An Improved RANSAC Surface Reconstruction Study

Ma Li, Zhang Jinming*, Ma Dongchao and Fu Yingxun
School of Information, North China University of Technology, Beijing China

*E-mail: jimzhangx@qq.com

Abstract. Surface reconstruction technology has always been an important research area in graphic image. However, efficiently processing massive amounts of high-precision point cloud data is still a problem worth studying. Based on the characteristics of point cloud data and the existing research results, this paper proposes a RANSAC algorithm based on voxel segmentation and applies it to 3D surface reconstruction. By analyzing the characteristics of the original point cloud data, the method improves the reconstruction speed of the 3D model while retaining a large amount of effective feature information and ensuring high precision. The experimental results show that compared with the reconstruction algorithm based on the original RANSAC algorithm, the proposed algorithm can effectively reduce the surface reconstruction time of 47.51%, and the model error after processing is only one thousandth compared with the original model.

1. Introduction

Surface reconstruction technology has been widely used in industrial high-precision parts error detection, digital cultural relics engineering, urban and rural construction and planning, precision medical analysis, virtual reality and augmented reality technology, military simulation training and other fields. With the rapid development of 3D laser scanning instruments, it is easy to collect a large amount of three-dimensional data on the surface of the object, that is, point cloud data, and the point cloud data set contains the spatial geometric features and surface characteristics of the measured object. Use a large amount of point cloud data to reconstruct a surface map to build a high-precision 3D model. However, due to various irresistible factors such as the error of actual operation, there are bound to be many redundant points and outliers in the original point cloud data set. Therefore, it has important practical significance to optimize the processing of point cloud data sets and improve the efficiency of surface reconstruction.

Based on the improved random sampling consistency algorithm of voxel segmentation, this paper analyzes the feature properties of point cloud data and retains a large number of effective feature information to ensure the high precision of the model and reduce the denoising time of point cloud. The method proposed in this paper is compared with the application of the original RANSAC algorithm in surface reconstruction. The experimental results show that the proposed method can reconstruct the high-precision 3D model more quickly and efficiently.

2. Related Works

Shi and other researchers proposed a simplified method based on clustering, which is to classify the point cloud data by 3D raster and iteratively calculate the normal vector deviation to ensure the point cloud data characteristics, but the calculation is very large[1]. Chu et al. proposed to use two-
dimensional SIFT to perform multi-view cloud data registration and splicing, extracting and matching feature points in effective texture images, and removing the interference of holes and invalid data in point cloud data. The calculation process is simple and improved accuracy and effect[2]. It is proposed in Simon Kriegel to iteratively determine the 3D surface model by boundary detection and surface trend estimation of the acquired model. When performing a new laser scan, point cloud data is integrated into the triangle mesh and the probabilistic voxel space in time. This method can quickly collect information on the surface of the object, but the algorithm takes a long time[3]. Anette Eltner summarizes the state of the art in SfM workflows in photogrammetry and geosciences, as well as the development of high-performance digital sensors and important software innovations developed in the field of computer-based vision and visual perception research. Processing extends to 3-D to generate point cloud data from a series of non-calibrated images[4]. For underwater scanning technology, Ricard Campos proposed a full 3D surface reconstruction method for underwater environment to denoise unstructured point cloud data[5]. Wu Mengqi et al. proposed the use of Point Feature Histograms (FPFH) and SIFT to achieve adaptive stitching of point clouds, while considering geometric features and image features, the algorithm has good stability[6]. JUAN LI et al. used random sampling consistency to eliminate erroneous point cloud data, and finally reconstructed the surface of point cloud data by singular value decomposition[7]. EDS and Raindrop companies in the United States developed Imageware software[8] and Geomagic Studio software[9] respectively. Imageware can quickly fit point cloud data directly into high-quality surfaces, while Geomagic Studio can efficiently use point cloud data to create polygonal grids and finally generate NURBS surfaces.

The surface reconstruction algorithm mentioned above and the existing common reconstruction methods (poisson reconstruction, etc.) can realize the prototype reconstruction of point cloud data from different degrees, but these methods do not consider improving the running speed of reconstruction.

3. Improvement of RANSAC Algorithm Based on Voxel Segmentation Idea

3.1 Algorithm Background and Problem
Fischler and Bolles first proposed the RANSAC (random sampling consistency) algorithm in 1981 [10]. They use the RANSAC algorithm to solve the LDP problem in the 3D model reconstruction. At present, RANSAC has been applied to geometric primitive detection, wide baseline stereo matching, motion segmentation, stitching, robust feature image matching and other fields. It estimates the parameters of a mathematical model from a set of observation data sets by iterative calculations. However, it is an indeterminate algorithm as it only has a certain probability to get a reasonable result, the number of iterations must be increased for better results.

The random sampling consistency algorithm is widely used in point cloud data preprocessing. It has a good effect on some isolated points and redundant points generated when processing the scan. It is a process of constructing a basic subset consisting of only intra-point data by randomly sampling out-of-office points. The basic idea is: in the parameter estimation, instead of treating all possible input point cloud data indiscriminately, firstly, a judgment criterion model is designed for the specific problem, and the judgment criterion is used to iteratively eliminate those and the estimated the point cloud data with inconsistent parameters is then estimated by the correct point cloud data. This algorithm is a well-established robust estimation method. In computer vision and many other research fields, robust estimation of model parameters is a core issue. Guo Jidong et al. mentioned that the random sampling consistency algorithm can still obtain ideal processing results for data with error rate exceeding 50%, which is one of the most effective robust estimation algorithms[11].

3.2 Basic Idea of Improved Algorithm
Based on the existing RANSAC random sampling consistency algorithm, this paper introduces the idea of voxel grid partition space, which realizes the effective speed improvement of point cloud denoising. Firstly, according to the existing original point cloud data N, spatial voxel segmentation is
performed in the three-dimensional virtual space, that is, the point cloud data set is wrapped by many
unit voxels, so that each point cloud data is included in the unit body. Within the prime, there may be
multiple point cloud data within each unit voxel. Figure 1 is a schematic diagram of voxel
segmentation of a point cloud data set; Figure 2 is a spatial state of point cloud data in a unit voxel.

Figure 1. Schematic diagram of voxel segmentation  Figure 2. Point cloud data in unit voxels

After the voxel partitioning is completed, the original point cloud data set is divided into several
point cloud data subsets. It is necessary to compress the point cloud data in each unit voxel according
to certain criteria, but at the same time as the point cloud compression, it is necessary to retain some of
the regional features described in the subset of point cloud data within the unit voxel. The algorithm
selects the data point \( G(x, y, z) \) closest to the unit voxel center of gravity \( Q(x, y, z) \) as the feature
information retention point of the data points in the subset of point cloud data.

Set the vertex closest to the origin of the coordinate system in each voxel as \( D(i, j, h) \), The voxel is the \( i \)-th voxel in the \( x \)-axis direction, the \( j \)-th voxel in the \( y \)-axis direction, the \( z \)-th voxel in the \( z \)-axis direction, the position vector is \( d = \left( \begin{array}{c} i \\ 0 \\ 0 \end{array} \right) \), and the coordinate vector is \( p = (x, y, z) \).

The voxel side length is \( l = (l_x, l_y, l_z) \), which represents the voxel side length in the \( x \), \( y \), and \( z \) axis
directions, respectively. Therefore, the vertex \( D(i, j, h) \) obtained by \( p = dl \) is obtained. The
center of gravity \( G(x, y, z) \) and the coordinate vector \( g = (x, y, z) \) in the unit voxel can be calculated by \( g = dl + 0.5l \).

The side length \( l = (l_x, l_y, l_z) \) of a unit voxel directly affects the amount of point cloud data within
a unit voxel, and thus requires a reasonable estimate. It can be seen from the conclusion of the
literature[12] that when the side length \( l \) is large, the efficiency of the point cloud compression is
affected. When the side length \( l \) is too small, an empty body element is generated. The unit voxel is a
cube, that is \( l_x = l_y = l_z \), and the calculation formulas (1) to (4) are as follows. Where \( \alpha \) is the scale
factor used to adjust the side length of the unit voxel; \( \theta \) is the scale factor; \( n' \) is the number of point
cloud data per unit volume in the ideal state; \( L_x, L_y, L_z \) are the maximum coordinate values of the
entire point cloud dataset along the \( x \)-axis, \( y \)-axis, and \( z \)-axis, respectively, and \( V \) is the minimum
voxel required to cover all point cloud data.

\[
l' = \left( \frac{\theta \alpha^3}{n'} \right)^{\frac{1}{3}} \tag{1}
\]

\[
n' = \frac{N}{V} \tag{2}
\]
The final formula of unit voxel side length is expressed as:

\[ l' = \left( \frac{\theta L_x L_y L_z \alpha^3}{N} \right)^{\frac{1}{3}} \]  

The center of gravity \( G(x, y, z) \) coordinate vector \( g = (x, y, z) \) is:

\[ g = ((i + 0.5)l', (j + 0.5)l', (h + 0.5)l') \]

Traversing each point cloud data \( Q_k (x_k, y_k, z_k) \), \( k = 1, 2, \ldots, n \) within the voxel. Calculate the spatial Euclidean distance \( s_k \), \( (k = 1, 2, \ldots, n) \) of each data point to the voxel center of gravity \( G(x, y, z) \).

\[ s_k = \sqrt{(x-x_k)^2 + (y-y_k)^2 + (z-z_k)^2} \]

Get the smallest \( s_k \), \( (k = 1, 2, \ldots, n) \), retain the point cloud data, and discard the remaining \( n - 1 \) point cloud data in the subset of point cloud data. Figure 3 shows the variation of voxel partitioning point cloud compression before and after a unit voxel.

\[ V = L_x \times L_y \times L_z \]

\[ l' = \left( \frac{\theta L_x L_y L_z \alpha^3}{N} \right)^{\frac{1}{3}} \]

\[ V = L_x \times L_y \times L_z \]

The final formula of unit voxel side length is expressed as:

\[ l' = \left( \frac{\theta L_x L_y L_z \alpha^3}{N} \right)^{\frac{1}{3}} \]

\[ g = ((i + 0.5)l', (j + 0.5)l', (h + 0.5)l') \]

Traversing each point cloud data \( Q_k (x_k, y_k, z_k) \), \( k = 1, 2, \ldots, n \) within the voxel. Calculate the spatial Euclidean distance \( s_k \), \( (k = 1, 2, \ldots, n) \) of each data point to the voxel center of gravity \( G(x, y, z) \).

\[ s_k = \sqrt{(x-x_k)^2 + (y-y_k)^2 + (z-z_k)^2} \]

Get the smallest \( s_k \), \( (k = 1, 2, \ldots, n) \), retain the point cloud data, and discard the remaining \( n - 1 \) point cloud data in the subset of point cloud data. Figure 3 shows the variation of voxel partitioning point cloud compression before and after a unit voxel.

**Figure 3.** Changes in data points within voxels

In the classic RANSAC formula, the key to solving the model is to find all valid interior points from a set of data points. The so-called inner points are the data points that match the "best" model. The number of valid interior points is usually unknown. The RANSAC algorithm finds all the required interior points and corresponding models with a certain probability by repeatedly drawing random samples from the input data point set. \( m \) is the number standard used to determine the suitability of the model in the data set.

Suppose \( t \) is the number of times the current iteration is calculated, and the initial value is 0.

1. When \( t \) is smaller than the target iteration number \( r \), \( num \) data points are randomly selected from the data set, and then a model \( mb\_model \) suitable for \( num \) is constructed.
2. The remaining \( (N - num) \) points are traversed, from which data points that satisfy the model \( mb\_model \) are found and counted in the number of data points applicable to the model.
3. When the number of data points applicable to the model is greater than the set standard number \( m \), it is considered that the optimal model \( b\_model \) in this iterative calculation is found.
4. Then through a certain strategy to determine the applicability of \( b\_model \) for all points in the data set.
5. If the applicability of \( b\_model \) is higher than the optimal model \( model \) in the previous \( (t-1) \) iterations, then the \( model \) and current best applicability are updated.
Repeat ① to continuously find the optimal model until the end of the iterative calculation.

In the original RANSAC algorithm, it is assumed that \( w \) is the probability that each time a point is selected from the point cloud data set \( N \) is exactly an inner point. The value of \( w \) is usually unknown, but can be roughly estimated by equation (7).

\[
w = \frac{m}{N}
\]

(7)

\( P \) is the ideal probability that the original RANSAC algorithm provides an effective model after running. According to the theoretical results of [13], the number of iterations \( r \) is a function of \( P \). \( \text{Min} \) is the minimum value for the number of data points in the model. Satisfy formula (8)

\[
r = \frac{\ln(1-P)}{\ln(1-w_{\text{min}})}
\]

(8)

In the improved RANSAC algorithm integrated with the voxel partitioning idea, the number of unit voxel \( N' \) in \( V \) can be calculated according to formula (1), (2) and (7), that is

\[
N' = V \div l^3 = V \div \frac{\theta \alpha^3 V}{N} = \frac{N}{\theta \alpha^3}
\]

(9)

Since the improved algorithm performs point cloud compression within the unit voxel, \( N' < N \) must hold. Under the premise that the model fitness degree \( m \) of the original RANSAC algorithm is constant, the probability \( w' \) of selecting the inner point from the point cloud data set is higher than \( w \). This means that after voxel partitioning, the applicability of the improved RANSAC algorithm generation model to the same point cloud dataset is effectively improved.

\[
w' = \frac{m}{N'} > w = \frac{m}{N}
\]

(10)

Similarly, voxel segmentation optimizes the selection of finite data points in a unit voxel. From the formula deduction, we can draw another conclusion: the number of iterations of the RANSAC algorithm after \( r' \) improvement, the ideal probability \( P \) of the effective model and the applicable Under the premise that the minimum number \( \text{min} \) of the model data points is constant, the number of iterative calculations is reduced, and the efficiency of the algorithm is improved.

\[
r' = \frac{\ln(1-P)}{\ln(1-w'_{\text{min}})} < r = \frac{\ln(1-P)}{\ln(1-w_{\text{min}})}
\]

(11)

The improved RANSAC algorithm based on the fusion of voxel segmentation can solve the effective de-noising model with higher probability and faster speed, it also provides a new point cloud data set that is more regular and easier to process for the subsequent point cloud reconstruction.

4. Experimental results and analysis

The experiment was implemented using the VS2010. In order to verify the excellent characteristics of the above algorithm, the optimization effect of the RANSAC algorithm based on voxel segmentation on the de-noising and surface reconstruction of point cloud datasets was investigated.

Figure 4 is a comparison of the effects of a set of point cloud data sets in the first set of experiments after the improved RANSAC algorithm. It can be seen from Figure 4(a) that the point cloud density of the set of point cloud data at the edge is higher than the internal density, and there are some separation points that are outside the overall point cloud data set. Figure 4(b) shows the processing effect of the improved RANSAC algorithm. The density of the edge point cloud data is greatly reduced, but it is basically consistent with the internal density, and the geometric features of the edge regions are preserved.

The second set of experiments uses the commonly used surface reconstruction algorithms, namely the Greedy Projection Triangulation reconstruction algorithm, the Poisson reconstruction algorithm and the Moving Cube reconstruction algorithm to compare and improve the optimization effect of the improved RANSAC algorithm on the surface reconstruction algorithm. The point cloud dataset used in
the experiment was acquired by industrial part scanning, and Figure 5 is the visualization effect of opening the point cloud data directly in Geomagic Studio software. Many redundant points and outliers can be seen very intuitively.

**Figure 4.** Improved RANSAC algorithm processing renderings

**Figure 5.** Original point cloud data

**Figure 6.** Greedy Projection Triangulation reconstruction algorithm point cloud and grid diagram

**Figure 7.** Poisson reconstruction algorithm point cloud and grid diagram

**Figure 8.** Moblie Cube reconstruction algorithm point cloud and grid diagram
Figure 6 (a)(b), 7 (a)(b) and 8 (a)(b) respectively show the Greedy Projective Triangular surface reconstruction algorithm, Poisson reconstruction algorithm and Mobile Cube reconstruction algorithm based on the original RANSAC point cloud data denoising processing point cloud renderings and grid renderings.

Figure 6(c)(d) 7(c)(d) and 8(c)(d) respectively shows the point cloud renderings and grid renderings of the Greedy Projection Triangulated surface reconstruction, Poisson reconstruction algorithm and Mobile Cube reconstruction algorithm based on the fusion voxel partitioning idea’s RANSAC algorithm.

In the comparison of the three sets of pictures, the optimization effect of Figure 6 and Figure 7 is the most intuitive. It can be clearly seen that the redundant area on the circumference surface is reduced, and the outliers in the center of the sphere are completely removed. Moreover, in Figure 9, it can be intuitively seen that the time taken by the algorithm is reduced. The visual effect in Figure 8 is not obvious, but the improved RANSAC algorithm is more efficient from the algorithm execution time comparison in Figure 9.

![Algorithm running time comparison](image)

**Figure 9.** Algorithm running time comparison

|                      | Greedy PT | Poisson | Mobile Cube |
|----------------------|-----------|---------|-------------|
| Time spent by original RANSAC algorithm(second) | 146.192   | 406.841 | 264.355     |
| Time spent by RANSAC algorithm based on voxel segmentation(second) | 335.22    | 743.768 | 606.785     |

**Table 1.** Error comparison between model based on improved RANSAC algorithm and original model

|                      | coordinates of the spheric center | radius of the spheric |
|----------------------|----------------------------------|-----------------------|
| original model       | (-0.038,0.085,1.173)             | 110.002               |
| model based on improved RANSAC algorithm | (-0.032,0.086,1.177) | 110.008               |

Through the time-consuming data obtained from the experiment, the average optimization time-consuming rate of the improved RANSAC algorithm is 47.51%. The relevant experimental data shows that the point cloud data set obtained by the improved RANSAC algorithm is used to calculate the
radius of the spheric and the coordinates of the spheric center. Compared with the result calculated by the original RANSAC algorithm, the error is only in the thousandth position, maintaining the accuracy of 96.48% of the model. The detailed data is shown in Table 1.

5. Conclusion

In the process of surface reconstruction, RANSAC (Random Sample Consensus) is frequently applied to denoise processing, but the traditional RANSAC algorithm takes a lot of time in this process. In this paper, the improved RANSAC algorithm based on voxel segmentation is proposed. The improved algorithm effectively reduces the number of iterative calculations and achieves the purpose of reducing time consumption. The experimental result shows the improved algorithm makes the model error after processing only in the thousands of decimal places and reduces the algorithm execution time by 47.51%. However, the algorithm proposed in this paper still needs to be further studied in point cloud data hole repair.

Foundation items

National Natural Science Foundation of China(6172023,61371143)
Key B of Natural Science Foundation of Beijing(KZ201810009011)
Natural Science Foundation of Beijing(4172019)
Science and technology innovation project of North China University of Technology(18XN053)

References

[1] Shi Baoquan, Liang Jin, and Zhang Xiaogiang. Research on Point Cloud Reduction Technology for Feature Preservation[J]. Journal of Xi’an Jiaotong University, 2010.
[2] Chu Wei, Nie Chunmei, and Wang wei. Multi-viewpoint Cloud Data Registration and Splicing Algorithm Based on SIFT Features[J]. Semiconductor Optoelectronics, 2011, 32(3): 442-447.
[3] Kriegel S, Rink C, and Tim Bodenmüller. Efficient next-best-scan planning for autonomous 3D surface reconstruction of unknown objects[J]. Journal of Real-Time Image Processing, 2015, 10(4):611-631.
[4] Anette E, Andreas K, Carlos C, et al. Image-based surface reconstruction in geomorphometry &ndash; merits, limits and, developments[J]. Earth Surface Dynamics, 2016, 4(2):359-389.
[5] Campos R, Garcia R, Alliez P, et al. A surface reconstruction method for in-detail underwater 3D optical mapping[J]. The International Journal of Robotics Research, 2015, 34(1):64-89.
[6] Wu Mengqi, Li Zhongwei, Zhong Kai, et al. Point cloud adaptive stitching method based on geometric features and image features[J]. Acta Optica Sinica, 2015, 35(02).
[7] Li J, and Jiang G. Point clouds registration with surfaces of low curvatures[C] 2015 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC). IEEE, 2015.
[8] Niu Jianwei. Imageware reverse modeling technology and 3D printing [M]. Publishing House of Electronics Industry, 2014.
[9] Guo Peizhen,and Du Liming. Surface reconstruction and error analysis of point cloud data using Geomagic Studio[J]. GeoInfo World, 2015(1): 57-60.
[10] Fischler M A, and Bolles R C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography[J]. Communications of the ACM, 1981, 24(6):381-395.
[11] Guo Jidong, and Xiang Hui. A Robust Estimation Algorithm for a Basic Matrix [C] National Conference on Image Graphics. 2005.
[12] Lengyel E S. Voxel-Based Terrain for Real-Time Virtual Simulations[J]. Dissertations & Theses - Gradworks, 2010.
[13] Chen Fuxing, Wang Runsheng. Fast Random Sampling Consistency Algorithm Based on Pre-Test[J]. Journal of Software, 2005, 16(8): 1431-1437.