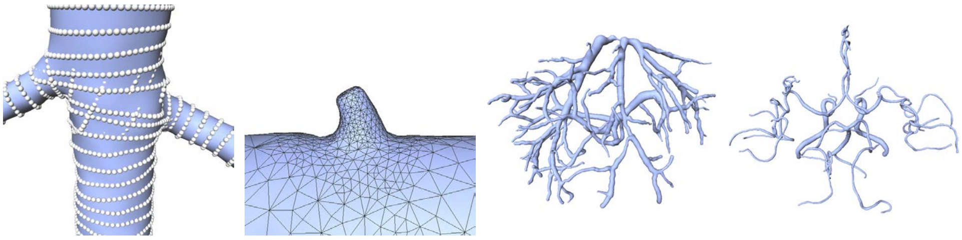
DOI: 10.1111/j.1467-8659.2012.03098.x
Eurographics Conference on Visualization (EuroVis) 2012
S. Bruckner, S. Miksch, and H. Pfister
(Guest Editors)

Volume 31 (2012), Number 3

Reliable Adaptive Modelling of Vascular Structures with Non-Circular Cross-Sections

Jan Kretschmer[13], Thomas Beck[23], Christian Tietjen[3], Bernhard Preim[4] & Marc Stamminger[1]

[1]Computer Graphics Group, University of Erlangen, Germany
[2]Institute for Anthropomatics, Karlsruhe Institute of Technology, Germany
[3]Siemens Healthcare Sector, Computed Tomography, Forchheim, Germany
[4]Department of Simulation and Graphics, Otto-von-Guericke University of Magdeburg, Germany



Abstract

Accurate visualizations of complex vascular structures are essential for medical applications, such as diagnosis, therapy planning and medical education. Vascular trees are usually described using centerlines, since they capture both the topology and the geometry of the vasculature in an intuitive manner. State-of-the-art vessel segmentation algorithms deliver vascular outlines as free-form contours along the centerline, since this allows capturing anatomical pathologies. However, existing methods for generating surface representations from centerlines can only cope with circular outlines. We present a novel model-based technique that is capable of generating intersection-free surfaces from centerlines with complex outlines. Vascular segments are described by local signed distance functions and combined using Boolean operations. An octree-based surface generation strategy automatically computes watertight, scale-adaptive meshes with a controllable quality. In contrast to other approaches, our method generates a reliable representation that guarantees to capture all vessels regardless of their size.

1. Introduction

The visualization of vascular structures is an important issue in many clinical applications. Due to their complex morphology and their vital function, vessels are of particular importance when assessing risks or evaluating different surgical strategies. In the field of oncological liver surgery planning the intrahepatic vasculature is of particular importance, since the functionality of the organ is in large parts dependant on the blood supply. A reliable visualization of *all* vessels is thus crucial for a solid risk assessment. In diagnostic procedures, such as virtual endoscopy, high-quality visualizations, which reliably model the interior of complex furcations, are required. To represent pathologies, accurate surface generation concepts that can handle non-circular vessel outlines are necessary. They are also of great importance in stent planning routines, since they require intersection-free surfaces that allow an inspection of the interior of vessels. Due to their similar morphology, surface visualizations of the respiratory tract are closely related to vascular methods which makes most concepts applicable in both scenarios.

Geometric representations, such as centerlines, constitute a common way to describe anatomical tree-like structures in a compact manner. At sample points along the center of a vessel information about the surface, like radii or samples of the surface outline, is provided. In contrast to segmentation masks, these models explicitly describe the shape *and* the topology of vascular structures. They are widely used for annotation- and classification tasks and provide key information for navigation during fly-through visualizations and cross-section analysis. A common way to derive centerlines is to skeletonize presegmented binary masks [KHH*04], which are visualized using so called *model-free* techniques that interpolate point clouds sampled from the boundary of the segmentation mask. Geometric segmentation algorithms [BB*09] [TdTF*07] [WNV00], however, explicitly avoid the costly generation of these masks. They rely on geometric models or active shape models which are fitted to the vascular surface on the fly. Unbranched vessel segments are often traced separately and composed to define the entire vascular tree. Interpolating the resulting outlines is, however, error-prone since the underlying models may cause segmentation outlines at furcations to partially run through adjacent vessels (see Figure 1). Traditionally, surface generation from centerlines is performed in a *model-based* manner. Unfortunately these methods resort to restrictive assumptions on the shape of cross-sections like circles or ellipses.

We present a reliable adaptive modelling approach for vascular structures (RAMVAS) that efficiently generates surface meshes from general centerline descriptions. It simplifies the reconstruction of vascular trees in their entirety by delegating the description to local segments that only have to provide implicit representations for their local volume. We introduce a novel thin plate spline interpolation-based technique to model smooth surfaces between pairs of expressive free-form outlines. In contrast to previous approaches we generate scale-adaptive meshes *and* guarantee watertight polygonizations at the same time. We present a refinement criterion for octrees that guarantees a topologically reliable extraction of vessels of any radius. This is an important feature, since the representation of *all* vessels is essential for many clinical applications. RAMVAS is independent of the smallest vessel diameter that is present and handles arbitrary variations in vessel size. It is capable of producing reliable low-poly meshes, which allow a fast rendering, as well as high-quality surfaces and is thus applicable in a wide range of scenarios. Our aim is not to smooth error prone segmentations or to improve their quality otherwise, but to represent the information contained in the centerline and the associated surface information as accurately as possible.

2. Related Work

The methods proposed to model vascular surfaces can roughly be categorized to be either *model-based* or *model-free*. In this section we will first review the most important model-free approaches before giving a brief overview over model-based techniques. For an extensive overview on the topic the reader is referred to [WWL*10] and [SOB*07].

2.1. Model-free Mesh Generation

Model-free methods are also referred to as *implicit methods*, since they make extensive use of implicit indicator functions derived from point clouds extracted from a binary segmentation masks. Common choices for interpolants include Radial Basis Functions [ZO03] and poisson surface reconstruction [KBH06]. Schumann et al. [SOB*07] employ Multi-level Partition of Unity implicit surfaces which construct a global implicit surface from locally supported surface patches that are blended together with smooth transitions. Iso-surfaces are usually polygonized using standard Marching Cubes (MC) [LC87] which is bound to a fixed triangle resolution for the whole dataset. This results in an oversampling of large vessels and an unreliable undersampling of small vessels. More recently, curvature adaptive vascular meshes have successfully been generated using advancing front methods [WWL*10]. These techniques, however, lack robustness as they are dependent on well behaved surfaces and cannot guarantee to produce watertight meshes. Moreover, in case of disjoint surface portions every surface part needs to be detected and provided with an entry point for reconstruction. To extract iso-surfaces Kazhdan et al. [KKDH07] describe an adaptive method which is capable of extracting watertight meshes from octrees. Due to the local adaptivity their technique is very well suited for the generation of scale-adaptive surface meshes of complex structures such as vessels.

2.2. Model-based Mesh Generation

Model-based approaches usually rely on the assumption that vessels have near-circular cross-sections. Most of them are categorized as *explicit* since they use centerlines to generate a surface representation in a constructive fashion. An early approach described in [HPSP01] explicitly constructs approximations based on tubular mesh primitives. Unfortunately, this ad-hoc technique leads to highly (self-)intersecting meshes that only provide means for a restricted set of visualization scenarios. Felkel et al. [FKF*02] and Volkau et al. [VZB*05] describe a method where topology is handled while constructing a quadrilateral base-mesh which is later refined by Catmull-Clark subdivisions. The authors state that the algorithm works for "natural" datasets. Self intersections, however, may occur, since their algorithm is sensitive to small angles between branches and the placement of centerline sampling points. Adler et al. [AMMP10] propose a low resolution meshing scheme, where centerlines

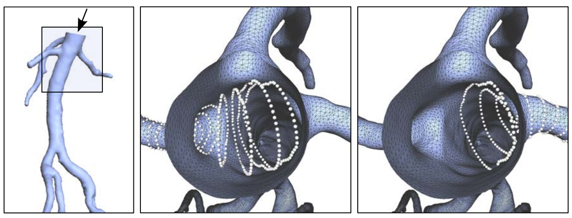


Figure 1: Left: An abdominal and pelvic artery mesh generated with RAMVAS. Center & Right: Interior views from the point of view indicated in the left image. Free-form contours (white dotted lines) of two unbranched vessel sections contributing to the bifurcation are displayed. Since contours may partially lie inside adjacent vessel, a simple interpolation leads to undesirable inner structures.

are polygonized by iteratively extending an initial triangular sphere. In a process similar to region growing, new vertices are generated at faces that are intersected by the centerline and are then projected onto the vessel surface. In [BRB05] simplex meshes are constructed and subsequently deformed by iteratively moving the generated vertices to the nearest boundary voxel. Results are smooth, but a preprocessing stage that explicitly removes overlaps at furcations and revises the centerline topology is needed which effectively discards information. The most elaborate model-based method was presented by [OP05] and is based on Convolution Surfaces (CS). To our knowledge, it is the only model-based approach that can be categorized as *implicit*. Radial information is interpreted as a 1D signal along the vessel, which restricts the approach to circular cross-sections. A global implicit function is then created by blending radii along vessels using a Gaussian filter with finite support.

3. Overview

Generally, the model assumptions and the explicit handling of furcations are the major drawbacks of most model-based approaches. Model-free methods produce good results when generating surface representations from binary masks. Since they are bound to point clouds they are, however, unsuitable for centerline-based visualizations. Figure 1 shows a set of free-form contours and the vascular surface reconstruction computed with RAMVAS. The geometric centerline description was generated using the method from [BB*09], which is able to capture complex vascular shapes. Because of the Active Shape Model used during the segmentation process a considerable part of the segmented contours is located *inside* the vessel lumen. Any model-free method that tries to generate a surface by interpolating these points inevitably produces unusable results. Additionally, converting radial or free-form centerline descriptions into point-clouds discards valuable topological information which can result in the fusion of vessels with close proximity. To prevent this, we introduce a level of abstraction that relies on implicit functions *and* uses the topological information provided with the centerline description.

In contrast to model-based vessel visualization methods which explicitly construct meshes, we employ implicit functions. We decompose the centerline description into segments S_i which are locally described by implicit functions and combine them using Boolean operations to describe vascular surfaces (Section 4). This resolves all intersection- and furcation-related issues independent from their complexity. We then construct an adaptive octree which is converted into a watertight scale-adaptive mesh (Section 5).

4. Admissible Distance Functions

Signed distance functions (SDFs) are a common tool to describe the volume and the surface of an object in a scalar field $d(x)\colon \mathbb{R}^3 \to \mathbb{R}$. The surface of the object is usually defined by the zero level-set $d(x) = 0$, the interior of the object is defined by $d(x) < 0$ and the exterior satisfies $d(x) > 0$. The values of d encode the distance to the surface and the gradient defines the direction that is normal to the surface.

Unfortunately, the analytic description of SDFs quickly becomes complex even for simple geometric primitives. Since we use a MC-based iso-surface extraction algorithm [KKDH07], however, we are able to simplify their definition. For MC to *detect* intersections with a surface described by a zero level-set $d(x) = 0$, the exact distance values are of no importance. The topological configurations of the MC cells exclusively depend on the signs of d at the cell vertices. This enables us to relax the distance property of SDFs by allowing $|d|$ to underestimate the distance to the surface. We explicitly prohibit an overestimation of the distance, since we will later use $|d|$ to define a safe perimeter that is guaranteed to contain no surface portions. Note, that relaxing the distance property does not imply a change of the zero level-set $d(x) = 0$. It leads to the following key properties

1. d encodes the union surface:

$$d(x) = 0 \Leftrightarrow x \text{ on the vascular surface}$$
$$d(x) < 0 \Leftrightarrow x \text{ inside the vasculature}$$
$$d(x) > 0 \Leftrightarrow x \text{ outside the vasculature}$$

2. d does not represent the exact distance to the union surface. It provides, however, an *admissible heuristic* of the signed distance, since it is never overestimated.

We thus refer to these functions as *admissible distance functions* (ADFs). They enable a very compact definition of primitive-based volumes by performing Boolean combinations using min and max operations as described in [FPRJ00]. To model the entire vasculature, we decompose it into segment functions $f_i\colon \mathbb{R}^3 \to \mathbb{R}$ that describe their local volumes using ADFs. To compute the global, intersection-

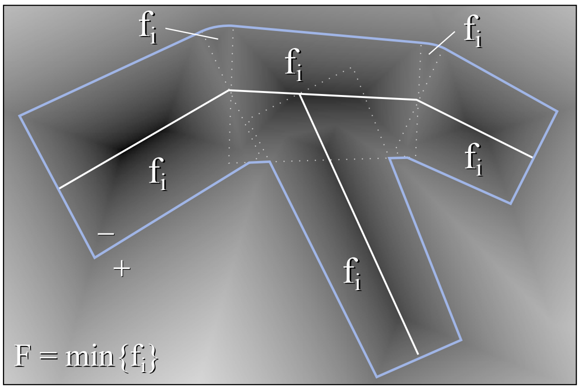


Figure 2: By computing their minimum the local ADFs f_i are combined to form the Boolean union volume F. Note the distance peaks of F inside the union volume.

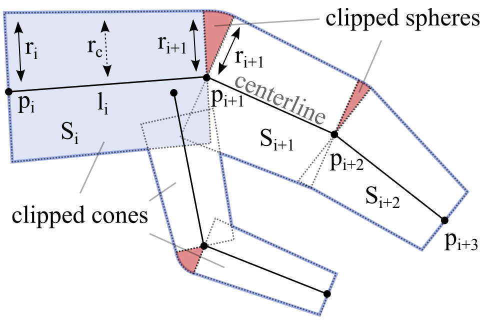


Figure 3: A vessels branch modelled with clipped cones (blue) and clipped spheres (red). One radius r_i is given per centerline node p_i. r_i and r_{i+1} are interpolated along the centerline. Gaps between clipped cones are filled by placing clipped spheres at the inner nodes.

free surface, we form the Boolean union volume as illustrated in Figure 2.

$$F(x) : \mathbb{R}^3 \rightarrow \mathbb{R} = \min_i f_i(x) \tag{1}$$

F may contain distance peaks on the contact surfaces of neighboring segments (see Figure 2) but since we select the minimum of all f_i we receive positive values if $\forall i, f_i(x) > 0$ and negative values if $\exists i, f_i(x) < 0$. This means the sign of F can be used to classify points as to lie inside or outside the union of all segments. Note, that no restrictions are imposed on the realization of the local segment functions themselves. We only require the vascular system to be decomposable into volumes whose union describes the whole tree. No explicit rasterization of the underlying implicit functions is required. They may describe simple analytic primitives or complex interpolated functions. The ADFs f_i are evaluated on the fly, which means that no artifacts are introduced by discretizing the domain of F.

4.1. Radial Vessel Models

Circular vessel cross-sections are a widely used model assumption. They constitute a valid simplification of the vascular outline in many applications. For example, when planning oncological liver interventions the spatial constellation of vessels with respect to lesions is the main aspect of interest while deviations from circular cross-sections are less important.

We choose a representation that uses truncated cones and spheres to model the tubular structures of the vessel tree as shown in Figure 3. Linear centerline segments with two radii r_i and r_{i+1} associated with the end points p_i, p_{i+1} are represented by cones which are clipped at the end of each segment. The radius $r_c = r_{i+1} \cdot t + r_i \cdot (t-1)$ with $t \in [0,1]$ is interpolated along the line segment. The linear segments usually coincide at centerline nodes p_i at a certain angle. When using clipping planes orthogonal to the corresponding line segment l_i a gap in direction of the larger angle at p_i is created (see Figure 3). We close this gap by placing spheres between segments which are clipped against the same planes as the adjacent cones. The clipped cone is described by a Boolean intersection of an infinite tube with two half-spaces. Thus, the local ADF f_i of a clipped cone can be computed as the maximum of three ADFs

$$D_c = \max\{D_{p0}, D_{p1}, D_{tube}\} \tag{2}$$

where D_{p0} and D_{p1} are the signed distances to the clipping planes, each with an outward facing normal and $D_{tube} = D_{centerline} - r_c$ is the distance to the centerline with the interpolated radius subtracted. An ADF of a clipped sphere is analogously described by

$$D_s = \max\{D_{p0}, D_{p1}, D_{sphere}\} \tag{3}$$

where $D_{sphere} = D_{center} - r_i$ is the signed distance to the surface of the sphere. In the vicinity of the surface these distance functions locally behave like SDFs. In the far away regions that are close to the clipping planes, the distance is underestimated. The union volume of all clipped spheres and cones describes the volume of the whole vascular tree and faithfully represents the radial information of the nodes.

4.2. Free-Form Vessel Models

Centerlines with associated free-form contours allow a very precise representation of the vascular geometry. This is particularly important when dealing with pathologies. In our case, the contour C associated with a centerline node is represented by a list of consecutive coplanar points $(q^1, \ldots, q^n)$ which uniformly sample the vascular contour. The number of surface samples increases with the circumference of the vessel. To describe a local segment S_i, we need to provide

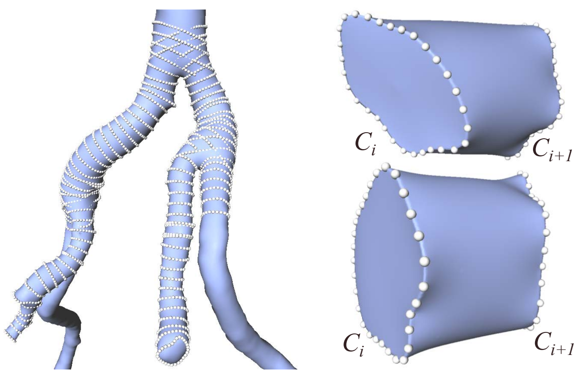


Figure 4: Left: RAMVAS reconstruction of an arterial tree computed form free-form contours. A subset of contours is shown in the image. Right: Reconstructions of two individual segments defined by pairs of contours C_i and C_{i+1}.

an ADF f_i of the volume which is described by two consecutive contours $C_i = (q_i^1, \ldots, q_i^n)$ and $C_{i+1} = (q_{i+1}^1, \ldots, q_{i+1}^m)$. For centerlines with circular cross-sections we interpolate the two end radii r_i and r_{i+1} as described in Section 4.1. Computing the distance to a smoothly interpolated surface between two free-form contours is far more challenging.

To describe the local ADF, we compute a mapping $TPS(x)\colon \mathbb{R}^3 \to \mathbb{R}^3$ based on thin plate splines ([Boo89], [RSS*01]) that transforms C_i and C_{i+1} into a normalized space where distance computations become feasible. This mapping process has to be topology-preserving to prevent changes of the interpolated surface. As stated in the original work of Bookstein [Boo89], the thin plate spline mapping is a diffeomorphism as long as it does not fold. To ensure a continuous mapping, we assign contour pairs to normalized target shapes that are homeomorphic to the shapes defined by the original configurations. The contours associated with centerlines can exhibit different configurations as shown in Figure 5. In most cases two consecutive contours define a cone-like volume. Since they are usually sampled in planes orthogonal to the centerline, they may however touch or even intersect if the centerline exhibits a high curvature or centerline nodes are close.

Under the assumption that the topology of a pair of neighboring contours is one of $TOP(C_i, C_{i+1}) \in$ {*non-intersecting*; *touching at 1 contour position*; *intersecting at 2 contour positions*} a system of generators is illustrated in Figure 5. Obviously, more complex relations are possible in theory but in practice this system of supported constellations has proven to be sufficient to model realistic vascular structures. After examining the relation between a pair of neighboring contours and assigning the appropriate generator element, a set of reference points $(\tilde{q}_i^1, \ldots, \tilde{q}_i^n, \tilde{q}_{i+1}^1, \ldots, \tilde{q}_{i+1}^m)$ is distributed on the target shape. The target shape of *non-intersecting* is a normalized cylinder with $radius = 1$ and $height = 1$, the target shape of *touching at 1 contour position* is a section of a horn torus with $radius_a = radius_b = 1$ and the target shape of *intersecting at 2 contour positions* is defined by two unit-circles whose intersection line goes through their centers (see Figure 5). In the case of the torus and the crossed circles it is important to map the points surrounding the intersections of C_i and C_{i+1} to the corresponding locations on the target shape to generate a well-behaved mapping (highlighted contour points in Figure 5).

The set of corresponding point pairs is then used to determine a smooth mapping $TPS(x)$ from world space W to normalized space N that satisfies $TPS(q_i^j) = \tilde{q}_i^j$ for $1 \le j \le n$ and $TPS(q_{i+1}^k) = \tilde{q}_{i+1}^k$ for $1 \le k \le m$. For this the displacement vectors $(\tilde{q}_i^1 - q_i^1, \ldots, \tilde{q}_i^n - q_i^n, \tilde{q}_{i+1}^1 - q_{i+1}^1, \ldots, \tilde{q}_{i+1}^m - q_{i+1}^m)$ are interpolated using thin plate splines to define a global vector deformation field. The calculation of coefficients for the interpolating basis functions leads to a system of linear equations as described in Rohr et al. [RSS*01]. To approximate the distance function $f_i(x)$ of a requested point $x \in W$, the point is transferred into the normalized space where the distance can be calculated $\tilde{f}_i(\tilde{x}) = \tilde{f}_i(TPS(x))$ with $\tilde{x} \in N$ based on the normalized primitive. An important remark on this distance approximation approach is that the transformation $TPS(x)$ is not distance-preserving due to the non-rigid deformation. However, the sign of distances and thus the topology of the object is preserved. Figure 4 shows an arterial tree reconstructed from free-form contours and two local segment surfaces described by contour pairs.

5. Adaptive Modelling of Vascular Structures

To efficiently generate a vessel mesh with a locally adapted triangle size, we construct an octree that serves two purposes. Firstly, it is used to establish a relevant subset of segment functions $f_i(x)$ for each octree cell to approximate $F(x)$. Secondly the cell size is adapted to the size of the

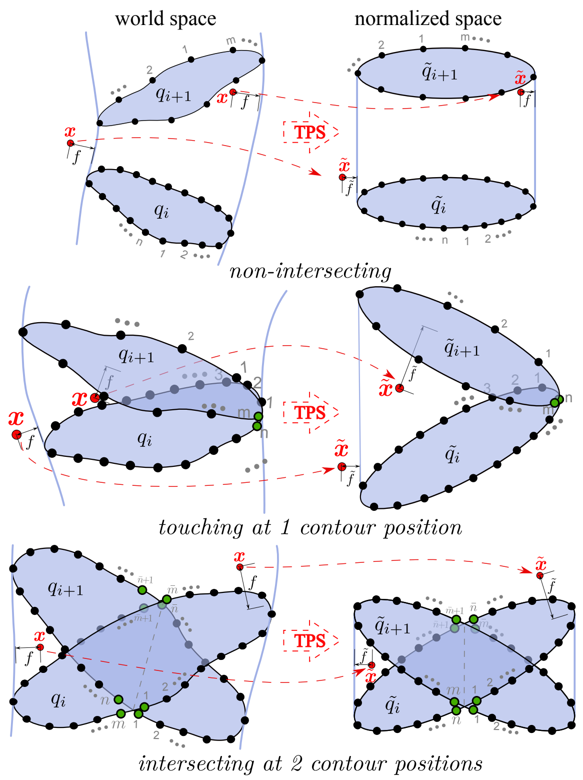


Figure 5: Mapping of neighboring free-form contour pairs to normalized shapes. A non-rigid thin plate spline transformation TPS to a topologically equivalent simple object is performed to compute local ADF f by an easily computable approximation $\tilde{f}$ in the normalized space. For our free-form vessel models the topology of each neighboring contour pair is classified as *non-intersecting*, *touching at 1 contour position* or *intersecting at 2 contour positions* as displayed.

smallest vessel that is locally present. Finally, we employ a modified version of the iso-surface extraction method described in [KKDH07] that uses a bisection method to determine iso-surface positions.

5.1. Octree Construction

To compute the global function $F(x)$, all segments S_i need to be evaluated. Since the evaluation of the implicit functions $f_i(x)$ is potentially expensive, we try to avoid the consideration of far-away segments. To establish locality, we enclose all segments with a bounding box B_i that must contain the entire local surface. The root cell is initialized with all segments and is recursively subdivided. When splitting cells, we distribute vessel segments to child cells if they intersect. Cells which intersect no segments cannot contain surface portions and are not examined any further. Instead of evaluating $F(x)$, we use the localized approximation $F'(x) = \min\{f_i(x) | B_i \text{ intersects } O\}$ for distance queries in octree cell O. Note, that $F'(x)$ does not equal $F(x)$. It is only defined in octree cells that contain segments and is not continuous, since segment functions may pop in or out based on their bounding boxes. However, the properties listed in Section 4, including admissibility, are preserved which makes F' a valid approximation.

5.1.1. Radius-sensitive Octree Refinement

To ensure an adequate sampling, we refine the octree until the local cell size is small enough to capture the shape of all relevant segments. For this purpose, we require every segment to provide a conservative approximation of its radius and determine the smallest radius r_{min} of all locally relevant segments S_c. Since r_{min} is required to be a conservative estimation of the vessel radius, the smallest object that can be present is a sphere of radius r_{min}. This means we have to choose the sampling distance (i.e. the cell size) small enough

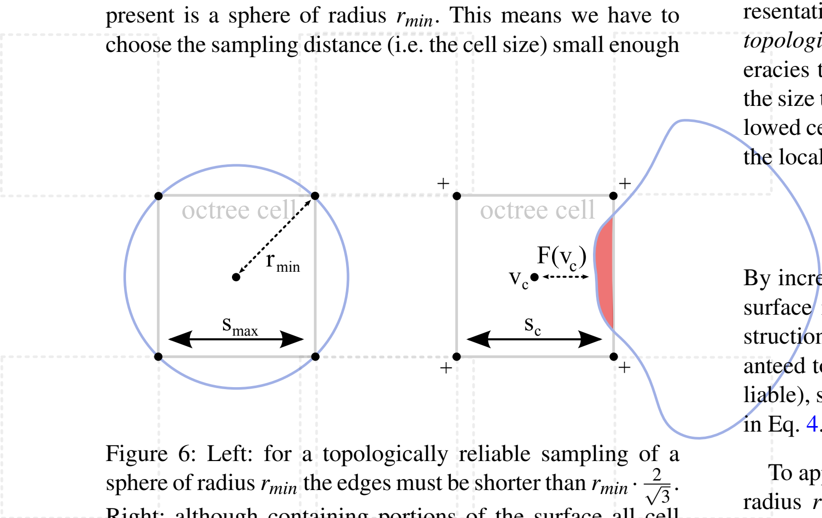


Figure 6: Left: for a topologically reliable sampling of a sphere of radius r_{min} the edges must be shorter than $r_{min} \cdot \frac{2}{\sqrt{3}}$. Right: although containing portions of the surface all cell vertices lie outside the volume. Pruning the cell prohibits the reconstruction of the highlighted area. If $|F(v_c)| > \frac{\sqrt{3}}{2} \cdot s_c$ holds the surface does *not* run through the cell.

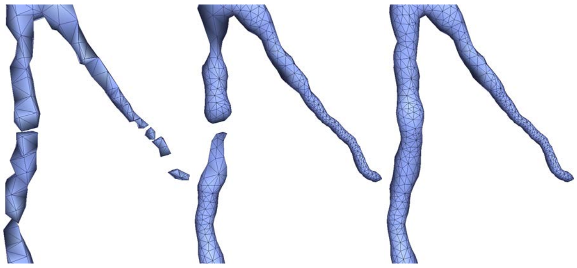


Figure 7: Topological degeneracies occur if the size threshold for the octree cells is violated. Quality parameters Q from left to right are 0,5, 0,9 and 1. A quality parameter of $Q >= 1$ guarantees a topologically reliable reconstruction.

to be able to capture the topology of a sphere with radius $r_{min} > 0$ (see Figure 6a). Since octree cells are cubic, the upper bound s_{max} for the cell size $s_c > 0$ is defined as follows

$$\sqrt{(\frac{s_c}{2})^2 \cdot 3} < \sqrt{r_{min}{}^2}$$
$$s_c < r_{min} \cdot \frac{2}{\sqrt{3}} = s_{max}. \tag{4}$$

All cells are thus subdivided until their size is below this local threshold. Note, that the resulting meshes are not necessarily topologically correct. If the distance between the surfaces of two vessels is smaller than s_{max} the two vessels may be merged if the gap is not sampled. Adhering to s_{max}, however, ensures that every vessel segment is sampled at least at one position inside the volume which guarantees a representation of every vessel. Thus, we refer to this method as *topologically reliable*. Figure 7 shows the topological degeneracies that occur when sampling F' with cells that violate the size threshold. In addition to describing the maximum allowed cell size, the size threshold s_{max} can be used to control the local surface quality by introducing a parameter $Q > 0$

$$s_c < \frac{s_{max}}{Q}. \tag{5}$$

By increasing Q, the local cell size is forced to adapt to the surface more closely which leads to more accurate reconstructions. For $Q >= 1$ the vascular reconstruction is guaranteed to contain a representation of every vessel (i.e. is reliable), since the local cell size s_c fulfills the condition stated in Eq. 4.

To approximate the radius for clipped spheres, we use the radius r_i of the sphere and for clipped cones we use the smaller of the two radii $\min\{r_i, r_{i+1}\}$. For free-form contours we use the radius of the maximum inscribed circle. It can be computed using a distance transform and selecting the maximum of the distance field inside the contour perimeter.

5.1.2. Cell Pruning

Bounding boxes provide an overestimation of vessel segments and allow no pruning of cells that lie completely inside the vessel lumen. When refining the octree to approximate the surface, this produces a huge cell-overhead. To avoid this, we need a safe criterion to check if surface intersections are present.

To safely prune cells, the signs of F' at the cell vertices cannot be used, since the vascular surface may intersect a cell without producing opposite signs at its vertices. Figure 6b shows an example where the values of F' at the cell vertices are all positive i.e. outside the volume. Pruning the cell leads to a reduced accuracy of the reconstructed surface, since the highlighted arc is lost. To guarantee the conservation of all cells that intersect the surface, we exploit the admissibility property of F'. As stated in Section 5.1, it provides an estimation of the distance to the surface that never overestimates the distance to the surface. If the absolute function value $|F'(v_c)|$ at the cell center v_c is larger than the distance from the center to the corner vertices $\frac{\sqrt{3}}{2} \cdot s_c$ the cell cannot contain any portions of the. This scheme enables us to prune cells based on a single evaluation of F'. If pruning fails there are two cases: the cell is either subdivided, or it is flagged for polygonization. In the former case $F'(v_c)$ is reused by the child cells, in the latter case $F'(v_c)$ is reused to compute a disambiguate MC index.

5.2. Calculating Function Roots

When extracting the zero iso-surface, marching cubes-based algorithms [LC87], [KKDH07] detect roots $F = 0$ by examining changes in the sign of a scalar field. The root position $r_{ab} \mid F(r_{ab}) = 0$ is then interpolated linearly between the octree vertices v_a and v_b which are of opposite sign. In our case, however, linear interpolation can lead to artifacts because F provides no reliable information on the distance to the surface. As shown in Figure 8, r_{ab} differs greatly from the actual root, since the value sampled at v_a underestimates the distance to the surface.

For a robust and fast approximation of the root position along the line segment $[v_a, v_b]$ we use a slightly modified bisection method. We iteratively split the interval $[v_a, v_b]$, which is known to bracket a root, at a point s_{ab}. To ensure the detection of roots that lie on the outer surface only, we terminate the search if the scalar value is small $|F(s_{ab})| < \varepsilon$ and positive $F(s_{ab}) > 0$. If not, we repeat the process with subsegment $[v_a, s_{ab}]$ or $[s_{ab}, v_b]$ depending on which of them has opposite signs and thus contains our root. When splitting the interval at its mid-point $\frac{v_a+v_b}{2}$ this process leads to linear convergence. Since F is piecewise linear on the line segment $[v_a, v_b]$, we accelerate the root search by splitting the interval at the interpolated root position $s_{ab} = v_a \frac{|F(v_a)|}{|F(v_a)|+|F(v_b)|} + v_b \frac{|F(v_b)|}{|F(v_a)|+|F(v_b)|}$. In regions where $F(v_a)$ and $F(v_b)$ both yield the distance to the same (locally plane) segment surface $i \mid F(v_b) = f_i(v_b) \wedge F(v_a) = f_i(v_a)$ the root finding problem is solved by applying the intercept theorem, which leads to immediate convergence.

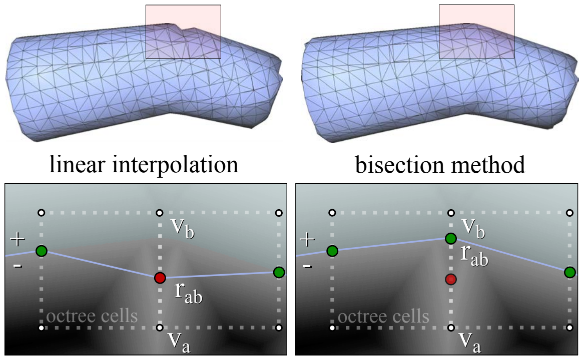


Figure 8: Top left: artifacts occur if the roots of F are linearly interpolated. Top right: using a bisection method the actual roots are identified and the reconstruction is correct. Bottom: the corresponding scalar fields. Left: r_{ab} is interpolated at the wrong position because of the distance peak inside the volume. Right: the bisection method iteratively finds the correct root positions indicated in green.

5.3. Computation of Normals

To compute gradients for a scalar function at arbitrary positions, a standard approach is to evaluate F in the surrounding of q and to approximate the partial derivatives by forming difference quotients. This is an expensive operation, since it implies at least four evaluations of F each resulting in multiple evaluations of underlying segment functions f_i. Since, our global indicator function F is not an SDF its gradient cannot be used to derive normals. We only need to compute normals at those surface positions which are calculated during the root finding scheme described in Section 5.2. Once the root position is found the normal can be derived directly. For this we simply keep track of the segment with the smallest signed distance and use its normalized gradient as the normal of the global surface. Since we are not dealing with SDFs the resulting normal is not exact. In practice, however, the error introduced by using our ADFs is marginal and can be compensated by one normal smoothing pass in a post-processing step.

6. Results and Discussion

To validate the proposed technique, we applied it on centerlines of various portal and venous liver trees (LT), a bronchial tree (BT), a cerebral vascular tree (CT) and an arterial tree (AT). The first three were described by centerlines with radial cross-sections and the arterial tree was described by free-form contours with $\approx 10-50$ surface samples per

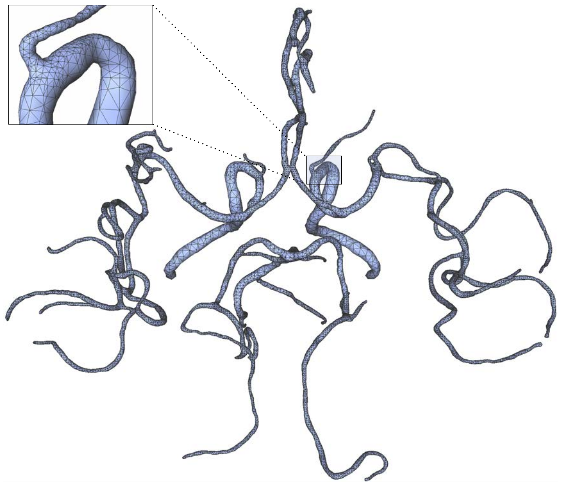


Figure 9: A low-poly reconstruction (70 812 triangles) of a cerebral tree with quality parameter $Q = 1$. The vascular branches are very thin, but due to the local refinement constraints all structures are reliably represented.

centerline node depending on the vessel diameter. All reconstructions consistently confirmed the reliability criterion described in Section 5.1.1. If the quality parameter Q was chosen $Q >= 1$ no single branch of any dataset was lost and no holes in the reconstruction occurred. Choosing $Q = 1$ thus leads to a fast low-poly reconstruction of the vascular surface independent of the variation of the vessel diameter. Figure 9 shows a reconstruction of a CT with $Q = 1$. The surface approximation is rather rough, but it is very well suited for applications such as intervention planning or therapy planning, since these focus on the spatial relations of vessels. It is also valuable in scenarios where a fast but highly representative rendering is necessary.

Our adaptive strategy proved to be powerful for datasets with a high variation in vessel diameter such as BT and CT. We compared it to an implementation of Convolution Surfaces (CS) which is shipped as a part of MeVisLab. It basically provides two quality parameters "smoothness at branchings" and "polygonal refinement" both providing three options (low, medium and high). The first one determines the size of the Gaussian kernel during implicit surface modelling and the latter one determines the MC resolution during polygonization. For CS to reproduce thin structures, we had to set the MC resolution to "high" leading to a substantial triangle overhead for large branches. Figure 10 shows a bronchial tree reconstructed with CS and RAMVAS ($Q = 1{,}5$). At equal triangle counts RAMVAS allows a more accurate representation of thin vessels by avoiding an oversampling of large structures.

Table 1 shows a comparison of the generation times of CS and RAMVAS with the corresponding resulting triangle counts. We set the smoothness parameter for CS to "low", which corresponds to the fastest setting for the algorithm. Generation times of RAMVAS are generally shorter except for the CT dataset. This is because it contains very thin vessels. The MC resolution of the CS implementation is too low in this case resulting in visible artifacts. The ability of RAMVAS to adapt to the local vessel size results in a denser sampling of the long thin vessels and thus in an increased computation time and triangle count.

Dataset		CS	RAMVAS	
			(Q=1)	(Q=2)
LT	*time*	14 070	2177	8 080
5146 segments	*tris*	510 056	95706	339 738
BT (Figure 10)	*time*	11 040	1 155	4 832
2580 segments	*tris*	261 664	49 418	221 168
CT (Figure 9)	*time*	2 048	1 716	4 756
3383 segments	*tris*	82 980	67 202	171 240

Table 1: Mesh generation times (in milliseconds) and triangle counts for CS [OP05] and RAMVAS, taken on an Intel Xeon CPU with 4 physical cores @2.80GHz and 8GB RAM.

Wu et al. [WWL*10] describe an advancing front meshing algorithm that generates curvature adaptive triangulations. Near-equilateral triangles are constructed on the surface and their size adapts smoothly to the local curvature. These properties make their meshes suitable for computa-

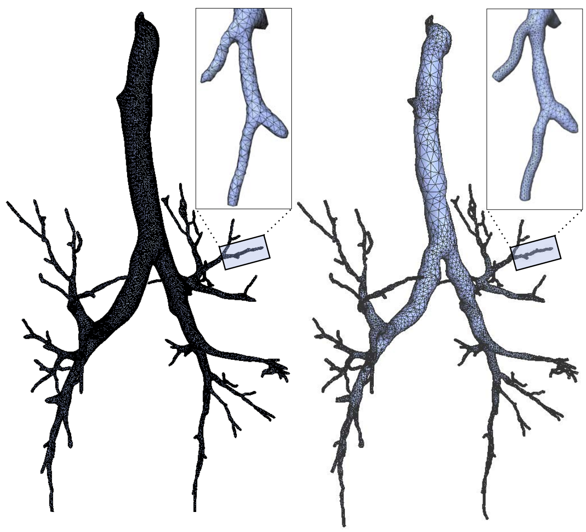


Figure 10: Two reconstructions of a bronchial tree with close-up views. Left: CS reconstruction. The main bronchus is sampled in unnecessary detail. Right: RAMVAS reconstruction with quality parameter $Q = 1{,}5$. Even though both meshes consist of ≈ 120 000 triangles RAMVAS is able to represent thin structures more accurately.

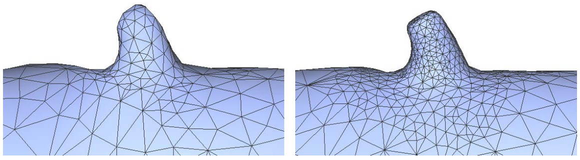


Figure 11: Two close-ups of a thin vessel branching from a large one (LT dataset, left: $Q = 1$, right: $Q = 2$). Our approach is able to handle such situations as the octree is refined to very small cells in the vicinity of the small branch.

tional fluid dynamics simulations. Due to the octree-based refinement strategy, our approach is inherently bound to power of two subdivisions in the cell size. This means our triangle size does not change smoothly. The projection-based approach described in [WWL*10], however, can fail if sudden changes in the curvature are present. For example a very small vessel branching from a large and low-curved structure leads to a sudden change in the local curvature. If the advancing front arrives at such a spot with a large current triangle size it might miss the small feature entirely. RAMVAS in contrast, is explicitly capable of handling such constellations, since small vessels force the local octree refinement to an adequate level (see Figure 11). Since we use axis-aligned bounding boxes of vessel segments to determine the smallest vessel that is present, more triangles than necessary are generated (recall Figure 11). Using oriented bounding boxes should greatly reduce this overhead.

6.1. Accuracy

CS implicitly generate rounded off caps at the vessel ends. Since our RAMVAS method allows the models for the individual segments to vary for one vascular tree, vessel ends can easily be modelled in different ways. By placing a half-sphere at the end, similar to Hahn et al. [HPSP01], we achieve rounded-off caps. By simply ending a vessel with a clipped cone, a plain, cut-off end is reconstructed (Figure 11, right and Figure 10, top right). This is useful to illustrate that a vessel does not actually end at a certain point but that the representation is discontinued due to an incomplete segmentation. To improve the quality of our triangulation, we smooth the generated meshes using the displacement corrected Laplacian proposed in [VMM99]. As stated in [BHP06] it is suited for meshes of elongated complex structures, since it reduces the shrinkage induced by ordinary Laplacian smoothing while maintaining a reasonable speed.

Figure 12: Projections of CS meshes displayed in solid black. The differences to the same surface generated with RAMVAS are color-coded. Red areas are not present in RAMVAS and green areas are not present in the CS mesh. Left: the differences at vessel ends arise from our ability to model non-spherical ends.

Since we had no influence on the polygonization of the CS implementation, a quantitative comparison that differentiates between the modelling and the meshing error was not feasible. Qualitative examinations, however, showed that our radial vessel model described in Section 4.1 delivers very similar results. When increasing the smoothness parameter (i.e. the kernel size) of the CS reconstruction, furcations are blended more smoothly but deviations from the original centerline radii dramtically increase. Figure 12 shows the projection of a surface computed with CS. The differences to the same dataset reconstructed with RAMVAS are color-coded. Differences between the results are marginal. Note, that the discrepancy at the vessel ends is not an error but results from our ability to model cut-off vessel ends (see Section 6.1).

We choose $\varepsilon = \min\{r_i\}/100.0$ to ensure that the root finding (i.e. the determination of the surface locations) is accurate enough not to compromise the reconstruction of the smallest vessel for arbitrary data sets. Experiments showed that an average of ≈ 2.2 iterations is needed and that $\approx 33\%$ of the roots are found at the first iteration. Using standard bisection (dividing $[v_a, v_b]$ at the mid-point) resulted in ≈ 3.3 iterations per root search, which validates our strategy that exploits the local SDF-like behavior of F'.

7. Conclusion

With RAMVAS, we present a flexible method that generates surface representations from centerline descriptions. Vascular systems described by free-form contours as well as radial cross-sections can be converted into watertight adaptive meshes with a controllable quality. Furcation-related issues are handled by employing implicit functions which results in intersection-free meshes for any complexity of the vascular furcations. We introduce an octree-based scale-adaptive refinement strategy that guarantees a *reliable* reconstruction of all vessels independent of their thickness. Our method is able to generate low-poly approximations and high-quality surfaces with one single approach. Compared to existing techniques our approach offers a superior reliability.

Acknowledgment

We want to thank Steffen Oeltze for providing the centerline datasets (CT, AT & BT) that we used to test our method and to compare it to Convolution Surfaces.

References

[AMMP10] ADLER S., MOENCH T., MECKE R., PREIM B.: Vessel surface generation for surgical simulations. In *Proceedings of CURAC* (2010), pp. 149–156. 2

[BB*09] BECK T., BIERMANN C., ET AL.: Robust model-based centerline extraction of vessels in CTA data. In *Proceedings of SPIE* (2009), vol. 7259, p. 72593O. 2, 3

[BHP06] BADE R., HAASE J., PREIM B.: Comparison of fundamental mesh smoothing algorithms for medical surface models. In *Simulation und Visualisierung* (2006), pp. 289–304. 9

[Boo89] BOOKSTEIN F.: Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence 11*, 6 (1989), 567–585. 5

[BRB05] BORNIK A., REITINGER B., BEICHEL R.: Reconstruction and representation of tubular structures using simplex meshes. *Proceedings of Winter School of Computer Graphics (WSCG)* (2005), 61–65. 3

[FKF*02] FELKEL P., KANITSAR A., FUHRMANN A. L., WEGENKITTL R., AG T. M., A L., GEBIRGE A.-B. A.: Smart - surface models from by-axis-and-radius-defined tubes. In *TR VRVis 2001 026, VRVis Research* (2002), pp. 122–124. 2

[FPRJ00] FRISKEN S. F., PERRY R. N., ROCKWOOD A. P., JONES T. R.: Adaptively sampled distance fields: a general representation of shape for computer graphics. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (2000), ACM Siggraph, pp. 249–254. 3

[HPSP01] HAHN H. K., PREIM B., SELLE D., PEITGEN H.-O.: Visualization and interaction techniques for the exploration of vascular structures. In *IEEE Visualization* (2001). 2, 9

[KBH06] KAZHDAN M., BOLITHO M., HOPPE H.: Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing* (2006), Eurographics Association, pp. 61–70. 2

[KHH*04] KIRALY A. P., HELFERTY J. P., HOFFMAN E. A., MCLENNAN G., HIGGINS W. E.: Three-Dimensional Path Planning for Virtual Bronchoscopy. *IEEE Transactions on Medical Imaging 23*, 11 (2004), 1365–1379. 1

[KKDH07] KAZHDAN M., KLEIN A., DALAL K., HOPPE H.: Unconstrained isosurface extraction on arbitrary octrees. In *Proceedings of the fifth Eurographics symposium on Geometry processing* (2007), Eurographics Association, pp. 125–133. 2, 3, 6, 7

[LC87] LORENSEN W., CLINE H.: Marching cubes: A high resolution 3d surface construction algorithm. *ACM Siggraph Computer Graphics 21*, 4 (1987), 163–169. 2, 7

[OP05] OELTZE S., PREIM B.: Visualization of vasculature with convolution surfaces: method, validation and evaluation. *IEEE Transactions on Medical Imaging 24*, 4 (2005), 540–548. 3, 8

[RSS*01] ROHR K., STIEHL H., SPRENGEL R., BUZUG T., WEESE J., KUHN M.: Landmark-based elastic registration using approximating thin-plate splines. *IEEE Transactions on medical imaging 20*, 6 (2001), 526–534. 5

[SOB*07] SCHUMANN C., OELTZE S., BADE R., PREIM B., PEITGEN H.: Model-free surface visualization of vascular trees. In *IEEE/Eurographics Symposium on Visualization* (2007), Eurographics Association, pp. 283–290. 2

[TdTF*07] TYRRELL J., DI TOMASO E., FUJA D., TONG R., KOZAK K., JAIN R., ROYSAM B.: Robust 3-D modeling of vasculature imagery using superellipsoids. *IEEE Transactions on Medical Imaging 26*, 2 (2007), 223–237. 2

[VMM99] VOLLMER J., MENCL R., MÜLLER H.: Improved laplacian smoothing of noisy surface meshes. *Computer Graphics Forum 18*, 3 (1999), 131–138. 9

[VZB*05] VOLKAU I., ZHENG W., BAIMOURATOV R., AZIZ A., NOWINSKI W.: Geometric modeling of the human normal cerebral arterial system. *IEEE Transactions on Medical Imaging 24*, 4 (2005), 529–539. 2

[WNV00] WINK O., NIESSEN W., VIERGEVER M.: Fast delineation and visualization of vessels in 3d angiographic images. *IEEE Transactions on Medical Imaging 19*, 4 (2000), 337–346. 2

[WWL*10] WU J., WEI M., LI Y., MA X., JIA F., HU Q.: Scale-adaptive surface modeling of vascular structures. *BioMedical Engineering OnLine 9*, 1 (2010), 75. 2, 8, 9

[ZO03] ZHAO H., OSHER S.: *Geometric Level Set Methods in Imaging, Vision, and Graphics*. Springer, 2003, ch. Visualization, Analysis and Shape Reconstruction of Sparse Data, pp. 361–380. 2