Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2014) 1–10

Procedia Computer Science

Surface Reconstruction From the Cross-Sections Based on FFD Matching

Abstract

Keywords:

1. Introduction

In the present work, a new framework has been developed to reconstruct a object from a series of parallel planar cross-sections(slices), each slice contains one or more contours, and a 3-D surface will be obtained as a result.

A high quality reconstruction is demanded by product designing, 3D-printing, nondestructive inspection, medical imaging, or where need a visual effect from the slices. Recently, with the improvements in data acquisition and imagine techniques like MRI, CT and ultrasound imaging, it's easy to get the slices which represent the object precisely, which make the reconstruction from slices more practicably in many fields.

The procedure of 3D reconstruction based on the 2D cross-sections usually consist of data acquisition, correspondence decision and surface fitting. The correspondence problem can subdivide into two steps, global correspondence and local correspondence. Global correspondence decides the portions connection, and solves the branching problem (many-to-many case), sometimes the branching problem is considered as a independent problem Fig.1a. And the local correspondence(also called the tiling problem) decide how to stitch the adjacent corresponding contours together,like Fig. 1b. An automatic solution of the correspondence is very difficult in its general form, since there is no constraints on the adjacent slices, and it's very hard to extract the correspondence from the adjacent dissimilar slices. The surface fitting step will generate a smooth surface, and it is hot topic in computer graphics.

There has been a lot of previous research on the correspondence problem, the most of them obtained the correspondence by advancing rules or a set of constraints, it's really hard to deal with the various difference between the adjacent slices.

This paper present a new frame work to do the reconstruction, all of which works on the 2D planar cross sections. The correspondence problem is solved by the free from deformation (FFD) matching, and the branching problem is solved by the medial axis. Once corresponding points are extracted, object surface can be quickly rendered by various classic algorithms. See more detail in Section 3. Our algorithm can solves the most general version of the problem, with successive parallel slices, as well as the branching problem. The generated surface is approximate to the contour's points, so is naturally smooth.

The remainder of this paper is organized as follows. In section 2, we review previews work on reconstruction. In section 3, we present in detail our reconstruction framework. This includes the FFD matching, branching problem and the detect of the capping region. In section 4, we show some example applications of our reconstruction framework, and analyze the accurate of the result.

(a) global correspondence (b) local correspondence

Figure 1: The correspondence decision

2. OVERVIEW OF PREVIOUS APPROACHES

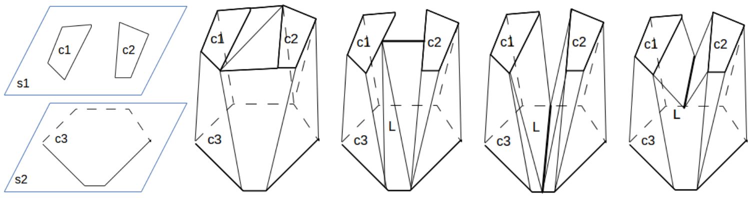
Many solutions have proposed for the surface reconstruction problem, some of them were suggested for the pure raster interpolation, the interpolation produces one or more intermediate raster images, and then extracted the surface from these raster images, like use the level set method to obtain the surface. Cline et al.[1, 2] use the marching cube technique convert the voxel to surface straightly, with a small size cube and a high quality input data, this method can have a good result, but the disadvantage of this technique is wrong surface and hole generation.[3]

Many other solutions, including the approach taken in this paper, use the contours extracted from each slice to represent the slice, and find the corresponding points on the adjacent contours, then generate the surface by stitching the contours together. At the beginning works were studied on the one-to-one case, which means one slice only contain one contour, in the simple case, studies used the global optimization of objective function or settled with a local tiling-advancing rule to get the correspondence or stitch the contour together straightly. Like Keppel[4] use the minimum path in a weighted toroidal graph to tile the triangle between the two contours. Fuchs et al.[5] proposed the Divide-and-Conquer algorithm to search the bottlenecks, and tiling with the minimum area, And Sloan[6] discussed on the cost metrics and made a improvement on Fuchs's method. But above works must work with the high degree of resemblance between adjacent contours. Christiansen and Sederberg[7] use a local advancing rule to deal with the no apparent resemblance case. And they also tried to solve the branching problem by adding a bridge, like Fig. 2a. But the drawback of this method is a conflicting bridge may be added like Fig. 2b. Barequet solved the branching problem by using the cleft, and stitching the cleft by dynamic programming technique which can minimizes the area of the triangles[8]. He also attempted some other cost metrics to do the triangulation[9, 10].Chandrajit do reconstruction by imposing a set of constraints on the reconstructed surface, and derived the correspondence and tiling rules from these constraints[11]. Qingqi Hong reconstructed the vessels surface depended on the Frenet Frame and blended the branches together by using the extended smooth maximum funtion[12].Liu et al[13],Barequet[14],Bermano[15] had extended their works to non-parallel and multilabel reconstruction.

(a) an acceptable bridge (b) a conflicting bridge

Figure 2: Bridges in simple branching case

For branching problem, Chandrajit et al classified the approaches into four methods, shown in Fig. 3. The main ideal of solving the branching problem is converting the one-to-one into a one-to-one case, Fig. 3b. shows that a composite contour replaced the branching contours[16], so it's converted into a one-to-one case. apparently, this method will loose details,like holes. Fig. 3c shows that a bridge added between the branching contours[8, 7], thus reducing to the simple one-to-one case. This method some times may add a conflicting bridge, or distortion may arise. Recently the solutions focused on method shown in the Fig. 3d 3e, with the developed of the scanning technique,we are able to reach a slice distance equal to the in-plane resolution. So interpolate a cure (or a contour) between the slice or in the bottom slice is no different between visual effects. Barequet[14], Chandrajit[11],Geiger[17]used the straight skeleton to do the interpolation, and Jeong[18] used the distance map to interpolate the contours between the slices.



(a) (b) (c) (d) (e)

Figure 3: Different reconstruction for branching contours:(a) branching contours on adjacent slices;(b)-(e) different surface reconstruction

3. Reconstruction Based on the FFD Matching

3.1. overview of our approach

To the best of our knowledge, all previous work used the advancing rule or a set of constraints to get the correspondence, and it is really hard to deal with the general form, because there are various types of differences between the adjacent slices. But if the adjacent slices have high degree of resemblance, almost all the previous works can get a perfect correspondence, so with this key-point we propose a new framework which can deform the adjacent contours to have a high degree of resemblance. Our proposed framework consists of the following steps,like fig4:

- Data acquisition:

Orienting all contours in each slices, and calculate the skeleton for the object.

- Branching problem:

Locating the branching locate by the skeleton, use the medial axis projection to subdivide the contour which is the one, in one-to-many case, and reducing the problem to one-to-one case. See detail in section 3.5.

- Capping Region:

Set the transformation direction, solve the degeneracy problem. See detail in section **??**.

- Matching Contour Based on the FFD:

Obtaining the correspondence based on the FFD. See detail in section 3.3.

3.2. Data Acquisition

like previous works, we are interesting in the contour's orientations, since all contours in consistent direction are good for the triangulation. And the data must consist of a sequence of slices, since we calculate the skeleton of the object base on the sequential slices. The skeletal points are the geometric centers of each contour. the skeleton is used to locate the branches.

Then distance transform (also know as distance maps) is compute for each slice by determining:

1. The nearest contour curve segment.
2. The distance to the nearest point of each pixel.

The distance transform will be used in the FFD matching and calculation of medial axis.

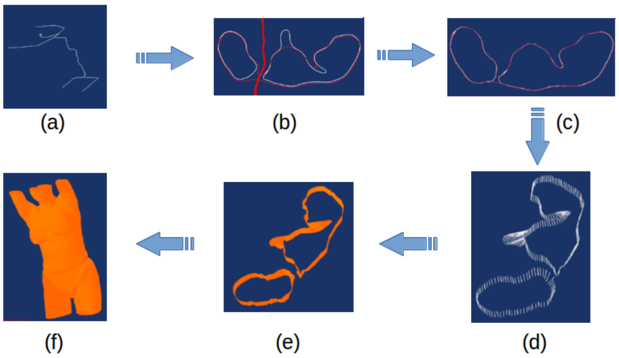


Figure 4: The basic steps of the reconstruction base on FFD: (a)It's the skeleton extract from the contours. (b) A para layers contain branches,the contour with white line is upper layer, and the contour with red line is lower layer, the bold red line is medial axis of the upper layer. (c) The deformation based on FFD. (d) The correspondence based on the FFD. (e)(f) The surface of the original data

3.3. Matching Contour Based on the Free-Form Deformation

In order to get a pair of contours with high degree of resemblance, we need perform deformation on one of the adjacent contours,local transformation models can represent pixel-wise deformations that deform a shape locally and non-rigidly, like optical flow [19][20],Thin Plate Splines(TPS)[21][22], and space deformation techniques such as Free Form Deformations[23]. The optical flow and FFD method can satisfy our demand that transform one contour to another one, both method can be used according to coarse-to-fine strategy which is more flexible to applications, FFD is able to implicitly enforce smoothness constraints, exhibits robustness to noise, and the recovered deformation field is continuous, preserves shape topology and guarantees a one-to-one mapping, which means a high degree of resemblance. Other than FFD, the methods based on optical flow is trying to estimate the motion of the flow filed, and minimize the object energy. Even with the advanced regularization and smoothness constraints, the model is not very suitable for reconstruction since it does not guarantee the preservation of topology and coherence of a shape after deformation[19], and it does not necessarily produce one-to-one correspondences.[23]. use the free-form deformation method to get the correspondence.

In this paper we chose the Incremental Free Form Deformation (IFFD)[23] to perform the contour's deformation. Its a B-spline based FFD model which used to minimize a Sum-of-Squared-Differences(SSD) measure and further recover a dense local non-rigid registration field.

For a given contour, a regular lattice P overlaid on its surface,the deformation of the contour is caused by manipulating the Lattice P. The deformation of the control lattice P consists of the displacements Θ as [Eq. 1] of all the control points $\{\mathbf{p}_{m,n}\}$ in the lattice, and from these sparse displacements Θ, a dense deformation field for every pixel in the embedding space can be acquired through interpolation using B-spline function.

$$\Theta = \{\delta \mathbf{p}_{m,n}\} = \left\{\delta p^x_{m,n}, \delta p^y_{m,n}\right\}; (m,n) \in [1, M] \times [1, N] \tag{1}$$

According to the B-splines property, for any pixel $\mathbf{x}$ on the original shape and its embedding space can represent as [Eq. 2]

$$L(\mathbf{x}) = \mathbf{x} = \sum_{k=0}^{3}\sum_{l=0}^{3} B_k(u)B_l(v)P^0_{i+k,j+l}, \mathbf{x} = \{x, y\} \tag{2}$$

where

- i= $\lfloor \frac{x-min_x}{max_x-min_x} \cdot (M-1)\rfloor$, $j = \lfloor \frac{y-min_y}{max_y-min_y} \cdot (N-1)\rfloor$, this is according to the definition of the cubic B-spline.
- $P^0_{i+k,j+l}$, $\mathbf{x} = \{x, y\}$ are the control points in the lattice, P^0 means original.
- $B_k(u)(B_l(v))$ are the k^{th} (l^{th}) basis function of the cubic B-spline.

The new position of pixel x after deformation can be given by

$$L(\Theta : \mathbf{x}) = \mathbf{x} + \sum_{k=0}^{3}\sum_{l=0}^{3} B_k(u)B_l(v)\delta P_{i+k,j+l} \tag{3}$$

With the definition of deformation, how to deform one contour to another one is equivalent to finding the control lattice deformation Θ. For example, we apply these to the upper a layer of the pair of the contours, and deformed the upper contour to coincides with the lower contour. According to [23], we can use the gradient descent optimization technique to find Θ. The energy function [Eq.6] consist of the data-driven term [Eq.4] and smoothness term [Eq.5].

$$E_{data}(\Theta) = \iint_D (\Phi_u(\mathbf{x}) - \Phi_l(L(\Theta : \mathbf{x})))^2 d\mathbf{x} \tag{4}$$

$$E_{smoothness}(\Theta) = \iint_D \left(\frac{\partial \delta L(\Theta : \mathbf{x})}{\partial x}^2 + \frac{\partial \delta L(\Theta : \mathbf{x})}{\partial y}^2 \right) d\mathbf{x} \tag{5}$$

$$\frac{\partial}{\partial \theta} E(\Theta) = \frac{\partial}{\partial \theta} E_{data} + \frac{\partial}{\partial \theta} E_{smoothness} \tag{6}$$

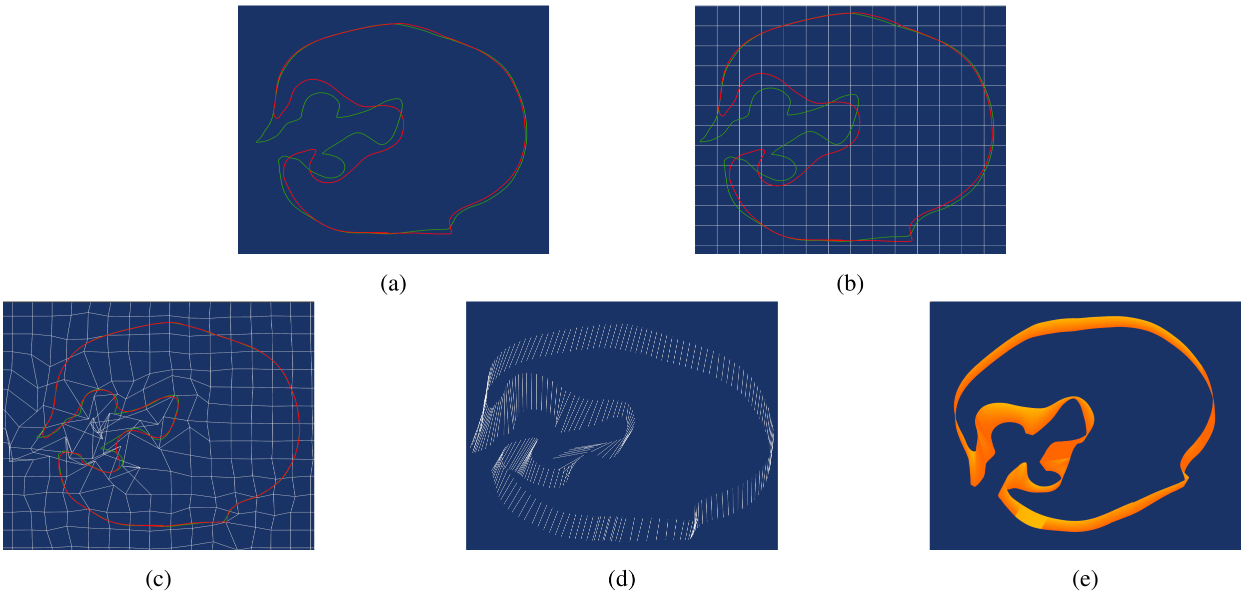


Figure 5: Matching based on the IFFD. (a) the original contours of a pair, the upper contour (in red), and the lower layer (green line) (b) the embedding space of the upper contour. (c) Final FFD control lattice configuration depicting the space warping to achieve a high degree of resemblance. (d) the correspondence of the adjacent contours. (e) The surface stitched based on the matching result.

As presented in Fig. 5, the surface is stitched based on the correspondence which is obtained by the IFFD matching. From fig. 5c 5d we can see that the correspondence is satisfied.

3.4. Capping Region

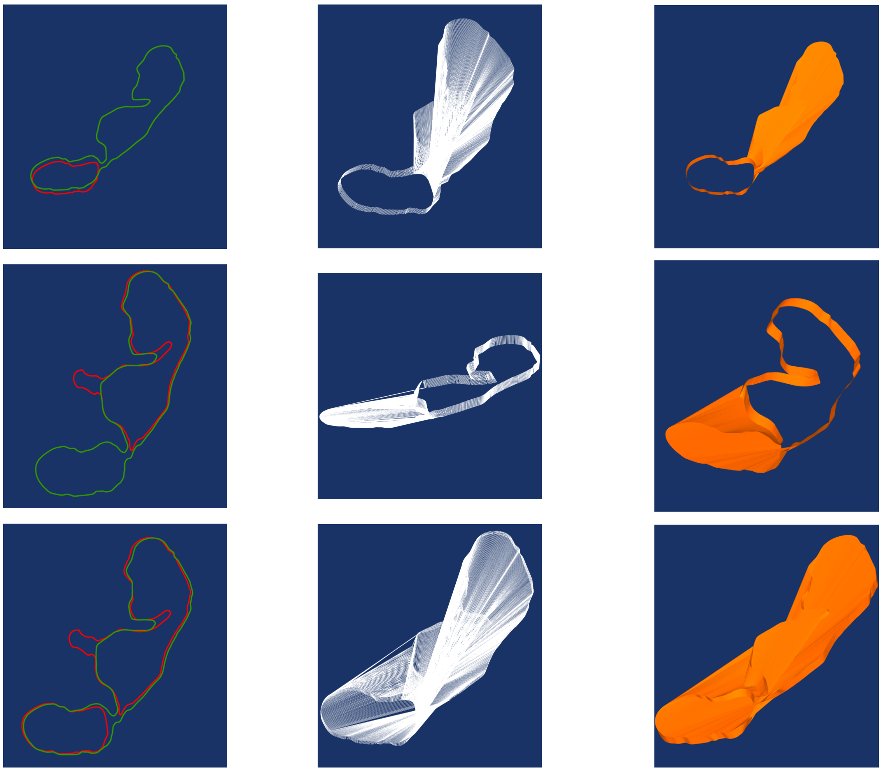
In capping region, where a contour has no counterpart in the neighbor layer, degeneracy occurs. This degeneracy is undesirable in the most time, like fig.6 shown.This degeneracy case is very common in reconstruction , some methods chose left this section as a plain surface, but the preferred way is that make a slant surface between the adjacent contours. In our framework, we can handle this degeneracy easily. In fig.6.(b) The upper layer was shrinked caused by the IFFD. From fig.6.(b)-(c),we can see that, the correspondence is incorrect, and the reconstructed surface has a wrong section. Actually, the shape shown in the fig.6 is a branching problem, and one branch represents a capping region ([18] call the capping region as single branching problem).

In order to solve this problem, we just need to measure the length of every contour, and make sure that we only transform the shorter contour to the longer one. Figure 7 demonstrates that the length check is very effective. As shown in fig. 7(b)-(c), the shrink was solved, and the correspondence is almost correct, expect the section in red square. There is a gap like a triangle in the red square, since there is no points as the counterpart in the green contour for the red contour. So the points in the red point corresponded to the nearest boundary points in the green contour. We will solve this problem in the section3.5.

1)

2)

3)



(a) (b) (c)

Figure 6: The degeneracy of the capping region: There are two layers, the lower layer has two contours in red, the upper layer has one contour. 1) the lower contour is smaller one in the lower layer involved in the process. 2) the lower contour is bigger one in the lower layer involved in the process. 3) both contours in the lower layer involved in the process. (a) the original contours, the upper layer contains one contour in green, the lower contains one contour in red. (b) the correspondence based on the IFFD. (c) the final surface

3.5. Medial Axis for Branching Problem

From fig.5, it can be seen that the advantage of the technique above is obviously — we can achieve a pair of resembled contours like fig.5c; But the defect is obviously too, it is only appropriate for no branching case, since the upper contour will be transformed as same as the lower contour depend on the energy function, illustrative examples is shown in fig. 8. A synthetic shape was shown in fig.8, the shape represents a classic branches problem, the upper layer contains two contours (both are circles), and the lower layer contains one contour is an ellipse. A intersection (fig.8(1)(b)) is introduced by the transform when we calculate the correspondence, which is unexpected in the most time. And even for one-to-one case, there may be a gap left on the surface, 7.

In order to avoid the appearance shown above, we introduce the medial axis to deal with the branching problem. As we explained above, the key point of the IFFD marching is the distance transform, the processing of transformation is mostly depending on the distance transform. So we use the medial axis to separate the lower layer. Generally, the lower layer will be divide into the same parts as upper layers.

As fig.8 shown, the medial axes of upper layer (fig.8.2(b) red points) is first calculated, and then project the medial axes into the lower layer inner-contour section, finally we recalculate the distance map of the lower layer, until now we can apply the reconstruction based on IFFD technique as usual. The result is more naturally, see fig.8. 2(c),8. 2(d).

4. EXPERIMENTAL RESULTS

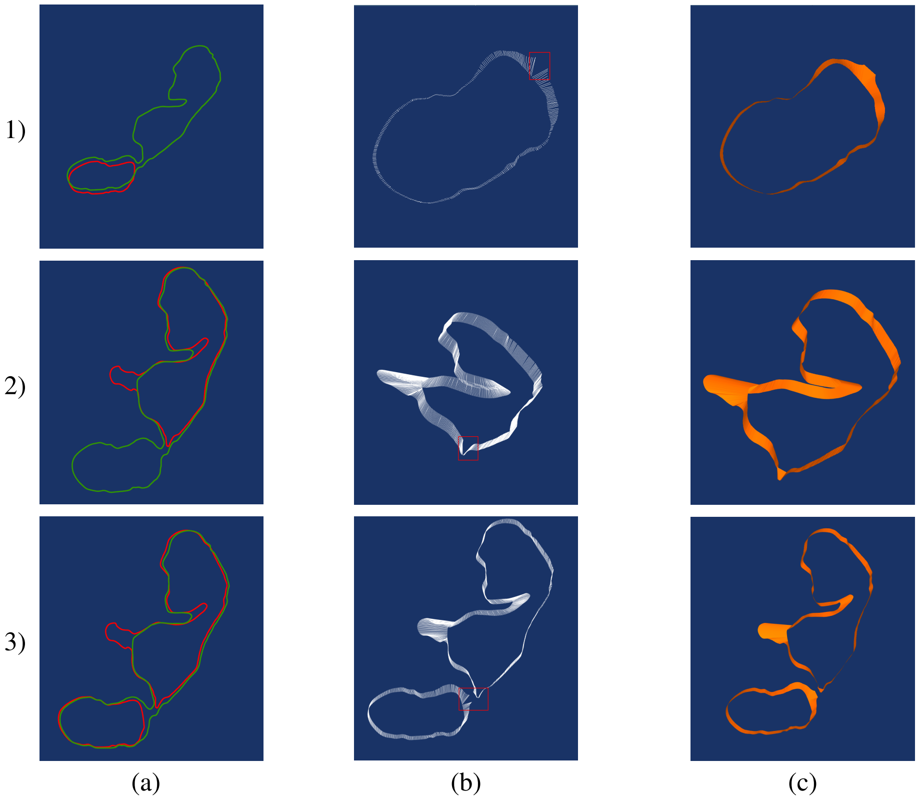


Figure 7: the capping region problem: 1) both layer contain one contour, the process don't consider the length; 2) lower layer contains contain two contour in red, upper layer contains one contour in green. the process don't consider the length; 3) lower layer contains contain two contour in red, upper layer contains one contour in green. the process consider the length. (a)the original contours ;(b) the correspondence based on IFFD; (c) the final surface.

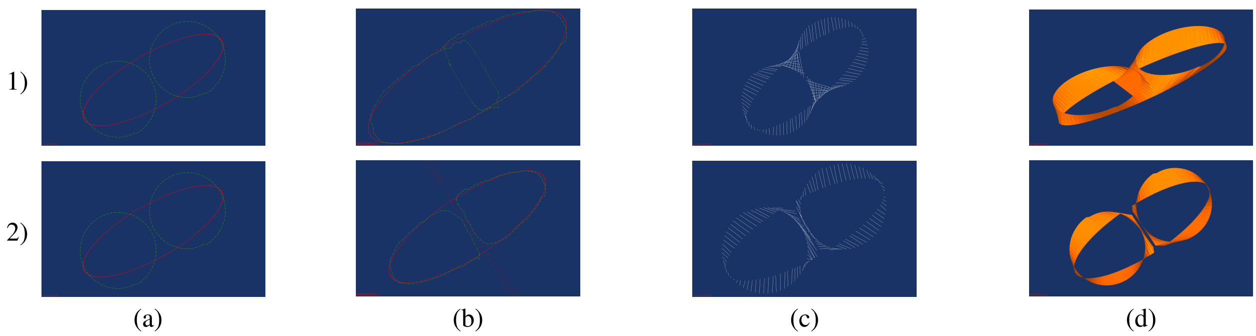


Figure 8: A synthetic dataset with branches. 1) reconstruct without medial axes, 2) reconstruction with medial axes. (a) the original contours, (b) contours after IFFD, (c) Established correspondences after IFFD, (d) the final surfaces

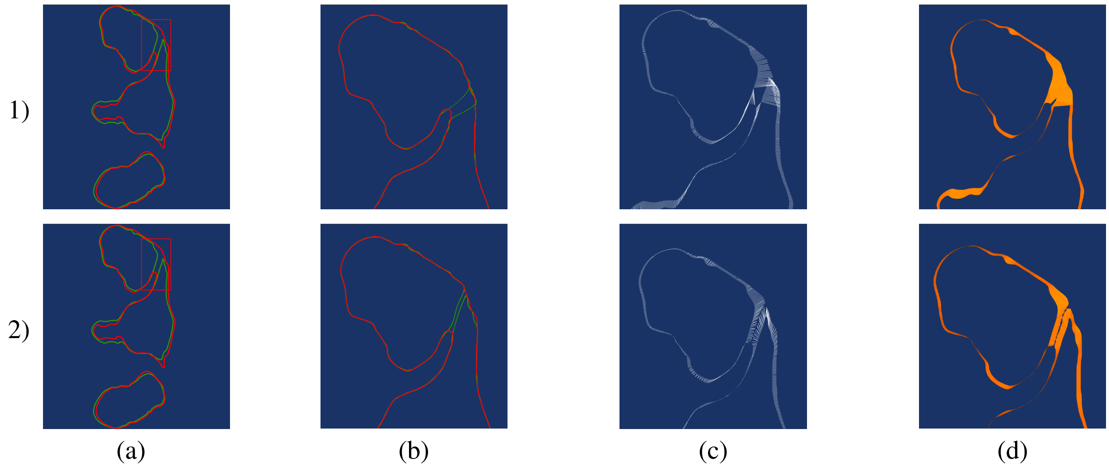


Figure 9: A real dataset with branches. (1) reconstruction without medial axes, (2) reconstruction with medial axes. (a) the original contours, (b) the contours after IFFD, (c) Established correspondence after IFFD, (d) The final surfaces

References

[1] H. E. Cline, W. E. Lorensen, S. Ludke, C. R. Crawford, B. C. Teeter, Two algorithms for the three-dimensional reconstruction of tomograms, Medical physics 15 (3) (1988) 320–327, PMID: 3043154.

[2] W. E. Lorensen, H. E. Cline, Marching cubes: A high resolution 3D surface construction algorithm, in: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '87, ACM, New York, NY, USA, 1987, p. 163169. doi:10.1145/37401.37422.
URL http://doi.acm.org/10.1145/37401.37422

[3] J. Jin, Q. Wang, Y. Shen, J. Hao, An improved marching cubes method for surface reconstruction of volume data, in: The Sixth World Congress on Intelligent Control and Automation, 2006. WCICA 2006, Vol. 2, 2006, pp. 10454–10457. doi:10.1109/WCICA.2006.1714052.

[4] E. Keppel, Approximating complex surfaces by triangulation of contour lines, IBM J. Res. Dev. 19 (1) (1975) 211. doi:10.1147/rd.191.0002.
URL http://dx.doi.org/10.1147/rd.191.0002

[5] H. Fuchs, Z. M. Kedem, S. P. Uselton, Optimal surface reconstruction from planar contours, Commun. ACM 20 (10) (1977) 693702. doi:10.1145/359842.359846.
URL http://doi.acm.org/10.1145/359842.359846

[6] K. R. Sloan, J. Painter, Pessimal guesses may be optimal: A counterintuitive search result, IEEE Trans. Pattern Anal. Mach. Intell. 10 (6) (1988) 949955. doi:10.1109/34.9117.
URL http://dx.doi.org/10.1109/34.9117

[7] H. N. Christiansen, T. W. Sederberg, Conversion of complex contour line definitions into polygonal element mosaics, in: Proceedings of the 5th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '78, ACM, New York, NY, USA, 1978, p. 187192. doi:10.1145/800248.807388.
URL http://doi.acm.org/10.1145/800248.807388

[8] G. Barequet, M. Sharir, Piecewise-linear interpolation between polygonal slices, Computer Vision and Image Understanding 63 (2) (1996) 251–272. doi:10.1006/cviu.1996.0018.
URL http://www.sciencedirect.com/science/article/pii/S1077314296900181

[9] G. Barequet, D. Shapiro, A. Tal, History consideration in reconstructing polyhedral surfaces from parallel slices, in: Proceedings of the 7th Conference on Visualization '96, VIS '96, IEEE Computer Society Press, Los Alamitos, CA, USA, 1996, p. 149ff.
URL http://dl.acm.org/citation.cfm?id=244979.245044

[10] G. Barequet, D. Shapiro, A. Tal, Multilevel sensitive reconstruction of polyhedral surfaces from parallel slices., The Visual Computer 16 (2) (2000) 116–133.

[11] C. L. Bajaj, E. J. Coyle, K.-N. Lin, Arbitrary topology shape reconstruction from planar cross sections, Graphical Models and Image Processing 58 (6) (1996) 524–543. doi:10.1006/gmip.1996.0044.
URL http://www.sciencedirect.com/science/article/pii/S1077316996900441

[12] Q. Hong, Q. Li, J. Tian, Implicit reconstruction of vasculatures using bivariate piecewise algebraic splines, IEEE Transactions on Medical Imaging 31 (3) (2012) 543–553. doi:10.1109/TMI.2011.2172455.

[13] L. Liu, C. Bajaj, J. O. Deasy, D. A. Low, T. Ju, Surface reconstruction from non-parallel curve networks, Computer Graphics Forum 27 (2) (2008) 155163. doi:10.1111/j.1467-8659.2008.01112.x.
URL http://onlinelibrary.wiley.com/doi/10.1111/j.1467-8659.2008.01112.x/abstract

[14] G. Barequet, A. Vaxman, Reconstruction of multi-label domains from partial planar cross-sections, Computer Graphics Forum 28 (5) (2009) 13271337. doi:10.1111/j.1467-8659.2009.01510.x.
URL http://onlinelibrary.wiley.com/doi/10.1111/j.1467-8659.2009.01510.x/abstract

[15] A. Bermano, A. Vaxman, C. Gotsman, Online reconstruction of 3D objects from arbitrary cross-sections, ACM Trans. Graph. 30 (5) (2011) 113:1113:11. doi:10.1145/2019627.2019632.
URL http://doi.acm.org/10.1145/2019627.2019632

[16] D. Meyers, S. Skinner, K. Sloan, Surfaces from contours, ACM Trans. Graph. 11 (3) (1992) 228258. doi:10.1145/130881.131213.
URL http://doi.acm.org/10.1145/130881.131213

[17] B. Geiger, Three-dimensional modeling of human organs and its application to diagnosis and surgical planning, Ph.D. thesis, INRIA Tech. Rep. 2105 (1993).
URL http://citeseer.uark.edu:8380/citeseerx/viewdoc/summary?doi=10.1.1.56.9652

[18] J. Jeong, K. Kim, H. Park, H. Cho, M. Jung, B-spline surface approximation to cross-sections using distance maps, The International Journal of Advanced Manufacturing Technology 15 (12) (1999) 876–885. doi:10.1007/s001700050145.
URL http://link.springer.com/article/10.1007/s001700050145

[19] N. Paragios, M. Rousson, V. Ramesh, Non-rigid registration using distance functions, Computer Vision and Image Understanding 89 (23) (2003) 142–165. doi:10.1016/S1077-3142(03)00010-9.
URL http://www.sciencedirect.com/science/article/pii/S1077314203000109

[20] C. Xu, J. J. Pilla, G. Isaac, J. H. G. Iii, A. S. Blom, R. C. Gorman, Z. Ling, L. Dougherty, Deformation analysis of 3D tagged cardiac images using an optical flow method, Journal of Cardiovascular Magnetic Resonance 12 (1) (2010) 1–14. doi:10.1186/1532-429X-12-19.
URL http://link.springer.com/article/10.1186/1532-429X-12-19

[21] S. Belongie, J. Malik, J. Puzicha, Matching shapes, in: Eighth IEEE International Conference on Computer Vision, 2001. ICCV 2001. Proceedings, Vol. 1, 2001, pp. 454–461 vol.1. doi:10.1109/ICCV.2001.937552.

[22] H. R. Roth, J. R. McClelland, T. E. Hampshire, D. J. Boone, Y. Hu, M. Modat, H. Zhang, S. Ourselin, S. Halligan, D. J. Hawkes, Registration of prone and supine CT colonography datasets with differing endoluminal distension, in: H. Yoshida, S. Warfield, M. W. Vannier (Eds.),

Abdominal Imaging. Computation and Clinical Applications, no. 8198 in Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2013, pp. 29–38.

[23] X. Huang, N. Paragios, D. Metaxas, Shape registration in implicit spaces using information theory and free form deformations, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (8) (2006) 1303–1318. doi:10.1109/TPAMI.2006.171.