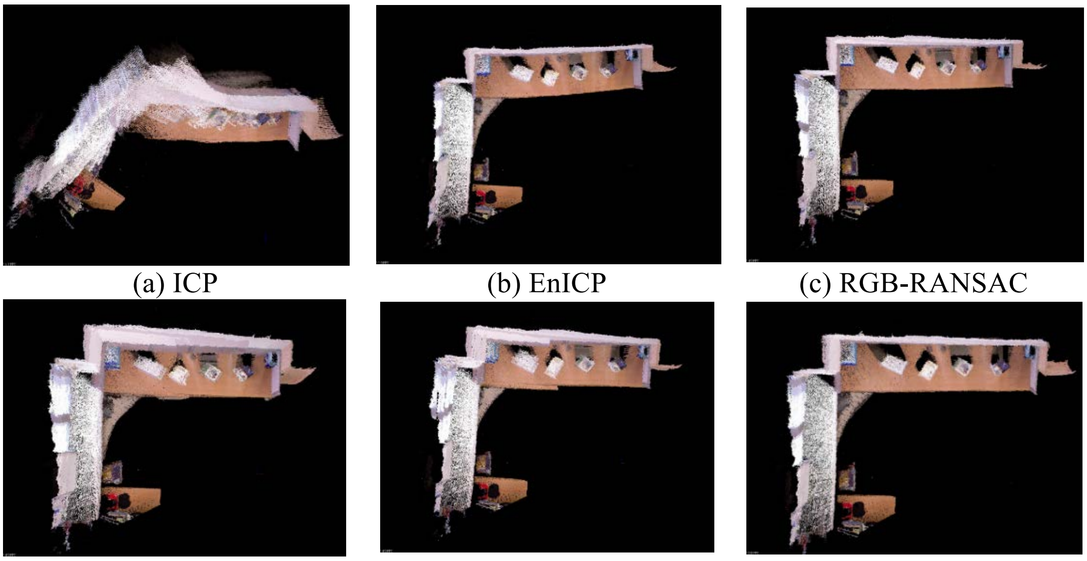
(a) (b) (c)

Figure 11: The reconstruction result by the RGB-RANSAC algorithm.
(a) The 3D map reconstructed by the RGB-RANSAC algorithm. (b) The misaligned place in the red circle. (c) The local view result.



(a) (b) (c)

Figure 12: The reconstruction result given by the camera localization.
(a) The 3D map reconstructed from first 10 data which has no rotation pose by the camera localization. (b) The misaligned 3D map, generated from the 24 data. (c) The local view.



(a) ICP (b) EnICP (c) RGB-RANSAC

(d) Camera localization (e) Command (f) Laser ground truth

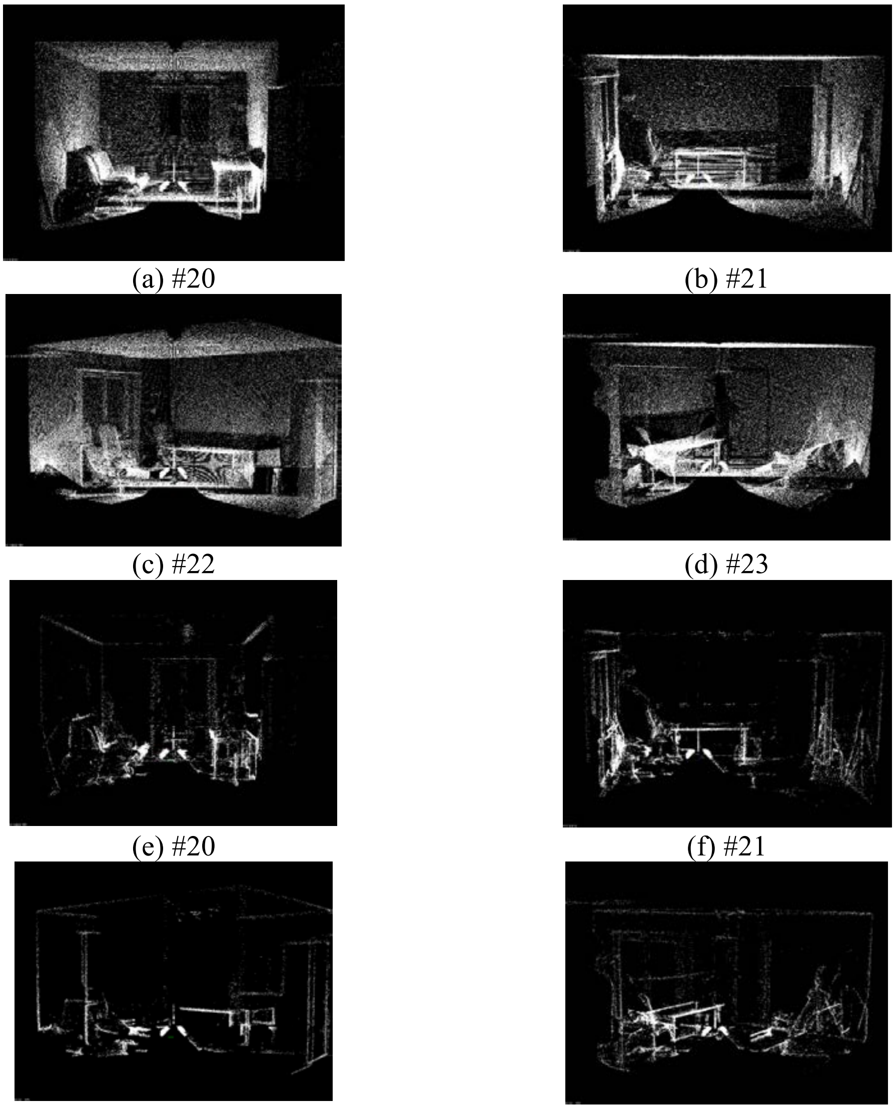
Figure 13: The top view of 3D map generated from all the 3D environment reconstruction methods: ICP, EnICP, RGB-RANSAC, camera localization, command, laser ground truth.

5 Testing and Comparison on ASL Datasets Repository

This section presents the testing result and related comparison on the dataset from the Autonomous Systems Lab (ASL) at the Swiss Federal Institute of Technology Zurich [36]. The proposed entropy-feature-based ICP algorithm, EnICP, is tested with laser registration datasets - apartment - recorded using laser Hokuyo UTM-30LX with scan angle 270°, angular resolution 0.025°, and the guaranteed range from 0.1 m to 30 m. The dataset represents a sequence of indoor apartment structured environment taken from the laser scanner. It is available online at [37].

The global position is recorded from a theodolite from Leica Geosystems, which measures one location at a time, and provides ground truth at each camera pose.

In order to evaluate the result of proposed EnICP algorithm, the ground truth which provided from ASL datasets repository is used for comparison, and the result of using traditional ICP algorithm is also compared. In this experiment, four samples, #20, #21, #22, #23, from the dataset are used, which are captured at the office of the apartment. The point cloud map of these four sets of data is shown in Figure 14 (a)-(d). The entropy of these data points are then computed and shown in Figure 14(e)-(h). These results demonstrate that the extracted region of the spatial structural change are the edge between walls and ceiling, the edge of window, the edge of desk, the edge of sofa, and the edge of chair.



(a) #20 (b) #21

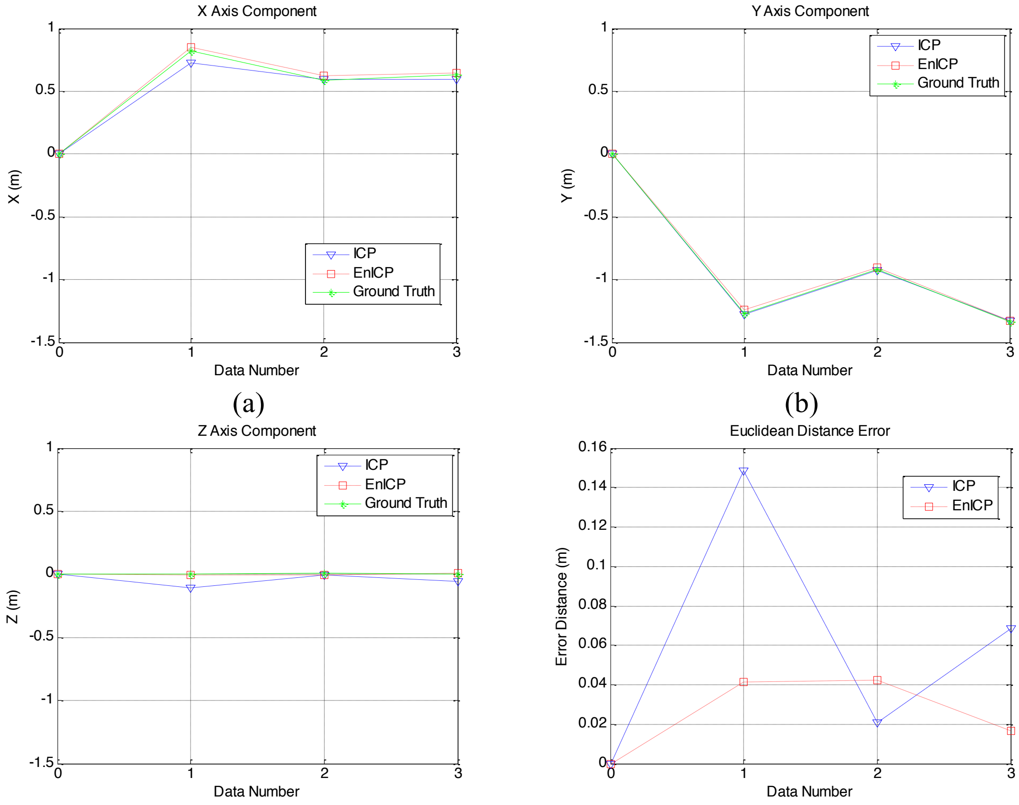
(c) #22 (d) #23

(e) #20 (f) #21

(g) #22 (h) #23

Figure 14: (a)-(d) The point cloud map captured from laser scanner and (e)-(h) The entropy of these data points. The extracted region of the spatial structural change are the edge between walls and ceiling, the edge of window, the edge of desk, the edge of sofa, and the edge of chair.

The numerical comparison is shown in Figure 15, where the results of using the ICP and EnICP localization approaches are individually compared with the ground truth. The X-, Y-, and Z-axis components of the laser position are estimated by different 3D environment reconstruction algorithms, i.e., the blue inverted triangle signs represent the ICP result, the red square signs represent the EnICP result, and the green star signs represent the ground truth provided by ASL datasets repository. The Euclidean distance error is shown in Figure 15(d). The performance of the proposed EnICP algorithm is better than that of the traditional ICP algorithm. The mean error by EnICP is 0.025m and the mean error by the traditional ICP algorithm is 0.0593m.

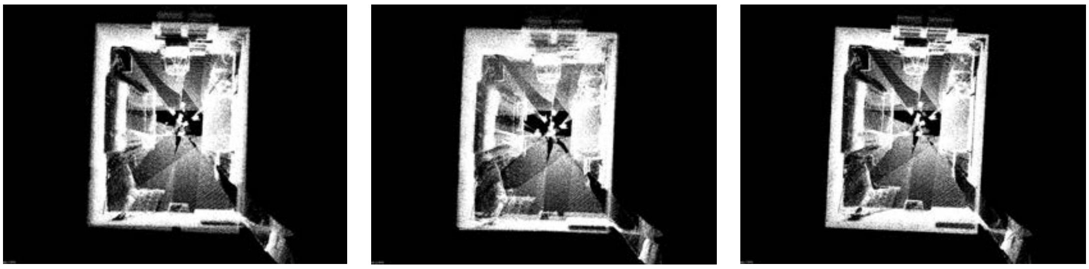


(a) (b)

(c) (d)

Figure 15: The results of ICP and EnICP localization comparing to the ground truth. (a) X-axis component, (b) Y-axis component, (c) Z-axis component, (d) the Euclidean distance error.

The point cloud generated by the traditional ICP and the proposed method EnICP are shown in Figure 16(a) and 16(b), respectively. The result shows that the performance of EnICP is better than traditional ICP.



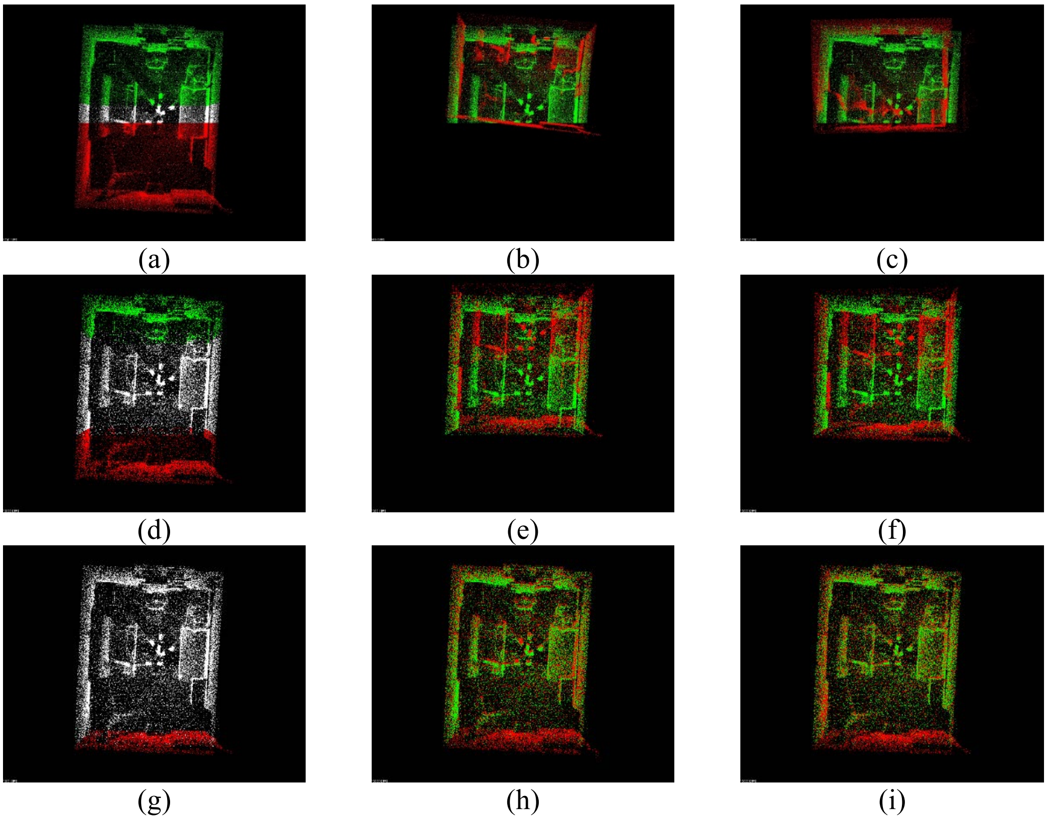
(a) (b) (c)

Figure 16: The top view of 3D point cloud map.
(a) by traditional ICP, (b) by proposed EnICP, (c) the ground truth.

Finally, to find the relative positions and features in the overlapping area between target cloud and source clouds, the performance of different overlapping ratios are discussed. Here, the overlapping ratios: 10%, 50%, and 90%, are tested. The dataset of the Office cloud has been divided into two parts, the target cloud is in the red region, and the source cloud is in the green region. The overlapping area between the target cloud and the source cloud is defined by the overlapping ratio and denoted as the white area. Figure 17 shows the top view of the mapping results. Figure 17(a), 17(d), and 17(g) show the target, the source, and the overlapping clouds, Figure 17(b), 17(e), and 17(h) show the mapping results by using the traditional ICP approach, and Figure 17(c), 17(f), and 17(i) show the mapping results by using the proposed EnICP approach. The numerical registration error is summarized in Table 1. These results show that, with increasing the overlapping ratio, the mapping results become more precise. In addition, the result of the proposed EnICP approach is more robust than the traditional ICP approach. Generally speaking, with the structure-entropy features, the performance affects by outliers can be improved.

Table 1: The numerical Registration Error (meter) of Euclidean Distance

Overlapping Ratios	By ICP Approach	By EnICP approach
10%	1.9826	1.6571
20%	1.6574	1.5457
30%	1.4562	1.5139
40%	1.3144	1.3899
50%	1.1320	0.9814
60%	0.8278	0.4706
70%	0.5318	0.2177
80%	0.2934	0.0519
90%	0.0399	0.0073
100%	0.0003	0.0000



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 17: The top view of mapping results with different overlapping ratios.
(a)-(c), (d)-(f), (g)-(i): 10%, 50%, 90%.
(a), (d), (g): The target, the source, and the overlapping clouds.
(b), (e), (h): By the traditional ICP approach.
(c), (f), (i): By the proposed EnICP approach.

6 Summary

In this paper, a three-dimensional surface reconstruction of indoor environment based on structure-entropy feature is proposed. The proposed method uses new concept to descript the feature of spatial change. The original frames can be replaced by points with the entropy energy, not only increasing the robust of the proposed method, but also reducing the computation. And the entropy image matching can reject outliers efficiently and provide a rough initial position for ICP. Experimental results demonstrate that the accuracy of the proposed method is better than traditional ICP algorithm. And the proposed method is unaffected by color or even in complete darkness compared to RGB-based feature mapping method. Moreover, the experiment of different overlapping ratio has also been presented. The result shows that the proposed method is more robust than the traditional ICP algorithm. The proposed method can not only be used on Kinect data, but also on other sensors which provide the spatial geometric information. Most complete mapping systems have a closed-loop form architecture, such as the detection of loop closures and globally consistent alignment of all frames. With the feedback of error, the mapping system performs more robust. Although the performance of the proposed method is better than the traditional ICP algorithm, the accumulating error still increases with number of frames. The proposed method can be improved with implement of global optimization algorithm, such as sparse bundle adjustment. Moreover, the proposed method only can be implemented in static environment. A framework of dynamic environment is the next work of this thesis. An experiment of localization can be conducted with a same scene. Through the same scene captured by different viewpoints, the effect of outlier can be reduced. Then, the result only can be affected by the characteristics of the mapping algorithms.

Acknowledgement

This work was supported in part by the Ministry of Education, Taiwan, and the Ministry of Science and Technology, Taiwan, under Grants, NSC-102-2221-E-002-247-MY3 and MOST 105-2221-E-002 -135 -MY3.

References

[1] Jungong Han, Ling Shao, Dong Xu, and Jamie Shotton, "Enhanced Computer Vision with Microsoft Kinect Sensor: A Review," IEEE Trans. on Cybernetics, 43(5), Oct. 2013.

[2] Tim Bailey and Hugh Durrant-Whyte, "Simultaneous Localization and Mapping (SLAM): Part II State of the Art," IEEE Robot. & Autom. Mag., 13(3): 108-117, Sep. 2006.

[3] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "RGB-D mapping - Using Kinect-style depth cameras for dense 3D modeling of indoor environments," The International Journal of Robotics Research, 31(5): 647-663, Feb. 2012.

[4] J. Biswas and M. Veloso, "Depth camera based indoor mobile robot localization and navigation," in Proc. of 2012 IEEE Int'l Conf. on Robot. & Autom., May 14-18, 2012.

[5] Serdar Gedik and A. Aydın Alatan, "3-D Rigid Body Tracking Using Vision and Depth Sensors," IEEE Transactions on Cybernetics, 43(5), Oct, 2013.

[6] João Emílio Almeida, Rosaldo J. F. Rossetti, and António Leça Coelho, "Mapping 3D Character Location for Tracking Players' Behaviour," in Proc.of 2013 8th Iberian Conf. on Information Systems and Technologies, June 19-22, 2013.

[7] A. Davison, I. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," IEEE Trans. on Pattern Analysis and Machine Intelligence, 29(6), Jun. 2007.

[8] David Nister, Oleg Naroditsky and James Bergen, "Visual Odometry," in Proc. of IEEE Conf.on Computer Vision and Pattern Recognition, Jun. 27-Jul. 2, 2004.

[9] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "RGB-D mapping: Using depth cameras for dense 3-D modeling of indoor environments," in Proc. of the 12th Int'l Symp. on Experimental Robotics, Dec. 18-21, 2010.

[10] E. Rosten, R. Porter, and T. Drummond, "Faster and Better: A machine learning approach to corner detection," IEEE Trans. on Pattern Analysis and Machine Intelligence, 32(1):105-119, Jan. 2010.

[11] N. Engelhard, F. Endres, J. Hess, J. Sturm, and W. Burgard, "Realtime 3-D visual SLAM

with A hand-held camera," in Proc. of RGB-D Workshop on 3D Perception in Robotics at the European Robotics Forum, 2011.

[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. on Robotics, 26(3): 424-441, Jun. 2010.

[13] R. Newcombe, A. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in Proc. of 10th IEEE Int'l Symp. on Mixed and AR, Oct. 26-29, 2011.

[14] S. Izadi, D. et al., "KinectFusion: Real-time 3D reconstruction and interaction using a moving depth camera," in Proc. of 24th annual ACM Symp. on User Interface Software and Technology, pp. 559–568, Oct. 2011.

[15] Romeil Sandhu, Samuel Dambreville, and Allen Tannenbaum, "Point Set Registration via Particle Filtering and Stochastic Dynamics," IEEE Trans. on Pattern Analysis and Machine Intelligence, 32(8), Aug. 2010.

[16] Michael Kazhdan and Hugues Hoppe, "Screened Poisson Surface Reconstruction," ACM Transactions on Graphics, 32(3), Jun. 2013.

[17] D. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, 2(60): 91-110, 2004.

[18] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-Up Robust Features (SURF)," Computer Vision and Image Understanding (CVIU), 110(3): 346-359, Jun. 2008.

[19] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary Robust Independent Elementary Features," in Proc. of 11th ECCV, Sep. 5-11, 2010.

[20] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in Proc. of IEEE Int'l Conf. on Computer Vision, Nov. 6-13, 2011.

[21] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary Robust Invariant Scalable Keypoints," in Proc. of IEEE Int'l Conf. on Computer Vision, Nov. 6-13, 2011.

[22] Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast Retina Keypoint," in Proc. of IEEE Int'l Conf. on Computer Vision and Pattern Recognition, June 16-21, 2012.

[23] Jens Berkmann and Terry Caelli. "Computation of Surface Geometry and Segmentation Using Covariance Techniques." IEEE Trans. on Pattern Analysis and Machine Intelligence, 16(11):1114-1116, Nov. 1994.

[24] A.E. Johnson and M. Hebert, "Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes," IEEE Trans. on Pattern Analysis and Machine Intelligence, 21(5): 433-449, May 1999.

[25] Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik, "Recognizing Objects in Range Data Using Regional Point Descriptors," in Proc. of 8th ECCV, May 11-14, 2004.

[26] F. Tombari, S. Salti, and L. Di Stefano. “Unique Signatures of Histograms for Local Surface Description.” in Proc. of 11th ECCV, Sep. 5-11, 2010.

[27] R. B. Rusu, N. Blodow, and M. Beetz, “Fast Point Feature Histograms (FPFH) for 3D Registration,” in Proc. of IEEE Int’l Conf. on Robot. & Autom., May 12-17, 2009.

[28] Hyoseok Hwang, Seungyong Hyung, Sukjune Yoon, and Kyungshik Roh, “Robust Descriptors for 3D Point Clouds using Geometric and Photometric Local Feature,” in Proc. of the IEEE/RSJ Int’l Conf. on Intelligent Robot. & Syst., Oct. 7-12, 2012.

[29] David Nister, Oleg Naroditsky, and James Bergen, “Visual Odometry,” in Proc. of 2004 IEEE Conf. on Computer Vision and Pattern Recognition, Jun. 27-Jul. 2, 2004.

[30] Gionis, P. Indyk, and R. Motwani, “Similarity Search in High Dimensions via Hashing,” in Proc. of 25th Int’l Conf. on Very Large Data Bases, Sep. 7-10, 1999.

[31] G. H. Golub, and C. Reinsch, “Singular Value Decomposition and Least Squares Solutions,” Numerische Mathematik, 14(5): 403-420, 1970.

[32] P. J. Besl and N. D. Mckay, “A Method for Registration of 3-D Shaped,” IEEE Trans. on Pattern Analysis and Machine Intelligence, 14(2): 239-256, Feb. 1992.

[33] Y. Chen and G. Medioni, “Object Modeling by Registration of Multiple Range Images,” Image and Vision Computing, 10(3): 145–155, 1992.

[34] M. Magnusson, A. Lilienthal, and T. Duckett, “Scan registration for autonomous mining vehicles using 3D-NDT,” Journal of Field Robotics, 24(10): 803–827, 2007.

[35] The CloudViewer. In PCL website. Retrieved June 6, 2014 from http://pointclouds.org/documentation/tutorials/cloud_viewer.php#cloud-viewer

[36] F. Pomerleau, M. Liu, F. Colas, and R. Siegwart, “Challenging Data Sets for Point Cloud Registration Algorithms,” Int’l J. of Robotic Research, 31(14): 1705-1711, Dec. 2012.

[37] ASL Dataset: Apartment, website. Retrieve June 18, 2014, from https://github.com/ethz-asl/libpointmatcher/blob/master/doc/ICPIntro.md