International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 5 Issue 11 ǁ November. 2017 ǁ PP. 37-42

^{3}D Fragment Matching Based on Contours in Reverse Engineering

Da Li [1],Xu Zhang [2]
[1,2]*(College of Mechanical Engineering, Shanghai University of Engineering Science,Shanghai,China)*
Corresponding Author: Da Li

ABSTRACT: With the development and application of digital scanning equipment, point cloud processing, data modeling and other reverse engineering technology, the restoration of cultural relics has been more convenient. Therefore, the study of how to use computer to assist the restoration, has important historical and practical significance.

In this paper, the three-dimensional fragments of the general shape as the object, and based on the boundary point cloud matching, to achieve debris more accurate splicing recovery. The thesis will study the following aspects:

1) In the case of fragments matching, we select the reasonable boundary feature and feature set, and combine the methods in the literature to measure and determine the similarity of two feature sets. To speed up the matching speed, the rational design of the search method, the greatest extent to find the matching part of the feature group.
2) Fragment matching stage, combined with the existing splicing algorithm, to study how to optimize splicing and error detection.

Keywords : Cultural relic restoration; contour;fragment matching;LCS;dynamic programming

---

Date of Submission:15-11-2017 Date of acceptance: 30-11-2017

I. INTRODUCTION

In the field of cultural relic's protection and restoration, matchingdebris fragmentshas always been a complex and arduous task. In order to more efficiently and accurately guide the repair work of the repair staff, the researchers carried out a long study on the problem of debris splicing. Among them, the use of debris boundaries of the geometric features is the majority of researchers to solve the problem of matching entry point.

Based on the debris matching, we mainly focus on the selection and matching algorithm of boundary feature. For boundary features, manyresearchers use the global and local features [1]. The boundary features involved in this paper are dominated by local features such as curvature, torsion and related invariants. Early physical fragmentation matching was developed based on the matching of 2D curves. Weixin Kong et al. [2] define the extension and bending costs of the curve, and calculate the optimal match by establishing the DP (dynamic programming) table of the two curve character descriptors. Thomas B. Sebastian et al. [3] treat the matching path of the two curves as a curve with an arc length, andimprove the arc length variable and limit the path template in the DP table so that a small number of local points can be matched. The Longin Jan Latecki et al. [4] improved DTW (dynamic time planning method) and LCS (the longest common substring), limiting the matching path and proposing MVM.

To match the solid fragmentsUsing debris boundary information. Many researchershave a very full study. Weixin Kong et al. [2] extended the plane curve matching to the spatial curve, using the interpolated third order ENO to calculate the curvature and torsion, and enhanced the accuracy of the eigenvalue. Andrew R. Willis [5] transforms the matching objective function into probabilistic form, based on the theory of Bayesian theory. Geoffrey Oxholm [6], by adding the image field descriptor on the basis of the curvature and torsion, and generating the multi-scale description graph of the boundary based on the multi-scale technique, so that the sequence matching is transformed into image matching. In order to identify and match the motion trajectories, Shao Zhanpeng[7] propose two integral invariants which are based on the integral kernel function. These two invariants are morerobust in matching process.

However, the above literature has several major problems:

1 The traditional curve matching method based on these two descriptors is very sensitive to the noise of the data because the curvature and torsion of the boundary discrete point cloud are related to the higher order derivative. The choice of descriptors is critical.

2 Based on the LCS development of the matching algorithm
Therefore, according to the boundary of the fragment, we add the integral invariant as the characteristic description of the boundary and improve the matching method to match the fragmentsbetter.

The ordered sequence of the boundary points

We use relate model of PCL (point cloud library) to extract the boundary points. However, theboundary points scatter disorderly, so the needs for sorting processing in order to match the debris have practical significance. The order of disordered points is arranged in the form of polygons. In this paper, we use the method mentioned in [8] to find the polygonal line. As shown in Figure 2, the disordered points can be arranged in an orderly polygon.Specific stepsare as follows:

1) Take some point in the boundary feature point as the starting point P_s of the polygon, take its nearest point as the ending point P_e of the polygon, and then multiply the edge along two sides.

2) Search the direction to the P_s.

Take P_s and find the nearest point P in the remaining points. d_s isthe distance calculated from the point P to P_s.And d_s is the distance calculated from the point P to P_s.If $d_s < d_e$,the point P becomes thestarting point P_s.

Otherwise,search another direction to the P_e:

3) Take P_e and find the nearest point P in the remaining points.Calculate d_s and d_e using the same method in 2).If $d_e < d_s$,the point P becomes the starting point P_e. Otherwise, search the new direction to the P_s demonstrated in 2).

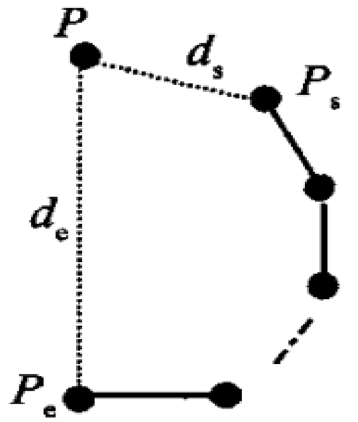

Fig 2 Sort the boundary points

1. Boundary matching based on multi-scale strategy

The matching for ordered boundary sequences is essentially a matching of finite set of points. The process can be summarized as follows: for a given two boundary sequences $C_1 = \{c_0^{(1)}, c_1^{(1)}, ..., c_n^{(1)}\}$ and $C_2 = \{c_0^{(2)}, c_1^{(2)}, ..., c_m^{(2)}\}$, we need to find and record the matching points or sections. However, due to the discrepancy of the fragmentation boundary point itself, the boundary sequence is regarded as the three-dimensional curve sampling point in Euclidean space. According to [9]:

Given a differentiable function $k(s) > 0$ and $\tau(s)$, $s \in I$. Then there is a regular parameter curve: C: $I \to \mathbb{R}^3$, s as the arc length of C, $k(s)$ as the curvature, $\tau(s)$ as the torsion. Moreover, any other curve $\overline{C}$ satisfying the same condition, is rigidlytransformed by C. That is, $\overline{C} = \rho \circ C + c$,an orthorhombic linear mapping ρ with a positive determinant and a vector c in $\mathbb{R}^3$.

Then the curvature and torsion can be used as an eigenvalue descriptor to determine the matching similarity of the three-dimensional curve. For the discrete sequence, this paper uses high-order B-splinecurve [10]for interpolation, which has good mathematical and algorithmic properties. For the algorithm, we can solve the high-order derivative of each point.

$$k = \frac{|C' \times C''|}{|C''|^{3/2}} \tag{1}$$

$$\tau = \frac{(C', C'', C''')}{(C' \times C'')^2} \tag{2}$$

$$n = \frac{(C' \times C'') \times C'}{|(C' \times C'') \times C'|} \tag{3}$$

In this paper, the matching strategy based on sampling scale is adopted: the boundary sequence C_δ of different scales δ is estimated and the corresponding eigenvalue sequence $P_\delta = \{p_0, p_1, ..., p_n\}$ is obtained. The smaller value of δ means more number of sampling points, more local information that the eigenvalue sequence can keep, and more suitable for the fine matching. On the other hand, the larger value of δ indicates the less the number of sampling points, and we can facilitate the calculation of eigenvalues and search the corresponding point, which is more suitable for the rough matching.

3.1 Rough match

At the maximum sampling scale δ, $P_{\delta,i} = \{p_0, p_1, ..., p_j, ...p_n\}$, the eigenvalue sequence of the rough matching phase,is calculated, where k_j and τ_j are respectively the curvature and torsion values at the sample point $p_j = (k_j, \tau_j)$ are represented.

The sequence of eigenvalues is analyzed. For the zero and extreme curvature, by setting the threshold and filtering to become a corner. Record the corner, and find the location of the corner in the initial boundary sequence, the initial sequence segmentation.

3.2 Fine matching

3.2.1 Distance Integral invariants

Obtained at a larger scale δ, the amount of data $C_\delta = \{c_0, c_1, ..., c_m\}$ is significantly increased, and the time-consuming characteristics of the sequence feature are increased. In addition, the method of solving higher order derivatives by interpolating higher order B-spline curves has the limitation: the algorithm is influenced by noise and data fluctuation, and the accuracy and reliability of sequence eigenvalues (Fig.3).

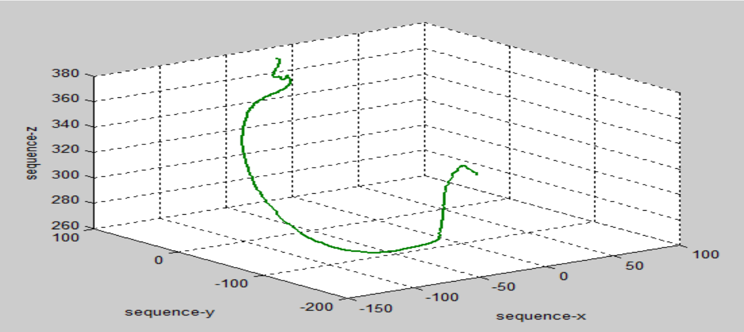

Fig.3a Boundary points

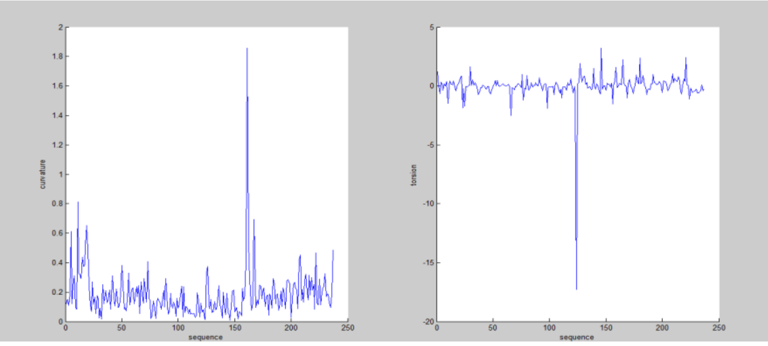

Fig.3b The fluctuation of curvature and torsion

In this paper, we introduce and improve the concept of integral invariants in [7]. For the spatial discrete sequence, using the integral invariant as the matching eigenvalue, it is possible to avoid the calculation of the higher order derivative and reduce the influence of the fluctuation of the data.

Definition 1: Let G be a set of rigid transformations on □^{3} Euclidean space. If the conditions are met:

$$I(C(s)) = I(g \bullet C(s)), \forall g \in G \tag{4}$$

Andthe function is a rigid invariant on the sequence.

Definition 2: Define the distance kernel function on the sequence and point, where there is a distance integral invariant

$$I(c_i) = \int_C d(c_i, C(s))ds \tag{5}$$

Among them, $d(c_i, C(s)) = \| c_i - C(s) \|$ is the space Euclidean distance.

Therefore, for the boundary sequence $C(s)$ and the feature sequence $P_{\delta,i} = \{p_0, p_1, ..., p_j, ...p_m\}$, its elements are rewritten as $p_j = \{k_j, \tau_j, I(c_j)\}$.

3.2.2 MVM-based matching algorithm

MVM matching algorithm is a flexible sequence matching algorithm. As an improved DTW algorithm, MVM allows to match sequences of different lengths and to be able to skip individual mismatched elements in both sequences. And unlike the LCS algorithm optimizing over the length of the longest common subsequence, MVM directly optimizes the sum of distances of corresponding elements (without any distance threshold).

For two finite feature sequences of real numbers $P = \{p_1, ..., p_i, ...p_m\}$ and $Q = \{q_1, ..., q_j, ...q_n\}$ with m < n, the goal is to find a subsequence Q' of Q oflength m such that P best matches Q. Thus, we want to find the best possible correspondence of sequence P to a subsequence Q' of Q.

Now, we describe the MVM to solve the matching of feature sequences. At first, establish the similarity matrix M. The elements $M_{i,j}$ mean the similarity and are calculated using the Euclidean distance, i.e. $M_{ij} = d(P(i), Q(j))$. Thus, there will be a lot of paths $M_{1,1}$ from to $M_{i,j}$, the price can be expressed as $D = d_1 + d_2 + ... + d_e$, the e represents the number of elements through the path as: $\max(m,n) \le e \le m + n - 1$.

Then, in order to find the best matching segments between P and Q, we need to find a path in the similarity matrixM that has the highest similarity and the best numerical result.To obtain the solution, we treat $M_{i,j}$ as a dynamic planning table with the following links: $M_{i,j}$ is directly linked to $M_{k,l}$, $k - i = 1$ and $j + 1 \le l \le j + n - m$.

When traversing the obtained directed graph, the meaning of both conditions is as follows: for any two consecutive points $M_{i,j}$ and $M_{k,l}$ in each path, $k - i = 1$means that we always go to the next row, while $j + 1 \le l \le j + n - m$means that we can skip some columns with certain elasticity (maximum n - m), but cannot go backwards. Fig 5shows the constructed DAG.

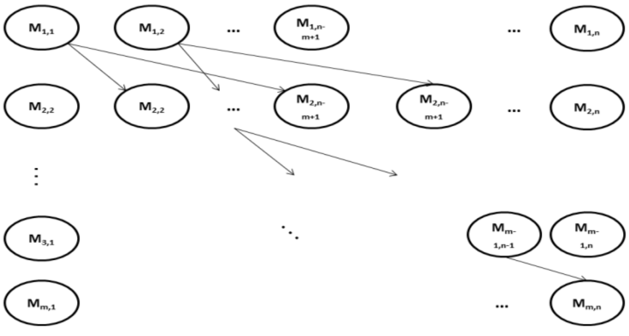

Fig.5 DAG

We want to have a least-value path with respect to the following cost function for each pair of nodes:

$$\mathrm{linkcost}(M_{i,j}, M_{l,k}) = \begin{cases} M_{l,k} = \|q_k - p_i\| & \text{if } k = i+1 \text{ and} \\ & j+1 \le l \le j+1+(n-m)-(j-i), \\ \infty & \text{otherwise.} \end{cases} \tag{6}$$

The conditions in (6) imply that we can obtain a DAG (directed acyclic graph) G whose nodes are theelements of (Mi,j)i,jand weights are defined by the function linkcost. Denoting a path leading to Mi,jasSP(i, j) and its cost as pathcost(i, j), we want to find a path with minimized pathcost(i, j) that satisfiestwo conditions:

- starting in the row, between columns 1 and n-m+1,i.e.,at M1,jfor j=1,…,n-m+1.
- Ending at some node in the last row$M_{m,j}$for j = m, . . . , n

Dynamic programming is a method efficient in solving problems exhibiting the properties of overlapping sub-problems and optimal substructure. In our case, the shortest path SP can be found based on the pathcostof every pair of reachable nodes in G. Taking pathcost(Mi,j) as the sub-problem, the optimal structure of our problem can be formally defined as:

$$\mathrm{pathcost}(M_{i,j}, M_{l,k}) = \begin{cases} M_{i,j} & \text{if } i = 1, 1 \le j \le n\,; \\ \min(\mathrm{pathcost}(i,j), & \text{if } 2 \le i \le m, i \le k \le i+n-m, \\ \mathrm{pathcost}(i-1,k) + & k+1 \le j \le k+1+(n-m)\,; \\ \mathrm{linkcost}(M_{i-1,k}, M_{i,j})) & \\ \infty & \text{otherwise.} \end{cases} \tag{7}$$

Experimental results

In this paper, we use HandyScan300 portable scanner to scan the debris and obtain point cloud data on the outer surface of the debris. In the preprocessing stage, this paper not only uses Imageware to patch the point cloud, de-noise and Gaussian sampling, but also implements the rapid extraction of the boundary based on the PCL open cloud library.

We chose a few pieces of artifacts as the experimental object. Fig 6 shows a few pieces of fragmentsscanned by scanner.

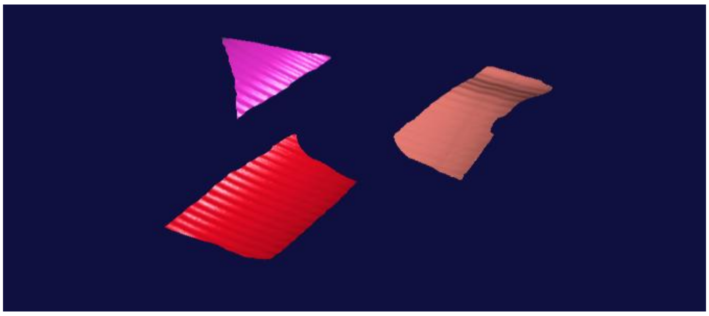

Fig.6 Scanned fragments

Fig7shows the contours of the pieces, which is searched and sorted by the method mentioned in chapter 2. Figure 8 shows the matching result by the multi-scale strategy.

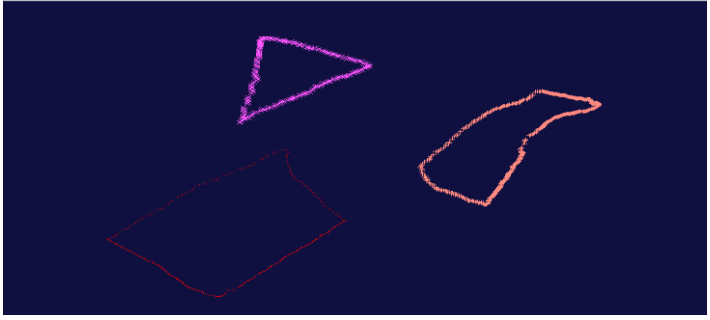

Fig.7 Searched and sorted boundary

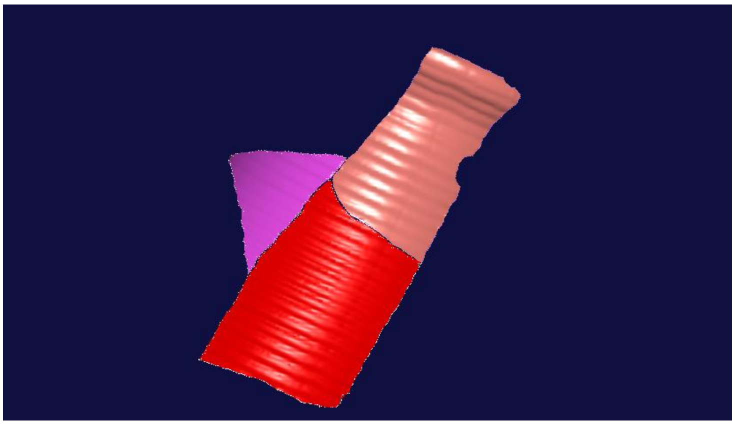

Fig.8 Matching result

II. CONCLUSION

In this article, we use boundaries to match fragments. First of all, the boundary is extracted and sorted out by using polygon search method. Then weuse the multi-scale matching strategy to fully match the fragments: in the coarse matching stage, we define the corner points and usethem to quickly filter the segments; in the fine matching phase, integral invariants are introduced to increase the accuracy and robustness of matching. Although the multi-scale matching strategyis more complicate than simply fine matching, this strategy reduces the unqualifiedpotential candidates. Using MVM-based algorithm, the matching achieves a good result.

REFERENCES

[1]. Zhou, Yu, J. T. Liu, and X. Bai. Research and Perspective on Shape Matching. ActaAutomaticaSinica 38.6(2012):889.

[2]. Kong, Weixin, and B. B. Kimia. On Solving 2D and 3D Puzzles Using Curve Matching. Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference onIEEE, 2001:II-583-II-590 vol.2.

[3]. Sebastian, Thomas B., et al. Constructing 2D Curve Atlases. Mathematical Methods in Biomedical Image Analysis, 2000. Proceedings. IEEE Workshop on IEEE, 2000:70-77.

[4]. Latecki, Longin Jan, et al. An elastic partial shape matching technique. Pattern Recognition 40.11(2007):3069-3080.

[5]. Willis, Andrew R., and D. B. Cooper. Bayesian Assembly of 3D Axially Symmetric Shapes from Fragments. Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on IEEE Xplore, 2004:I-82- I-89 Vol.1.

[6]. Oxholm, Geoffrey, and Ko Nishino. Aligning Surfaces without Aligning Surfaces. 2011 IEEE Workshop on Applications of Computer Vision, WACV 2011. N.p., 2011. 174–181.

[7]. Shao, Zhanpeng, and Y. Li.On Integral Invariants for Effective 3-D Motion Trajectory Matching and Recognition. IEEE Transactions on Cybernetics 46.2(2016):511.

[8]. QIAN,Jin-fengBoundary Feature Detection in Point Cloud Data Compression.Journal of Image and Graphics.2005 , 10 (2) :164-169

[9]. DoCarmo, Manfredo P. Differential geometry of curves and surfaces.(New Jersey:Prentice-Hall, 1976). 13-17

[10]. Piegl, L., &Tiller, W. (1997). The NURBS, Book. (BeiJing:Tsinghua University Press.2010). 35-90

Da Li . “3D Fragment Matching Based on Contours in Reverse Engineering.” International Journal of Research in Engineering and Science (IJRES), vol. 05, no. 11, 2017, pp. 37–42.