ICIC Express Letters
Part B: Applications
Volume **8**, Number **4**, April **2017**

ICIC International ©2017 ISSN 2185-2766
pp. **689–694**

A REGISTRATION ALGORITHM FOR THREE-DIMENSIONAL POINT CLOUDS BASED ON LOCAL APPROXIMATION SURFACES

Yanhui Zhang[1], Dianzhu Sun[1] and Yanrui Li[2]

[1]School of Mechanical Engineering
Shandong University of Technology
No. 266, Xincun West Road, Zhangdian District, Zibo 255049, P. R. China
59743593@qq.com

[2]School of Mechanical Engineering
Xi'an Jiao Tong University
No. 28, Xianning West Road, Xi'an 710049, P. R. China

Received September 2016; accepted December 2016

Abstract. *The point clouds obtained in multiple measurements are inconsistent, so it is inapplicable to the iterative closest point (ICP) algorithm and its variants that directly establish correspondences through the original points. For this problem, the local approximation surface is utilized in this paper to help establish correspondences, converting the initial point-to-point registration to the surface-to-surface registration. This algorithm strives for a tradeoff between efficiency and accuracy. The detail process of building the surfaces and the overall procedure of this algorithm are elaborated in this paper. Three-dimensional registration tests of multi-view point clouds show that this algorithm performs better in terms of registration accuracy.*
Keywords: Point cloud, Registration, ICP, Local approximation surface

1. **Introduction.** In industrial applications, blades, drills and other complex industrial parts are susceptible to damage due to the severe, harsh working conditions and they are always hugely expensive; therefore, the restoration of these used parts is an urgent demand in relative areas [1,2]. In the rebuilding process based on reverse engineering (RE), the most significant step to ensure the repairing accuracy of complicated parts is to improve the accuracy of damaged area geometric modeling, which is then determined by registration accuracy [3,4]. The registration technology of three-dimensional point clouds has been one of the hottest researches in recent years.

A typical rigid-registration method of three-dimensional point clouds is the iterative closest point (ICP) algorithm [5] which is proposed by Besl and McKay in 1992. The ICP algorithm defines the point clouds to be registered as the source and the target. During the registration process, the target keeps fixed and the source is transformed to the target progressively. In each iteration process, the ICP algorithm searches for the nearest point in the target for each point in the source, generates corresponding points and computes the coordinate transformation matrix transforming the source to the target. The original ICP algorithm provides a prototype, but it takes too much time when dealing with a large number of points because it traverses all the points. Moreover, when the point numbers in the source and the target are unequal or the point clouds contain noises, the original ICP algorithm cannot achieve precise registration. As a result, many ICP algorithm variants have been proposed to solve this problem.

For optimizing the efficiency, [6,7] establish a KD-tree index for querying the nearest point, [8] adds a small perturbation to the translation matrix for fast convergence, and [9,10] replace the original objective function. For raising the accuracy, [11,12] employ a weighted linear combination of position distance and characteristic distance to find correspondence, and [13] proposes an iterative re-weighting method to tackle the problem of evaluating corresponding points. In order to generalize the ICP algorithm to make it available in a broader field, [14] proposes an algorithm that works well while processing point clouds with noises. [15] proposes an algorithm performing well in the registration of point clouds from damaged parts.

The 3D point clouds to be registered are acquired from surfaces of parts by measurement equipments; there is no guarantee that different measurement processes produce points of the same number. Consequently, the target point cloud is inappropriate to be directly used to establish correspondence in the registration process. However, the original ICP algorithm and its variants mentioned above compute the corresponding points directly from the target point cloud, which causes deviation between the selected point and the real corresponding point, generates an improper transformation matrix and finally influences the registration accuracy. In addition, because the optimization algorithm is often accompanied by huge time consuming, it is necessary to ensure the registration efficiency within an acceptable range while at the same time optimizing the registration accuracy.

In order to further improve the registration accuracy to meet the industry needs, this paper proposes an algorithm from the perspective of optimizing the corresponding points. This algorithm constructs approximation surfaces of local neighborhoods of the target to search for more reasonable potential corresponding points for each point in the source. Since building the surfaces is time-consuming and inefficient, this algorithm first applies a highly efficient algorithm to achieving a fast convergence, and then exploits the local approximation surfaces to improve the accuracy of the final registration.

The remaining content of the paper is summarized as follows. Section 2 elaborates the algorithm of building the approximation surface, followed by the whole registration algorithm detailed in Section 3. In Section 4, this algorithm is compared to others to validate this proposed method. Finally, a conclusion is given in Section 5.

2. **Building Approximation Surface of Local Neighborhood.**

2.1. **Parameterization of local neighborhood data points.**

3D point clouds to be registered are usually obtained by contact or non-contact measurement equipments and are always uniformly distributed; therefore, the k nearest neighbors algorithm can be used to obtain the points representing the local neighborhood area. In this algorithm, the k value should be selected based on the sampling density of the three-dimensional measuring device. If the k is exceptionally large, the approximation surface cannot be guaranteed to be an open surface. Conversely, if it is extremely small, more reasonable corresponding points cannot be acquired. Owing to the fact that 3D measurement equipments provide a large number of points, the R*-tree or the KD-tree should be applied as an index to enhancing the k nearest neighbors query efficiency. Assume the source and the target to be S and M, and S_i to be a point in S. With the given neighborhood querying number k, the small scale local neighborhood points M are parameterized as follows.

Step 1. Construct a spatial index for M using KD-tree.

Step 2. For each S_i, compute the closest point M_i from M, and find the k-nearest points of M_i from M. The resulting points constitute the point set X.

Step 3. Let c_i $(i = 1, 2, 3, 4)$ be the constant coefficients. Then the equation of the least square plane of X is

$$p = f(x, y, z) = c_1x + c_2y + c_3z + c_4 = 0 \tag{1}$$

Compute the least square plane p.

Step 4. Project the points in X onto the plane p. Let a be an arbitrary point in X, and its projection on plane p be a'. Let $\vec{n}$ be the normal vector of p, and then $\vec{n} = (c_1, c_2, c_3)$.

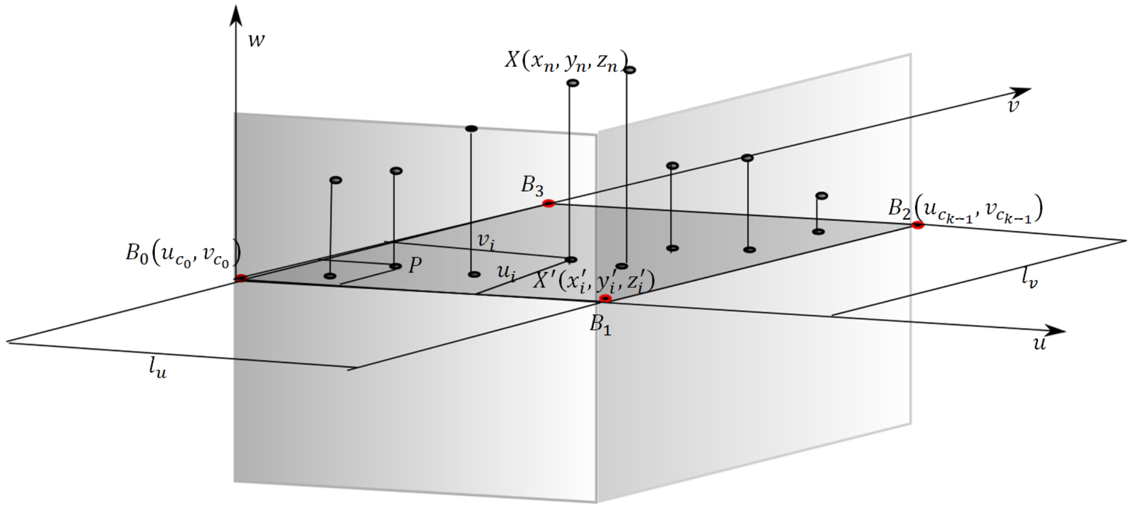


FIGURE 1. Parameterization of local neighborhood data points

The projection point set X' of X is computed by the follow formula:

$$a' = a - \frac{\vec{n} \cdot a + c_4}{||\vec{n}||||\vec{n}||}\vec{n} \tag{2}$$

Step 5. Use Graham Algorithm [16] to obtain the convex hull of X', based on which the algorithm in [17] is implemented to obtain the minimum bounding rectangle (MBR), which is represented by R.

Step 6. As shown in Figure 1, let one of the vertices $B_0(u_{c_0}, v_{c_0})$ of R be the origin, and its diagonal vertex be $B_2(u_{c_{k-1}}, v_{c_{k-1}})$. Establish the local coordinate system C. Transform all the data points of X' to the coordinate system C. l_u and l_v are the two side lengths of R. The formula to parameterize the local neighbor points is

$$u_i' = \frac{u_i - u_{c_i}}{l_u}, \ v_i' = \frac{v_i - v_{c_i}}{l_v} \tag{3}$$

2.2. **Approximation surface construction.** The $m \times n$ tensor product surface is defined as follows:

$$S(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u) B_{j,m}(v) P_{i,j} \tag{4}$$

where $P_{i,j}$ are the control points, and $B_{i,n}(u)$ and $B_{j,m}(v)$ are Bernstein Polynomials.

Assume $Q_t \in X$, $t = (0, 1, 2, \ldots, k)$, then

$$S(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u) B_{j,m}(v) P_{i,j} = Q_t \tag{5}$$

Let $D_{i,j}(u, v) = B_{i,n}(u) B_{j,m}(v)$, and

$$D = \begin{bmatrix} D_{0,0}(u_0, v_0) & D_{0,1}(u_0, v_0) & \cdots & D_{0,m}(u_0, v_0) & \cdots & D_{n,0}(u_0, v_0) & D_{m,n}(u_0, v_0) \\ D_{0,0}(u_k, v_k) & D_{0,1}(u_k, v_k) & \cdots & D_{0,m}(u_k, v_k) & \cdots & D_{n,0}(u_k, v_k) & D_{m,n}(u_k, v_k) \end{bmatrix} \tag{6}$$

then,

$$D \times \begin{bmatrix} P_{0,0} \\ \cdots \\ P_{0,m} \\ \cdots \\ P_{n,m} \end{bmatrix} = \begin{bmatrix} Q_0 \\ Q_1 \\ \cdots \\ Q_k \end{bmatrix} \tag{7}$$

Use singular value decomposition method (SVD) to solve Equation (7), and compute all the control points $P_{(i,j)}$ of the parametric surface. Therefore, the approximation surface $S(u, v)$ is determined.

3. **Registration Algorithm Based on the Local Neighborhood Approximation Surface.** The approximation surface of the local neighborhood reflects the actual surface of the real parts. Resample the approximation surface with a fixed interval, search for the nearest point from the discrete points in term of Euclidean distance, and then establish optimized corresponding points. Based on that, a more reasonable transformation matrix is obtained and then used to transform the source. Although this algorithm reduces the total number of iteration required for convergence, it is time-consuming in each iteration process when dealing with a large amount of points, which finally reduces the overall algorithm efficiency. In order to optimize the operating efficiency, the precise registration method is accomplished in two steps. To begin with, align the source to the target rapidly using a highly efficient algorithm, such as algorithm [6]. Secondly, apply the algorithm based on the approximation surface to further improving the registration accuracy. The whole process is detailed as follows.

Step 1. For each point S_i, construct the approximation surface according to Section 2, which is:

$$S = S(u, v), u \in [u_{c_0}, u_{c_{k-1}}], v \in [v_{c_0}, v_{c_{k-1}}] \tag{8}$$

With the discrete parameters u and v, compute the discrete points according to Equation (8). Then the Euclidean distance from S_i to all the discrete points d_i $(i = 1, 2, \ldots, m)$, where m is the number of the discrete points, can be obtained. Let $d_{\min} = \min\{d_1, d_2, \ldots, d_m\}$, and then $d_{\min}$ is the distance from S_i to the approximation surface. The data point M_i which generates $d_{\min}$ and S_i constitutes a pair of corresponding points.

Step 2. Let the translation matrix be $\vec{q}_t$, the rotation matrix be $\vec{q}_r$, and the total number of points in the source be N. Build the error function:

$$F(\vec{q}_r, \vec{q}_t) = \frac{1}{N} \sum_{i=1}^{N} ||M_i - (\vec{q}_r S_i' + \vec{q}_t)||^2 \tag{9}$$

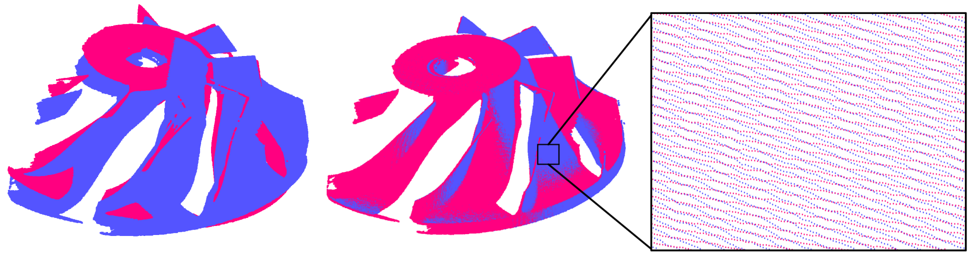
Solve Equation (9) by using the SVD method to get the optimal transformation matrix $\vec{q} = \vec{q}_r | \vec{q}_t$.

Step 3. Transform S_i using the acquired transformation matrix $\vec{q}$, that is $S_i' = S_i \times \vec{q}$.

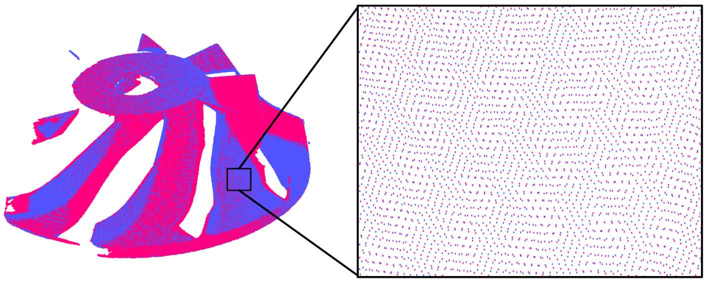
Step 4. Let $E^n = \sum_{i=1}^{n} ||S_i' - M_i||^2$, where n is the iteration number. The algorithm terminates when any one of the following conditions is satisfied. (1) $|E^n - E^{n-1}| < \delta$, in which δ is the preset threshold. (2) $n > I_{\max}$, in which $I_{\max}$ is the preset iterative limit. If any one of these two conditions is unsatisfied, the algorithm jumps to Step 1 for the next iteration.

4. **Experiment and Results.** Use 3D-CaMega-CPC flow binocular optical scanning device to scan the experimental part twice, with an interval degree of 15°, to get two point clouds of different coordinates. After removing the extra data and noises, the numbers of the two point clouds are 164538 and 165425. Perform the test in a hardware environment of Intel (R) core (TM) i3-4150-CPU@3.50GHz-4.00GB memory. The main aim is to verify whether the method proposed here is relatively more accurate than other methods that establish correspondence through the points in the target. Three algorithms, variants of the ICP algorithm, are tested with the same data for contrast. The mean distance of corresponding points is computed to value the registration accuracy, that is $E = \frac{1}{N} \sum_{i=1}^{N} ||S_i' - M_i||$ where N is the number of the source.

The initial point clouds are shown in Figure 2(a). Record the original mean distance between the corresponding points. Write programs according to the corresponding documents, using c++ language and applying KD-tree to improve algorithm efficiency. In



(a) Point clouds for registration (b) Point clouds after performing algorithm [15]



(c) Point clouds after performing the algorithm proposed in this paper

FIGURE 2. The results of the algorithms

TABLE 1. The data comparison of different algorithms

	number of points		initial E/(mm)	total time/(s)	final E/(mm)
	source	target			
algorithm [7]	164538	165425	92.293	205	1.36335
algorithm [14]	164538	165425	92.293	3211	1.00341
algorithm [15]	164538	165425	92.293	587	1.01121
this algorithm	164538	165425	92.293	3881	0.83611

order to avoid generating false correspondence caused by non-overlapping regions, set the point cloud with fewer points to be the source and the one with more points to be the target. The comparison algorithms are applied to the two initial point clouds, and force the source gradually close to the target. Record the operating time and the mean distance of the corresponding points of each algorithm when the program exits. The registration result of algorithm [15] is shown in Figure 2(b), from which we can see that the sharp region is unsatisfying and thus this registration result cannot meet the actual requirement.

Select the bicubic Bezier curved surface to be the local approximation surface and set the parameters $k = 30$, $u = 0.1$, $v = 0.1$ and $I_{\max} = 10$. Program according to the algorithm described in this article using c++ language and the existing PCL and CGAL libraries. Apply this program to the data above, and the registration result is shown in Figure 2(c) at the same magnification, which clearly demonstrates that the algorithm proposed here performs better. All the result data is shown in Table 1. This algorithm performs better in term of accuracy in contrast to the comparison algorithms.

5. **Conclusion.** Point clouds acquired by contact or non-contact 3D measurement devices are under different coordinate systems; therefore, the original ICP registration algorithm will cause registration errors. The proposed algorithm constructs approximation surfaces to represent the real local surface of the real part, and attempt to get more reliable points through the surfaces as the corresponding points. The points in local area of

the target are firstly parameterized, based on which a Bezier surface, the approximation surface, can be determined. Then the optimal correspondence is obtained with the benefit of this surface, and thus the more precise transformation matrix is obtained. Even though this algorithm is a little more time-consuming, it is fully functional when accuracy is the top priority.

The principle of this algorithm does not conflict with the existing variants of the ICP algorithm, so it could be combined with the existing algorithms to further improve the performance. Following study of this algorithm could focus on two aspects: 1) improve the precision of building local approximation surfaces, and 2) optimize the operation efficiency of this algorithm so that the algorithm can be combined with the existing reverse engineering software for more widespread use.

Acknowledgment. This work is partially supported by the National Natural Science Foundation of China (NSFC) (Grant No. 51575326) and the Natural Science Foundation of Shandong Province, China (NSFS) (Grant No. ZR2015EM031). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] J. Jones, P. McNutt and R. Tosi, *Remanufacture of Turbine Blades by Laser Cladding, Machining and in-Process Scanning in a Single Machine*, University of Texas, 2012.
[2] R. Subramoniam, D. Huisingh, R. B. Chinnam and S. Subramoniam, Remanufacturing decision-making framework (RDMF): Research validation using the analytical hierarchical process, *Journal of Cleaner Production*, vol.40, no.12, pp.212-220, 2013.
[3] E. Bagci, Reverse engineering applications for recovery of broken or worn parts and re-manufacturing: Three case studies, *Advances in Engineering Software*, vol.40, no.6, pp.407-418, 2009.
[4] J. Li, F. Yao and Y. Liu, Reconstruction of broken blade geometry model based on reverse engineering, *International Conference on Intelligent Networks and Intelligent Systems*, pp.680-682, 2010.
[5] P. J. Besl and N. D. McKay, Method for registration of 3-D shapes, *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol.14, no.2, pp.586-606, 1992.
[6] M. Greenspan and M. Yurick, Approximate k-d tree search for efficient ICP, *International Conference on 3-D Digital Imaging and Modeling*, pp.442-448, 2003.
[7] A. Nchter, K. Lingemann and J. Hertzberg, Cached k-d tree search for ICP algorithms, *International Conference on 3-D Digital Imaging and Modeling*, pp.419-426, 2007.
[8] X. Zhang, J. Xi and J. Yan, Research on digital measurement technology based on point cloud data of complex surfaces, *Computer Integrated Manufacturing Systems*, vol.11, no.5, pp.727-731, 2005.
[9] T. M. Iversen, A. G. Buch and N. Krger, Shape dependency of ICP pose uncertainties in the context of pose estimation systems, *Computer Vision Systems*, pp.303-315, 2015.
[10] J. Xie, Y. F. Hsu, R. Feris and M. T. Sun, Fine registration of 3D point clouds with ICP using an RGB-D camera, *IEEE International Symposium on Circuits and Systems*, pp.2904-2907, 2013.
[11] Y. Ren and F. Zhou, A 3D point cloud registration algorithm based on feature points, *International Conference on Information Sciences, Machinery, Materials and Energy*, 2015.
[12] X. Qin, J. Wang and H. Zheng, Point clouds registration of 3D moment invariant feature estimation, *Chinese Journal of Mechanical Engineering*, vol.49, no.1, pp.129-134, 2013.
[13] Y. Liu, L. D. Dominicis, B. Wei and L. Chen, Regularization based iterative point match weighting for accurate rigid transformation estimation, *IEEE Trans. Visualization and Computer Graphics*, vol.21, no.9, pp.1058-1071, 2015.
[14] S. Bouaziz, A. Tagliasacchi and M. Pauly, Sparse iterative closest point, *Computer Graphics Forum*, vol.32, no.5, pp.113-123, 2013.
[15] C. Li, W. Xiao and X. Gu, Precise registration method of damage parts based on improved ICP algorithm, *Computer Integrated Manufacturing Systems*, vol.1, 2015.
[16] F. P. Preparata and M. I. Shamos, Computational geometry: An introduction, *Texts & Monographs in Computer Science*, vol.47, no.176, 1985.
[17] G. T. Toussaint, Solving geometric problems with the rotating calipers, *Proc IEEE Melecon*, vol.83, pp.A10, 1983.