Point Cloud Registration Algorithm Based on Overlapping Region Extraction

Jun Li¹, Fei Qian¹ and Xianfu Chen¹

¹ Department of Electronic Science and Technology, University of Science and Technology of China, Hefei, Anhui, 230026, China

*Xianfu Chen’s e-mail: xfchen@ustc.edu.cn

Abstract. During the process of the point cloud registration, the problem that the accuracy is not high is due to the unknown relative position of the multi-view point clouds and the diversity of the target structure. This paper proposes a point cloud registration method based on extracting overlapping regions to solve it. First of all, according to the characteristics of the target geometric structure, each point cloud is divided into blocks and the ESF multi-dimensional shape descriptors of each point cloud block are constructed, the region with the greatest similarity between the descriptors, namely the overlapping region between the point clouds. Then our algorithm uses the Super Four Point Fast Robust Matching (Super4PCS) algorithm to execute a coarse registration of point clouds in overlapping regions, the initial positions of the point clouds after the coarse registration are obtained according to the consistency constraints, and finally uses the iterative closest point algorithm (ICP) to precisely register the overlapping region and obtain the final point cloud model. Compared with traditional Super4PCS algorithm, the experimental results show that the method proposed in this paper effectively improves the accuracy and accelerates the process of point cloud registration.

1. Introduction

Point clouds can well represent the geometric characteristics and spatial position of targets, and are widely used in the field of 3D reconstruction [1] and robotics [2]. Point cloud registration is an important part of it, point cloud registration can calculate the best rigid transformation matrix of multi-view point clouds and affect the accuracy of model reconstruction.

At present, there are many types of point cloud registration algorithms, which are roughly divided into coarse registration and precision registration according to the steps of point cloud registration. Coarse registration is used to solve large unknown relative conversions, and it is commonly used in global registration algorithms [3-6]. The LORAX algorithm [7] uses machine learning for point cloud registration, and uses neural networks to perform feature compression on the depth map to obtain a 5 by 2 matrix as a descriptor for point cloud registration. The probabilistic registration methods [8-9] have demonstrated promising results on datasets, Magnusson et al. [10] relies on the normal distribution transform (NDT), which represents the density of the scans as a structured GMM. Besides, point cloud registration algorithms tend to extract local features, obtain matches between features in the two point clouds, and use RANSAC [11] or other robust estimators to estimate the relative pose. Four-Points Congruent Sets(4PCS) algorithm is a fast and robust point cloud registration method that uses coplanar four-point affine invariance, its performance is stable to noise, but in a point cloud with a large amount of data, its calculation cost is high. Super4PCS algorithm is an algorithm using
intelligent index to accelerate 4PCS introduced by Nicolas Mellado et al. [12], which reduces the algorithm complexity and performs registration faster and better. The above algorithms use the entire point cloud for feature point extraction and matching, which requires a large amount of calculation and may cause many mismatches.

The purpose of precision registration is to minimize spatial location differences between point clouds on a coarse registration basis. Even if there are only a small number of accurately matched point pairs, compared with a large number of mismatched point pairs, it can better describe the rotation and translation of the two point clouds. Iterative Closest Point (ICP) algorithm proposed by Besl and McKay et al. [13] and its improved algorithm [14-16] are classic precision registration algorithms, and the ICP algorithm has requirements for good initial position and high overlap rate. Using overlapping regions for feature point extraction and matching can effectively improve the registration efficiency and accuracy. On this basis, this paper proposes a point cloud registration algorithm based on extracting overlapping regions of the point clouds, and compares this method with the registration using Super4PCS and FPFH algorithm [17] directly, it can obtain a highly accurate point cloud registration when the point cloud overlap rate is low, and improve the running speed of the algorithm.

2. Acquisition of overlapping regions

2.1. Normal Vector and Curvature Estimation

The normal and curvature values of each point in the point cloud do not change with the movement of the object, and can well represent the geometric characteristics of the object. Principal Component Analysis (PCA) is a commonly used method for calculating normal vector and curvature. Using KD-Tree to quickly search for the k-neighborhood nbhb(p_i) of the query point p_i, construct a covariance matrix Cov(p_i) for each query point and other points in its domain, and decompose its features.

\[
Cov(p_i) = \sum_{j \in nbhb(p_i)} (p_j - \bar{p})(p_j - \bar{p})^T
\]

(1)

\[
Cov(p_i) \cdot \lambda_j = \lambda_j \cdot \mathbf{v}_j, \quad j \in \{0, 1, 2\}
\]

(2)

Formula (1) represents the construction of the covariance matrix. Where p_i is the neighborhood point; \( \bar{p} \) is the center of mass of nbhb(p_i); \( \lambda_j \) is the jth eigenvalue of the covariance matrix, and the corresponding eigenvector is represented by \( \mathbf{v}_j \).

Formula (2) represents the feature decomposition of the covariance matrix, three eigenvalues \( \lambda_0, \lambda_1, \lambda_2 \) are obtained after the covariance matrix C is solved, Where \( \lambda_0 < \lambda_1 < \lambda_2 \), using these three eigenvalues to calculate the surface curvature \( \rho \) of p_i, where \( \rho = \frac{\lambda_0}{\lambda_0 + \lambda_1 + \lambda_2} \).

2.2. Point cloud segmentation

Point cloud segmentation is the basis of extracting overlapping regions of point clouds. There are differences between point clouds of different viewing angles, but geometric structures such as the normal angle and curvature characteristics of the surface of the overlapping region don’t change with the different viewing angle. The specific point cloud segmentation process is showed as follows:

Step 1: According to the method in section 2.1, calculating the normal vector and curvature of each point in the two point clouds P, Q, and sorting them according to the curvature value.
Step 2: Taking the point with the smallest curvature value as the initial seed point and adding it to the seed point queue.

Step 3: Using KD-Tree to search the neighboring points of the current seed point. When the angle between the normal vector of the seed point and a point is less than the threshold value $\sigma_1$, and the difference between the point’s curvature and the seed point’s is less than the threshold $\sigma_2$, adding the point to the clustering region. At the same time adding the field point to the seed point queue.

Step 4: Deleting the current seed point, taking out the new seed point and executing STEP 3 until the seed point queue is empty to get a point cloud block.

Step 5: For the remaining points, executing STEP 2, 3, 4 and looping execution, and finally outputting all the point cloud blocks.

2.3. Constructing Feature Descriptors for Point Clouds Blocks

Dividing point clouds $P$ and $Q$ from different perspectives of the same point cloud into two collection of regions $\{P_1, P_2, ..., P_m\}$, $\{Q_1, Q_2, ..., Q_n\}$. Constructing feature descriptors and comparing the similarity of feature descriptors of different regions, we can find the overlapping region of the two point clouds. The higher the similarity, the more likely the region is an overlapping region.

Ensemble of Shape Functions (ESF) is a commonly used descriptor in point cloud recognition. It is composed of three different global shape features. It describes regional features from three aspects of distance, angle and region. It is good to classify the geometric structure of objects, and it is robust to noise and incomplete surfaces. ESF uses a voxel grid as an approximation of the real surface, selects three points randomly as a unit to traverse all points in the region, and uses $D_2$, $D_3$, $A_3$ and $D_2$ ratio distribution functions to solve the distance, angle, square root of triangle region between pairs of points. Combining with the relationship between the connection line and the surface of the region, a shape function combined histogram representing a set of three angles (outgoing angle, incoming angle, mixed angle), three regions (incoming point cloud region, outgoing point cloud region, mixed point cloud region), three distances (point-in distance, point-out distance, mixed distance) and a distance ratio (line-to-line ratio) is constructed.

The ESF descriptor is a 640-dimensional vector composed of the probability distribution values of the above 10 histograms. After obtaining the ESF descriptor of each region, the region with the most similar shape features between the point clouds of two different view is the overlapping region. Fig 1 shows the ESF descriptor of each region of Bunny.

![Figure 1. ESF descriptors for different regions](image)

3. Our approach

In the original 3D point cloud data, using the Super4PCS algorithm and ICP algorithm tends to spend a large amount of calculation and consume a long time to search the corresponding point. In order to achieve better performance, this paper extracts the overlapping regions of point clouds from different views as a pre-processing, so that the overlapping regions are used as initial data to be matched in the
Super4PCS algorithm, and the transformation matrix is applied to the global transformation. It makes the algorithm more targeted and greatly reduces the time of searching the matching points. At the same time, the ICP algorithm can obtain better convergence under a good initial value, and reduce the time complexity of matching.

3.1. Coarse Registration
The global registration algorithm Super4PCS uses the invariance of the intersection ratio of intersecting line to search the corresponding four-point set in the global region, calculates the optimal transformation matrix, and completes the point cloud registration according to the principle of consistency constraints.

Taking the overlapping regions in the two point clouds extracted above as initial registration data and recording them as \( P_c, Q_c \). First, selecting three points randomly which are not collinear from the source point cloud \( P_c \) and finding a fourth point which is approximately coplanar with the previous three points. The distance between the points should be as large as possible. The above four points form a coplanar four-point base \( B = \{ P_a, P_b, P_c, P_d \} \). Then, using equations (3) and (4) to find the distance and intersection ratio of the two straight lines.

\[
\begin{align*}
    d_1 &= \| P_a - P_b \|, \quad d_2 = \| P_c - P_d \| \\
    r_1 &= \frac{\| P_a - e \|}{\| P_a - P_b \|}, \quad r_2 = \frac{\| P_c - e \|}{\| P_c - P_d \|}
\end{align*}
\] (3)

Finally, according to \( r_1 \) and \( r_2 \), a four-point set approximately congruent to \( B \) is searched in the target point cloud \( Q_c \), and the error is within the range of \( \delta \). But there may also be wrong corresponding point sets, as shown in Fig 2, but according to \( < r_1, r_2, d_1, d_2, \theta > \), the only corresponding four point set can be determined in the target point cloud.

![Figure 2. Non-uniqueness of coplanar four-point set](image)

3.2. Precise Registration
After the above coarse registration, the initial transformation matrix of two point clouds is obtained, but the registration accuracy of the point cloud of coarse registration is not high, and there will be a little deviation between the two point clouds, so we use the ICP algorithm to perform precise registration. Selecting the point \( p_i \) in the point cloud \( P_c \), and finding the point \( q_i \) with the shortest Euclidean distance from the point cloud \( Q_c \), and using \( (p_i, q_i) \) as the closest point pairs to solve the transformation matrix. Through continuous iteration, the error function (5) is minimized, and the optimal transformation matrix is finally obtained, so that the two point clouds become coincident.

\[
E(R, t) = \sum_{i=1}^{n} \left\| q_i - (Rp_i + t) \right\|^2
\] (5)

Where \( n \) is the number of nearest point pairs, \( p_i \) is a point in the target point cloud, \( q_i \) is the
closest point to \( p_i \) in the source point cloud, \( R \) is the rotation matrix, and \( t \) is the translation vector.

The algorithm proposed in this paper can narrow the search range in the registration process, extract a part of the overlapping region that can be used for registration, and improve efficiency and accuracy of the registration.

4. Experiments and analysis
We conduct three sets of experiments to test the accuracy and efficiency of the proposed method. Our hardware platform is VS2013 and PCL1.8.1, the operating system is Windows 7, Intel(R) Core™ i5-4590 CPU @3.30GHz 16GB. The selected experimental data comes from Stanford's 3D scan repository, and Geometry-Hub.

4.1. Overlapping Region
Using the method introduced in the second section to divide the point cloud into different regions, the result is shown in Fig 3, different colored regions represent different point cloud regions. The first two graphs of Fig 3(a) represent two point clouds from different perspectives, and the latter two graphs represent the overlapping regions of the two point clouds, Fig 3(b) and Fig 3(c) are showed in the same way.

![Figure 3. The result of extracting overlapping regions](image)

4.2. Experimental Results and Analysis
The optimal transformation matrix of overlapping regions registration is applied to the global point cloud according to the principle of consistency constraint to obtain the final point cloud registration model. Performing point cloud registration on three sets of point cloud data with similar structures, large overlapping regions, and small overlapping regions.

![Figure 4. The result of point cloud registration](image)

Stanford Bunny: The result of aligning two partial scans of the Stanford Bunny with relative viewpoint difference 90° are shown in Fig 4. The number of source point cloud is 79312, the number of target point clouds is 59629, and the average distance between the points in the point cloud is
0.00060m. Fig 4 (a) and Fig 4 (b) show the registration result using traditional FPFH algorithm and SAC-IA, they tend to fall into local optimization, resulting in point clouds mismatch. The registration result obtained by the Super4PCS algorithm is shown in Fig 4 (c), the back of the Bunny model is more dislocated. As shown in Fig 4 (d), the back of the Bunny model is more relevant and significantly improved than the traditional Super4PCS and ICP algorithm.

Stanford Dragon: The result of aligning two partial scans of the Stanford Bunny with relative viewpoint difference 24° are shown in Fig 5. The number of source point cloud is 41841, the number of target point cloud is 43467, and the average distance between the points in the point cloud is 0.00058m. The Stanford Dragon’s structure is complex, using the global algorithm Super4PCS to match two point clouds, the model's belly and melon generate mismatch, as shown in the Fig 5(c). The other three algorithms perform well, as shown in the Fig 5(a), Fig 5 (b), Fig 5 (d).

Geometry-Hub Bed: Selecting overlapping regions for registration, and the algorithm matches well, as shown in Fig 6. The number of source point cloud is 110372, the number of target point cloud is 92517, and the average distance between the points in the point cloud is 0.00440m. The global registration algorithm Super4PCS is prone to fall into mismatch. And the search for coplanar four-point set is fast, the running time of different algorithms on different data sets is shown in Table 1. The algorithm proposed in this paper reduces the search time for matching point pairs, and accelerates the running time of the algorithm.

| Data     | Overlapping regions Number | FPFH Time(s) | SAC-IA Time(s) | Super4PCS Time(s) | Our Time(s) |
|----------|-----------------------------|-------------|----------------|-------------------|-------------|
| Bunny    | 997/1904                    | 121.08      | 21.32          | 42.40             | 28.67       |
| Dragon   | 727/1062                    | 96.89       | 40.04          | 15.99             | 6.33        |
| Bed      | 11359/15301                 | 100.56      | >200           | 32.47             | 23.38       |

5. Conclusion
In this paper, a point cloud registration method based on extracting overlapping regions is proposed, and the experimental results show that the method can not only improve the running speed, but also have higher registration accuracy, and can show stability when the point cloud difference is large. However, the method proposed in this paper requires the target point cloud to be with certain geometric structure characteristics, so it should focus on studying the more efficient overlapping region extraction methods in the future work, so that the point cloud registration method is not only highly accurate but also universal.
References
[1] Ma Y, Wang Y, Mei X, et al. Visible/Infrared Combined 3D Reconstruction Scheme Based on Nonrigid Registration of Multi-Modality Images With Mixed Features[J]. IEEE Access, 2019:1-1.
[2] Kim P, Chen J, Cho Y K. SLAM-driven robotic mapping and registration of 3D point clouds[J]. Automation, 2018, 89(MAY):38-48.
[3] Lei H, Jiang G, Quan L. Fast Descriptors and Correspondence Propagation for Robust Global Point Cloud Registration[J]. IEEE Transactions on Image Processing, 2017, 26(8):3614-3623.
[4] Aiger D, Mitra N J, Cohen-Or D. 4-points congruent sets for robust pairwise surface registration[J]. Acm Transactions on Graphics, 2008, 27(3):1-10.
[5] Liu Y, Wang C, Song Z, et al. Efficient Global Point Cloud Registration by Matching Rotation Invariant Features Through Translation Search[C]// European Conference on Computer Vision. Springer, Cham, 2018.
[6] W.Wohlkinger and M. Vincze, " Ensemble of shape functions for 3D object classification[C]"//2011 IEEE International Conference on Robotics and Biomimetics, Karon Beach, Phuket, 2011, pp. 2987-2992.
[7] G. Elbaz, T. Avraham and A. Fischer, "3D Point Cloud Registration for Localization Using a Deep Neural Network Auto-Encoder," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 2472-2481, doi: 10.1109/CVPR.2017.265.
[8] Lawin, Felix Järemo, Danelljan M, Khan F S, et al. Density Adaptive Point Set Registration[J]. 2018.
[9] HGMR: Hierarchical Gaussian Mixtures for Adaptive 3D Registration[J]. 2018, 10.1007/978-3-030-01267-0(Chapter 43):730-746.
[10] Magnusson M, Andreasson H, Nuchter A, et al. Appearance-based loop detection from 3D laser data using the normal distributions transform[C]// IEEE International Conference on Robotics & Automation. IEEE, 2012.
[11] Schnabel R, Wahl R, Klein R. Efficient RANSAC for Point-Cloud Shape Detection[C]// John Wiley & Sons, Ltd. 2007:214-226.
[12] Mellado N, Aiger D, Mitra N J. Super 4PCS Fast Global Pointcloud Registration via Smart Indexing[J]. Computer Graphics Forum, 2014, 33(5):205-215
[13] Besl P J, Mckay H D. A method for registration of 3-D shapes IEEE Trans Pattern Anal Mach Intell[J].IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, 14(2):239-256
[14] Yang J, Li H, Campbell D, et al. Go-ICP: A Globally Optimal Solution to 3D ICP Point-Set Registration[J].IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(11):2241-2254
[15] Lu W, Wan G, Zhou Y, et al. DeepICP: An End-to-End Deep Neural Network for 3D Point Cloud Registration[J]. 2019.
[16] Serafin J, Grisetti G. NICP: Dense Normal Based Point Cloud Registration[C]. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2015:742-749.
[17] Rusu R B, Blodow N, Beetz M. Fast Point Feature Histograms (FPFH) for 3D registration[C]// IEEE International Conference on Robotics & Automation. IEEE, 2009.