

Surface reconstruction of complex contour lines based on chain code matching technique

Jiang Xiaotong

(Department of Instrument Science and Technology, Southeast University, Nanjing 210096, China)

Abstract: A new method for solving the tiling problem of surface reconstruction is proposed. The proposed method uses a snake algorithm to segment the original images, the contours are then transformed into strings by Freeman's code. Symbolic string matching technique is applied to establish a correspondence between the two consecutive contours. The surface is composed of the pieces reconstructed from the correspondence points. Experimental results show that the proposed method exhibits a good behavior for the quality of surface reconstruction and its time complexity is proportional to mn where m and n are the numbers of vertices of the two consecutive slices, respectively.

Key words: chain code; string matching; surface reconstruction; local shape feature

Three-dimensional surface reconstruction is widely used in medical imaging to visualize and manipulate 3-D structures, especially from a set of tomographic cross-sections. Meyers^[1] subdivided the problem of the surface reconstruction into four subproblems: the correspondence problem, the tiling problem, the branching problem and the surface-fitting problem. A solution to the tiling problem for a pair of planar polygonal contours is a surface that is composed of a series of triangles. Each triangle is formed by joining an apex situated on one of the contours to a vertex located on the other one. In the past three decades, many algorithms have been proposed to solve this tiling problem^[2-5]. Among them, some make use of an optimization process to propagate the local information to a more global level^[2,3]. The tiling problem is always well handled for contours having similar shape, but most of the methods often fail in producing good results when contours have dissimilar shapes, or different orientations. The main reason, as indicated by Meyers^[1], is that the cost function used to compute the tiling does not use enough information to reach a good solution. In complex situations (strong dissimilarities between the shapes, split or merge contour), the resolution of the problem requires a more global analysis, with the use of prior knowledge of the structure.

Our work aims at bringing some solution in defining more robust criteria for solving the tiling problem. Our approach relied on a string matching technique to measure the similarity of two contours and establish a correspondence between points of these contours. A global optimization scheme based on a dynam-

ic programming process allowed wearing local information at a global level. The surface was then piecemeal built from the set of matched points^[4].

1 Contour Matching Based on String Matching Technique

1.1 String matching algorithm

A number of different algorithms for string matching have been proposed in Refs. [6 – 11]. They aim at measuring the similarity between two strings through a metric derived from the Damereau-Levenstein's method and based on edit operations. The distance is defined as the minimum cost operations sequence to convert a string A into a string B . The considered operations are classically the insertion, deletion, and substitution of symbols. A cost function γ is assigned to each basic edit operation. The total cost of the transformation is then given by the cumulated sum of the elementary costs applied to each operation in the sequence.

An edit distance between string A and string B can then be derived:

$$\delta(A, B) = \min \{ \text{cost}(O) \mid O \text{ is an edit operations sequence to convert } A \text{ into } B \} \quad (1)$$

Its computation is performed using a dynamic programming algorithm^[6] and goes through the edit matrix computation $D_{0,1, \dots, |l_n|; 0,1, \dots, |l_m|}$ where l_n and l_m characterize the respective size of each string of symbols A and B and $D_{i,j}$ the minimum number of operations needed to match the symbols $a_{1,2, \dots, i}$ to $b_{1,2, \dots, j}$.

$$\left. \begin{aligned} D_{i,0} &= i, \quad D_{0,j} = j \\ D_{i,j} &= \min \left\{ D_{i-1,j} + \gamma(a_i \rightarrow \lambda), D_{i-1,j-1} + \right. \\ &\quad \left. \gamma(a_i \rightarrow b_j), D_{i,j-1} + \gamma(\lambda \rightarrow b_j) \right\} \end{aligned} \right\} \quad (2)$$

Received 2005-04-22.

Biography: Jiang Xiaotong (1975—), male, doctor, jxtong@seu.edu.cn.

where $\gamma(a_i \rightarrow b_j)$, $\gamma(a_i \rightarrow \lambda)$, $\gamma(\lambda \rightarrow b_j)$ respectively represent the substitution, deletion and insertion costs. The edit distance $\delta(A, B)$ is given by the term D_{l_n, l_m} . Its computation is achieved with a time complexity equal to $O(l_n l_m)$. Edit distances can also be defined in terms of “trace”, where a trace from A to B , $T_{A, B}$, is a set of ordered pairs of integers (i, j) corresponding to the indices of symbols which have been associated.

1.2 Contour matching by means of symbolic string matching technique

String matching has been previously used in Refs. [10, 11] by considering the pairing of contours and centerlines in vascular imaging (coronarography) in order to build high level entities, i. e., vascular branches. It also facilitates the 3-D reconstruction of the vascular tree from two views. The contour lines were preliminarily extracted by applying an active contours (snake) algorithm^[12] on each slice of the data set. Let $(C_{l_n}^1, C_{l_m}^2)$ be two arbitrary contours so that $C_{l_n}^1 = \{p_i^1 \mid i=0, 1, \dots, l_n\}$ and $C_{l_m}^2 = \{p_j^2 \mid j=0, 1, \dots, l_m\}$. Let A and B be the two strings respectively associated with each curve $C_{l_n}^1$ and $C_{l_m}^2$. The symbolic representation was given as a sequence of numbers representing the tangent computed at each point on each curve: $a_i = \{t_i^1\}$, $b_j = \{t_j^2\}$. We then derived a cost function from the angles computed between the tangent at each point $p_i^1 \in C_{l_n}^1$ ($p_j^2 \in C_{l_m}^2$) and the line d_{ij} joining the two points p_i^1 and p_j^2 (see Fig. 1). The cost function associated with the substitution operation, is defined as

$$\gamma(a_i \rightarrow b_j) = \omega \left(|\cos \alpha_i - \cos \beta_j| + (1.0 - \omega) \cdot (|\cos \alpha_i| + |\cos \beta_j|) \right) \quad (3)$$

where

$$\cos \alpha_i = \frac{t_i^1 \cdot d_{ij}}{|t_i^1| |d_{ij}|}, \quad \cos \beta_j = \frac{t_j^2 \cdot d_{ij}}{|t_j^2| |d_{ij}|} \quad (4)$$

The insertion and deletion costs are equal and set to 1.

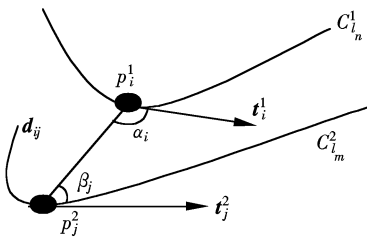


Fig. 1 Perpendicularity and symmetry factor

The substitution cost introduces a parallelism property through a symmetry and perpendicularity measure, performed on pairs of associated points. The costs are variable and continuous on the interval $[0, 2]$. The perpendicularity factor is given by $|\cos \alpha_i| + |\cos \beta_j|$. This factor provides information on the local

perpendicularity of the traces with regard to the contours. The symmetry factor is given by $|\cos \alpha_i - \cos \beta_j|$. It represents the local symmetry of the contours with regard to a median axis (it equals 0 if the tangent vectors are symmetric). ω is a weighting coefficient, which allows balancing these two factors.

We had to match close curves with each other. We do not *a priori* know the starting points to initiate the matching procedure on each curve. We overcame this problem by considering the cyclically shifted string^[8]. It involves duplicating one of the curves, B for instance, to obtain $B^2 = \{b_1, b_2, \dots, b_{l_m}, b_1, b_2, \dots, b_{l_m}\}$. We are then sure that if the string B is partially similar to the string A , B^2 will contain A as a substring independent of the starting position. Some change needs to be brought to the basic algorithm to handle this situation. In particular, when searching for the least cost trace by applying the backtracking process in the edit matrix, we have to look for the element $D_{l_n, j}$, $j = l_m, l_m + 1, \dots, 2l_m - 1$, which equals 0 and will be the starting point of the backtracking procedure (see Ref. [8] for more details). Fig. 2 illustrates this scheme.

B/A	b_1, b_2, \dots, b_{l_m}	b_1, b_2, \dots, b_{l_m}
	0 0, 0, ..., 0	$\infty, \infty, \dots, \infty$
a_1	1	
a_2	2	
\vdots	\vdots	
a_{l_n}	l_n	

Fig. 2 Illustration of string distance computation using cyclically shifted strings

1.3 Triangulation process

Once the correspondence was established between the two contours, the trace (graphically corresponding to the line joining the paired points) provided a first ground for the surface triangulation. The matching allows a one-to-one correspondence but some points on the curves may not have been paired (see Fig. 3 (a)). So we completed the triangulation process in the following way: ① When two successive points on one of the contours were matched with two successive points on the other contour, each point was considered as the vertex of a quadrangle. The latter one was then divided into two triangles by spanning a line on the shorter diagonal. ② When meeting unmatched points, we completed the matching by applying Christiansen and Sederberg's method^[4] to generate a triangulation mesh. For four points forming a quadrangle, the method proposed in Ref. [4] consists of

choosing the shorter of the two possible spans at each stage during construction of the tiling. Fig. 3(b) illustrates the triangulation refinement result.

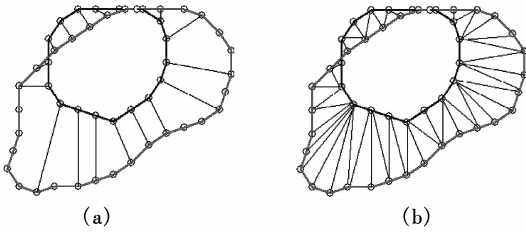


Fig. 3 Triangulation result

2 Experimental Results

We compared our procedure with two classical methods, based on “surface area minimum” and “heuristic” methods, on simulated contours. The minimum area method consists in choosing a triangulation whose surface area is minimum^[2]. The heuristic method proposed by Ganapathy and Dennhy can be outlined as follows: The perimeter length of the contours being tiled was normalized, and the positions of the contour vertices along the perimeter were expressed relative to the normalized length. The span was chosen so that the results provide the least difference in the normalized arc length between the current points on the two contours^[5].

No ground truth is available to measure the quality of a triangulation and the quality of the reconstructed surface. A good triangulation should be smooth, contain long skinny triangles and avoid “waves”. Some criteria were defined in Ref. [13] to evaluate the quality of the triangulation. We applied them to compare the three methods:

Area: Sum of the areas of all the triangles. It measures the total surface area of a particular triangulation. A minimum-surface triangulation is generally considered the “best” guess of a 3-D object surface.

Average dihedral angle: Average of the dihedral angles among all triangles. It can be used to estimate the degree of “smoothness” of the surface.

Average ratio of edges: Average ratio of the longest edge to the shortest edge of the triangles. It indicates the existence of undesired long skinny triangles (Note that for a perfect equilateral triangulation this measure is 1).

Maximum ratio of edges: Maximum ratio of the longest edge to the shortest edge of the triangles. Unlike the previous measures, this one indicates the existence of a single long skinny triangle.

Number of inversions: A triangle forms an inversion when two of its edges are internal to the triangulation

and the dihedral angle of one of them is more than π whereas that of the other edge is less than π . It measures the degree to which the surface is rippled.

These criteria, computed for each of the methods, are reported in Tab. 1. When comparing these figures, we see that they locate close to each other. The “string matching” method reveals good performances if we refer to the criteria values, which appear optimal in most cases. Its time complexity ($2l_n l_m$) is less than the minimum area method’s one ($O(l_n l_m \log_2 l_m)$), but greater than the heuristic method’s one that requires only $l_n + l_m$ operations. Fig. 4 shows the set of contours.

Tab. 1 Comparison of three methods for five pairs of contours

Method	Area	Average angle	Average ratio	Maximum ratio	Inversion numbers
String matching	1 783. 57	182. 91	4. 04	6. 13	18
Heuristic	1 864. 12	186. 19	4. 70	7. 14	23
Minimum area	1 779. 42	184. 24	4. 44	6. 54	17

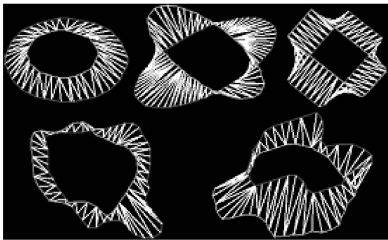


Fig. 4 Triangulation process based on string matching technique on simulated data

We then applied the string matching process on a CT data set to extract the surface mesh of a tumor localized in the lung. The images were acquired on a Siemens system. The acquisition parameters are as follows: slice thickness is 1. 5 mm; pixel size is 0. 33 mm \times 0. 33 mm; matrix size is 512 \times 512; resolution is 12 bit, number of slice is 8. Six successive slices are depicted in Fig. 5. The segmentation results (by deformable model^[12]) appear superposed on the original data. Fig. 6 shows the 3-D linear-interpolated triangulation mesh resulting from the application of the string matching technique.

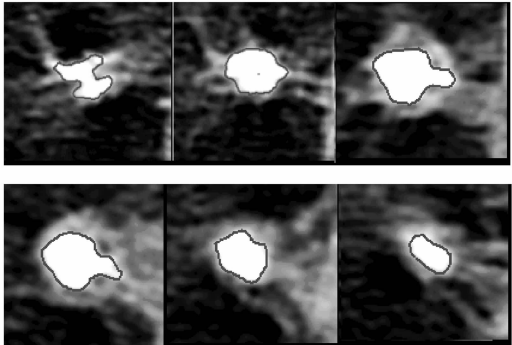


Fig. 5 Six adjacent CT slices of lung showing a tumor

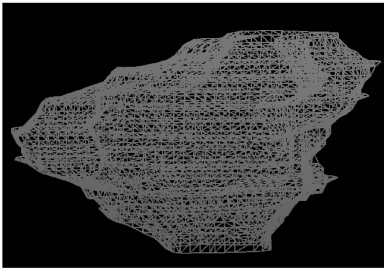


Fig. 6 Reconstructed triangular mesh of the lung tumor

3 Conclusion

A new method based on a string matching technique was proposed to solve the tiling problem in the surface reconstruction. It established an optimal correspondence between points of contours, based on local constraints of perpendicularity and symmetry of contours. The optimization process wore the local information at a global level, producing a more accurate and detailed surface tiling. We evaluated the quality of the “string matching” technique with regard to other methods such as “heuristic” and “minimum area” methods, from a set of criteria we applied on the triangulation results. They underlined the good performance of the “string matching” method. This approach was then applied on a CT data set to reconstruct the surfaces of a tumor localized in the lung. The results provided a good description of the shape. We dealt with a one-to-one contour matching. Our further work will be devoted to the extension to multiple contours matching to take into account the contour split or merge between two slices.

References

- [1] Meyers D. Reconstruction of surfaces from planar contours [D]. Seattle, WA, USA: Department of Computer Science and Engineering of University of Washington, 1994.
- [2] Keppel E. Approximating complex surfaces by triangulation of contour lines [J]. *IBM Journal of Research and Development*, 1975, **19**: 2 – 11.
- [3] Fushes H, Kedem Z, Uselton S P. Optimal surface reconstruction from planar contours [J]. *Communication of the ACM*, 1977, **20**(10): 693 – 702.
- [4] Christiansen H N, Sederberg T W. Conversion of complex contour line definitions into polygonal element mosaics [A]. In: *Proceedings of SIGGRAPH'78, Computer Graphics* [C]. New York, 1978, **12**: 187 – 192.
- [5] Ganapathy S, Dennehy T G. A new general triangulation method for planar contours [A]. In: *Proceedings of SIGGRAPH'82, Computer Graphics* [C]. Boston, Massachusetts, 1982, **16**: 69 – 75.
- [6] Wagner R A, Fisher M J. The string to string correction problem [J]. *Journal of the ACM*, 1974, **21**(1): 168 – 173.
- [7] Marzal A, Vidal E, Airbar P. Fast computation of normalized edit distances [J]. *IEEE Trans Pat Anal Machin Intel*, 1995, **17**: 899 – 902.
- [8] Bunke H, Bühler U. Application of approximate string matching to 2D shape recognition [J]. *Pattern Recognition*, 1993, **26**(12): 1797 – 1811.
- [9] Maes M. Polygonal shape recognition using string matching techniques [J]. *Pattern Recognition*, 1991, **24**(5): 433 – 440.
- [10] Brieva J, Toumoulin C, Wendling F. Extraction of vascular segments in coronarographic image by means of string matching [A]. In: *19th Annual Conference of the IEEE Engineering in Medecine and Biology Society (EMBS)* [C]. Chicago, 1997. 814 – 817.
- [11] Toumoulin C, Brieva J, Bellanger J, et al. String matching techniques for high level primitive formation in 2-D vascular imaging [J]. *IEEE Trans Information Technology in Biomedicine*, 2003, **7**(4): 291 – 301.
- [12] McInerney T, Terzopoulos D. Medical image segmentation using topologically adaptable snakes [A]. In: *Proc of the First International Conference on Computer Vision, Virtual Reality, and Robotics in Medicine (CVRMed'95)* [C]. Nice France, 1995, **905**: 92 – 101.
- [13] Barequet G, Shapiro A. History consideration in reconstructing polyhedral surface from parallel slices [A]. In: *Proceeding of Visualization* [C]. San Francisco, California, 1996. 149 – 156.

基于链码匹配技术的断层间复杂轮廓线三角曲面重建方法

姜晓彤

(东南大学仪器科学与工程系, 南京 210096)

摘要: 利用链码理论对已获取的层间轮廓线进行编码, 将二维的轮廓曲线转化为包含轮廓形状信息的一维链码; 以包含轮廓形状相似信息的“链间距离”为目标函数, 采用链码匹配技术完成相邻层轮廓特征点的匹配, 将复杂轮廓线分割为若干简单的曲线段, 重构的曲面由这些分片构造的三角片曲面拼接而成. 实验表明该算法能够找出复杂轮廓线上恰当的对应该特征点, 从而构造出较真实的曲面.

关键词: 链码; 串匹配; 表面重构; 局部形态特征

中图分类号: TP391. 41