

Automatic Registration Using Virtual Polar Ball

H. T. Yau¹, L. S. Tsou² and H. M. Tseng³

¹National Chung Cheng University, imehty@ccu.edu.tw

²National Chung Cheng University, ltsou@cad.me.ccu.edu.tw

³National Chung Cheng University, hmtseng@cad.me.ccu.edu.tw

ABSTRACT

In reverse engineering applications, multiple scanned datasets need to be registered and integrated. However, automatic registration without user intervention is still an open research problem. In this paper, we propose a novel approach, the polar ball algorithm, which can find the corresponding features for automatic registration. Based on the Voronoi diagram and Delaunay triangulation, our approach is unique and invariant to viewpoint, which is critical in finding corresponding features. We utilize this property to obtain automatic registration without user interaction. The result of our registration algorithm is more robust to noise and stable than traditional characteristic point approach. Because the feature polar ball, or “feature ball” for short, can represent a curved surface region, it can reduce and smooth out the noise effect and represent a feature more reliably.

Keywords: Surface registration, surface matching, automatic 3D modeling.

1. INTRODUCTION

Optical three-dimensional measurement technology has been used in a wide variety of applications in the past few years. In the model building procedure, proper utilization of reverse engineering techniques can speed up the design process and achieve better efficiency than the use of normal design procedures. To reverse engineer an existing physical part into a geometric model, optical three-dimensional measurement equipment can be used to acquire 3D data points. Due to model complexity and measurement limits, it is necessary to acquire 3D range images from multiple viewpoints to cover the entire geometry of the physical part. Since these datasets are originally represented in local measurement coordinates, the registration and integration of multiple scanned data become important steps for the construction of complex 3D models. Fig. 1 shows an overview of the complex 3D model construction process.



Fig. 1: Overview of the complex 3D model construction process.

Traditionally, registration relies on accurate mechanical positioning devices or requires manual processing. A common manual matching method is generally divided into two steps. The first step is to quickly find the initial transformation between two data sets. To apply this initial transformation, two surfaces are approximately aligned. This step is usually known as crude registration or initial alignment. Then, second step is to refine the initial alignment using the Iterative Closet Point (ICP) algorithm [1-2]. Because the ICP algorithm is a locally convergent method that requires parameter initialization close to the aligned position, the initial alignment becomes very important. In commercial software, for the purpose of obtaining robust initial matching, crude registration is achieved manually. Commercial software manually or automatically extracts characteristic points from range images and then lets the user manually click corresponding characteristic points on the data. In order to avoid user interaction, using 3D range images to automatically execute the registration is attractive because automation is the trend in modern 3D model construction systems.

Nowadays, automatic registration is still an ongoing research subject. In the past, a popular method used for automatic registration was the feature-based approach. This approach extracts invariant features and usually converts the

problem of 3D matching into 1D or 2D matching. Huber and Hebert [7] used a combination of discrete and continuous optimization methods to construct a globally consistent model from a set of pair-wise registration results. Maier [8] exploited characteristic points of the Gaussian image that are invariant against facial form changes caused by surgical operations. Zhang and Hebert [9-10] proposed using the harmonic shape image (HSI). HSI maps a 3D surface patch with disc topology to a 2D domain and encodes the shape information of the surface patch into the 2D image. There are also some approaches that use a spin-image in surface matching [11-13]. The concept of a spin-image developed from ideas in basic geometric hashing and structural indexing. Sun et al. [14] proposed an interesting method, which is called point fingerprint. It is a set of 2D contours that are the projections of geodesic circles onto the tangent plane. A more detailed survey of surface matching can be found in [6].

In this paper, we propose a polar ball [3-5] algorithm for automatic crude registration. Traditionally, when we use the coordinate measuring machines (CMM) to measure objects such as mechanical parts or freeform surfaces, we usually place certain standard parts on the object. The standard parts often utilize the standard ball and can be regarded as the function which sets the measurement coordinate system. Thus, the basic idea of this algorithm is to search for the polar ball from the geometric model. A polar ball is the center of the Voronoi ball at a pole. The well-known Voronoi diagram and Delaunay triangulation of a set of 3D points are both unique and invariant to viewpoint. We use this property to automatically specify corresponding feature points in pairs of data sets. Then the initial transformation will be obtained simultaneously. The ICP algorithm can be applied to achieve fine registration. An overview of automatic registration is shown in Fig. 2. Our objective is to avoid user interaction and to achieve automatic registration.

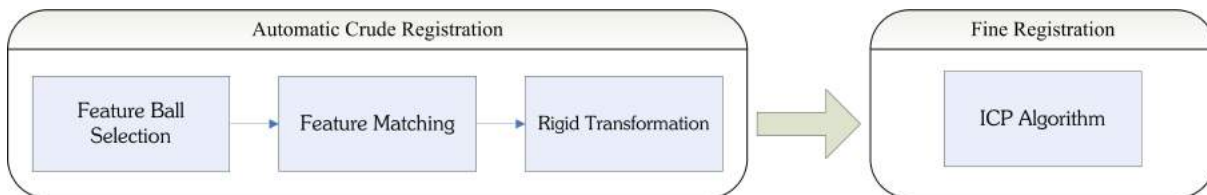


Fig. 2: Overview of the automatic registration process.

The content of this paper is organized as follows. The mathematical background for the polar ball is introduced in Section 2. Section 3 outlines the procedure of our polar ball algorithm. Section 4 shows the experimental results. Conclusions are given in Section 5

2. PRELIMINARIES

Before the proposed algorithm is presented, a brief review of the basic mathematical background used in our algorithm is given as follows. In addition, for a clear and complete presentation, the definition of poles from Amenta's works [3-5] is also briefly described.

2.1 Voronoi Diagram and Delaunay Triangulation

Let O be a real object and ∂O be its boundary surface. A finite set of distinct sample points acquired from ∂O is denoted as S . Let $S = \{s_1, s_2, \dots, s_n\}$ in the Euclidean space E . A three-dimensional Voronoi diagram $V(S)$ of the point set S is a division of the space into convex regions, called Voronoi cells $Vc(s_i)$, in which every sample point s_i is associated with a $Vc(s_i)$. Each Voronoi cell $Vc(s_i)$ consists of a set of points $x_i \in E$ that are closer to s_i than to any other points of S and its vertices are called the *Voronoi vertices*. A three-dimensional Delaunay triangulation, which is dual to the three-dimensional Voronoi diagram, is a tetrahedralization that divides the convex hull $conv(S)$ of the points set S into a collection of tetrahedra. Each tetrahedron has a circumsphere called the *Voronoi ball* that centers at a Voronoi vertex with the vertices of this tetrahedron located on its sphere boundary. The Voronoi ball contains no point s_i and no other Voronoi balls in its interior. One of most important properties is that a three-dimensional Delaunay triangulation is unique under the assumption that no 5 points of S can be found on a common sphere.

2.2 Medial Axis

The medial axis $M(O)$ of an object O is the locus of the centers $c_i \in E$ of the maximum empty balls, called *medial balls*, which contains no points s_i and no other empty ball. For an enclosed object in R^3 , the medial axis is obviously divided into the internal and the external medial axis. The medial axis transformation (MAT) is the set of medial balls. That is, the MAT does not only consist of the centers c_i of the medial ball but also their radii r .

2.3 Poles

Amenta et al. [3-5] is the first one that proposes the definition of poles, which are a subset of the Voronoi vertices of the point set S and represent the approximated medial axis when S is a sufficiently dense sample. Each sample point s_i is associated with two poles, denoted by p_i^+ and p_i^- respectively. These two poles are the farthest Voronoi vertices of $Vc(s_i)$ from s_i . If pole p_i^+ is one farthest Voronoi vertex of $Vc(s_i)$ from s_i , the other pole p_i^- must have a negative dot-product of $\overline{S_i P_i^+}$ and $\overline{S_i P_i^-}$, which is illustrated in Fig. 3.

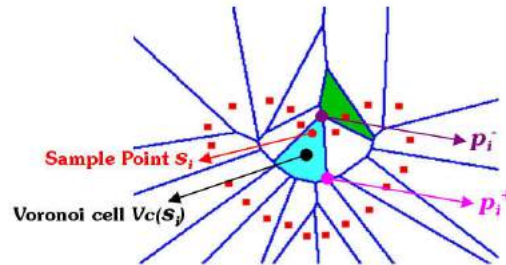


Fig. 3: Each Sample point is associated with one Voronoi cell and two poles.

The Voronoi ball centered at a pole is called the *polar ball*. When S is a sufficiently dense sample, the Voronoi cell $Vc(s_i)$ appears naturally long and slim, and is approximately perpendicular to the boundary ∂O of the object O . Therefore, two poles p_i^+ and p_i^- are roughly in the opposite direction and the *pole direction* (from sample point s_i to its poles) approximates the surface normal at s_i under that dense sampling condition. In Fig. 4., the finite union of the polar balls forms a good discrete approximation to the *MAT*, which is a representation of the object as an infinite union of balls.

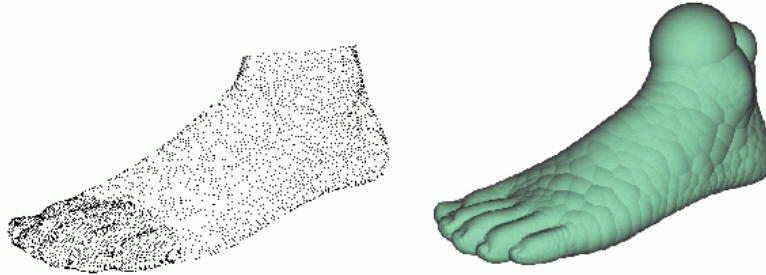


Fig. 4: Union of polar balls [5].

3. ALGORITHM DESCRIPTION

As shown in Fig. 2, the automatic crude registration algorithm is composed of three procedures: feature ball selection, feature matching, and transformation computing. Outlined below are the complete steps of the proposed polar ball algorithm.

Step 1. Select the feature balls.

Step 1.1. Compute all polar balls from the scanned datasets.

Step 1.2. Mark polar balls to obtain candidate balls.

Step 1.3. Merge candidate balls to obtain feature balls.

Step 2. Match corresponding feature balls between two objects.

Step 2.1. Calculate similarity criterion.

Step 2.2. Eliminate wrong corresponding pairs.

Step 3. Compute rigid transformations (R_0, T_0) : R_0 is initial rotation matrix and T_0 is the initial translation vector.

3.1 Feature Ball Selection

The feature balls are the polar balls that can represent the characteristics or features of a surface. Traditionally, the characteristic points are extracted in the scanned datasets, but the noises and different sampling resolutions deeply affect the registration results. The use of feature balls are better than the use of characteristic points because a feature ball can represent a curved surface region rather than just a single characteristic point. Thus, the polar ball algorithm can smooth out the noise effect and is independent of sampling resolutions.

The power crust algorithm [5] can be applied to calculate all polar balls from a geometric model, but it is impossible to select all polar balls to be feature balls. If the geometric model has the features that approximate a sphere, the polar balls will congregate very near to each other and have similar radii. Fig. 5. illustrates the selection idea in 2D plane. There are two features that approximate a circle in the model. The left circle has high sampling resolution and the right circle has low sampling resolution. The color circles will crowd together even though the datasets have different sampling resolutions. Thus these circles can be merged into a feature ball individually.

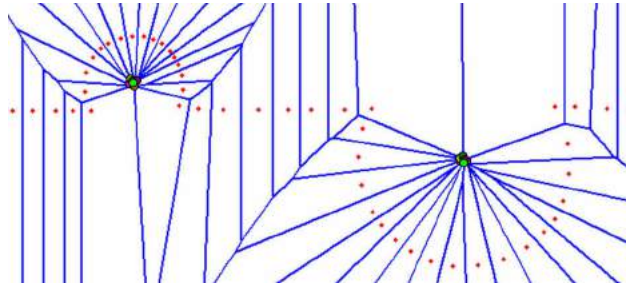


Fig. 5: An illustration of the selection idea in 2D plane.

Fig. 6 demonstrates the special model with three half balls and a radius histogram of all the polar balls. In the histogram, there are some peak values. This means that this radius range may have feature balls. In this case, only three feature balls are extracted to represent the characteristics of the special model. Thus the important step of our registration approach is to search for all feature balls.

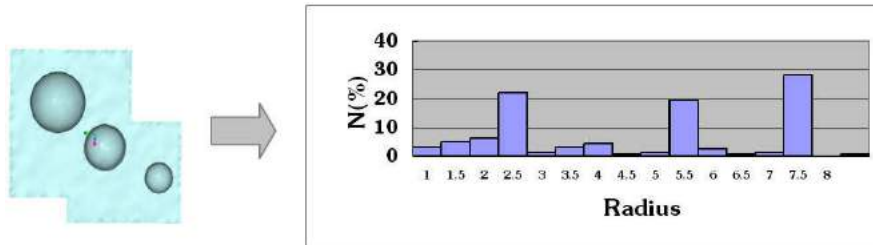


Fig. 6: The geometric model with three half balls and a radius histogram of all the polar balls.

3.1.1 Mark Polar Balls to Obtain Candidate Balls

In Fig. 6, the feature balls can be selected easily because the features of the special model contains standard ball features. Furthermore, the polar balls that locate at the feature areas will congregate very near to each other. But if the features are of an approximate ball or ellipsoid, the polar balls will not congregate very near to each other, as shown in Fig. 7. In order to mark all gathering polar balls to be candidate balls, the first step is to create a KD-tree from all polar balls. Afterward, each polar ball must search for the neighboring polar balls with a specified radius from the KD-tree. If the searched result shows that there are other polar balls located within this scope and the quantity of the neighboring polar balls is larger than the specified number (in our experience, this number is usually set to 5), the current polar ball will be marked as a candidate ball.

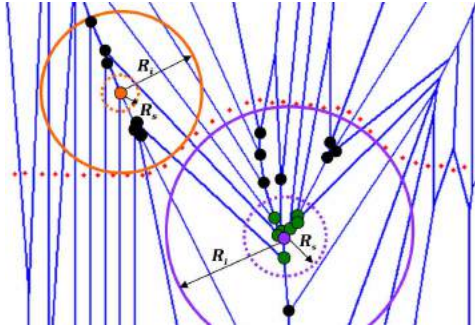


Fig. 7: An illustration of the marking idea in 2D plane.

In Fig. 7, if the radius R_i of the current polar ball is a searched radius, the orange and purple polar ball will be marked as candidate balls. In fact, only the purple polar ball is a candidate ball. Thus to achieve good and robust marking results, the searched radius should change dynamically. Our basic idea is very simple. If the current polar ball has large radius, its neighboring polar balls will be distributed more widely so the searched radius must be larger. On the contrary, if the current polar ball has little radius, the searched radius must be smaller. Therefore, the searched radius R_s can be formularized as follows:

$$R_s = r \left[1 + f \left(\frac{R_i}{R_{max}} \right)^2 \right] \quad (3.1)$$

where

- r : is the minimum searched radius.
- R_i : is the radius of the current polar ball.
- R_{max} : is the maximum radius of all the polar balls.
- f : is the feature coefficient.

In Eqn. (3.1), if the feature coefficient increases, the searched radius will increase and then more and more polar balls may be marked. Although many candidate balls can be obtained, the next step can also merge these balls to be feature balls. A drawback is to reduce the efficiency of our algorithm because the marking step and merging step will need more time to complete the work. In order to improve the efficiency, we want to obtain appropriate candidate balls. Since the geometric model has a lot of curved features, it means that many polar balls will gather together. Thus the feature coefficient can be decreased. However, if the geometric model is flat, it means that many polar balls will distribute widely. Thus the feature coefficient must be increased.

3.1.2 Merge Candidate Balls to Obtain Feature Balls

The candidate ball is only the candidate feature polar ball. If there exists a feature ball, the polar balls will congregate near to each other and have similar radii; moreover, there needs to be only one feature ball that expresses these candidate balls, as shown in Fig. 8.

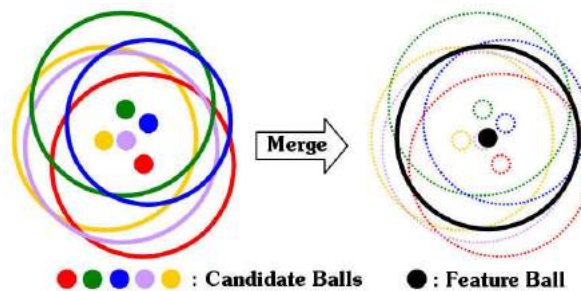


Fig. 8: An illustration of the merging idea in 2D plane.

In our merging algorithm, one ball is chosen from among the candidate balls as an initial ball to start the merging process. This step is important because a good initial ball can obtain good merge results. In Fig. 8., the purple candidate ball is the best initial ball because it is located in the approximate center of the candidate balls in this feature area. Furthermore, it can merge with more candidate balls than other balls can. In order to get the initial ball, we build a KD-tree of all the candidate balls. Each of the candidate balls uses the same searching algorithm as mentioned above to find its neighboring candidate balls and to sort the search result by counting the number of neighboring balls. The candidate ball that has the maximum number of neighboring balls is the best initial ball. When the initial ball is determined, its neighboring balls will all be checked for the merging criterion to ascertain whether they can merge or not. The idea of using the merging criterion comes from lemma 2 of the power crust algorithm [5]. It says that any inner polar ball and any outer polar ball can only intersect shallowly. We observe the polar balls to find out when the polar balls will intersect deeply when they are in the feature area. Fig. 9. shows the merging criterion. The intersecting angle θ can be calculated using the law of cosines, given as follows:

$$\theta = \pi - \alpha = \cos^{-1} \left[-\frac{R_1^2 + R_2^2 - dist^2(C_1, C_2)}{2 \times R_1 \times R_2} \right] \tag{3.2}$$

where

- R_1 : is the radius of the initial ball
- R_2 : is the radius of the neighboring ball
- $dist(C_1, C_2)$: is the distance of these two balls

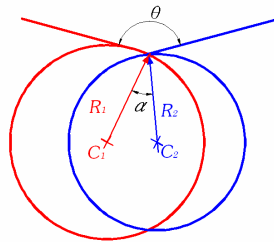


Fig. 9: An illustration of the merging criterion.

Since the intersecting angle is close to π , the polar balls will intersect more deeply. Thus these two candidate balls can be merged together. When all of the neighboring balls of the initial ball have been checked and the initial ball merges N of its neighboring balls simultaneously, the feature ball can be obtained and its center coordinate and radius will be recomputed using Eqn. (3.3).

$$\left\{ \begin{array}{l} C_x = \frac{\sum_{i=1}^N C_{xi}}{N}, C_y = \frac{\sum_{i=1}^N C_{yi}}{N}, C_z = \frac{\sum_{i=1}^N C_{zi}}{N} \\ R = \frac{\sum_{i=1}^N R_i}{N} \end{array} \right. \tag{3.3}$$

Thus, the flow chart of the merging algorithm is shown in Fig. 10. and Fig. 11. shows the feature balls of a kitty model

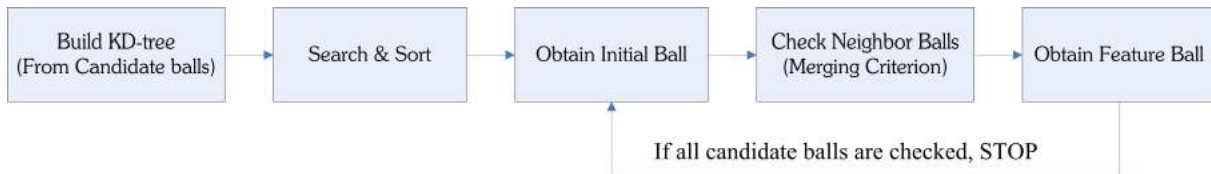


Fig. 10: Flow chart for the merging algorithm.

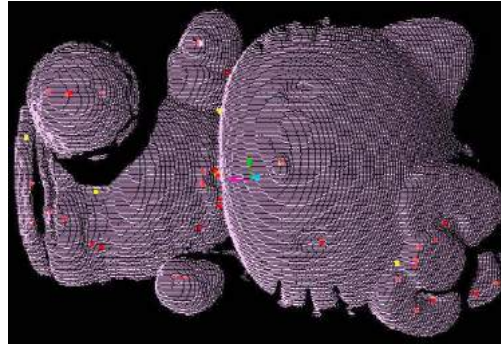


Fig. 11: The feature balls of a kitty model.

3.2 Feature Matching

Suppose the feature balls f_1 and f_2 are already calculated from two surfaces S_1 and S_2 respectively. If f_1 and f_2 is a pair of corresponding balls, their neighboring polar balls will be similar because the Voronoi diagram is both unique and invariant to viewpoint. The matching algorithm is to compare with local histograms of all feature balls in the corresponding model and the reference model. The local histogram can express the local property of this feature; therefore, the local characteristic value can be calculated using the histogram. Fig. 12. shows that each of the feature balls will search for all the polar balls to find the polar balls which are inside the respective feature ball and will then utilize these polar balls to build the local histograms. Suppose the histogram has m regions and each region has N_i polar balls. The corresponding radii of these polar balls are R_{ij} . Thus, the local characteristic value can be expressed as follows:

$$c = \frac{\sum_{i=1}^m \sum_{j=1}^{N_i} N_i R_{ij}}{\sum_{i=1}^m N_i} \tag{3.4}$$

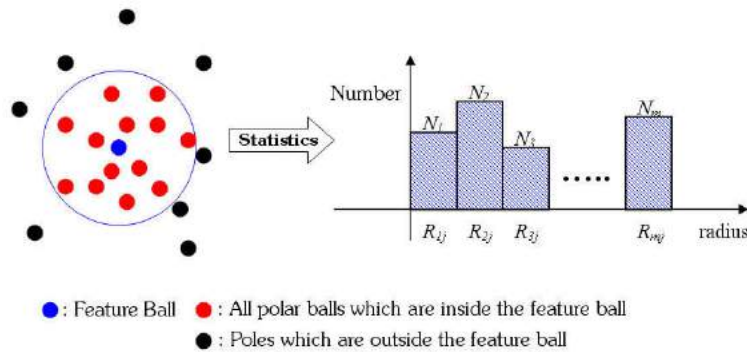


Fig. 12: An illustration of the local histogram.

In order to match the feature balls, the following formula is used to compute SIM_{ij} which is the similarity measure between the i th feature ball on S_1 and the j th feature ball on S_2 . If SIM_{ij} is close to 100%, the two feature balls can be matched with a pair of corresponding balls. Fig. 13. shows the corresponding balls of the Stanford Bunny model

$$SIM_{ij} = \left(1 - \frac{|c_i - c_j|}{c_i + c_j}\right) \times 100\% \tag{3.5}$$

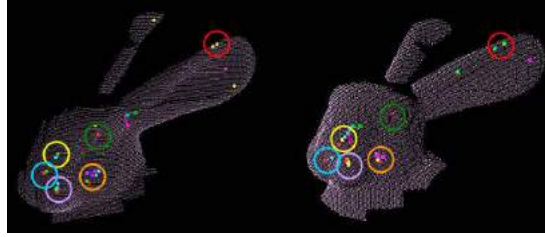


Fig. 13: The corresponding balls of the bunny model.

3.2.1 Eliminate Wrong Corresponding Pairs

Since some geometric models may have symmetrical features, the similarity measure will fail and wrong corresponding pairs will result. Thus, we proposed the elimination process to let the registration algorithm become more robust. Let us assume that $\{p_i, q_i\}_1^n$ are n pairs of feature balls that correspond to each other. A transformation matrix T exists, and its error function and average error are as follows:

$$F_{err} = \sum_{i=1}^n (p_i - Tq_i)^2 \tag{3.6}$$

$$E_{avg} = \frac{\sum_{i=1}^n (p_i - Tq_i)^2}{n} \tag{3.7}$$

In order to decrease the error function, wrong pairs must be eliminated. The process of eliminating wrong corresponding pairs is shown in Fig. 14.

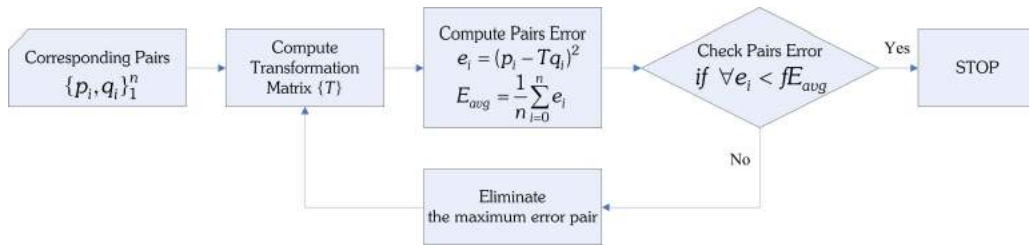


Fig. 14: The process of eliminating wrong corresponding pairs.

3.3 Transformations Computing

To obtain the initial rotation matrix and translation vector [15], there exists a transformation relationship in each pair, as follows:

$$R_0 p_i + T_0 \cong q_i \tag{3.7}$$

where R_0 is the initial rotation matrix and T_0 is the initial translation vector. The problem of finding a suitable R_0 and T_0 for all the pairs, $i = 1, \dots, n$, can be formulated as a least squares problem, and has been solved using the singular value decomposition method. Here, for completeness, the method is as follows:

Step 1. Calculate p_c and q_c , which are the centroids of $\{p_i\}$ and $\{q_i\}$, respectively, where

$$p_c = \frac{1}{n} \sum_{i=1}^n p_i, q_c = \frac{1}{n} \sum_{i=1}^n q_i \tag{3.8}$$

Step 2. Let

$$\{p'_i\}_{i=1}^n = \{p_i - p_c\}_{i=1}^n \tag{3.9}$$

$$\{q'_i\}_{i=1}^n = \{q_i - q_c\}_{i=1}^n \tag{3.10}$$

Here, we have the new sets of the two characteristic points which have the same original coordinates.

Step 3. Calculate the 3×3 matrix:

$$H = \sum_{i=1}^n p_i q_i^t \quad (3.11)$$

Step 4. Find the SVD of H:

$$H = U \Lambda V^t \quad (3.12)$$

Step 5. Calculate $R=VU^t$. If $\det(R)=1$, then $R_0 = R$.

Step 6. Calculate the translation vector T_0 :

$$T_0 = q_c - R_0 p_c \quad (3.13)$$

4. EXPERIMENTAL RESULTS

Our proposed polar ball algorithm was implemented in C++. We used the computational geometry algorithms library CGAL [16] to compute 3D Delauny triangulation. Fig. 15, Fig. 16 and Fig. 17 show the result of automatic crude registration using our polar ball algorithm. Fig. 18 shows the automatic registration process of a Buddha model. Crude registration and fine registration results are shown in Fig. 18(b) and Fig. 18(c), respectively.



Fig. 15: Automatic crude registration of a bunny model.



Fig. 16: Automatic crude registration of a kitty model.

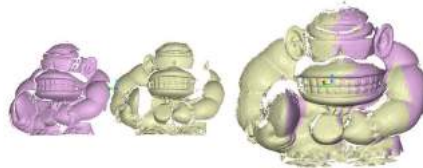


Fig. 17: Automatic crude registration of an orangutan model.

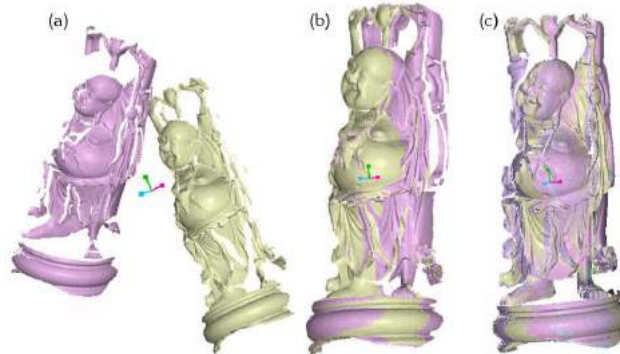


Fig. 18: Automatic registration of a Buddha model.

5. CONCLUSION

A novel surface registration approach based on the Voronoi diagram and Delaunay triangulation was proposed. The well-known Voronoi diagram and Delaunay triangulation of a set of 3D points are both unique and invariant to viewpoint. Our registration approach is able to register completely or partially overlapping shapes, and can automatically match the corresponding feature balls in pairs of datasets. The feature balls can represent the characteristics or features of a surface. The use of a feature ball is better than the use of characteristic points because a feature ball can represent a curved surface region rather than just a single characteristic point. Thus the polar ball algorithm is robust to noise and independent of sampling resolutions. Experimental results demonstrated that the method can provide a good initial alignment of real scanned datasets for further ICP refinement,

6. REFERENCES

- [1] Besl, P. J.; McKay, N. D.: A Method for Registration of 3D Shapes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), 1992, 239-256.
- [2] Rusinkiewicz, S.; Levoy, M.: Efficient Variants of the ICP Algorithm, *3D Digital Imaging and Modeling*, 2001, 145-152.
- [3] Amenta, N.; Bern, M.; Kamvysselis, M.: A New Voronoi-based Surface Reconstruction Algorithm, *Proc. SIGGRAPH'98*, 1998, 415-421.
- [4] Amenta, N.; Bern, M.: Surface Reconstruction by Voronoi Filtering, *Discrete Comput. Geom.*, 22, 1999, 481-504.
- [5] Amenta, N.; Choi, S.; Kolluri, R. K.: The Power Crust, *ACM Symposium on Solid Modeling and Applications*, 2001, 249-260.
- [6] Campbell, R.; Flynn, P.: A Survey of Free-Form Object Representation and Recognition Techniques, *Computer Vision*, 2, 2001, 166-210.
- [7] Huber, D. F.; Hebert, M.: Fully Automatic Registration of Multiple 3D Data Sets, *IEEE Computer Society Workshop on Computer Vision Beyond the Visible Spectrum*, 2001.
- [8] Maier, T.; Benz, M.; Schon, N.; Nkenke, E.; Neukam, F. W.; Vogt, F.: Automatic Coarse Registration of 3D Surface Data in Oral and Maxillofacial Surgery, *Medical Robotics, Navigation and Visualization*, 2004, 51-58.
- [9] Zhang, D.; Hebert, M.: Harmonic Maps and their Applications in Surface Matching, *Computer Vision Pattern Recognition*, 2, 1999, 524-530.
- [10] Zhang, D.: Harmonic Shape Images: A 3D Freeform Surface Representation and Its Applications in Surface Matching, Ph. D. Thesis, Carnegie Mellon Univ., Pittsburgh, PA, 1999.
- [11] Johnson, A. E.; Hebert, M.: Surface Registration by Matching Oriented Points, *Recent Advances 3D Digital Imaging Modeling*, 1997, 12-15.
- [12] Yamany, S. M.; Farag, A. A.: Freeform Surface Registration Using Surface Signatures, *Computer Vision*, 1999, 1098-1104.
- [13] Johnson, A. E.; Hebert M.: Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes, *Pattern Analysis and Machine Intelligence*, 21, 1999, 433-449.
- [14] Sun, Y.; Paik, J.; Koschan, A.; Page, D. L.; Abidi, M. A.: Point Fingerprint: A New 3D Object Representation Scheme, *IEEE Trans. On Systems, Man, and Cybernetics-Part B: Cybernetics*, 33, 4, 2003, 712-717.
- [15] Yau, H. T.; Chen, C. Y.; Wilhelm, R. G.: Registration and Integration of Multiple Laser Scanned Data for Reverse Engineering of Complex 3D Models, *International Journal of Production Research*, 38, 2, 2000, 269-285.
- [16] CGAL, <http://www.cgal.org>, Open Source of Geometric Algorithms.