Simplex-Mesh Based Surface Reconstruction and Representation of Tubular Structures

Alexander Bornik, Bernhard Reitinger and Reinhard Beichel

[1] Institute for Computer Graphics and Vision,
Graz University of Technology, A-8010 Graz, Austria
Email: bornik@icg.tu-graz.ac.at

Abstract. Modelling and reconstruction of tubular objects from graph representations is frequently needed in medical visualization. The constructed geometrical models need to be consistent and compact at the same time for interactive applications. In this paper we present a new method for generating compact topologically consistent 2-manifold surfaces of branching tubular objects. The proposed two-stage approach based on deformable simplex-meshes can handle higher order furcations.

1 Introduction

Many anatomical structures like vessels or airway segments are of tubular type. Technically speeking arrangements of such structures like the portal vein tree inside the liver or airway trees are highly complex and heavily branched objects. New algorithms are able to detect even very thin tubular structures [1] in e.g. CT images, leading to graph representations, which are not directly suitable for high-quality visualization in medical applications such as surgical planning.

Rendering of boundary representation is well supported by graphics hardware. However, one has to ensure that the surface model is consistent. There must not be gaps or self-intersections in the model. In case the model is a polygonal or triangular mesh, it should be a 2-manifold. The mesh must be adaptive to account for changing tube radii. The overall primitive count should be small, while still accurately representing small geometrical details.

The presented work was done in the context of a research project with the goal to develop a virtual liver surgery planning system [2]. The planning system provides surgeons and radiologists with tools to carry out planning and staging of liver tumor resections in a virtual reality (VR) environment. This also involves visualization of surface models of the portal vein inside the liver. The graphical representation in the VR system needs to be compact in order to meet the real-time requirement, while still providing the physicians with the correct topological information needed for planning a surgical intervention.

2 Related Work

The probably most popular surface reconstruction algorithm is the Marching Cubes (MC)[3], which is frequently used in the medical context. However, it relies on input data given as a hexahedral grid. It is not adaptive and produces complex geometry. For VR applications mesh compression must be employed, introducing processing overhead. Staircase artifacts produced by MC must be removed using filtering techniques. Connectivity problems for 26-connected structures, which are not acceptable in surgical planning, require a modified algorithm [4]. Besides the MC there are generic algorithms involving deformable models directly [5], which is time consuming for tubular objects. A second class of algorithms involves surface tiling from cross-sections and algorithms connecting tubular objects to a single model [6, 7, 8]. Most of these algorithms have problems representing furcations of higher order correctly, some do not support changing radii along branches. An algorithm related to our can be found in [9], however the technical solution is different. Since the approach is not based on a deformable model, mesh refinement is not possible. An approach mainly focussing on rendering performance, not mesh consistency, can be found in [10].

3 Methods

The proposed algorithm generates compact surfaces representations of branched tubular objects. It can handle bifurcations, trifurcations and higher order furcations. The output mesh is topologically consistent, meaning that the output is a single 2-manifold mesh. Geometrically our approach is based on a simplex mesh structure [5]. The mesh for the tubular object is constructed in two stages:

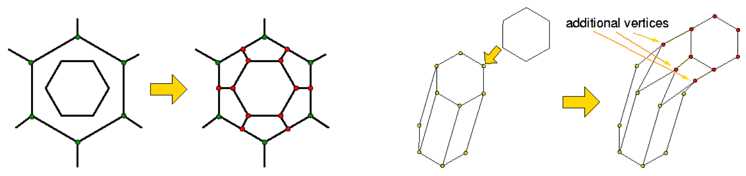
1. *An initial mesh is constructed* connecting cross-section polygons along the input graph. This step produces a single 2-manifold mesh.
2. *Mesh Refinement* based on simplex meshes as deformable model, improves the visual quality of the produced mesh, especially in the vicinity of furcations.

3.1 Input Data and Preprocessing

We use a directed acyclic graph data structure as input data for mesh initialization. The nodes of the graph are either branching points or endpoints. Each node stores its location, local radius, cross-section normal and connected links. Links (branches) store their endpoints, direction and a list of intermediate centerline points with cross-section radii and normals. This data is provided by a vessel-mining approach [1]. Since a deformable model is employed in the second stage of our algorithm, limiting to circular cross-sections during the initialization is possible without loss of generality.

There is no restriction on the radii. Consequently, spheres spanned by neighboring centerline points and radii can intersect. Sometimes a whole child branch is located within the circumscribed sphere of its parent. Our algorithm requires input data to be free from such cases. Preprocessing by thinning leads to initial meshes almost free of visual artifacts, since possibly overlapping cross-sections are removed. Thinning also makes initial meshes compact, because the number of remaining cross-sections directly influences the number of mesh polygons.

Fig. 1. (a) Polygon insertion in 2D (b) Polygon insertion in 3D: additional vertices are placed at the barycenter.



(a) (b)

3.2 Initial Mesh Setup

The initial mesh is set up by traversing input data in a breadth-first-search manner starting with a simplex mesh cylinder at the root node.

Connection of cross-sections. Further cross-sections of the root branch are connected to the existing geometry at the top polygon. This is achieved by connecting each vertex of the cross-section polygon to the existing geometry as shown in Figure 1(a). For maximum visual quality, polygons are oriented so that the overall connection distance is minimized. The visual quality can be improved by placing the additional vertices at the barycenter of the triangles formed by the base edge of the polygon to connect to and the connecting cross-section vertex. Figure 1(b) shows a 3D example.

Connection of branches. When starting a new branch the optimum polygon is found among a number polygons located in the vicinity of the the branching point, the *hot spot region* polygons. When a polygon is to be connected to a branching point, normal vectors for all the polygons of the hot spot regions are calculated using a singular value decomposition, since simplex mesh polygons are not necessarily planar. The best polygon is found maximizing a criterion based on the distance between the inserted cross-section and the hot spot region polygons and the corresponding normal vectors. Figure 2 shows the branch connection operation in detail.

3.3 Mesh Deformation

The overall visual quality of the mesh and particularly the quality near high order furcations is improved using the simplex mesh as a deformable model. Each vertex is displaced based on internal regularizing forces and external forces pulling the mesh towards some boundary. Since deformation can cause degenerated polygons, simplex meshes provide operations to restructure themselves according to different criteria by splitting or merging polygons. More details on simplex meshes can be found in [5]. External forces are calculated either towards sampling points defined by the cross-section radii of the input graph or, if

Fig. 2. (a) Simplex mesh of a tubular objec before connecting a new branch. The hot spot region is marked bright. (b) Mesh after the new branch has been added.

Table 1. Results for test datasets: $\#b$: number of graph branches. $\#cs$: total number of cross-sections. $\#vtx$, $\#pol$: number of vertices/polygons. t_i: mesh initialization time in milliseconds. t_r time for one iteration in the refinement stage. n: number of iterations performed for the images in Figure 3.

DS	Description	#b	#cs	#vtx	#pol	t_i	t_r	n
1	Artificial	8	64	950	477	28	25	40
2	Human portal vein	693	2680	24199	48394	527	1738	10

a voxel-based segmentation is available, towards boundary voxels. Several iterations result in energy-minimizing geometrical changes, mainly near branching points. The visual quality increases.

4 Results

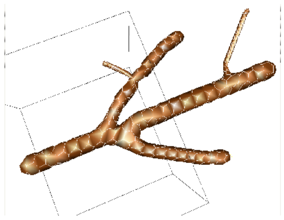
We have tested our algorithm with different types of medical datasets, including the vasculature of the liver, sheep and human airway trees and data from high-resolution scans of portal vein corrosion casts. Our approach produced consistent and visually appealing meshes in all cases. The number of polygons produced by our approach is the same order of magnitude as output produces by a MC approach followed by mesh filtering and simplification. Figure 3 and Table 1 show results including mesh complexity and timings for two datasets.

5 Conclusion

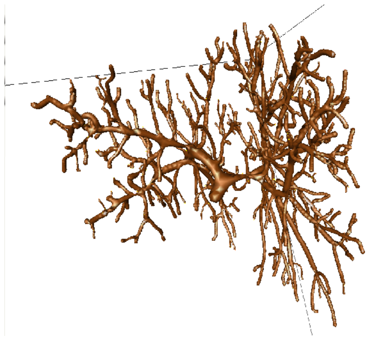
This paper presented an algorithm for generating compact consistent 2-manifold meshes of branched tubular objects based on contour-connection using simplex meshes. The visual quality of the output mesh is improved in an optional refinement step.

This work was supported by the Austrian Science Foundation (FWF) under grant P17066-N04.

Fig. 3. Results for the test datasets from Table 1: (a) Test dataset 1: Small artificial tubular dataset. (b) Test dataset 2: Human portal vein tree.



(a)



(b)

References

1. Beichel R, Pock T, Janko Ch, et al. Liver segment approximation in CT data for surgical resection planning. In: Fitzpatrick JMichael, Sonka Milan, editors. In SPIE Medical Imaging '04. vol. 5370. San Diego: SPIE Press; 2004. p. 1435–1446.
2. Bornik A, Beichel R, Reitinger B, et al. Computer Aided Liver Surgery Planning: An Augmented Reality Approach. In: Jr RLGalloway, editor. In SPIE Medical Imaging '04. vol. 5029. SPIE Press; 2003. p. 395–406.
3. Lorensen WE, Cline HE. Marching cubes: A high resolution 3D surface construction algorithm. Computer Graphics 1987;21(4):163–169.
4. Lewiner T, Lopes H, Vieira AW, Tavares G. Efficient Implementation of Marching Cubes: Cases with Topological Guarantees. Journal of Graphics Tools 2003;8(2):1–15.
5. Delingette H. General Object Reconstruction Based on Simplex Meshes. International Journal on Computer Vision 1999;32(2):111–146.
6. Oliva JM, Perrin M, Coquillart S. 3D Reconstruction of Complex Polyhedral Shapes from Contours using a Simplified Generalized Voronoi Diagram. Computer Graphics Forum 1996;15(3):397–408.
7. Meyers D, Skinner S, Sloan K. Surfaces from contours. ACM Trans Graph 1992;11(3):228–258.
8. Lluch J, Vivó R, Monserrat C. Modelling tree structures using a single polygonal mesh. Graphical Models 2004;66(2):89–101.
9. Felkel P, Wegenkittl R, Bühler K. Surface Models of Tube Trees. In: Proceeding of the Computer Graphics International CGI'04; 2004.
10. Hahn HK, Preim B, Selle D, Peitgen HO. Visualization and interaction techniques for the exploration of vascular structures. In: Proceedings of the conference on Visualization '01. IEEE Computer Society; 2001. p. 395–402.