124 Y. Sun and M. A. Abidi, "Surface matching by 3D point's fingerprint," Proc. IEEE International Conference on Computer Vision, Vol. II, pp. 263-269, Vancouver, Canada, July 2001.

Surface Matching by 3D Point's Fingerprint

Y. Sun and M. A. Abidi
Department of Electrical and Computer Engineering
The University of Tennessee
Knoxville, TN 37996
yiyong@iristown.engr.utk.edu

Abstract

This paper proposes a new efficient surface representation method for the application of surface matching. We generate a feature carrier for the surface point, which is a set of 2D contours that are the projection of geodesic circles onto the tangent plane. The carrier is named point's fingerprint because its pattern is similar to human fingerprint and discriminating for each point. Each point's fingerprint carries the information of the normal variation along geodesic circles. Corresponding points on surfaces from different views are found by comparing the fingerprints of the points. This representation scheme includes more local geometry information than some previous works that only use one contour as the feature carrier. It is not histogram based so that it is able to carry more features to improve comparison accuracy. To speed up the matching, we use a novel candidate point selection method based on the shape irregularity of the projected local geodesic circle. The point's fingerprint is successfully used to register both synthetic and real $2\frac{1}{2}D$ data.

1. Introduction

Surface matching has two direct important applications in the area of computer vision. The first is 3D image registration [8, 16, 18], which is also known as pose estimation. When 3D images are taken at different viewpoints and data fusion is necessary, the rigid transformation between each view needs to be computed. From Horn's work [7], given three or more pairs of non-coplanar corresponding 3D points, the rigid transformation between the point pairs has a closed form solution. The pose estimation problem becomes the surface point matching problem. The other application is object recognition [4, 5, 12, 17]. The model library stores the surface features of each object and the corresponding scene object is found by comparing those features.

Discrete surface matching is difficult because two surfaces may have self occlusions and different sampling resolutions, and are only partial representation of the same object, which make the statistical based features such as moment difficult to apply. Because the local surface geometry characteristic around each surface point contains information that is not sensible to sampling resolutions, previous works in this area tried to encode this information and use it for point matching, especially with different surface representation schemes which changed the problem into 1D or 2D feature matching [4, 8, 12, 16, 18].

Previous works encoded the point's local surface geometry characteristics using either a single contour on the surface or a 2D image of a neighborhood near the encoded point. Stein and Medioni [12] used the Splash to represent the normals along one geodesic circle of a center point. The relationship between the surface normal of the center point and the surface normal at each point on the splash is established to produce a 3D curve. Straight line segments are fitted to the 3D curve. The contour is encoded using the curvature and torsion angles and serves as the feature of the center point. Chua and Jarvis [4] used a sphere to cut a surface patch around a center point. A plane is fitted to the contour of the surface patch and then translated to the center point along the normal direction of the center point. The signed distances between the points on the contour and the plane can be plotted as a function of the angle from 0 to 2π. This function is used as the Point Signature (PS) of the center point. Johnson and Hebert [8] proposed the Spin Image that is a 2D histogram. One parameter of the histogram is the distance to the tangent plane of the center point from all the surface points. The other parameter is the distance to the normal of the center point. The system finds the corresponding points by comparing the Spin Images. Yamany and Farag [16] proposed a modified version of the Spin Image, which is called the Surface Point Signature (SPS). One parameter of the histogram is the distance between the center point and every surface point. The other parameter is the angle between the normal of the center point and the seg-

0-7695-1143-0/01 $10.00 © 2001 IEEE

ment created by the center point and every surface point. The reason for using a 2D histogram is that it is hard to create a one-to-one map from the surface patch to a 2D image. Although 2D histogram loses some local geometry information, it has an advantage in that two histograms of corresponding points are independent of rotation. An application of one-to-one harmonic maps to surface registration was reported by Zhang and Hebert [18]. The Surface patch enclosed by the geodesic circle is mapped to a unit disk by the harmonic map. Curvature values approximated by the simplex angles are textured onto the harmonic image to generate a Harmonic Shape Image (HSI) for the point matching.

In this work, we developed a new surface representation scheme called 3D point's fingerprint and applied it to surface registration. It is a set of 2D contours that are the projection of geodesic circles onto the tangent plane. We named it point's fingerprint because it looks like human fingerprint and can be used as discriminating feature for the surface point. The advantage of this method is that it can carry more information than the schemes that use only one contour or 2D histogram and the computation is cheaper than the schemes using 2D image representation. Although the projection on the tangent plane is not a one-to-one map and makes the contours possibly intersected, each contour can be traced back to each geodesic circle so that the method has the same advantage as the one-to-one map.

The paper is organized as follows: In Section 2, we describe the generation of the geodesic circles, 3D point's local fingerprint, and the candidate point selection method. In Section 3, we introduce the global fingerprint combining the normal variation and the feature matching method. Results of 3D image Registration are given in Section 4 and the paper concluded in Section 5.

2. Fingerprint of the surface point

The view independent 2D feature of a surface point must be on the tangent plane of this point, as shown in Fig. 1(a). In previous works, Splash [12], Spin Image [8], and SPS [16] are defined on the tangent plane, while PS [4] and HSI [18] are not. Some works use the geodesic measure while others use the Euclidean measure. The problem of using the Euclidean measure is that the neighborhood of a surface point is sometimes ambiguous. In Fig. 1(b), only surface patch 1 is considered as the neighborhood of P if the geodesic measure is used, while surface patch 2 is a confusion if the Euclidean measure is used. In previous works, only Splash [12] and HSI [18] consider the geodesic measure. In this work, we projected geodesic circles onto the tangent plane to generate point's fingerprint. The geodesic circle is formed by the points that have the same geodesic distance to the center point. We describe how to compute geodesic distance on a triangulated surface and generate 3D point's fingerprint.

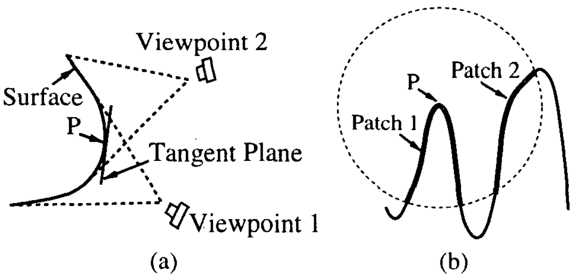

Figure 1. (a) View independent feature is on the tangential plane. (b) Geodesic vs Euclidean. Patch 1 is the neighborhood of P under geodesic measure; Patch 2 is an ambiguous neighborhood if Euclidean measure is used.

2.1. Geodesic circle generation

We need to compute the geodesic distance on the discrete surface, especially on a triangulated surface, which is the most popular representation of 3D objects. The Dijkstra's algorithm [1] is often used for finding the shortest path on a network with prescribed weights for each link between nodes. The problem of this method is that it is inconsistent with the underlying continuous problem. To compute the geodesic distance, we use a modified version of Kimmel's work [10]. Kimmel developed two methods to find geodesic path on surfaces, which are based on Sethian's level set methods and fast marching methods [11]. One method [9] is for the surfaces that is in the form of the height map on a rectangular grid and the other method [10] is for an arbitrary surface in the form of triangular mesh. The basic idea is to evolve a contour with unit speed from a starting point P on the surface and the fast marching method is used to solve the Eikonal equation that describes the evolution of the contour. When the contour passes a 3D point, the reaching time, i.e., the geodesic distance, is stored for that point. After the evolution stops, the geodesic distance between every point within the final contour and the point P is known. Since Kimmel's two methods can have the same computation complexity as the Dijkstra's algorithm ($O(N_P \log N_P)$, where N_P is the number of points within the final contour) if a heap data structure is used, but with more accuracy, it begins to draw much attention from different research areas. Most recently the method has been used to solve surface matching problems [14, 15]. (These two works tackle the problem using different methods from this paper).

The fast marching algorithm works as follows: (1) We tag the center point as *Alive* and all the other surface points as *Far*, and (2) we tag all the neighboring points of the center point as *Close*. The geodesic distance to the center point is denoted as T and the following loop is then executed.

1. Change the tag of the point in *Close* with the smallest T from *Close* to *Alive*.
2. Tag all neighboring points of this point as *Close*.
3. Recompute the T of these neighboring points, using only values of points that are *Alive*. T is renewed only if the recomputed result is smaller.
4. Go back to 1.

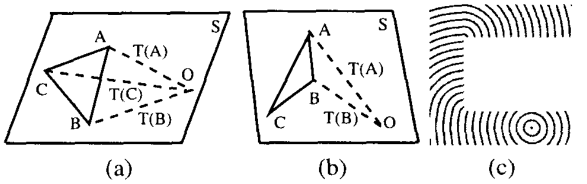

Figure 2. (a) Update the T value inside the triangle. (b) Update the T value outside the triangle. (c) Geodesic circles under occlusion.

The fast marching algorithm guarantees that the marching process is always from the point with the smallest geodesic distance to the point with the largest distance. Kimmel and Sethian [10] proposed a scheme of updating the T values in step 3 on an arbitrary triangular mesh. The scheme is extended from the scheme of marching on a regular triangulated planar domain. It is derived for the triangular mesh that contains only acute triangles. And the obtuse triangles need to be split at first.

Here we propose a simple new scheme to update the T values. Assume the geodesic distances at A and B have already been calculated as $T(A)$ and $T(B)$, i.e., A and B are *Alive*. In order to update $T(C)$, we create a virtual triangle OAB with lengths of two edges equal to $T(A)$ and $T(B)$, as shown in Fig. 2(a) for the update inside the triangle. Point O is the virtual center point in the same plane as the triangle ABC. We assign the new $T(C)$ as the length of the segment OC.

$$T(C) = \sqrt{|AC|^2 + T^2(A) + 2|AC|T(A)\cos{(\alpha + \theta)}}$$

where

$$\alpha = \arccos \frac{|AB|^2 + |AC|^2 - |BC|^2}{2|AC||AB|}$$

and

$$\theta = \arccos \frac{T^2(A) + |AB|^2 - T^2(B)}{2|AB|T(A)}$$

When the update is outside the triangle, we let

$$T(C) = T(B) + |BC|$$

, as shown in Fig. 2(b). Usually, the vertex may have update inside the triangle from at least one direction. The algorithm degenerates to the Dijkstra's algorithm for the vertex that does not have update inside the triangle. Locally, we changed the computation of the distance on the surface to the simple calculation in planar triangles. The validity of this method is that the length of the curve on the surface is the integration along the tangent direction and the tangent surfaces are locally isometric to the planes [3].

We need to point out that although the geodesic measure is more natural than the Euclidean measure in differential geometry, it has two shortcomings. The geodesic distance is more sensitive to the surface sampling noise. To overcome this, the data should be smoothed [13] in the preprocessing stage. The other drawback is when we generate the geodesic circles on the self occluded objects, the calculated geodesic distance may be different from the exact value. As shown in Fig. 2(c), when a part of the plane is occluded, the geodesic circles are no longer co-centric circles. One method to solve this problem is that the marching is stopped whenever a step discontinuity is met.

In Fig. 3, we plotted geodesic circles on a head model. The surfaces shown in Fig. 3(a) and Fig. 3(b) are created from two different views of a head model. The geodesic circles on two surfaces look very similar in spite of the existence of some self occlusion on the nose in Fig. 3(b). This inspires the use of fingerprints as the feature of 3D points.

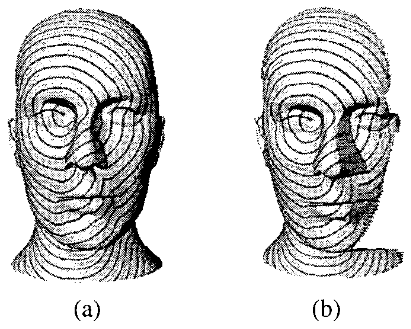

Figure 3. Geodesic circles obtained by fast marching methods. (a) and (b) are parts of a head model from two different views.

2.2. Local fingerprint

Before projecting the geodesic circles onto the tangent plane, the local coordinate system is defined at first, as shown in Fig. 4(a). The normal direction N at the point P defines one coordinate axis, which is computed as the average normal of the neighboring triangles. We arbitrarily choose one of the neighbor points Q and the other two axes are defined as

$$V_y = N \times \overrightarrow{PQ}/||\overrightarrow{PQ}||$$

and

$$V_x = V_y \times N$$

For a certain point M, the coordinate (x, y) of its projection on the tangent plane are

$$x = ((N \times \overrightarrow{PM}) \times N) \cdot V_x$$

and

$$y = ((N \times \overrightarrow{PM}) \times N) \cdot V_y$$

Only the contours are used in surface matching. Fig. 4(b) shows two projected geodesic contours that are generated for the same 3D point, but on surfaces from two different views respectively. The upper one has fewer numbers of contours because the marching is stopped when a step discontinuity is met. We call these contours the local fingerprint of a certain point. The fingerprints obtained from different views look very similar. Fig. 4(c) plots the radius variation for the third pair of contours in (b). Not only the fingerprints are discriminating themselves, but they also can carry other features. The reason for this is as follows: The projection on the tangent plane may cause many surface points mapped to the same point in the fingerprint. It seems impossible to let each pixel carry a feature of a surface point, but when we use the fingerprint representation, each pixel in a certain contour corresponds to one surface point, although the contours may be intersected with each other on the tangent plane. This is one advantage of using fingerprints over some previous works that project the whole surface patch on the tangent plane. The other advantage of our representation is that the comparison of several contours is much more efficient than the comparison of a pair of images.

2.3. Candidate point selection

Because it is time consuming to compare all pairs of points in two surfaces, and the points in the flat area provide little information in the point matching, we choose to only compare some candidate points. Some previous works argue that all point pairs should be compared in the case of free-form surface matching, where it may not have easily detectable landmark features such as edges and vertices. In our work, we find that some feature points can be extracted even for the free-form surfaces as long as a large enough neighborhood is considered. But the most popular method that extracts high curvature points is no longer appropriate, because the curvature is defined in an infinitesimal surface area. Also the computation of stable and accurate curvature values is not easy on the discrete surface.

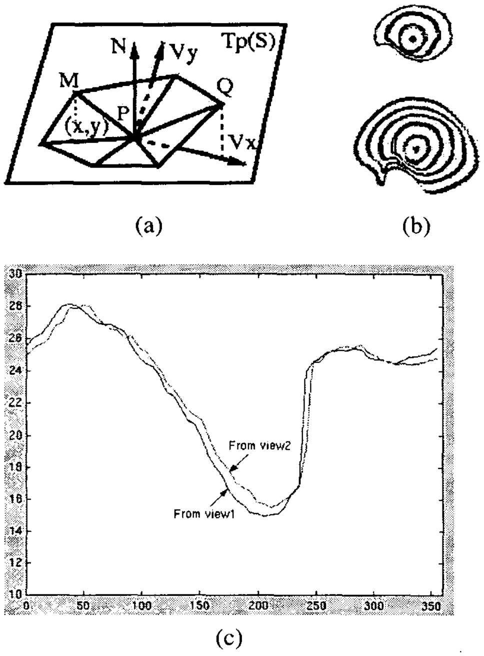

Figure 4. (a) Local coordinate system definition (b) Local fingerprints of the same point from two different views. (c) Plot of the radius variation from 0 to 2π for the third pair of contours in (b).

Considering that the points we are interested in have discriminating fingerprints, we find the candidate points that have irregular contour shapes in the fingerprint. Here the irregularity is defined by the radius variation of the contour. Only one contour is used and the size of the geodesic circle generating the contour depends on the type and the size of the surface under matching. We use relatively large contours for free-form surfaces or large scale surfaces. The candidate points are labeled whenever the ratio between the largest and the smallest contour radius is over some threshold. To operate locally on the triangular mesh, most previous works only consider a simplex [17], or some nearest points obtained by KD-tree implementation that is based on the Euclidean measure. We argue that using the geodesic measure to find the nearest neighboring points is the correct way. One advantage is shown in Fig. 1(b) and the other advantage is that the neighborhood is independent of the surface sampling resolution. Fig. 5(a)(b) show the candidate point selection on the surfaces from two different views of a synthetic head model and Fig. 5(c)(d) show the candidate point selection on the surfaces from the real USGS Digital Elevation Model (DEM), which are similar to free-form surfaces. The fingerprints of the extracted candidate points are compared to find correspondences.

The complexity of candidate point selection is $O(N_1 N_2 \log(N_2))$, where N_1 is the number of the points in surface mesh, N_2 is the number of neighboring points considered for each point. On an SGI OCTANE it takes 4 seconds to extract the candidate points from the surface shown in Fig. 5(d), which has 4392 points.

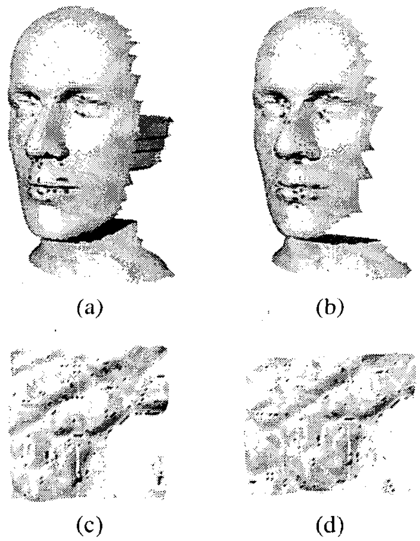

Figure 5. Feature point selection based on irregular local geodesic circles. (a) and (b) Extracted points on the synthetic head model from two different views. (c) and (d) Extracted points on a pair of partially overlapped surfaces from the real USGS DEM. Selected feature points are shown as black dots.

3. Feature matching

Some candidate points are located near the surface boundary. The contours of their fingerprints may not be closed, but these contours still contain useful information. In our work, we marched the whole surface to create the fingerprint for each candidate point. We call it the global fingerprint, as shown in Fig. 6(a)(b).

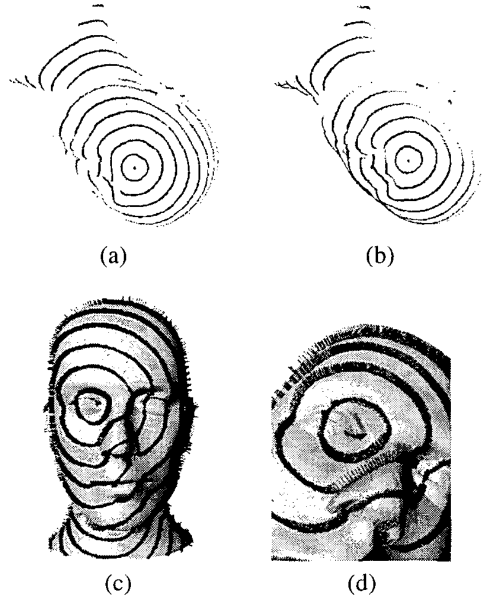

Figure 6. (a) and (b) show the same 3D point's global fingerprint on a head model from two different views. (c) The normal variation on the geodesic circles. (d) A zoom view of (c) near the center point.

The fingerprint can carry different features. Currently we are using only the contour radius variation and the normal variation for surface matching, as shown in Fig. 6(c)(d). On each contour of the fingerprint, we sample with incremental angle of $2\pi/K$ to represent the whole contour. Because each surface may have L candidate points and each candidate point's fingerprint may have M contours, we used a three dimensional ($L \times M \times K$) data structure to store the information for each surface. Usually we choose $L < 100$, $M < 20$ and $K = 30$.

The fingerprints from different views are up to a 2D rotation and the samples along each contour are periodic. We used the following formula to compute R_{ij} that is the dissimilarity measure between the ith candidate point in the first surface and the jth point in the second surface. The formula is similar to the form of cross correlation.

$$R_{ij} = \min_{l=1}^{K}[\sum_{m=1}^{M} \sum_{k=1}^{K} (n_{1,i,m,k} \cdot N_{1,i} - n_{2,j,m,k+l} \cdot N_{2,j})^2]$$

where $n_{1,i,m,k}$ is the normal at the kth point on the mth contour of the ith fingerprint from the first surface and $N_{1,i}$

is the normal at the center point of the ith fingerprint from the first surface, and similarly for $n_{2,j,m,k}$ and $N_{2,j}$ from the second surface. The ith candidate point in the first surface and the kth candidate point in the second surface are corresponded if

$$k = argmin_j R_{ij}$$

and R_{ik} is below a threshold. The contour radius variation is used similarly to confirm the correspondences.

4. Experiment results on 3D image registration

After the corresponding points are found, we solved the rigid transformation between two surfaces' coordinate systems using Horn's method [7]. Fig. 7(a) shows the misaligned bunny model and Fig. 7(b) shows the registered result. The bunny models were under translation before registration. Fig. 7(c) shows the misaligned head model and Fig. 7(d) shows the registered result. The head models were under both translation and rotation before registration. The method is also applied to align the real USGS DEM data. Fig. 7(e) shows the misaligned surfaces and Fig. 7(f) shows the registered result. Note that the two datasets only overlap in some area. Our method successfully found the corresponding point pairs in the overlapping area.

Due to the nature of the discrete sampling, the corresponding points can not be exactly the same. So the registration generally has a small error. The Iterative Closest Point (ICP) algorithm [2] can be used to refine the results.

5. Conclusion

We presented a new surface representation method – 3D point's fingerprint. It is based on a set of geodesic circles generated on the triangular mesh. The geodesic circles' projection onto the tangent plane forms the discriminating feature similar to a human's fingerprint. The point's fingerprint can carry other features such as the normal variation. We also proposed a novel method to extract the candidate surface points, which is based on one of the small contours of the fingerprint. The fingerprints and the embedded normal variations from a pair of surfaces are compared to find the corresponding points. Experimental results of the application to pose estimation demonstrate that the method performs well for synthetic and real data. Much work remains to be done in the future, especially in surface matching under occlusion and stability analysis of the algorithm.

6. Acknowlegement

This work was supported by the University Research Program in Robotics under grant DOE-DE-FG02-86NE37968, by the DOD/TACOM/NAC/ARC Program,

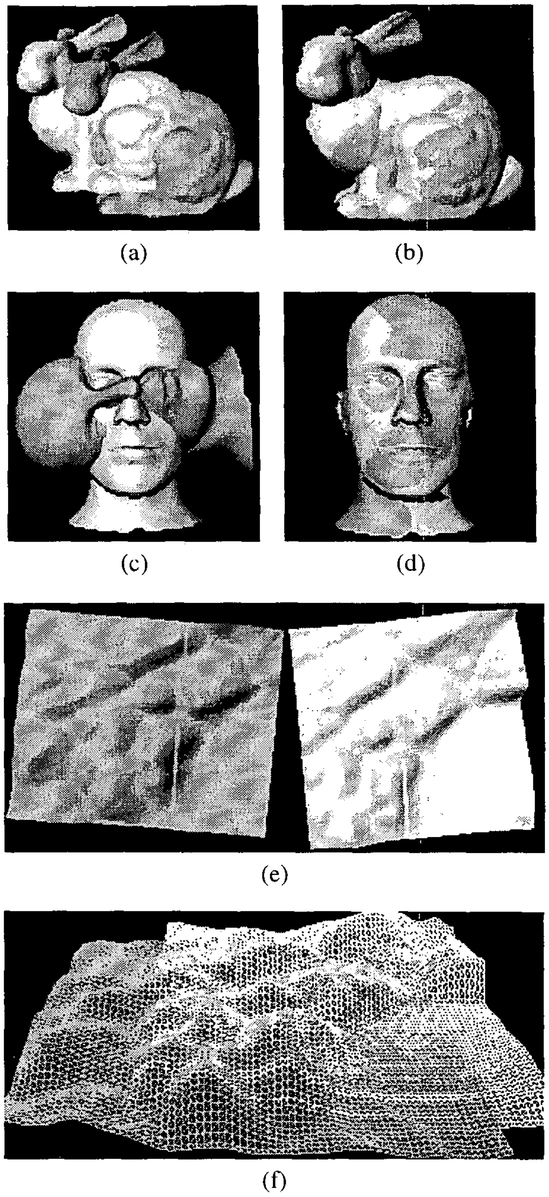

Figure 7. Application of point's fingerprint to 3D registration. (a)(c) Unregistered synthetic surfaces from the bunny model and the head model. (b)(d) Surface registration results of (a) and (c) by point's fingerprint matching. (e) Misaligned surfaces from the real USGS DEM data with only partial overlapping region between each other. (f) The registration result of (e) by point's fingerprint matching, shown in wireframe.

R01-1344-18, and by FAA/NSSA Program, R01-1344-48/49. We would like to thank Profs. A. Freire and C. Plaut for helpful discussions.

References

[1] A. Aho, J. E. Hopcroft, and J. D. Ullman. *Data structures and algorithms*. Addison-Wesley, Reading, Mass., 1983.

[2] P. Besl and N. McKay. "A method for registration of 3-D shapes". *IEEE Trans. Pattern Analysis Machine Intelligence*, 14(2):239–256, 1992.

[3] M. P. Do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice Hall, 1976.

[4] C. Chua and R. Jarvis. "Point signatures: a new representation for 3D object recognition". *International Journal of Computer Vision*, 25(1):63–85, 1997.

[5] C. Dorai and A. K. Jain. "COSMOS – a representation scheme for 3D free-form objects". *IEEE Trans. Pattern Analysis Machine Intelligence*, 19(10):1115–1130, October 1997.

[6] M. Hebert, K. Ikeuchi, and H. Delingette. "A spherical representation for recognition of free-form surfaces". *IEEE Trans. Pattern Analysis Machine Intelligence*, 17(7):681–690, July 1995.

[7] B. Horn, H. Hilden, and S. Negahdaripour. "Closed-form solution of absolute orientation using orthonormal matrices". *Journal-of-the-Optical-Society-of-America-A-(Optics-and-Image-Science)*, 5(7):1127–1135, July 1988.

[8] A. E. Johnson and M. Hebert. "Surface registration by matching oriented points". In *International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, pages 12–15, May 1997.

[9] R. Kimmel, A. Amir, and A. M. Bruckstein. "Find shortest paths on surfaces using level sets propagation". *IEEE Trans. Pattern Analysis Machine Intelligence*, 17(6):635–640, June 1995.

[10] R. Kimmel and J. A. Sethian. "Computing geodesic paths on manifolds". In *Proc. National Academy of Sciences*, pages 8431–8435, July 1998.

[11] J. A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Material Sciences*. Cambridge University Press, second edition, 1998.

[12] F. Stein and G. Medioni. "Structural indexing: Efficient 3-D object recognition". *IEEE Trans. Pattern Analysis Machine Intelligence*, 14(2):125–145, February 1992.

[13] Y. Sun, J. K. Paik, J. R. Price, and M. A. Abidi. "Dense range image smoothing using adaptive regularization". In *IEEE International Conference on Image Processing*, volume II, pages 744–747, 2000.

[14] Y. Wang, B. Peterson, and L. Staib. "Shape-based 3D surface correspondence using geodesic and local geometry". In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume II, pages 663–668, 2000.

[15] H. Yahia, E. Huot, I. Herlin, and I. Cohen. "Geodesic distance evolution of surfaces: a new method for matching surfaces". In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume I, pages 644–651, 2000.

[16] S. M. Yamany and A. A. Farag. "Free-form surface registration using surface signatures". In *IEEE International Conference on Computer Vision*, pages 1098–1104, September 1999.

[17] D. Zhang. *Harmonic Shape Images: A 3D Free-Form Surface Representation and Its Application in Surface Matching*. PhD thesis, Carnegie Mellon University, 1999.

[18] D. Zhang and M. Hebert. "Harmonic maps and their applications in surface matching". In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume II, pages 524–530, 1999.