

## Research Article

# A Point Cloud Registration Algorithm Based on Feature Extraction and Matching

Yongshan Liu,<sup>1</sup> Dehan Kong ,<sup>2</sup> Dandan Zhao,<sup>1</sup> Xiang Gong,<sup>1</sup> and Guichun Han<sup>1</sup>

<sup>1</sup>Department of Information Science and Engineering, Yanshan University, Hebei Street, No. 438, Qinhuangdao, Hebei 066000, China

<sup>2</sup>Department of Information Engineering, Hebei University of Environmental Engineering, Jingang Street No. 8, Qinhuangdao, Hebei 066000, China

Correspondence should be addressed to Dehan Kong; [kdh0312@163.com](mailto:kdh0312@163.com)

Received 23 August 2018; Revised 18 November 2018; Accepted 9 December 2018; Published 24 December 2018

Academic Editor: Kauko Leiviskä

Copyright © 2018 Yongshan Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The existing registration algorithms suffer from low precision and slow speed when registering a large amount of point cloud data. In this paper, we propose a point cloud registration algorithm based on feature extraction and matching; the algorithm helps alleviate problems of precision and speed. In the rough registration stage, the algorithm extracts feature points based on the judgment of retention points and bumps, which improves the speed of feature point extraction. In the registration process, FPFH features and Hausdorff distance are used to search for corresponding point pairs, and the RANSAC algorithm is used to eliminate incorrect point pairs, thereby improving the accuracy of the corresponding relationship. In the precise registration phase, the algorithm uses an improved normal distribution transformation (INDT) algorithm. Experimental results show that given a large amount of point cloud data, this algorithm has advantages in both time and precision.

## 1. Introduction

The continuous development of three-dimensional (3D) scanning equipment has promoted the application of 3D point cloud reconstruction technology in the fields of mechanical manufacturing, medicine, robot navigation, and other industries. Registration technology plays an important role in 3D point cloud reconstruction. Due to the influence of the measuring equipment, object shape, and environment, a complete point cloud data model must be measured from multiple perspectives. It is an important step to integrate the point cloud in different coordinate systems into a unified coordinate system to obtain the complete data model of the object being tested.

With a large amount of point cloud data, the existing registration algorithms either improve the registration accuracy at the cost of time or increase the registration speed. How to achieve a balance between time and accuracy when dealing with a large volume of cloud data is a problem worth studying. In this paper, a point cloud registration algorithm based on feature extraction and matching is proposed to improve the speed while maintaining the accuracy of the registration of

a large amount of cloud data. The algorithm uses the fast point feature histogram (FPFH) descriptor [1] and Hausdorff distance [2] to extract and locate the corresponding feature points in order to minimize the number of registration points, thereby improving its speed. We propose an improved normal distribution transform (INDT) algorithm for precise registration and a mixed probability density function (PDF) to replace the single PDF, which further improves the accuracy of the algorithm.

The remainder of this article is organized as follows. The research situation is introduced and analyzed in Section 2. Section 3 introduces our rough registration algorithm. Section 4 provides the improved algorithm of the normal distribution transformation. Section 5 verifies the speed and accuracy of the registration algorithm based on experimental simulation data.

## 2. Related Works

The research on registration technology started in 1960, and it experienced a transition from the two-dimensional image

registration to a 3D model registration. Automatic registration is one of the three types of point cloud registration methods, which includes instrument-based and manual registration. Automatic registration has greater flexibility and does not require human intervention and auxiliary equipment. Therefore, most researchers study automatic registration. Point cloud registration technology mainly has two parts: rough registration and accurate registration. The registration algorithm in this paper is also composed of two parts, rough registration and precise registration, while registration of point clouds is mainly divided into rigid registration and nonrigid registration. The deformation of point clouds, which will directly affect the extraction of feature points, must be considered in nonrigid registration. This study focuses mainly on rigid registration.

*2.1. Initial Registration of Point Cloud.* The four current types of rough registration of point clouds are based on the following: (1) the measurement equipment; (2) human-computer interaction; (3) exhaustive thinking; and (4) features. The first two methods rely on the limitations of the auxiliary equipment and the need for manual intervention. The latter two constitute the main research direction.

The registration method based on exhaustive thinking can be divided into methods of (1) maximizing the point of rigid body transformation to obtain registration parameters and (2) finding the minimum transformation of the error function by traversing the transformation space quasi-parameter. These include the voting method [3], geometric hashing, search transformation space method [4], and RANSAC (RANdom SAMple Consensus) algorithm. In 1981, Fischler proposed the RANSAC algorithm, which is a method of fitting a mathematical model based on a sample. Meng [5] proposed the application of a sampling sphere and improved the original registration algorithm based on it; this method is based on the RANSAC idea. By setting up a sampling sphere model, the searching efficiency of the corresponding point pair is improved, and the time complexity of the algorithm is reduced. In 2015, Zhang Yonging [6] proposed a point cloud rough registration method based on the feature descriptors of a point feature histogram and FPFH, along with the principle of sampling consistency.

The feature-based registration method obtains the coordinate transformation parameters by performing a corresponding point pair search on the overlapping regions of the point clouds, thereby completing the point cloud registration. When looking for a corresponding point, the method chooses the point with the most similar geometry. In 2000, Chua [7] and others proposed the point signature (PS) method. This method constructs a point signature through extensive calculation; it is a complex process that is sensitive to noise. In 2005, through the unremitting efforts of Gelfand [8] and others, feature descriptors based on integral invariants were applied to the field of 3D registration, thereby ensuring the reliability of the registration results. Yan Jianfeng [9] proposed a method of rough registration using the curvature of the point cloud to extract feature points and using similarity to find pairs of points; this shortened the time of

registration, but the accuracy remained low. Yang Xiaoqing [10] improved the point cloud coarse registration algorithm based on the normal vector based on a point geometric feature extraction algorithm in 3D space. The key points are selected by comparing the characteristic degree of each point in the point cloud data (i.e., the angle between the normal points and adjacent points) and the threshold value. The main curvature is then calculated only for the key points, thereby increasing the speed of calculating the curvature and the efficiency of the corresponding point pair search. However, this method, facing the registration of a large volume of point cloud data, has a long running time or low precision. To solve this problem, we propose a rough point cloud registration algorithm to ensure a shorter running time and more accurate calculations.

*2.2. Point Cloud Accurate Registration.* The most classic point cloud accurate registration technology is the iterative closest point (ICP) algorithm. However, because ICP is used to find the corresponding point using the nearest neighbor search, the operation is too long, and the initial position of the point cloud is too high, so the normal distributions transform (NDT) algorithm is the main precision Quasi-technical research.

The NDT algorithm was proposed in 2003 by Biber et al. [11]. In 2006, Martin Magnusson [12] summarized 2D-NDT and extended it to the registration of 3D data through 3D-NDT. Magnusson's algorithm is faster than the current standard for 3D registration and is often more accurate. In 2011, Cihan [13] proposed a multilayered normal distribution transformation algorithm called MLNDT. The algorithm divides a point cloud into  $8n$  cells, where  $n$  is the number of layers, replaces the original Gaussian probability function with the Mahalanobis distance function as a fractional function, and uses the Newton and Levenberg-Marquardt (LM) optimization method to optimize the fractional function, with better registration speed than the previous NDT algorithm. In 2013, Cihan [14] further explained the MLNDT algorithm. In 2015, Hyunki Hong and B.H. Lee [15] proposed the key-layered normal distributions transform algorithm (KLNDT) based on the key layer normal distribution transformation; it has a good success rate and accuracy. In 2016, Hyunki Hong and B.H. Lee [16] proposed to convert the reference point cloud into a disk-like distribution suitable for the point cloud structure to improve the registration accuracy of the NDT algorithm. However, there are few studies of NDT algorithms in China. Zhang Xiao [17] performed a feature point search based on the speeded up robust features (SURF) algorithm, an improvement of the 3D-NDT algorithm. Hu Xiuxiang [18] proposed the normal aligned radial feature (NARF) algorithm, which improved NDT. Chen Qingyan [19, 20] study is based on curvature feature of NDT registration method; Zheng Fen [21] et al. improved 3D scale invariant feature transform (3DSIFT) algorithm and 3D-NDT algorithm. The above methods all improve the original NDT algorithm in different ways, with various accuracy and time advantages. However, it has always been a challenge to obtain a balance between time and accuracy when registering large

amounts of point cloud data. The registration algorithm in this paper improves the normal distribution transformation IND algorithm in the accurate registration stage, improves the speed of the registration algorithm, and ensures good registration accuracy.

Existing initial and precise registration algorithms face problems when dealing with a large volume of point cloud data. This paper proposes an algorithm that is based on features and matching, which improves the registration speed while ensuring good registration accuracy.

### 3. Rough Registration of Point Clouds

The main ideas of the initial algorithm are as follows. First, the feature points are extracted from the source point cloud and the target point cloud, respectively, reducing the number of points involved in registration and removing the redundant points and thus greatly reducing the registration time. Second, the high dimensional descriptor FPFH is used not only as a standard, but also the Moorhouse multihusband distance to further determine the correspondence between points. The RANSAC algorithm is then used to remove the corresponding pairs of errors. Finally, we use singular value decomposition (SVD) to solve the coordinate transformation matrix and apply it to the source point cloud to complete the coarse registration. Next, we introduce the main links of the rough registration algorithm.

**3.1. Feature Point Extraction.** The extraction of feature points reduces the number of point sets involved in registration, thereby improving the speed of registration. With fewer point cloud data points, the introduction of feature points greatly improves the algorithms speed. But a large volume of point cloud data will increase the running time of feature point extraction. This can easily burden the entire registration algorithm, which leads to long running time. To solve this problem, we judge the retention points before judging the feature points. The judgment of reserved points cannot only remove obvious outliers but can improve the efficiency of the whole feature point extraction algorithm. Moreover, the more point cloud data the greater the time advantage of the feature point extraction algorithm.

The method of extracting feature points in this paper is divided into two steps: (1) removing some points according to the mean curvature of the point, i.e., selecting the reservation point, and (2) determining the feature points by the concave and convex points, as described below.

The first step is the judgment of the reservation point. The mean curvature of each point in the point cloud is calculated, and the mean  $\mu$  and variance  $\sigma$  of the point average mean curvature are calculated. Second, the points with average curvature greater than  $\mu + p\sigma$  are chosen as reservation points to remove the other points in the point cloud. Among them,  $p$  is the proportional coefficient, whose value controls the number of reserved points. When  $p$  is large, there are fewer reserved points, and vice versa.

Curvature generally includes the principal, normal, mean, and Gaussian curvature. We use the mean curvature

to filter feature points. The normal curvature measures the degree of curvature of a surface in a certain direction, and the principal curvature represents the extreme value of the normal curvature. Gaussian curvature is an intrinsic measure of curvature; i.e., its value depends on how the distance on the surface is measured, rather than how the surface is embedded in space. Average curvature is an "external" bending measurement standard, which describes the curvature of a surface embedded in the surrounding space locally. Therefore, it can better reflect the degree of change and particularity of the current point, so in this paper, average curvature is used as the measure of screening feature points.

The second step is the judgment of the concave and convex points. If a point on any surface is concave or convex, it is taken as a feature point [22]. Based on this principle, we calculate the value of  $S(p)$  ( $s(p) \in (0, 1)$ ) using the following equation:

$$S(p) = \frac{1}{2} - \frac{1}{\pi} \arctan \frac{k_1(p) + k_2(p)}{k_1(p) - k_2(p)} \quad (1)$$

If  $S(p_i) > \max(S(p_{i1}), S(p_{i2}), \dots, S(p_{ik}))$ , then point  $p_i$  is a local convex point. If  $S(p_i) < \min(S(p_{i1}), S(p_{i2}), \dots, S(p_{ik}))$ , then  $p_i$  is a local concave point. In this, the  $S(p_{i1}), S(p_{i2}), \dots, S(p_{ik})$  are the  $S(P)$  value of the point  $p_i$  neighborhood points. Points in point cloud data that are concave points or bumps are characteristic points.

The steps to extract feature points from point cloud  $P$  are as follows:

- (1) Calculate the mean curvature of each point.
- (2) Calculate the mean  $\mu$  and variance  $\sigma$  of mean curvature and select points with mean curvature greater than  $\mu + p\sigma$  as retention points ( $p$  is proportional to the coefficient and the  $p$  is used to control the number of reserved points).
- (3) The concave points are judged by the reserved points, and the feature points are extracted.

**3.2. Corresponding Point Pair Lookup Based on FPFH and Hausdorff Distance.** To find the corresponding point is to find the nearest point of the query point in another point cloud. If we calculate the Euclidean distance from each point in the other point cloud to the query point, we will find the corresponding point pair, but this requires extensive calculation with little accuracy in the results. The accuracy of the corresponding point pairs directly affects the registration effect of point clouds. In this paper, the robust FPFH descriptor is added to the corresponding point pair search algorithm. The fast point feature histogram (FPFH) is based on the relationship between the sampling point and the neighborhood point normal, which is represented by 33 dimensional descriptors. In addition, to find the corresponding points more accurately and reduce the effect of error correspondences, the Hausdorff distance is introduced to the algorithm.

We use FPFH points and Hausdorff distances to find corresponding points. First, the feature points set in the source point cloud and target point cloud are described by FPFH. Next, we find the point of minimum difference between the FPFH points of each point of the source point

```

Input: source cloud  $P$ , target cloud  $Q$ 
Output: Registered point cloud  $outCloud$ 
1: Extract feature points from the source cloud and the target cloud separately
2: for all points  $p_i \in P$  do
3:   if  $p_i$  is a key point then
4:      $P_{key} \leftarrow p_i$ 
5:   end if
6: end for
7: for all  $p_i \in P_{key}$  do
8:    $P_{des-key} \leftarrow p_i$ 
9: end for
10: gets  $Q_{key}$  and  $Q_{des-key}$  in the same way
11: for point  $q_j \in Q_{des-key}$  do
12:   gets  $T_q$ 
13: end for
14: for point  $p_i \in P_{des-key}$  do
15:    $S \leftarrow$  most similar point to  $p_i$  in  $T_q$ 
16:    $C_{con} \leftarrow S_i$ 
17: end for
18: getRemainingCorrespondences();
19: estimateRigidTransformation( $P_{key}, Q_{key}$ );
20: transformPointCloud( $P, Q, transform\_matrix$ );
21: Return  $RS$ ;

```

ALGORITHM 1: Point cloud initial registration algorithm.

cloud feature point through the nearest neighbor search in the target point cloud feature point set. Finally, we calculate the Hausdorff distance; if it is less than the threshold value, then the two are corresponding points.

**3.3. RANSAC Culling Error Corresponding Point Pair.** The corresponding error pairs will affect the estimation of the final rigid transformation matrix, thus affecting the point cloud registration. Therefore, wrong corresponding point pairs must be eliminated. By finding the corresponding points based on FPFH and the Hausdorff distance in the previous section, we obtained a corresponding point pair with a higher matching degree, however, with a certain gap in the actual application of the distance point cloud registration. This is mainly due to noise in the gathering process of the point cloud, which leads to the topology of the point cloud data of the two times of obtaining the same area being not completely consistent, so there are some wrong correspondences. In this paper, the RANSAC (random sample consistency) algorithm is used to eliminate error correspondences.

**3.4. Description of the Algorithm.** The rough point cloud registration algorithm for feature extraction and matching mainly uses the FPFH description, Hausdorff distance, and RANSAC algorithm to perform pairwise registration of point clouds, aiming to provide their accurate registration of point clouds and good initial position. The specific algorithm description is shown as Algorithm 1.

In lines 1-6, the feature points are extracted according to the method mentioned in Section 3.1. The reserved points

are obtained in lines 2-4, and concave and convex points are judged in lines 5 and 6. Finally, the feature points of the source point cloud are obtained. Line 7 performs similar operations on the target point cloud to get the feature points set. Lines 8-12, as mentioned in Section 3.2, perform a lookup based on corresponding points. Line 13 uses RANSAC to eliminate the corresponding error points. Line 14 solves the rigid body transformation matrix. Line 15 transforms the source point cloud to complete the initial registration.

## 4. Accurate Registration Based on Improved NDT Algorithm

We use the improved NDT algorithm, INDT, for accurate registration. First, only the feature point sets of the source cloud and the target cloud are involved in the algorithm during the registration process. Second, the algorithm replaces a single probability density function with a mixed probability density function. Finally, the algorithm updates the Newton iterative algorithm through a linear search. We will describe the INDT algorithm in detail.

**4.1. Mixed Probability Density Function.** For noise points, the negative logarithm probability function of the normal distribution will grow without limit, which will greatly affect the result. Therefore, using a single probability density function, as shown in Formula (1), the registration effect is not good for noisy point cloud data. Like Lu Ju [23], we use

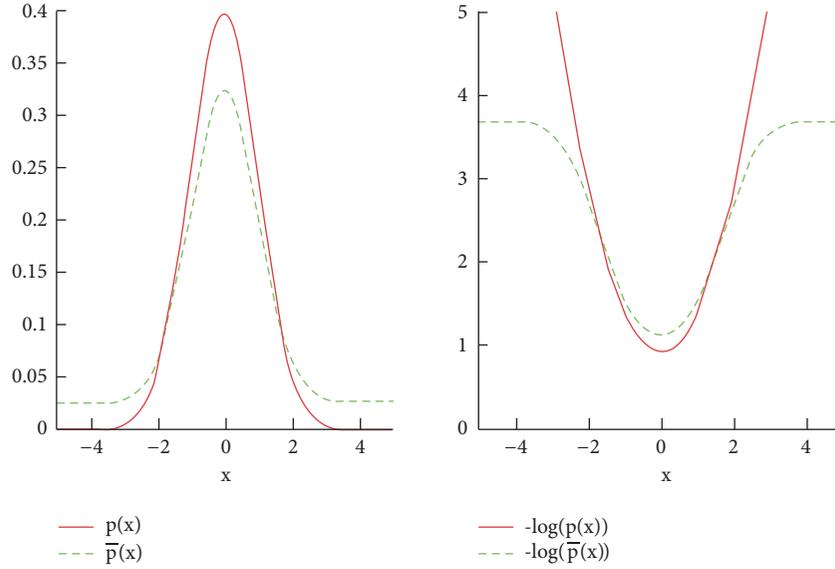


FIGURE 1: Comparison of single probability density function and mixed probability density function.

the PDF of the PDF, which consists of a normal distribution and a uniform distribution:

$$\bar{p}(\bar{x}) = c_1 \exp\left(\frac{(\bar{x} - \bar{\mu})^T \Sigma^{-1} (\bar{x} - \bar{\mu})}{2}\right) + c_2 p_0 \quad (2)$$

$p_0$  is the expected ratio of noise and  $c_1, c_2$  are used to determine the normalization of the probability density function by the spatial span of voxels. The effect of noise when using the mixed probability density function is bounded, as shown in Figure 1.

The form of  $\bar{p}(x) = -\log(c_1 \exp((\bar{x} - \bar{\mu})^T \Sigma^{-1} (\bar{x} - \bar{\mu})/2) + c_2)$  is a logarithmic function to be optimized. These terms have no simple first and second derivatives. However, the negative logarithm probability function can be approximated by a Gaussian function. This form of function is similar to the Gauss function,  $\tilde{p}(x) = d_1 \exp(-d_2 x^2 / 2\sigma^2)$ . The required fitting parameters should be consistent with  $x=0$ ,  $x=\sigma$  and  $x=\infty$ , as shown in Formula (3). Using the Gaussian function to estimate, the influence of a point in the current point cloud on the NDT fractional function is shown in Formula (4). The NDT fractional function is simpler than the derivative of Formula (2), but it still has the same general properties in the optimization. Since  $d_3$  only adds a constant offset to the fractional function when registering with NDT and does not change its shape or its optimized parameters,  $d_3$  is rounded off from Formula (4).

$$d_3 = -\log(c_2)$$

$$d_1 = -\log(c_1 + c_2) - d_3$$

$$d_2 = -2 \log\left(\frac{-\log(c_1 \exp(-1/2) + c_2) - d_3}{d_1}\right) \quad (3)$$

$$\bar{p}(\bar{x}_k) = -d_1 \exp\left(-\frac{d_2}{2} (\bar{x}_k - \bar{\mu}_k)^T\right) \sum_k^{-1} (\bar{x}_k - \bar{\mu}_k) \quad (4)$$

In Formula (4),  $\bar{x}_k - \bar{\mu}_k$  is the average number of NDT cells;  $\sum_k^{-1} (\bar{x}_k - \bar{\mu}_k)$  is the covariance of the NDT cell.

**4.2. Newton Iteration Algorithm for Parameter Solution.** Given point set  $x = \bar{x}_1, \bar{x}_2, \dots, \bar{x}_n$ , pose  $\bar{p}$ , and the transform function  $T(\bar{p}, \bar{x})$  that transforms  $\bar{x}$  by  $\bar{p}$ , for the current parameter vector, the NDT fractional function is shown in

$$s(\bar{p}) = -\sum_{k=1}^n \bar{p}(T(\bar{p} - \bar{x}_k)) \quad (5)$$

This corresponds to the possibility of point  $\bar{x}_k$  being located on the surface of the target point cloud after  $\bar{p}$  transformation.

The probability function requires the inverse of the covariance matrix  $\Sigma^{-1}$ . Where the points in a cell are completely coplanar or collinear, the covariance matrix is singular and cannot be inverted. In the 3D case, the covariance matrix calculated from three or fewer points is singular. Therefore, PDFs only calculate cells that contain more than three points. In addition, to prevent numerical problems, as long as it is found that the  $\Sigma$  is almost singular,  $\Sigma$  is slightly enlarged. If the maximum eigenvalue of  $\Sigma$ ,  $\lambda_3$ , is 100 times greater than the eigenvalue  $\lambda_1$  or  $\lambda_2$ , the smaller eigenvalue,  $\lambda_j$ , will be replaced by  $\lambda_j = \lambda_3/100$ . Replace the covariance matrix  $\Sigma$

```

Input: two point clouds  $P$  and  $Q$  to be registered
Output: Point cloud  $P'$  and  $Q'$  after accurate registration;
1: //Initialization
2: Divide the ell structure  $B$ 
3: for all points  $q_i \in Q$  do
4:   find the cell  $b_i$  containing the point  $q_i, b_i \in B$ 
5:   store  $q_i$  in  $b_i$ 
6: end for
7: for all cells  $b_i \in B$  do
8:    $p' \rightarrow$  all points in  $b_i$ 
9:    $\bar{m}u_i \rightarrow \frac{1}{n} \sum_{k=1}^m \bar{y}'_k$ 
10:   $\Sigma_i \rightarrow \frac{1}{m-1} \sum_{k=1}^m (\bar{y}'_k - \bar{\mu})(\bar{y}'_k - \bar{\mu})^T$ 
11: end for
12: //Registration
13: while not converged do
14:    $score \rightarrow 0$ 
15:    $\bar{g} \rightarrow 0$ 
16:    $H \rightarrow 0$ 
17:   for point  $p_i \in P_{des-key}$  do
18:     find the cell  $b_i$  containing the conversion function  $T(\bar{p}, \bar{x}_k)$ 
19:      $score \rightarrow score + \tilde{p}(T(\bar{p}, \bar{x}_k))$ 
20:     update  $\bar{g}$ 
21:     update  $H$ 
22:   end for
23:   solve  $H\Delta\bar{p} = -\bar{g}$ 
24:    $\bar{p} \rightarrow \bar{p} + \Delta\bar{p}$ 
25: end while

```

ALGORITHM 2: INDT algorithm.

with a matrix  $\Sigma' = \nu \wedge' \nu$ , where  $\nu$  contains the eigenvalue of  $\Sigma$  and  $\wedge'$  is shown in

$$\wedge' = \begin{bmatrix} \lambda'_1 & 0 & 0 \\ 0 & \lambda'_2 & 0 \\ 0 & 0 & \lambda'_3 \end{bmatrix} \quad (6)$$

The Newton algorithm can be used to find the parameter  $\bar{p}$  for optimization  $s(\bar{p})$ . The algorithm iteratively solves the equation  $H\Delta\bar{p} = -\bar{g}$ , where  $H$  and  $\bar{g}$  are the Hessian and the gradient vector of the fractional function  $s(\bar{p})$ , respectively. During each iteration,  $\bar{p} \rightarrow \bar{p} + \Delta\bar{p}$  (the increment is  $\Delta\bar{p}$ ). Let  $\bar{x}_k \equiv T(\bar{p}, \bar{x}_k) - \bar{\mu}_k$ ; in other words  $\bar{x}_k$  is the transformation of point  $\bar{x}_k$  by the current pose parameter, relative to the center of the cell PDF to which it belongs. The element  $g_i$  of the gradient vector  $\bar{g}$  is as shown as

$$g_i = \frac{\delta_s}{\delta_{p_i}} = \sum_{k=1}^n d_1 d_2 \bar{x}'_k \sum_k \frac{\delta_{\bar{x}'_k}}{\delta_{p_i}} \exp\left(-\frac{d_2}{2} \bar{x}'_k T \sum_k \bar{x}'_k\right) \quad (7)$$

The element  $H_{ij}$  of the Hessian matrix is shown in

$$\begin{aligned} H_{ij} = \frac{\delta_s^2}{\delta_{p_i} \delta_{p_j}} = \sum_{k=1}^n d_1 d_2 \exp\left(-\frac{d_2}{2} \bar{x}'_k T \sum_k \bar{x}'_k\right) \\ \cdot \left(-d_2 \left(\bar{x}'_k \sum_k \frac{\delta_{\bar{x}'_k}}{\delta_{p_i}}\right) \left(\bar{x}'_k \sum_k \frac{\delta_{\bar{x}'_k}}{\delta_{p_j}}\right)\right) \\ + \bar{x}'_k \sum_k \frac{\delta_{\bar{x}'_k}}{\delta_{p_i} \delta_{p_j}} + \frac{\delta_{\bar{x}'_k}}{\delta_{p_j}} \sum_k \frac{\delta_{\bar{x}'_k}}{\delta_{p_i}} \end{aligned} \quad (8)$$

Regardless of the dimension of registration, the gradient of the NDT fractional function and the expression of the Hessian matrix are the same and are similarly independent of the transformation representation.

**4.3. Description of the Algorithm.** The algorithm uses the feature point set of the source point cloud and the target point cloud to register and replaces the single probability density function with the mixed probability density function with a normal distribution and a uniform distribution and uses the Newton algorithm to iteratively solve the transformation parameters. The algorithm is shown as Algorithm 2.

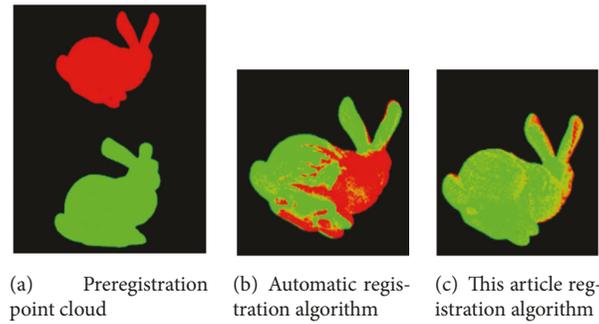


FIGURE 2: Result of point cloud registration.

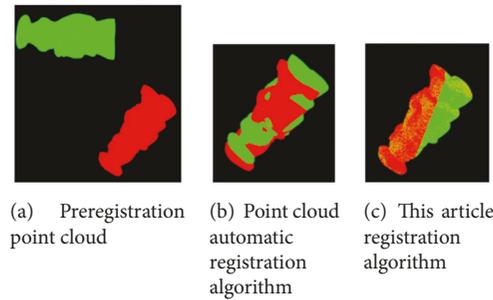


FIGURE 3: Results of point cloud chef registration.

## 5. Experimental Results and Analysis

The algorithm was compiled on CMake3.7.1 and debugged in Microsoft Visual Studio 2013 under the Win7 64-bit operating system. We ran the C++ development language and combined the PCL point cloud library to develop and maintain the algorithm. Because there is no need for libraries in PCL version 1.6, we used PCL1.8. Version 1.8 has many more cloud processing algorithms, and some modules have different usage.

Experiment 1: the point cloud data *rabbit.pcd* and *rabbit<sub>t</sub>.pcd* of Stanford University consist of 35,947 points. A comparison between the algorithm and the point cloud automatic registration algorithm based on feature extraction in reference 25 is shown in Figure 2, and the comparison of registration data is shown in Table 1.

The running time of this algorithm was 76.605 s (average of 10 results), and the root mean square error was 0.000600173 m. The point cloud automatic registration algorithm time was 129.57 s, and the root mean square error was 0.00367109 m. The time difference of the two methods was about 53 s, but the registration errors were quite different. From Figure 2 and Table 1, we can see that our algorithm has obvious advantages in both time and the accuracy of registration.

Experiment 2: Stanford University's *cheff.pcd* and *cheff<sub>t</sub>.pcd* point cloud data consist of 176,920 points. A comparison of the registration algorithm's registration results is shown in Figure 3, and the comparison of the registration data (including the running time and RMSE) is shown in Table 2.

Figure 3(a) shows the preregistration point cloud, and Figures 3(b) and 3(c) are, respectively, the registered point cloud. As can be seen from Figure 3, the registration effect of this algorithm is better than that of the point cloud automatic registration algorithm. The number of points in the Cheff model is 176,920, which is about five times that of the point cloud data in our first experiment. It can be seen from Table 2 that the time advantage of the algorithm is more obvious. In addition, the accuracy of this algorithm is still better.

Experiment 3: in this experiment the chicken point cloud is used in initial registration and accurate registration. In the initial registration stage, the SAC-IA (Sample Consensus-Initial Alignment) algorithm [24], the Huang's algorithm [25], and our algorithm are compared and the results are shown in Figure 4 and Table 3. In the accurate registration stage, NDT algorithm, Curvature NDT algorithm, and INDT algorithm are compared and the results are shown in Figure 5 and Table 4.

In the initial registration stage, in our algorithm, the source cloud and target cloud are almost overlapped while the RMSE is only 1.13234. Besides, the running time of 95.607 seconds is much shorter than the other two comparison algorithms.

In the accurate registration stage, the difference between the results of the three algorithms is almost indistinguishable from the naked eye. However, from the data in Table 4, the accuracy of INDT algorithm is slightly behind other two comparison algorithms and the difference is 3.0% and 4.3%, respectively, while the running time is only 2.1% and 5.2% of these two algorithms. Thus, the proposed algorithm has obvious advantages in terms of time while ensuring accuracy.

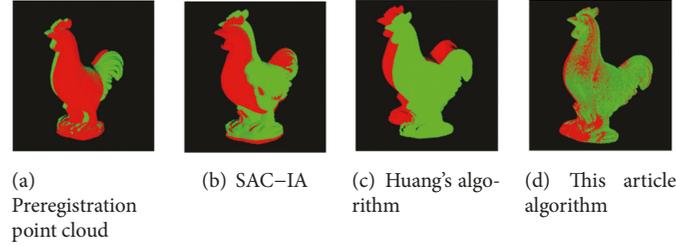


FIGURE 4: Results of point cloud chicken initial registration.

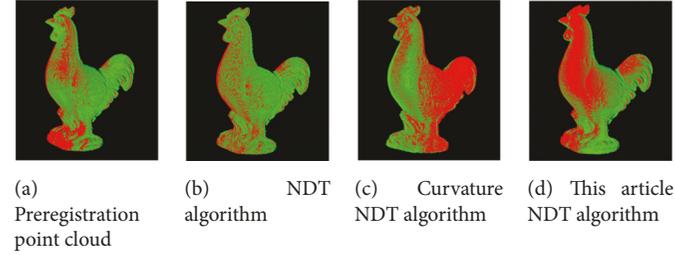


FIGURE 5: Results of point cloud chicken accurate registration.

TABLE 1: A comparison of the two registration methods.

Registration method	Time(s)	RMSE(m)
Point cloud automatic registration algorithm	129.57	0.00367109
This article's registration algorithm	76.605	0.000600173

TABLE 2: A comparison of the two registration methods.

Registration method	Time(s)	RMSE(m)
Point cloud automatic registration algorithm	2343.82	26.5153
This article's registration algorithm	184.75	0.0942159

TABLE 3: A comparison of three registration methods in initial registration.

Registration method	Time(s)	RMSE(m)
SAC - IA	992.982	4.96372
Huang's algorithm	484.648	25.3336
Our algorithm	95.607	1.13234

TABLE 4: A comparison of three registration methods in accurate registration.

Registration method	Time(s)	RMSE(m)
NDT	757.285	1.13069
Curvature NDT	309.705	1.117
INDT	16.236	1.16484

From the above experiments, we can find that the greater the amount of point cloud data, the more obvious the time advantage of the algorithm.

## 6. Conclusions

In this paper, a point cloud registration algorithm based on feature extraction and matching is proposed. The algorithm treats the feature point of the registration point cloud, uses FPFH and Hausdorff to find the corresponding point pair, and uses the RANSAC algorithm to remove the error corresponding point pair, thus completing the rough registration of the point cloud. In the stage of precise registration, the improved normal distribution transformation algorithm is used. This method has achieved remarkable results, especially when processing a large volume of point cloud data, where the advantages in registration time and accuracy are obvious. However, some parts of the algorithm require further study. We plan to carry out the following work. (1) In the experiment, the algorithm uses multiple experiments to determine the threshold. Next, we can study the automatic selection of various thresholds. (2) In this paper, the accuracy of the INDT algorithm in the registration of a large amount of point cloud data is improved, but there is still room for improvement, and we plan to study this.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This work is supported by the Natural Science Foundation of Hebei Province, China under Grant no. F2017203019.

## References

- [1] R. B. Rusu, N. Blodow, and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D registration," *IEEE International Conference on Robotics & Automation IEEE Press*, pp. 183–189, 2009.
- [2] M.-P. Dubuisson and A. K. Jain, "A modified Hausdorff distance for object matching," in *Proceedings of the International Conference on Pattern Recognition IEEE*, vol. 1, pp. 86–95, 2002.
- [3] I. Lysenkov and V. Rabaud, "Pose estimation of rigid transparent objects in transparent clutter," in *Proceedings of the 2013 IEEE International Conference on Robotics and Automation, ICRA 2013*, pp. 162–169, Germany, May 2013.
- [4] S. Ramalingam and Y. Taguchi, "A theory of minimal 3D Point to 3D Plane registration and its generalization," *International Journal of Computer Vision*, vol. 102, no. 1-3, pp. 73–90, 2013.
- [5] Y. Meng and H. Zhang, "Registration of point clouds using sample-sphere and adaptive distance restriction," *The Visual Computer*, vol. 27, no. 6-8, pp. 543–553, 2011.
- [6] Y. Zhang, *Research on Registration of Unorganized Point Cloud*, Changan university, 2015.
- [7] C. S. Chua, F. Han, and Y. K. Ho, "3D human face recognition using point signature," in *Proceedings of the Fourth IEEE International Conference*, pp. 233–238, 2000.
- [8] N. Gelfand, N. J. Mitra, L. J. Guibas et al., "Robust global registration," in *Proceedings of the Third Eurographics Symposium on Geometry Proceeding*, pp. 197–206, 2005.
- [9] Y. Jianfeng and D. Kezhong, "Point Cloud Registration Algorithm Based on Extracting and Matching Feature Points," in *Bulletin of Surveying and Mapping*, vol. 9, pp. 62–65, 2013.
- [10] X. Yang, *Study on Three-Dimensional Point Cloud Registration Method Based on The Normal Vector*, North University of China, 2016.
- [11] P. Biber and W. Straßer, "The normal distributions transform: a new approach to laser scan matching," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '03)*, vol. 3, pp. 2743–2748, IEEE, October 2003.
- [12] M. Magnusson, *3D Scan Matching for Mobile Robots with Application to Mine Mapping*, Örebro University, Örebro, Sweden, 2006.
- [13] C. Ulas and H. Temeltas, "A 3D Scan Matching Method Based On Multi-Layered Normal Distribution Transform," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 11602–11607, 2011.
- [14] C. Ulas and H. Temeltas, "3D Multi-Layered Normal Distribution Transform for Fast and Long Range Scan Matching," *Journal of Intelligent & Robotic Systems*, vol. 71, no. 1, pp. 85–108, 2013.
- [15] H. Hong and B. H. Lee, "Key-layered normal distributions transform for point cloud registration," *IEEE Electronics Letters*, vol. 51, no. 24, pp. 1986–1988, 2015.
- [16] H. Hong and B. Lee, "A method of generating multi-scale disc-like distributions for NDT registration algorithm," *International Journal of Mechanical Engineering and Robotics Research*, vol. 5, no. 1, pp. 52–56, 2016.
- [17] X. Zhang, A. W. Zhang, and Z. H. Wang, "Point Cloud Registration Based on Improved Normal Distribution Transform Algorithm," *Laser & Optoelectronics Progress*, vol. 51, no. 4, pp. 96–105, 2014.
- [18] H. U. Xiuxiang and L. Zhang, "Multi-view Point Cloud Registration Based on Improved 3 D-NDT Combining The Feature of NARF," *Journal of signal processing*, vol. 13, no. 12, pp. 1674–1679, 2015.
- [19] Y. C. Fang, Q. Y. Chen, N. Sun et al., *Three-dimensional normal distribution transformation point cloud registration method based on curvature feature*, CN 105069840, 2015.
- [20] Q. Chen, *The crane working environment oriented static scene 3D reconstruction*, Nankai University, 2015.
- [21] F. Zheng, RyadChellali, M. Y. Dai et al., "Research on Normal Distribution Transform Registration Method Based on developed 3D SIFT," *Journal of Computer Applications*, vol. 10, pp. 2875–2878, 2017.
- [22] Y. Wu, W. Wang, K. Lu, Y. Wei, and Z. Chen, "A New Method for Registration of 3D Point Sets with Low Overlapping Ratios," *Procedia CIRP*, vol. 27, pp. 202–206, 2015.
- [23] J. Lu, W. Liu, D. Dong, and Q. Shao, "Point cloud registration algorithm based on NDT with variable size voxel," in *Proceedings of the 34th Chinese Control Conference, CCC 2015*, pp. 3707–3712, China, July 2015.
- [24] X. Chen and Y. Zhu, "Point cloud registration technology based on SAC-IA and improved ICP algorithm," *Journal of Xian Engineering University*, vol. 31, no. 3, pp. 395–401, 2017.
- [25] H. Yuan, F. Da, and H. Tao, "An Automatic Registration Algorithm for Point Cloud Based on Feature Extraction," *Chinese Journal of Lasers*, vol. 42, no. 3, pp. 242–248, 2015.

