Hybrid Model for Vascular Tree Structures

Anna Puig, Dani Tost, and Isabel Navazo

Software Department, Avda. Diagonal, 647, 8 floor, 08018 Barcelona, SPAIN,
`anna@lsi.upc.es`,
WWW home page: `http::://www.lsi.upc.es/~anna/`

Abstract. This paper proposes a new representation scheme of the cerebral blood vessels. This model provides information on the semantics of the vascular structure: the topological relationships between vessels and the labeling of vascular accidents such as aneurysms and stenoses. In addition, the model keeps information of the inner surface geometry as well as of the vascular map volume properties, i.e. the tissue density, the blood flow velocity and the vessel wall elasticity.

The model can be constructed automatically in a pre-process from a set of segmented MRA images. Its memory requirements are optimized on the basis of the sparseness of the vascular structure. It allows fast queries and efficient traversals and navigations. The visualizations of the vessel surface can be performed at different levels of detail. The direct rendering of the volume is fast because the model provides a natural way to skip over empty data. The paper analyzes the memory requirements of the model along with the costs of the most important operations on it.

1 Introduction

Vascular diseases represent a 10% of the clinical examinations. Current diagnosis methods are based on vessel images obtained by: X-Rays, DSA (Digital Subtraction Angiography), contrasted CT (Computer Tomography) or MRA (Magnetic Resonance Angiography). From these images, physicians must mentally reconstruct the 3D shape of the vessels in order to detect lesions, such as stenoses and aneurysms. This is a difficult task because the brain vascular system has a complex tree-like structure, and the vessels are small, narrow and sparse in comparison to the surrounding volume. The reconstruction and visualization of blood vessels three-dimensional model from a set of slices provides better means of diagnosing and treating vascular pathologies.

Previous papers on this topic address three main approaches:

– Direct visualization of the data, generally performed with the Maximum Intensity Projection (MIP) [1],
– Blood vessels surface extraction using Marching Cubes (MC) [2] and Dividing Cubes (DC) [3] algorithms,
– Construction of a topological description of the vascular system (symbolic model) [4, 5, 6].

The former approach does not require any pre-processing. Even the segmentation, which is required for most operations on MRA data [7] can be avoided with MIP projection. However, MIP lacks depth perception, and therefore it produces ambiguities in

the overlapping of vessels, diameters reduction and loss of the smallest vessels. Depth cue can be added to the rendering [8] but, for clinician's use, it is generally necessary to compute in batch several different MIP views and record them as a film. Other visualization strategies require a good segmentation [9] and they are generally too slow for routine diagnosis, although they can be speeded-up by using hardware 3D texture mapping [10].

The main drawback of the second approach is that due to the nature of the vessels, the surface model is composed of a huge amount of tiny faces and the connectivity is difficult to guarantee.

This paper addresses the third approach: the construction of a symbolic model of the vascular tree providing information of the medial line of the vessels and of their diameters. This strategy provides a better understanding of the structure than the two former ones. In addition, it may allow the generation of a simpler and smoother surface model. However, the automatic construction of such a model is complex, because it requires the medial line extraction and diameters computation.

Several symbolic models have been previously proposed. In [4], individual vessels are modeled as a set of cylinders joined onto common circular sections. The model is extracted manually. Its main drawback is that it does not provide a global representation of the topological relationships between vessels. In [5] the vascular structure is represented as a symbolic tree, composed of numbered branches connected at bifurcation nodes, where the diameters of the vessels and the axes orientations are stored. The model computed in [6] is a tree of spline curves marking the center lines of the vessels, plus a sequence of cross-section contours perpendicular to the axis. A smooth surface model composed of tensor product rectangular patches can be fitted on this symbolic model. Both models ([5], [6]) can be constructed automatically. As a drawback, they do not provide a specific representation of aneuryms and stenoses. In addition, they do not give simultaneous information of the surface geometry and the internal property values. The hybrid model presented herein is intended at filling this gap. It is first described in section 2 and next evaluated in section 3 in terms of memory requirements and efficiency of its visualizations.

2 Model representation

The proposed model of the vascular structure provides three layers of information: the global structure of the map (the set of topological relationships between the different components of the map), the vascular surface (the global shape of the internal vascular wall) and the inner volume (the inner properties of the vessels). The volume properties are the density and the blood velocity, as all other relevant physical properties can be defined from them. The blood pressure, for instance, can be derived from the velocity and the vessel diameter and the elasticity can be computed from the density.

2.1 Global structure

Since the physical vascular structure is tree-like, an abstract graph with nodes representing branchings and edges representing vessels presents itself as the obvious choice

for the logical definition of the vascular area. The graph is cyclic because the vascular map can present closed paths of nodes, such as the Circle of Willis into the brain, i.e. a particular cerebral arteries network. In addition, the graph is undirected because an explicit relationship of order between nodes does not exist. The flow direction is not represented as the graph direction because in particular regions of the vein system and under specific circumstances, the flow may reverse its direction. Finally, although some vascular pathologies may produce disconnections of the vascular structure, the cerebral map is generally connected. Herein, each connected structure is considered separately as a connected graph.

The edges and the nodes of the graph are composed of a set of labeled segments (see Figure 1). Three different types of vessel segments (features) are considered: normal segments, stenoses and aneurysms. This classification eases the visual detection of accidents, and, as explained later, it reduces the memory requirements of the surface model. All the nodes and the edges segments of the graph are linked with surface information and property values, as described in the next section.
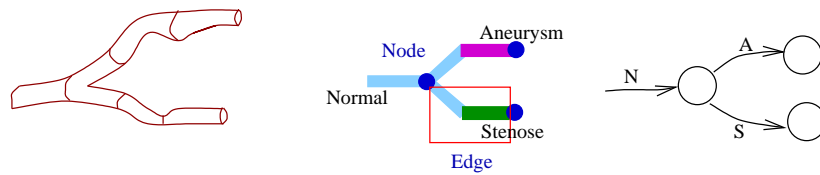

Fig. 1. Representation scheme

As well known, there are two common ways to implement graphs [11]: sequential and linked representations. The simulations performed in this paper are implemented on the basis of the linked representation.

2.2 Surface representation

A natural representation of the inner surface of a blood vessel segment is by sweeping contour cross-sections, generally circles, along a trajectory. The shape of the vessels is such that there are no intersection between contours and its medial line does not self intersect.

Therefore, the surface model is defined as the union of generalized cylinders [12] which represent each segment of the graph. This proposed model fulfills the following properties:

– The skeleton curve $SKLT(t)$, i.e., the sweep trajectory, is the medial axis of the vessel segment. It is a linear approximation of the curve segment which guarantees that every approximated point has a distance less than K to the curve. Different representations of the surface can be obtained varying the value of K. However, to simplify the data structure, in the implementation, K is assumed constant.

– Each section of the generalized cylinder is defined by a Frenet reference Frame which defines a coordinate system based on three orthogonal unity vectors (see Figure 2): a tangent vector $T(t)$, a normal vector $N(t)$ and a bi-normal vector $B(t)$.
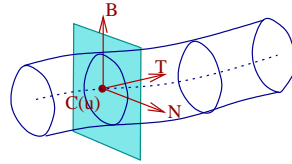

Fig. 2. Frenet Frame illustration.

Thus, the normal plane $L(t)$ at a point t of the skeleton $SKLT(t)$ can be defined with the vectors $N(t)$ and $B(t)$, and the vectors $B(t)$ and $T(t)$ of the normal plane specify the orientation of the curve.
– In each section, the contour curve $C(v)$ of vessel is approximated as a set of points on the normal plane $L(t)$. This approximation allows different levels of precision in the reconstruction from a simple polygon, a circle, an ellipse, to a spline [13]. In regular segments, the contour representation is implicit because the regular sections are circles and thus, it is enough to store their diameters. Cross sections in stenoses, aneuryms and branchings, generally more complex, are represented explicitly.
– It is sufficient to store sections only at points of the skeletal curve where a significant change of curvature or diameter exists and at end-points of the different features. In the regular segments, as the diameter of the cross section varies linearly, only one diameter per cylindrical segment and two diameters per conical segment need to be stored.
– There is a unique correspondence between the points of two consecutive sections. In the branching nodes, this correspondence is based on the criteria defined in [14] and [15].
– There is no blending between two consecutive cross-sections, as vessels do not present twists in their shape. Therefore a linear interpolation between section suffices [12].

The surface model distributed into each element of the graph structure is defined as the clipping region of the generalized cylinders against the node spatial region. Then, the surface data attached to a node is the skeletal curves and the set of end contour curves of the anatomical features which are concatenated in the node.

This surface representation enables the evaluation of the geometry of the boundary model with different levels of detail. The simplest evaluation is a union of generalized cylinders between nodes. A smoother one consists in polygonalizing the surface by tiling between contour curves ([14], [15]). Finally, it can be approximated by tensor product spline surfaces. This property of the model makes it suitable to the variety of operations that must be realized on it. Some operations, such as fluid simulations, require only simple cylindrical approximations while other ones, such as the navigations, need smoother surfaces.

2.3 Volume representation

The volume data represent the inner properties of the vessels, i.e. the density and the velocity. The most usual way of representing volume properties is the voxel model. This model allows direct access to a point, given its coordinates, and ordered traversals according to a coordinate direction. On the contrary, it does not provide a direct traversal of the vascular graph structure. From the occupancy point of view, it has high memory requirements. Even though, considering the sparse structure of the vessels, the voxels corresponding to the vascular map occupy only around a 2% of the model.

An alternative model is a compression of the voxel model using a run-length encoding [16], [17]. It is a suitable model for low occupancy data sets, especially if the original data is full of long constant property values sequences. A first difficulty with this coding is that the direct access to a voxel must be performed as a search on the run-length structure. [16] proposes to use additional indexed tables with pointers to the first voxel of each slice and each row in order to reduce the search to a column length. A second difficulty with a run-length encoded volume is that the volume can only be traversed optimally in the order in which the voxels are encoded. Thus, only a coordinate axis traversal of the model is optimal. Again, [16] solves this problem by precomputing three run-length encoding, one for each of the three coordinate axes. Finally, as the run-length is a spatial enumeration scheme, the volume traversal according to the topological order is, as in the voxel model, computationally expensive. This is a major drawback because topological traversals of the graph are needed in blood flow simulations as well as navigations.
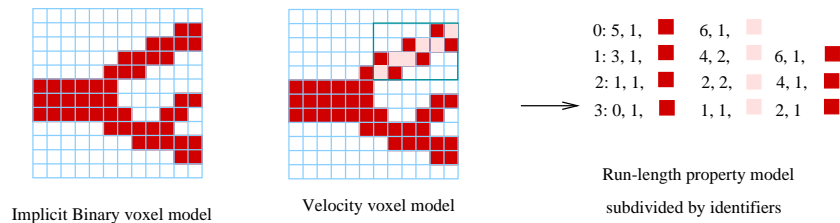

Implicit Binary voxel model   Velocity voxel model   Run-length property model subdivided by identifiers

Fig. 3. Run-length encoding of a property of the original model.

To solve the latter problem the volume representation proposed herein keeps separate volume representation of all the segments of the graph elements (features). This allows a random access to the features, as required in navigations, while taking benefit of the spatial ordering of each separate voxel model. Specifically, the volume representation of each node and segment is a subset of the original voxel model enclosing the bounding box of the feature, such that only the voxels belonging to the feature have a property value (see Figure 3). This representation involves overlapping of spatially close features of the graph although the property values of vascular voxels are stored only once in the union of all these subsets. The volume models of each feature are coded as run-length according to the ownership of each voxel to the feature. It allows direct accesses to the voxel data of each feature, as well as ordered traversals according to

a coordinate direction. Empty voxels of the overlapping regions are no coded several times. It takes advantage of the spatial structure of the voxel model. Thus, it allows a direct access to the property values of any voxel of a segment or a branch.
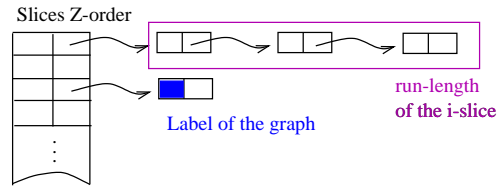

Fig. 4. Run-length encoding of the voxel model in relation to the features of the model.

Keeping the original voxel model in addition to the features volume schemes is redundant and unaffordable in most applications environment because of the memory limitations. Thus, the initial voxel model is segmented and it is compacted by a run-length encoding method on the basis of the vessel ownership property value of each voxel (see Figure 4). This model is necessary to keep the implicit spatial order of the voxel model and to compress the sparseness of the data, without loosing the direct access of a voxel.

2.4 Management of the three information layers

The main characteristic of the three layers model is that it provides simultaneous access to the topology, the surface and the properties. These different queries are managed as follows:

– All the accesses from the graph to the surface (**graph-to-surface**) and to the volume data (**graph-to-volume**) are direct because the surface and the voxels are stored into the corresponding features of the graph. These queries are necessary for the topological traversals of the structure, particularly in navigations and flow simulations.
– In order to establish the relationship between a voxel and a feature of the graph (**voxel-to-graph**), each voxel stores a simple identifier of the segment of the graph to which it belongs. This identifier defines a new volume property of ownership of each voxel to a segment of the vascular structure.
– The access **voxel-to-surface** is solved implicitly. To know if a voxel is traversed by the vessel surface, the feature to which the voxel belongs can be accessed. Next the surface of the feature should be evaluated. This can be done in several ways: at low surface resolution (cylinder test), with more complex test (B-splines test). A different approach consists in simply exploring the set of neighbor voxels and analyzing the gradient value. This latter solution is the classical one used in simple volume models.
– The access **surface-to-voxel** can be performed directly with a given surface point. The voxel to which the point belongs is directly retrieved from the point coordinate values and then, from the voxel, the identifier of the segment of the graph is found (**surface-to-graph**).

3 Evaluation

In the following sections, the occupancy and efficiency of the model are analyzed in terms of theoretical and experimental terms.

All simulations have been performed on a Sun Sparc, Ultra-1 with Solaris 5.5.1. Five data sets have been used. Three data sets are phantom data of different sizes. Two medical data sets have been tested: a MRA data and a SCTA (Spiral Computed Tomography Angiography) data sets. A segmentation preprocess is performed before building the model.

3.1 Occupancy

Let n^3 be the number of voxels of the original data set, and n_b^3 the number of voxels of the bounding box. Let n_v the number of voxels which contains blood vessels. n_v is $0.02 \times n_b^3$. Let n_f be the number of total features of the vascular data with a mean size of n_{ss}^3 voxels.

A classical voxel model of the whole volume requires $28 \times n^3$ bytes (1 integer for the density value and 3 floats for the velocity value at each voxel). A run-length compression of the boundary box representation needs $28 \times 0.02 \times n_b^3 + 6 \times 4 \times n_b^2$ bytes, where the last adder is the occupancy of the three tables ordered according to the three main axis respectively and assuming that the number of features can be coded with one integer. If a double structure of volume data is stored in order to keep both spatial and ordered traversal, the run-length model of the whole data, according to the identifier of the features, takes $0.02 \times n_b^2 \times 8 + 6 \times 4 \times n_b^2$, where the first adder is a coarse approximation of the real cost $n_b^2 \times (log_2 n_f + log_2 n_b)$ bits of a general case of run-length.

In addition to the primary model (i.e. the run-length of labeled voxels) storage requirements, a secondary model for each feature must be coded. If these secondary models are coded with boundary boxes, the mean global occupancy is $28 \times n_f \times n_{ss}^3$ bytes, whereas if these secondary models are run-length code according the ownership of the feature, the memory needed is $28 \times 0.02 \times n_b^3 + 6 \times 4 \times n_{ss}^2$ bytes.

The memory space required to store the graph and the surface is the following. First, the occupancy of the graph is $2 * e$, where e is the number of the edges between branchings. In addition, if each feature has, in mean, three control points and the number of points for each contour is at most 20, the surface representation needs $20 \times 3 \times n_f \times 16$ bytes.

Table 1 shows the memory requeriments of the symbolic model of different input data sets, as well as, the memory requeriments of the classical voxel model and of the mesh of triangles model.

The reduction of the memory requirements ranges from 37.74% to $95,85\%$. The maximum efficiency is achieved when the occupancy of the data is low and when the distribution of the data in the volume is more concentrated. This is the case of the model MRA which represents blood vessels. By opposite, model Phantom 3, which represents a skull has also a relatively low occupancy but the data are spread over the whole model. Therefore the compression rate is lower. Finally SCTA produces an overhead of memory requirement because the data, interleaved surface layers, occupy the whole original

volume. Summarizing, the model proves to be efficient for sparse and concentrated structures.

Table 1. Comparison of memory requeriments with different set of data.

Data set	Original resolution	Occupancy	Voxels size (Kbytes)	Model size (Kbytes)	Memory reduction
Phantom1	32x32x32	12.72%	32.768 Kb	20.400 Kb	37.74%
Phantom2	64x64x27	8.0%	110.592 Kb	113.484 Kb	102.61%
Phantom3	96x96x69	0.7%	635.904 Kb	129.308 Kb	79.66%
SCTA	128x128x42	21.0%	9633.792 Kb	1597.584 Kb	83.41%
MRA	256x256x124	0.2%	8126.464 Kb	640.080 Kb	92.12%

3.2 Efficiency

The efficiency of the model is next measured in terms of the costs of the visualizations. These costs do not account the pre-process of model construction, which involves a first segmentation step, the medial axis extraction, its processing to automatically detect its features and the construction of the data structures ([18]). The run time of the construction algorithm depends on the model occupancy and its original size. It ranges from seconds for small models to minutes for the SCTA and the MRA data sets.

Figures 5 and 6 show a simplified visualization of topology of the model. Edges segments are approximated by polylines with a circle contour curve at the significant points of the axis. Branching are shown with spheres. This visualization, very fast, is used to manage the user interface of the vascular model and to render the topological relations between different blood vessels.

The visualization of the vascular surface can be performed at different levels of detail such as: simple cylinders at the segments and spheres at the nodes, polygonal approximation of the surface by interpolating contour curves between the curves stored in the model and tiling between curves or by fitting biquadratic splines parameterized with rectangles between successive contours [6].

Figure 7, shows two levels of refinement of the same vessel segment. The first image uses 5 points per contour and no interpolation between the contours stored in the model. The second image uses 20 points per contour and it interpolates 5 contour curves per pair of stored contours. The number of faces is about 100 and 900 respectively. These different levels of detail are useful when the whole model is visualized, and also in navigation where the level of refinement can be computed in relation to the distance of the feature from the observer. Figure 8 shows an image of a walkthrough sequence. The navigation takes advantage of the topological model by clipping nodes and edges that are outside the viewing frustum. Experimentally, the number of edges and nodes to be render simultaneously in a navigation inside the vessels is almost constant, about 4 features. The navigation is therefore much more faster than a complete projection of the model.

The visualization of volume model may be performed by slice-to-slice composition, by MIP projection and by semi-transparent shaded rendering. As an example of the shaded rendering, Figure 9 shows a BTF (Back-To-Front) visualization. Figure 10 shows several boundary boxes of the edges of the graph. Each voxel has a different color depending on the feature to which it belongs. It can be seen that although the boxes may overlap, every vascular labeled voxel belongs only to one box.

Table 2 shows the performance of the volume visualization algorithm, based on the run-length encoding in comparison to full voxel models visualization. The efficiency depends on the level of compactness of the run-length. Results show that the cost decreases a lot in MRA data sets, providing real time rendering. Table 3 compares the cost of a walkthrough driven by the topological model versus a direct rendering of the whole data structure at each frame. The same reasoning as for the surface walkthrough can be done: the approximate number of features visualized per frame is constant and small.

Table 2. Performance of the volume BTF-visualization algorithm.

Data set	Original resolution	Voxels size (Kbytes)	Run time voxels	Run-length size (Kbytes)	Run time run-length
Phantom1	32x32x32	32.768 Kb	0.4 sec.	15.992 Kb	0.127 sec.
Phantom2	64x64x27	110.592 Kb	1.48 sec.	108.444 Kb	0.36 sec.
Phantom3	96x96x69	635.904 Kb	2.62 sec.	110.588 Kb	0.17 sec.
SCTA	128x128x42	9633.792 Kb	7.50 sec.	384.676 Kb	2.61 sec.
MRA	256x256x124	8126.464 Kb	28.68 sec.	487.280 Kb	0.38 sec.

Table 3. Performance of the volume walkthrough algorithm.

Data set	Original resolution	Projection whole data	Walkthrough model
Phantom1	32x32x32	0.127 sec.	0.127 sec.
Phantom2	64x64x27	0.36 sec.	0.25 sec.
Phantom3	96x96x69	0.17 sec.	0.045 sec.
SCTA	128x128x42	2.61 sec.	0.304 sec.
MRA	256x256x124	0.38 sec.	0.18 sec.

4   Conclusions and future work

The main contribution of this paper is the proposal of a representation scheme for blood vessels. It is a multiresolution hybrid model, which stores symbolic information on

the topology and geometry as well as volume and surface data. These characteristics provide flexibility and they enable the realization of all the physicians requirements in a Computed Assisted Neurovascular System.

The model has been implemented and tested with different visualization strategies. Results show that for MRA data a compression rate of 90% can be achieved. Visualizations reduce impressively their cost, especially navigations driven by the topological model.

Work on progress is the integrated visualization of the blood flow velocity with the vascular surface. Another question is how to improve the control of the refinement level by automatically tuning the parameter, which determines the number of control points stored in the coding of the axis model.

References

[1] A. Savopoulos G. Sakas, M. Grimm. Optimized maximum intensity projection (mip). *6th Eurographics Workshop on Rendering*, pages 81–93, June 1995.

[2] G. Gerig, R. Kikinis, and F.A. Jolesz. Image processing of routine spin-echo mr images to enhance vascular structures: Comparison with mr angiography. *3D Imaging in Medicine: Algorithms, Systems and Applications*, pages 121–132, 1990.

[3] H.C. Cline, W.E. Lorensen, R. Kikinis, and F. Jolesz. Three-dimensional segmentation of mr images of the head using probability and connectivity. *Journal of Computer Assisted Tomography*, 14(6):1037–1045, November/December 1990.

[4] C. Barillot, B. Gibaud, J. Scarabin, and J. Coatrieux. 3d reconstruction of cerebral blood vessels. *IEEE Computer Graphics and Applications*, 5(12):13–19, December 1985.

[5] C. Zahlten, H. Jurgens, and H.O. Peitgen. Reconstruction of branching blood vessels from ct-data. *Proceedins of Rostock, Eurographics Workshop on Visualization in Scientific Computing*, 1994.

[6] H.H. Ehricke, K. Donner, W. Koller, and W. Straßer. Visualization of vasculature from volume data. *Computer and Graphics*, 18(3):395–406, 1994.

[7] D. Vandermeulen, R. Verbeeck, L. Berben, D. Delaere, P. Suetens, and G. Marchal. Continuous voxel classification by stochastic relaxation: Theory and application to mr imaging and mr angiography. *Internal Research Report, Medical Imaging Research Lab, ESAT, Belgium*, 1994.

[8] W. Heidrich, M. McCool, and J. Stevens. Interactive maximum projection volume rendering. *Proceedings of the IEEE Visualization 95*, pages 11–18, October 1995.

[9] K.H. Hohne, M. Bomans, A. Pommert, M. Riemer, C. Schierds, U. Tiede, and G. Wiebecke. 3d visualization of tomographic volume data using the generalized voxel-model. *The Visual Computer*, 6(1):28–36, February 1990.

[10] K.J. Zuiderveld. Vr in radiology - first experiences at the university hospital utrecht. *ACM Computer Graphics*, pages 47–48, November 1996.

[11] A.V. Aho, J.E. Hopcroft, and J.D. Ullman. *Data Structures and Algorithms*. Addison-Wesley, 1983.

[12] M/S. Kim, E.J Park, and H.Y. Lee. Modelling and animation of generalized cyllinders with variable radius offset space curves. *The Journal of Visualization and Computer Animation*, 5:189–207, 1994.

[13] T. Maekawa, N.M. Patrikalakis, T. Sakkalis, and G. Yu. Analysis and applications of pipe surfaces. *Computer Aided Geometric Design*, 1997.

[14] J.D. Boissonnat. Surface reconstruction from planar cross section. *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 393–397, 1985.

[15] K. Anjyo, T. Ochi, Y. Usami, and Y. Kawashima. A practical method of constructing surfaces in three-dimensional digitized space. *The Visual Computer*, 3(1):4–12, February 1987.

[16] P. Lacroute. Fast volume rendering using a shear-warp factorization of the viewing transformation. Technical report, Departments of Electrical Engineeering and Computer Science, Stanford University, Setember 1995.

[17] G.T. Herman and H.K. Liu. Three-dimensional display of human organs from computed tomograms. *Computer Graphics and Image Processing*, 9(1):1–21, 1979.

[18] A. Puig. *Contribution to Volume Modeling and to Volume Visualization*. PhD thesis, Software Department, Universitat Politecnica de Catalunya, Barcelona, Spain, October 1998.
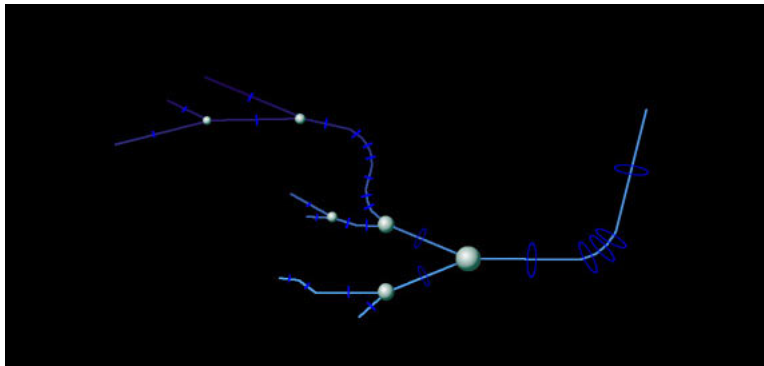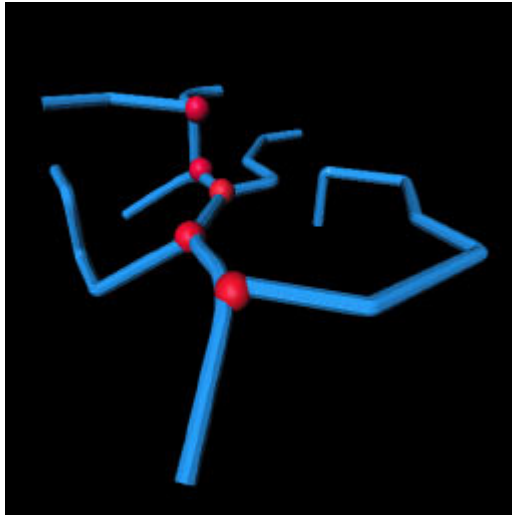
Fig. 5. Symbolic Model Visualization: Polylines.
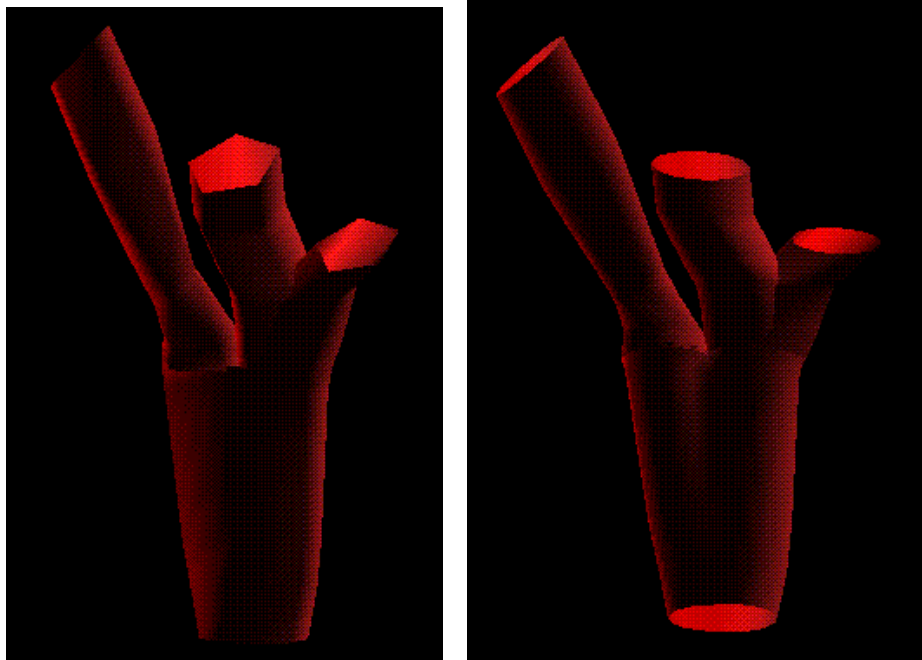
Fig. 6. Symbolic Model Visualization: Pipes and spheres.
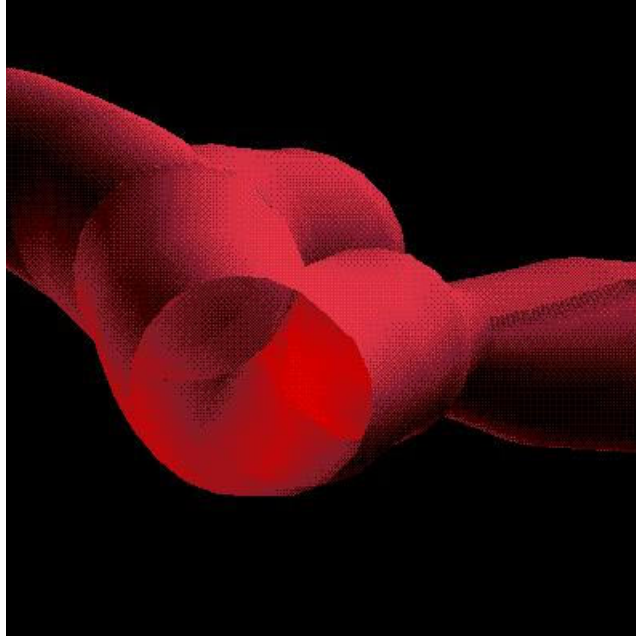

Fig. 7. Surface Visualization.

Fig. 8. Inner Surface Navigation.

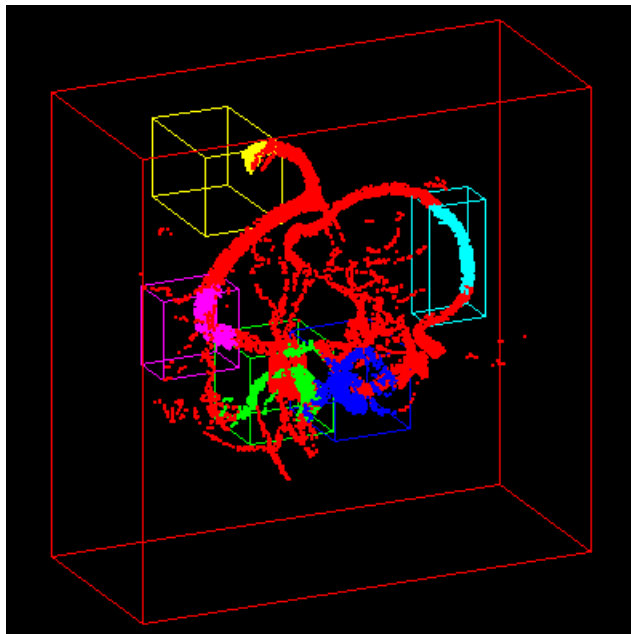**Fig. 9.** Shaded volum visualization with a BTF-method.

Fig. 10. Volume features visualization with a BTF-method.