IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 26, 2017, accepted January 22, 2018, date of publication February 7, 2018, date of current version March 15, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2802478

Centerline Extraction of Vasculature Mesh

MINGQIANG WEI[1], QIONG WANG[2], YICHEN LI[4], WAI-MAN PANG[5], LUMING LIANG[1], JUN WANG[1], KELVIN KIAN LOONG WONG[3], DEREK ABBOTT[6], (Fellow, IEEE), JING QIN[7], AND JIANHUANG WU[3]

[1]Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China
[2]Shenzhen Key Laboratory of Virtual Reality and Human Interaction Technology, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzen 518055, China
[3]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzen 518055, China
[4]Nanjing Normal University, Nanjing 210097, China
[5]Caritas Institute of Higher Education, Hong Kong, Hong Kong
[6]University of Adelaide, Adelaide, SA 5005, Australia
[7]The Hong Kong Polytechnical University, Hong Kong

Corresponding author: Jianhuang Wu (jh.wu@siat.ac.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61502137, Grant 61672510, and Grant 61233012, in part by the China Postdoctoral Science Foundation under Grant 2016M592047, in part by the Shenzhen Science and Technology Program under Grant JSGG20150602143414338 and Grant JCYJ20160429190300857, and in part by the Guangdong Science and Technology Program under Grant 2016A020220016.

ABSTRACT Mesh representation of vasculature is fundamental to many medical applications. The benefit is a clean and tidy appearance in terms of visualization, as well as the possibility of applying computer-assisted intervention and preoperative planning for patients. A vasculature mesh is often reconstructed by iso-surfacing its segmented volume data. Clinicians are usually interested in both the vasculature and its centerline. In this paper, we introduce a mesh centerline extraction approach in the case that volume data are unavailable. The extraction method is inspired by an observation that the vasculature is generally composed of piecewise cylindrical shapes. This observation leads to a conceptually simple but effective strategy to tackle the challenging problem of vasculature centerline extraction, which gracefully combines a branch segmentation scheme and a series of advanced techniques in discrete geometry processing. Our method competes favorably with three state-of-the-art methods in the completeness and accuracy of the extracted centerlines from real human vessels, including the pathological vasculature. Our method also usually leads to maximal and mean extraction errors of less than 1% and 0.5%, respectively.

INDEX TERMS Vasculature mesh, centerline extraction, rotational symmetry axis, vasculature segmentation, discrete geometry processing, interventional radiology simulation.

I. INTRODUCTION

Vasculature centerlines are important cues for medical applications. In the interventional radiology simulation [1] as an example, the simulator requires operators to move both the guidewire and catheter, minimally colliding with a vessel's inner walls. It can be achieved by encouraging the operators to move them along the vessel's centerline. Fig. 1 shows our system for the assessment of trainee performance, where the vasculature centerlines are equipped to provide guidance information.

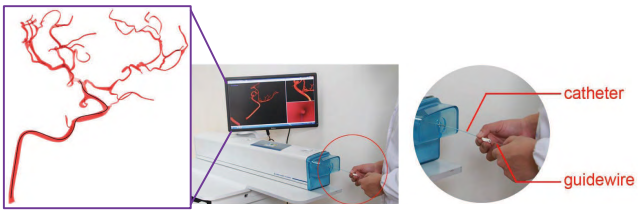


FIGURE 1. Our vascular intervention simulator is performed on a cerebral vessel. If we manipulate the virtual guidewire (see the black curve within the vasculature mesh) to move along the extracted centerline, colliding with vessel walls can be largely avoided.

The vasculature centerlines are mostly extracted on volume data [2], [3], such as the commercial software MeVisLab, and Materialise Mimics (MedCAD). However, these software provides few choices for users to construct a vascular centerline, when volume data are unavailable. In this paper, we focus on another important representation of 3D data, i.e., the vasculature mesh, when the volume data are absent. The other reasons for us to perform centerline extraction on

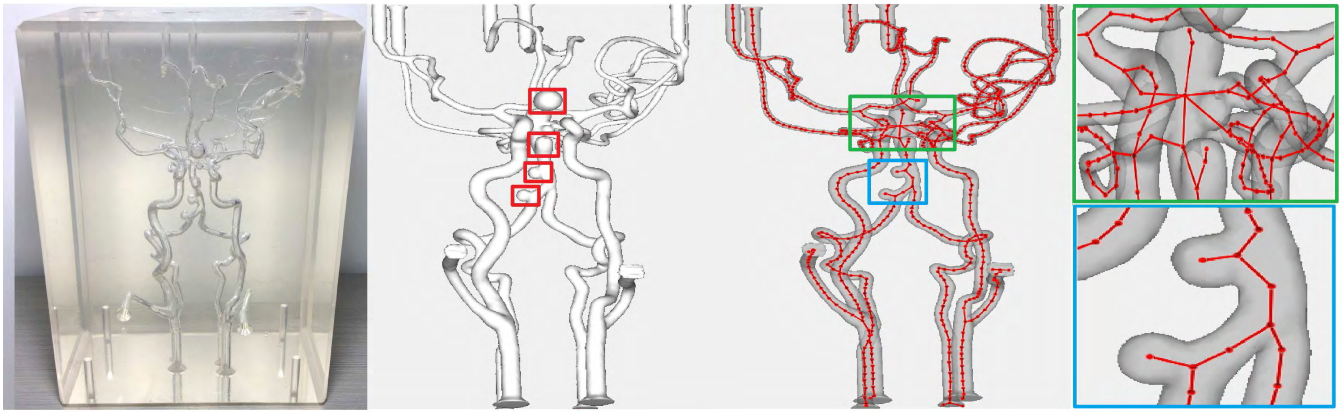


FIGURE 2. Centerline extraction for the complex vasculature with tumors (an anterior and posterior communicating artery). From left to right: the prosthesis object with a same-size ratio of a patient's blood vessel, the iso-surfaced model by Marching Cubes and the postprocessing technique in [6] (with tumors highlighted in red rectangle frames), the extracted centerline, and the corresponding magnified fragments. The centerline in these tumors is also captured.

the vasculature mesh are that: First, when an iso-surfaced mesh of a vascular network is reconstructed from the segmented volume data, it usually needs to be re-aligned to another coordinate system for specific applications, e.g., non-rigid and rigid organ registration [4]. In this case, the centerline extracted from volume data cannot be applied to the re-aligned mesh, unless the corresponding transformation is maintained. Otherwise, the mesh and the centerline require to be re-registered. Second, vasculature models generated by modeling tools or obtained from vascular prothesis by laser scanners usually lack volume data, which cannot be processed by the traditional volume-based extraction methods [5]. Third, special operations, e.g., collision detection, become possible on meshes, because they are watertight. Last, the triangular mesh representation of 3D surfaces is perfectly supported by modern graphics hardware.

Extracting vasculature centerlines is a non-trivial task. This is because, the vasculature possesses complicated geometry (e.g., vascular tumors) and topology (e.g., thin structures). In addition, artifacts (e.g., staircases and noise) exist in these vasculature meshes that also hinder the centerline extraction [6].

We propose a simple but effective strategy to meet these challenges. Our method is inspired by an observation that, the vasculature is generally composed of piecewise cylindrical shapes, except at joint regions. This observation is closely related to the rotational symmetry axis (ROSA) that is proposed in [7], in which the observation is applied to extract a curve-skeleton from an incomplete point cloud. We can segment the vasculature into multiple branches. It avoids the mutual interference of branches when extracting each branch's centerline. We have demonstrated that our method outperforms three state-of-the-art methods on the individual case. We have further demonstrated the effectiveness of the proposed method by employing it in a virtual reality based simulation system for interventional radiology training.

Fig. 2 shows a result of centerline extraction on a complex vascular network. Our contributions are two-fold:

- We propose an improved method for the vasculature centerline extraction from surface meshes based on the piecewise cylinder assumption.
- The integrated smoothing, thinning, and re-centering algorithms, which are tailored for the vasculature with complex geometry and topology, contribute to construct a complete and accurate 1D centerline.

II. RELATED WORK

A centerline is closely related to curve skeletons [8]. However, the centerline/curve-skeleton extraction from 3D shapes is not a well-defined problem [9]. It has resulted, over the decades, in the development of many different techniques, each attempting to comply with special requirements. We divide existing techniques into three categories: (i) volume-based methods, where the centerline extraction is based on a discretization of volume data; (ii) surface mesh based methods, and (iii) point cloud based methods, where the centerline/skeleton extraction is directly on primitives that define the surface.

A. VOLUME-BASED METHODS

Voxel-thinning methods obtain a centerline through the iterative removal of the boundary voxels that satisfy certain topological and geometric constraints of the voxel object [10]–[12], [5]. Based on a distance transformation, some methods first compute the minimum distance to the boundary for internal voxels to get a distance field, and then find ridge voxels through the distance field, which are used as candidate voxels for centerline construction [13]–[17]. Livesu *et al.* [18] presented a fundamentally different approach based on the visual hull: They extract curve skeletons from a set of 2D views of a 3D shape. This approach requires only a set of 2D views of the input shape, which can deal with an incomplete 3D shape model.

A triangular mesh of an object is usually watertight. We can first transform it into a voxelized representation, and then follow the pipeline of volume-based methods to construct

its centerline. Unfortunately, this is prone to errors [19] in constructing an object's internal geometry and topology.

B. POINT CLOUD-BASED METHODS

Sharf *et al.* [20] achieve the curve skeleton extraction via Laplacian-based contraction, and Tagliasacchi *et al.* [7] to achieve its extraction through a ROSA-based method. These two methods can deal with point clouds with moderate amounts of missing data. Huang *et al.* [9] introduce the medial curve skeleton of a point cloud. However, this method does not intend to distinguish points from different structures. For complex shapes that contain close-by surface sheets (e.g., vasculature), it, therefore, produces curve skeletons with incorrect topology. Kurlin [21] proposes a homologically persistent curve skeleton based on subgraph construction, which focuses solely on 2D point clouds.

These methods can usually produce quality results for simple objects that are represented by point clouds. Once they are deployed on complicated 3D vasculature models, where artifacts and extensive small structures exist, the extracted centerlines are also not optimal. Since the trend to represent vasculature with a watertight surface mesh, these extraction methods are not suitable for the vasculature meshes.

C. SURFACE MESH-BASED METHODS

Wang and Lee [22] perform an iterative least-squares optimization scheme to shrink models to extract a curve skeleton. Hassouna and Farag [23] first select a centerline automatically as the position of global maximum Euclidean distance from the boundary, and then extract the centerline robustly by means of a level set algorithm. Au *et al.* [24] perform mesh contraction for skeletonization. This method contracts the mesh geometry into a zero-volume skeletal shape by applying implicit Laplacian smoothing with global positional constraints. Pascucci *et al.* [25] use a robust on-line computation of Reeb graphs to extract a curve skeleton. Li *et al.* [26] use the well-known quadratic error minimization to compute a structurally simple, geometrically accurate, and compact representation of the medial axis transform. In summary, these methods are designed to handle a series of shapes for a wider applicability that produce quality extraction results. However, they are usually complicated and not robust to be directly performed on vasculature meshes.

Wang *et al.* [8] extract curve skeletons from the vasculature meshes using a variant of mesh contraction [24]. However, an important drawback of their method is that the operation of mesh contraction leads to vascular tumors collapsing to a centerline point, which runs the risk of losing clinically important topology information of vascular structures. In contrast, our method can handle tumors as other parts of the vasculature, since a tumor is treated as a small vascular branch. In this regard, the topology information of this small tumor branch is successfully included in the centerline. Our method can also obtain a smooth centerline with the following properties:

- Thinness: A centerline is a compact 1D representation of a vasculature.
- Centeredness: A centerline is the rotational symmetry axis of a vasculature which is invariant in any affine transformation (rotation, symmetry).
- Homotopy: A vasculature and its centerline are homotopically equivalent.

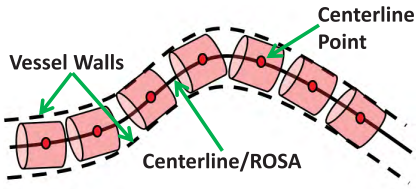


FIGURE 3. A diagram to illustrate the rotational symmetry axis (ROSA) and the centerline point (CP) in a vasculature branch.

III. PROBLEM STATEMENT

The general premise of our method is that a vascular system is composed of short cylinders, as shown in Fig. 3. The rotational symmetry axis (ROSA) from these cylinders thus forms the vasculature centerline. Considering an infinitesimal point on the centerline, namely ***centerline point (CP)***, we have its associated rotational symmetric cylinder with an infinitesimal area surrounding it. This cylinder lies on a cut-plane/cross-section of the vasculature, where the cut plane is perpendicular to the direction of the vasculature centerline. In a vasculature mesh, the infinitesimal cylinder is formed by oriented mesh vertices. We define a subset S consisting of mesh vertices from the same cut plane, and the CP as $c = (x_{cp}, v_{cp})$ with its position x_{cp} and normal vector v_{cp}. c is rotationally symmetric about S.

For each mesh vertex, we associate a subset S with it to compute its corresponding CP. However, determining the subset S for each vertex is not easy for the following three reasons: 1) we should find the optimal one which contains vertices consistently forming a good cut plane from all of the candidate subsets; 2) the cut plane intersects with multiple vasculature branches, it makes the subset S become "fat" and leads to the failure in the estimation of CPs; 3) we cannot employ the approach in [7] directly, which will search for the optimal cut plane containing vertices of a cross-section of the vasculature.

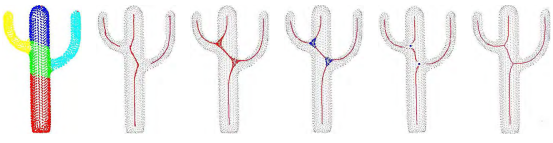


FIGURE 4. The steps of the proposed method: (a) the input vasculature mesh is segmented; (b) the CPs (red) of each branch are computed and then (c) smoothed; (d) the CPs at the joint region are detected and the thinning is performed; (e) the CPs are re-centered and each joint collapses to a unique joint center (blue); and (f) sub-sampling and connecting the samples with short curve segments to obtain a 1D centerline.

The steps of our method are illustrated in Fig. 4. We perform segmentation on the vasculature to separate branches as

the first step (Fig. 4(a)). This segmentation divides vertices into groups, where each group contains vertices from a single branch. We then determine an optimal cut plane as well as the corresponding CP for each mesh vertex according to the ROSA-based method reported in [7]. However, instead of searching the optimal cut plane from the whole vasculature surface [7], we narrow down the searching range to each branch that the vertex lies on, thereby making the calculation more efficient and precise. The generated CPs are shown in Fig. 4(b). To bridge the CPs at the endpoints of multiple branches, we employ Laplacian smoothing to initially encourage the CPs in each branch's endpoints to move to the joint region (Fig. 4(c)). We detect the CPs in the joint regions, and leverage the filter proposed in [27] to thin the CPs that are only associated with the branches (performing the filter to the CPs in the joint regions is not necessary). The detected CPs in the joint regions (blue points) and the thinning results on the branches are illustrated in Fig. 4(d). We then fine tune the centredness of these CPs, especially these located in the joint regions. The adjusted results are shown in Fig. 4(e), whereby in a joint, the CPs collapse to a unique joint center. Finally, we construct a 1D centerline by sub-sampling and connecting the samples with short curve segments (Fig. 4(f)).

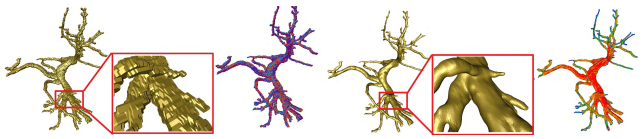


FIGURE 5. The poor portal vein model is optimized using the context-aware filter in [6]. Left: the polygonized iso-surface extracted from an inhomogeneous binary volume with noise and staircase artifacts, the middle colored model is rendered by discrete mean curvature to show the artifacts. Right: the optimized model with its morphology preserved, the right-most colored result shows that the model has become consistent after the optimization.

IV. VASCULATURE CENTERLINE EXTRACTION

Our method inputs an optimized iso-surfaced mesh obtained from the context-aware filter in [6]. An example of the optimized results by Wei *et al.* is shown in Fig. 5. We generate a centerline point for each mesh vertex in this section, and in the next section these centerline points are refined to form a 1D centerline. In order to avoid the interference from other branches, the vascular model is first segmented meaningfully by clustering techniques and interactive tools. Based on the segmentation results, the centerline points can be extracted effectively.

A. VASCULATURE SEGMENTATION

We employ the K-means fuzzy clustering algorithm to segment the branches with the geodesic distance as the dissimilarity measurement. We leverage the shortest paths algorithm [28] to calculate the geodesic distance of each vertex pair. We first assign each edge of the mesh a distance weight (Euclidean distance in our implementation) to construct a weighted graph for the input mesh, and then utilize the shortest path between two vertices on the weighted undirected graph to approximate the geodesic distance of the two vertices. We further adopt the newly proposed shortest path faster algorithm (SPFA) [29] to accelerate the computation.

Since our target is to form a set of segments that have approximate cylindrical shapes, the initial cluster number K in the K-means fuzzy clustering algorithm must be carefully selected. The most straightforward way is to set K as the number of branches of the input vasculature. It is observed that the endpoint(s) of a branch can be considered as a salient feature for a branch. We apply the method reported in [30] to extract the tips of the branches in the input mesh. The extracted branch tips from the input mesh are, therefore, used for the initialization of the K-means fuzzy clustering algorithm. We also apply the method proposed in [31] to optimize the tips to yield the optimal initial clustering centers, since the locations of the initial clustering centers are important for achieving satisfactory results.

After the initial clustering centers are determined, the $K \times n$ probability matrix U is computed using the following function:

$$u_{ij} = \left[\sum_{t=1}^{K} \left(\frac{d(i,j)}{d(i,t)} \right)^{2/(m-1)} \right]^{-1}, \tag{1}$$

where K is the number of the clustering centers; n is the vertex number of the triangular mesh; $d(i, j)$ and $d(i, t)$ denote the geodesic distances from the mesh vertex p_i to the clustering centers c_j and c_t, respectively; $m \in (1, \infty)$ is the weight index, and we obtain satisfactory results by setting $m = 2$ in our implementation.

The probability u_{ij} indicates the possibility that a vertex i belongs to a cluster j. Vertices are assigned to a cluster, if its probability u_{ij} exceeds a certain threshold μ. We experimentally set $\mu = 0.5$, which performs very well in our experiments. All other vertices form a fuzzy area. The centroid of the fuzzy area represents the center of the $k + 1$ cluster. The algorithm repeatedly computes the probability matrix and relocates the clustering centers, until no cluster centers move any more, and the objective function is defined as:

$$J = \sum_{i=1}^{K} \sum_{j=1}^{n} u_{ij}^2 d(i, j), \tag{2}$$

and the cluster centers are computed by

$$c_i = \frac{\sum_{j=1}^{n} u_{ij} v_j}{\sum_{j=1}^{n} u_{ij}}. \tag{3}$$

The segmentation results can be further refined by an interactive method [32]. Two typical segmentation results are shown in Fig. 6. Our segmentation method can segment the complicated vasculature into a set of cylinder-like segments that serve for the following centerline extraction.

B. OPTIMAL CUT PLANE AND CENTERLINE POINT

After the vasculature segmentation, we avoid ambiguities when finding cross-section vertices subsets for the CP generation, such as the inclusion of vertices from many branches.

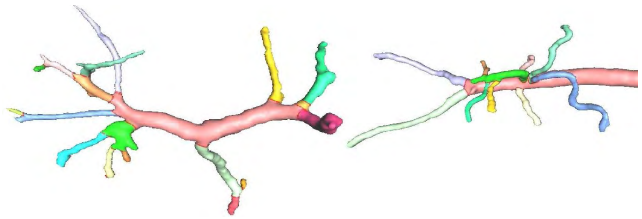


FIGURE 6. Segmentation results on two complex vasculature models. These segmentation results supply important information for extracting a complete and accurate centerline.

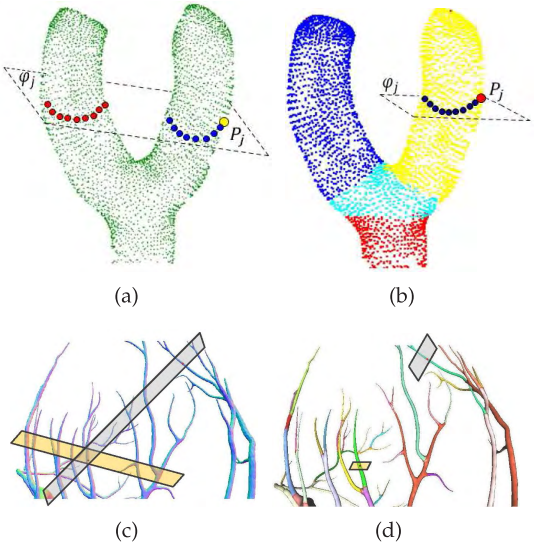


FIGURE 7. Comparison between traditional method [7] and our method: (a,c) the cut plane that is calculated by the vertex subsets generated by [7] and (b,d) the cut plane that is calculated by the vertex subsets generated only from the segmented branches in our method.

For example, as demonstrated in Fig. 7(a), the cut plane φ_j associated with a mesh vertex p_j is cutting through two branches at the same time. However, we only need the blue mesh vertices. The red mesh vertices are considered as artifacts. In Fig. 7(b), mesh vertices are separated according to branches (colored differently) and the cut plane only cuts through the branch where p_j lies. This problem is especially serious in real vascular newworks, when branches are close, as shown in Fig. 7(c). The segmentation facilitates our searching of the optimal cut plane, as shown in Fig. 7(d).

We need to estimate a cut plane φ_i and a CP $c_i = (x_{cp}, v_{cp})$ for each vertex $p_i = (x_i, v_i)$ (x_i and v_i denote its position and normal respectively) on the vasculature mesh. The estimation of the cut plane and the CP is actually simultaneous, because the normal of the cut plane is the same as the CP's normal v_{cp}. After the cut plane is fixed, the CP's position x_{cp} can be determined by solving a least-squares optimization. Therefore, the normal v_{cp} and position x_{cp} of a CP c_i are solved separately [7], as shown in Fig. 8, given a subset S_i. More details are given in the Appendix.

As a result, we can start from a random vertex p_i in an arbitrary branch. The cut plane φ_i for p_i can be represented

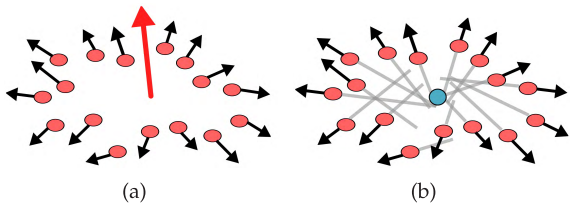


FIGURE 8. Finding the CP c with its position x_{cp} and normal v_{cp} from the subset S simultaneously, potentially leads to a non-convex problem [7]. Fortunately, the position and normal of a CP can be solved separately. The red points consisting of oriented mesh vertices from a cross-section of a vasculature denote the local subset S_i. (a) The normal v_{cp} (the red arrow) is obtained by minimizing the sum of angular variations from the CP's normal to its surrounding normals of those oriented mesh vertices. (b) The position x_{cp} (green dot) is determined by minimizing the sum of distances from the CP's position to the normals extensions (gray lines).

by $(x_i - x) \cdot v_{cp} = 0$. The objective is to find a v_{cp} to construct a cut plane φ_i through p_i, which best reflects its local rotational symmetry. Specifically, v_{cp} should be most rotationally symmetric about the mesh vertex normals of subset S_i surrounding the vertex p_i. Here, S_i is formed by adding vertices which are close to φ_i within a threshold d_{cut}. In our implementation, setting $d_{cut} = 0.025L$ yields optimal results, where L is the length of the bounding box diagonal of the vasculature model.

The optimal cut plane is obtained by an iterative algorithm proposed in [7]. We set the initial cut plane normal to be v_{cp}^0, which satisfies $v_{cp}^0 \cdot v_i = 0$. In the k^{th} iteration, the normal of the cut plane is then updated according to the local oriented subset S_i^k that is given by

$$v_{cp}^{k+1} = \arg\min \sum_{j=1}^{N} var\langle v_{cp}^k, v_j \rangle, \tag{4}$$

where N is the size of the local oriented subset S_i^k associated with the current cut plane after the k^{th} iteration. The iteration is stopped when the cut plane normal is not changed. Therefore, the cut plane φ_i can be obtained with the optimal normal v_{cp}. Fig. 9(a-b) demonstrates the iteration procedure.

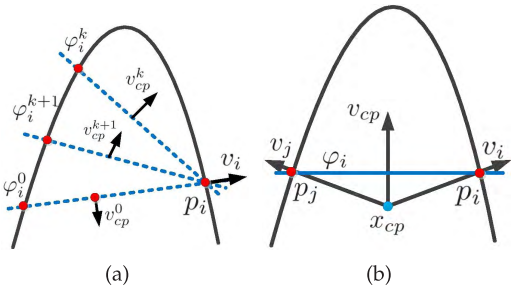


FIGURE 9. 2D example for illustrating the iterative procedure. (a) The process for obtaining the optimal cut plane from an initial cut plane φ_i^0 (dashed lines denote the candidate cut planes). (b) The optimal cut plane φ_i (the blue line) associated with p_i is obtained at the end of the iterations, and the corresponding CP is obtained simultaneously.

C. SMOOTHING AND THINNING

We adopt Laplacian smoothing to filter the CPs: it makes those CPs on a branch's intermediate region smoother, and

pushes the other CPs located at the region of a branch's endpoints to move to a joint region which is useful for bridging CPs on separated branches.

We thin all CPs, except the ones that have moved to joint regions, because these regions are not cylindrical. Thus, before thinning, we need to distinguish the CPs in the joint regions from those in the branches. To do this, we examine a standard linearity measure

$$\beta(c_i) = \frac{\lambda_i^{(1)}}{\lambda_i^{(1)} + \lambda_i^{(2)} + \lambda_i^{(3)}} \tag{5}$$

at a CP c_i, where $\lambda_i^{(j)}$ is the j^{th} largest eigenvalue from the principal component analysis (PCA) at c_i. Typically, if $\beta(c_i)$ is larger than a threshold ω, we consider c_i that belongs to a branch, otherwise it belongs to a joint.

We employ the filter proposed in [27] for the CPs belonging to the branches. For a random CP c_i, we first obtain its neighboring CPs c_j within a spherical region of radius h, i.e. $c_j - c_i < h$. The filter is then defined as

$$c_i^{new} = \frac{\sum \sigma_j c_j}{\sum \sigma_j}, \tag{6}$$

where the kernel function is $\sigma_j = \sin(\pi d)/\pi d$, and $d = \| c_i - c_j \|$.

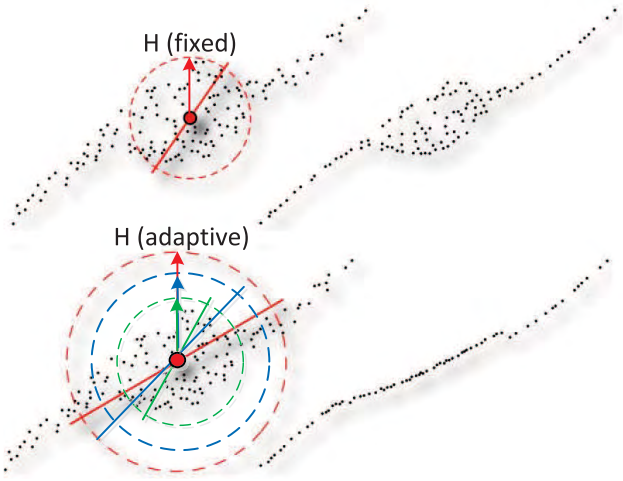


FIGURE 10. Thinning filtering by constant (top) and adaptive (bottom) values of radius h. The corresponding filtering results are shown on the right. Near joint regions, the radius h is adjusted by iteratively enlarging the neighborhood until $\rho(X, Y)$ exceed a threshold.

The above filter performs well in thinning when the CPs along the target thinning line are distributed uniformly enough. An exception occurs near the joint regions, as shown in Fig. 10 (upper), where a fixed and small radius can easily obtain an unsatisfactory thinning, since neighboring CPs are not exhibiting a strong directional relationship. To solve the problem, we propose a scheme to adaptively adjust the neighborhood radius h. As a result, more robust and satisfactory results can be achieved, as shown in Fig. 10 (bottom).

Given a CP c_i and the initial neighborhood radius h_0, we first obtain the CP set c_j. The covariance matrix of c_i is then calculated to obtain the eigenvalues $\lambda_i^{(k)}$ and their corresponding eigenvectors $v_i^{(k)}$.

We project these c_j onto the plane defined by the two eigenvectors $v_i^{(1)}$ and $v_i^{(2)}$ with the largest corresponding eigenvalues. All projected points are then reduced to a 2D local coordinate system. If the coordinates of the projected points are interpreted in terms of a distribution of two random variables X and Y, we can analyze their cross-correlation coefficient defined by

$$\rho(X, Y) = \frac{Cov(X, Y)}{SD(X)SD(Y)}, \tag{7}$$

where $Cov(X, Y) = E[X - E(X)]\,[Y - E(Y)]$ is the covariance between X and Y, $SD(\cdot)$ denotes the standard deviation, $\rho(X, Y)$ has a value between $[-1, +1]$ representing the degree of linear dependence between X and Y. A larger absolute value for $\rho(X, Y)$ indicates that there is a strong linear relationship between X and Y. Thus, we can adjust the neighborhood radius according to $\rho(X, Y)$, i.e., if $\rho(X, Y)$ is less than a pre-defined threshold δ_h, we will increase the radius value until $\rho(X, Y) > \delta_h$. After the adaptive adjustment of radius h, the CPs of branches will be thinned by the filter. Algorithm 1 gives the pseudo-code of the proposed thinning algorithm.

Algorithm 1 Algorithm of the CPs Thinning

Input: CP set C, initial neighborhood size h_0, step size ϵ_h, threshold δ_h
Output: Thinned CPs c^{new}
for (each c_i in C) **do**
  $h \longleftarrow h_0$
  Compute neighborhood set c_j
  Project c_j to plane sustained by $v_i^{(1)}$, $v_i^{(2)}$
  Compute correlation $\rho(X, Y)$
  if ($\rho(X, Y) > \delta_h$) **then**
    Compute c_i^{new} according to Eq. 6
  else
    $h \longleftarrow h + \epsilon_h$
    return to Step 5
  end if
end for

D. RE-CENTERING AND 1D CENTERLINE CONSTRUCTION

The steps taken so far, e.g., smoothing and thinning, may distort the centeredness of the CPs. Thus, we re-center the CPs according to their types (belonging to a branch or a joint). We exploit spatial coherence: close-by samples over the underlying shape of the triangular mesh should correspond to close-by CPs. Re-centering within a branch follows Eq. 10. A branch CP is re-calculated according to all the mesh vertices which lie on any cut plane of a small neighborhood c_j of c_i. Within a joint, the CPs collapse to a unique joint center according to Eq. 10, using all the mesh vertices which lie on any cut plane of all CPs in the joint. After the re-centering,

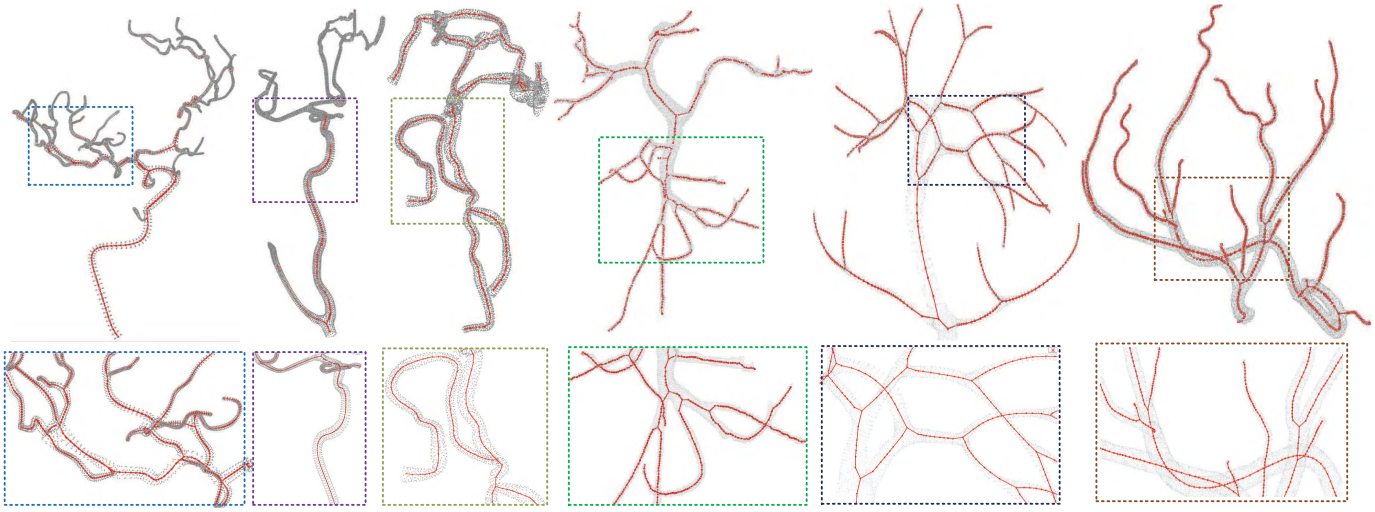


FIGURE 11. Centerlines of typical vasculatures. The top row shows the extracted results (red curves), and the bottom row shows the corresponding magnified fragments.

the centerline points are sufficiently close to be 1D. We apply sub-sampling and connect the CPs with short curve segments in order to obtain a final 1D centerline representation.

V. RESULTS AND DISCUSSIONS

We first assess our method on some complicated human vascular systems. In addition to visualization on these models, we adopt three metrics, i.e., $O_{\max}$, O_{mean}, and O_{var}, to calculate the errors relative to the manual centerline extraction results from volume data by an experienced clinician for the numerical analysis. However, it is nearly impossible to obtain a ground-truth centerline from a vasculature mesh. We use the extracted centerline from volume data as the ground truth. There are two main reasons for doing this. (1) Volume data having inner structures contain more information than an empty surface mesh. (2) An experienced clinician can help to extract centerlines by using commercial software, e.g., the VTK toolkit and the ITK toolkit. Thus, we consider the extracted centerlines of corresponding volume data by an experienced clinician as the ''ground truths", although they inevitably have deviations from real ones. The three metrics are based on Hausdorff distance. The $O_{\max}$ computes the maximal deviation distance between our extraction result and the manual result; the O_{mean} computes the averaged deviation distance from our result to the manual result; and the O_{var} computes the mean deviation variation. We then compare our results to three state-of-the-art methods in the individual case. In addition, the extracted centerlines are also performed on an application of computer-assisted simulation.

A. PARAMETERS SETTING

To address the problem of centerline extraction, we combine a branch segmentation scheme and a series of advanced techniques in discrete geometry processing. It, therefore, involves a set of parameters, mainly the weight index m in Eq. 1, the probability threshold μ to determine whether a vertex belongs to a cluster in Eq. 1, the threshold d_{cut} to measure the distance of a vertex to a cut plane, the threshold ω used in Eq. 5, the radius h, and ϵ_h and δ_h. Fortunately, five of these parameters can be empirically fixed in advance according to the references; for examples, $m = 2$, $\mu = 0.5$, $d_{cut} = 0.025L$, $h = 1$, and $\epsilon_h = 0.1$. We leave only two parameters for users to adjust. Here we list their value ranges for reference: We use local covariance analysis to distinguish the CPs between the vasculature branches and joint regions. Satisfiable results are obtained when the threshold ω is set in the range of $[0.7, 0.8]$. Another threshold is the correlation coefficient δ_h. In practice, any $\delta_h \geq 0.6$ brings desirable filtering results.

B. COMPLICATED VASCULATURE

From Fig. 2 we observe that our method is sensitive to vascular tumors, where centerlines in these tumors are well-captured (check the magnified fragments), thanks to the branch segmentation used (a tumor has been considered as a small branch by our method).

From Fig. 11 it can be observed that complete and smooth centerlines are obtained for the six kinds of vasculature. In addition, as shown in Fig. 12, we construct a vasculature centerline as the ground truth, and reconstruct a 3D vasculature mesh from it by using the method in [33]. We finally employ our extraction algorithm to obtain a centerline. We can observe that the extracted centerline is highly faithful to the constructed one.

In addition to visualization, we present the numerical analysis on these models in Figs. 11 and 12. From Table 1 we know that the deviation errors using our method are usually less than 1% of the diagonal length (denoted as L_{dia}) of a mesh's bounding box.

TABLE 1. Centerline errors relative to "ground truths".

Models	L_{dia}	$O_{\max}$	O_{mean}	O_{var}
Fig. 11 (1)	3.1702	0.02765	0.001717	0.008808
		(0.8722%)	(0.05419%)	(0.27785%)
Fig. 11 (2)	9.8725	0.06552	0.03214	0.01769
		(0.6637%)	(0.3256%)	(0.1792%)
Fig. 11 (3)	8.2614	0.08473	0.007658	0.01914
		(1.0256%)	(0.0927%)	(0.2317%)
Fig. 11 (4)	6.6704	0.02287	0.004322	0.008784
		(0.3428%)	(0.0648%)	(0.1317%)
Fig. 11 (5)	5.2036	0.060996	0.016952	0.031238
		(0.3174%)	(0.08821%)	(0.16255%)
Fig. 11 (6)	10.3814	0.076367	0.023288	0.03009
		(0.7928%)	(0.24176%)	(0.31235%)
Fig. 12	8.3278	0.05609	0.01155	0.021922
		(0.4731%)	(0.09618%)	(0.18256%)

FIGURE 12. Our extracted centerline is highly faithful to the constructed one. From left to right: the manual centerline, the 3D reconstruction from the manual centerline by using the method in [33], and our extraction result from the 3D reconstruction.

C. COMPARISONS

Our method is inspired by the generalized ROSA proposed in [7], which is employed in centerline extraction from objects with generally cylindrical shapes. Therefore, we first compare our method with this method. It is observed in Fig. 13 that our method can successfully avoid the ambiguities introduced by thin, close and long vascular structures and produce convincing results that are consistent with the result from volume data. By contrast, the method proposed in [7] produces a centerline with the incorrect topology. The main reason leading to the incorrect topology in [7] is that extensive non-relevant vertices from other different branches contribute to calculate the optimal cut planes and CPs. In contrast, by incorporating the branch segmentation, we can reduce the non-relevant vertices involved in the calculation and successfully create desirable results.

Second, we perform a comparison with Huang *et al.*'s $L_1-medial$ method [9]. The results are illustrated in Fig. 14. The vasculature centerline obtained from Huang *et al.*'s method misses a number of branches, including the bottom branch at the cross junction and a few small branches on the right part of the input vasculature (look inside the orange box for details). This reason is that it uses only position information in distinguishing points from different parts of the vasculature. It is insufficient since the position information is easily interfered from many nearby structures. In contrast, we achieve a more satisfactory result.

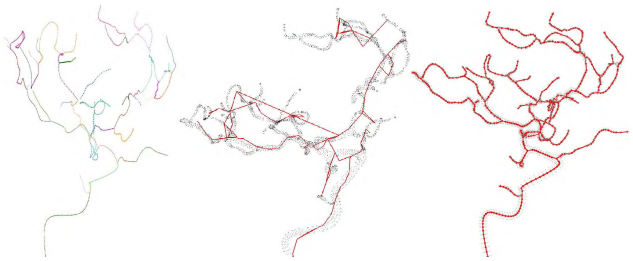


FIGURE 13. Comparison with [7]. The left is considered as the ground truth manually generated from a volume data. The middle is the result of Tagliasacchi *et al.* [7]. The right is our result. Tagliasacchi *et al.* produce a centerline with the incorrect topology, because of extensive non-relevant vertices from different branches involved. By reference to the ground truth centerline, our result is improved, due to the branch segmentation and series of advanced techniques in discrete geometry processing used.

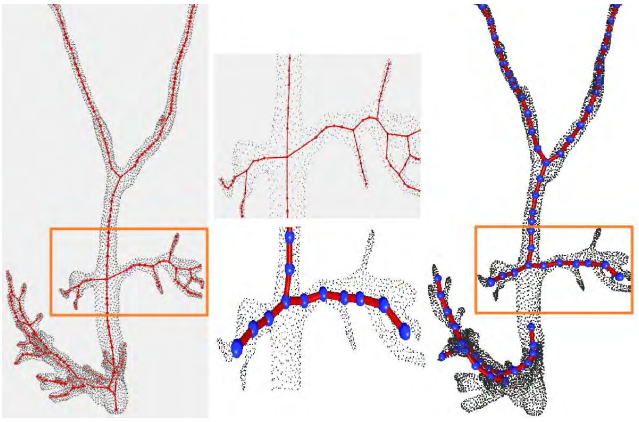


FIGURE 14. Comparison with [9]. The left is our result, the right is Huang *et al.*'s result [9]. Huang *et al.* produce an incomplete centerline due to insufficient information to distinguish vertices from different parts of the vasculature (see the magnified fragments in the middle column). We employ vasculature segmentation and PCA to carefully extract geometric and topological information of vertices, in order to achieve a more satisfactory result.

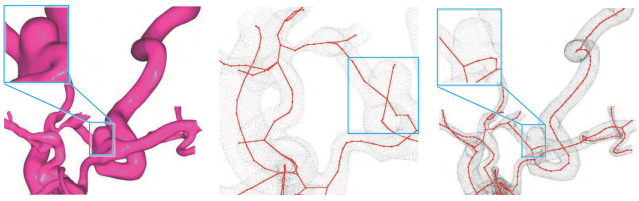


FIGURE 15. Comparison with [8] on a vasculature with a tumor. (a) The surface geometry of the vasculature and the tumor is highlighted. (b) Our method is sensitive to vascular tumors. Our method can handle the vascular tumor well by segmenting the tumor as a small branch. (c) Wang *et al.*'s [8] method produces no curves in this area (different/incorrect topology compared to the original data), because the mesh contraction scheme that is used tends to collapse the tumor to one centerline point in their method.

Third, we further compare with the method of Wang *et al.* [8] on a vasculature model with tumors. Our method can effectively handle the tumors as other parts of the vasculature, as shown in Fig. 15(b), since the tumor is

segmented as a small branch in the segmentation stage. In this regard, the topology of this small tumor branch is successfully included in the centerline. This is very important for some virtual medical diagnosis and treatment procedures, such as intervention radiology, where surgeons usually need to locate tumors along the centerline of the vasculature. Whereas, the tumor collapses to one point in Wang *et al.*'s [8] result.

Fourth, we compare the simulation quality of our centerline with the result of Wang *et al.* [8] (which is tailored for handling the vasculature). in a computer-assisted diagnosis and treatment application. Interventional radiology is widely used in the treatment of cardiovascular diseases. Computer-based training simulators provide solutions to overcome some drawbacks of the traditional apprenticeship training [34]. The simulation of interventional radiology is very complicated, involving many computation-intensive tasks. Among these tasks, the angiography, which is based on the contrast agents diffusion process, is an important cue in interventional radiology that can allow surgeons to manipulate the catheters and guidewires in a vasculature. Therefore, the realistic simulation of the contrast agents diffusion process is essential for a successful simulator.

We simulate this diffusion process based on smoothed particle hydrodynamics (SPH). The blood flow in each vasculature is modeled as an incompressible viscous fluid flowing through the vasculature. Based on biomechanical research, the velocity of blood flow usually shows a quadratic decrease from the vessel center to the vessel wall. In this case, we should realistically simulate this flow pattern when contrast agents flow with blood, once they are injected into the vasculature.

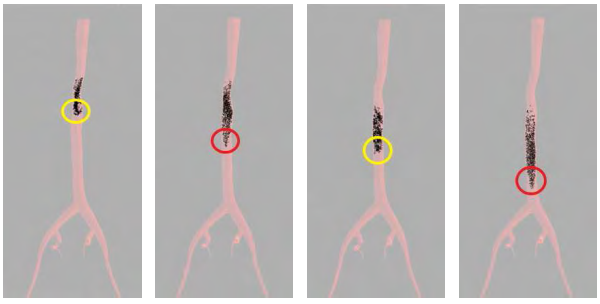


FIGURE 16. The simulation results of contrast agents with the repulsive forces calculated based on the centerlines generated from Wang *et al.*'s method: (a) and (c), and based on the centerlines generated from our method (b) and (d). (a) and (b) are one group for comparison, while (c) and (d) are another group for comparison. Regions highlighted with yellow circles and red circles show that our method can produce more realistic simulation results.

In order to simulate this effect, according to the distances between particles and the centerline of the vasculature, we can construct a varying repulsive force to each particle in the flow. In this case, the centerline plays an important role in ensuring the realism of the flow simulation. As shown in Fig. 16, compared with the cases where repulsive forces are calculated based on the centerline generated from Wang *et al.*'s method [8], the distribution of particles and the flow pattern are more realistic when exerting the repulsive forces calculated based on the centerline generated by our proposed method.

D. DISCUSSION

First, our algorithm seamlessly integrates a number of advanced techniques in discrete geometry processing to produce centerlines of the complex vasculature. However, some of these techniques involve complicated, and computationally intensive steps (e.g., CP thinning). Therefore, the whole processing pipeline is relatively time-consuming. The current implementation of the algorithm is only suitable for the off-line extraction of the high-quality vasculature centerlines (see the timing on the vascular models in Table 2, the approach is implemented by using VC++ and OpenGL; the experiments are performed on a PC with a 2.9 GHz Intel core i5 and 8 GB of RAM). In the future, we will attempt to accelerate some parts of the proposed approach by employing GPUs.

TABLE 2. Timing (minute) for models in Fig. 11 (from left to right).

#V&T	#V&T	#V&T	#V&T	#V&T	#V&T
(15.6k, 2.3)	(23.6k, 5.5)	(8.7k, 1.7)	(91.6k, 10.6)	(41k, 5.6)	(48.9k, 6.1)

Second, our algorithm has to work with the vasculature containing enough samples on the vessel walls, so that the shape of the general cylinder can be maintained. Otherwise, it fails to extract the centerlines, as shown in Fig. 17. A potential solution to this is to explicitly re-sample the vessel wall mesh with denser points.

FIGURE 17. An undersampled vascular structure does not work well with our method, since it can easily violate our assumption of having enough sample points to reveal the cylindrical shape of vessel branches.

Third, we assume that, the vasculature is generally composed of piecewise cylindrical shapes. Based on this assumption, we improve an existing K-means fuzzy clustering to segment it into multiple branches. The clustering method adopted here may not be the only choice, it is possible to refer to [35] for inspiring more ideas on the vasculature segmentation problem.

Fourth, medical surface meshes reconstructed by iso-surfaced methods (such as Marching Cubes) contain a high

degree of noise and staircase artifacts. A smart method, which is performed on these coarse data without sensitivity of artifacts, is more welcome, because several preprocessing steps may be avoided by users.

Fifth, bridging the CPs at the endpoints of multiple branches is somewhat arbitrary (bifurcation/trifurcation bias). The Laplacian smoothing may introduce the artifacts, i.e., the centerlines at around joints may prefer to branches, as shown in 4(f).

In addition, when discussing flow dynamics, we take into account only the laminar and not the turbulent flow. Calculations, such as tortuosity, hydraulic diameter or hydraulic ratio, would be helpful.

VI. CONCLUSION

Extracting centerlines of the vasculature represented by surface meshes is a challenging problem in the fields of digital medicine and computer-aided diagnosis. This work is intended to support vasculature-related virtual surgery, and to potentially supply doctors with more vasculature information in the progress of disease diagnosis with the possibility of reducing human subjective errors. We have proposed an effective centerline extraction approach for 3D vasculature surface meshes. Our approach is inspired by an observation that the vasculature consists of meaningful components in the form of general cylinders, and a vascular system can be decomposed into a variety of piecewise cylindrical branches. Experimental results demonstrate that our method can completely and accurately extract centerlines from complicated vascular models. The proposed method has significant potential for use in computer-assisted interventions for vascular diseases.

APPENDIX

CENTERLINE POINTS (CPs) GENERATION

In order to make the CP rotationally symmetric about the mesh vertex normals of S, the normal v_p of one CP is calculated by minimizing

$$\arg\min \sum_{i=1}^{N} var\langle v_p, v_i\rangle, \tag{8}$$

where v_i is the normal of an arbitrary vertex in S, var represents variations, and $var\langle\cdot\rangle$ measures the angle between two vectors. Eq. 8 has a closed form solution which can be dealt by singular value decomposition (SVD). Eq. 8 can be re-written as one which minimizes the quadratic form $v_p^T M v_p$ with matrix

$$M = \begin{bmatrix} \overline{X^2} - \overline{X}^2 & 2\overline{XY} - 2\overline{X}\,\overline{Y} & 2\overline{XZ} - 2\overline{X}\,\overline{Z} \\ 2\overline{XY} - 2\overline{X}\,\overline{Y} & \overline{Y^2} - \overline{Y}^2 & 2\overline{YZ} - 2\overline{Y}\,\overline{Z} \\ 2\overline{XZ} - 2\overline{X}\,\overline{Z} & 2\overline{YZ} - 2\overline{Y}\,\overline{Z} & \overline{Z^2} - \overline{Z}^2 \end{bmatrix}, \tag{9}$$

where X denotes a random variable for the x-component of the point normals in S and $\overline{X}$ denotes the average of these x-components. That is the same for Y, $\overline{Y}$, Z and $\overline{Z}$, respectively. The quadratic problem can be solved analytically using singular value decomposition (SVD).

To guarantee the centeredness of the centerline point, the position x_{cp} is calculated by minimizing the sum of squared distances from the centerline point to the line extensions of the mesh vertex normals in a subset S:

$$\arg\min \sum_{i=1}^{N} \| (x_{cp} - x_i) \times v_i \|^2, \tag{10}$$

where $p_i = (x_i, v_i)$ is a vertex of an oriented mesh S, N is the size number of S, and $(x_{cp} - x_i) \times v_i$ is the cross product of two vectors. Eq. 10 can be easily solved by straightforward differentiation.

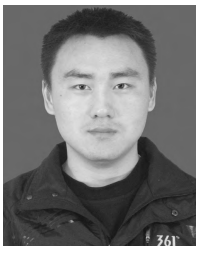
ACKNOWLEDGMENT

The authors thank the anonymous reviewers for their valuable comments. *M. Wei and Q. Wang equally contributed to this work.*

REFERENCES

[1] H. Wang, J. Wu, M. Wei, and X. Ma, "A robust and fast approach to simulating the behavior of guidewire in vascular interventional radiology," *Comput. Med. Imag. Graph.*, vol. 40, pp. 160–169, Mar. 2015.
[2] M. Schneider, S. Hirsch, B. Weber, G. Székely, and B. H. Menze, "Joint 3-D vessel segmentation and centerline extraction using oblique Hough forests with steerable filters," *Med. Image Anal.*, vol. 19, no. 1, pp. 220–249, 2015.
[3] B. V. Stimec, J. H. D. Fasel, and D. Ignjatovic, "3D reconstruction of a primary aortoenteric fistulaŰcenterline calculation and measurements," *Current Med. Imag. Rev.*, vol. 11, no. 2, pp. 127–131, 2015.
[4] L. Liang *et al.*, "Nonrigid iterative closest points for registration of 3D biomedical surfaces," *Opt. Lasers Eng.*, vol. 100, pp. 141–154, Jan. 2018.
[5] G. Bertrand and M. Couprie, "Isthmus based parallel and symmetric 3D thinning algorithms," *Graph. Models*, vol. 80, pp. 1–15, Jul. 2015.
[6] M. Wei *et al.*, "Morphology-preserving smoothing on polygonized isosurfaces of inhomogeneous binary volumes," *Comput.-Aided Des.*, vol. 58, pp. 92–98, Jan. 2015.
[7] A. Tagliasacchi, H. Zhang, and D. Cohen-Or, "Curve skeleton extraction from incomplete point cloud," *ACM Trans. Graph.*, vol. 28, no. 3, p. 71, 2009.
[8] S. Wang, J. Wu, M. Wei, and X. Ma, "Robust curve skeleton extraction for vascular structures," *Graph. Models*, vol. 74, no. 4, pp. 109–120, 2012.
[9] H. Huang *et al.*, "L_1-medial skeleton of point cloud," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 65:1–65:8, 2013.
[10] K. Palágyi and A. Kuba, "A parallel 3D 12-subiteration thinning algorithm," *Graph. Models Image Process.*, vol. 61, no. 4, pp. 199–221, 1999.
[11] K. Palágyi, "Parallel 3D 12-subiteration thinning algorithms based on isthmuses," in *Proc. 9th Int. Symp. Vis. Comput. (ISVC)*, Rethymno, Greece, Jul. 2013, pp. 87–98.
[12] R. J. T. Sadleir and P. F. Whelan, "Fast colon centreline calculation using optimised 3D topological thinning," *Comput. Med. Imag. Graph.*, vol. 29, no. 4, pp. 251–258, 2005.
[13] M. Wan, F. Dachille, and A. E. Kaufman, "Distance-field based skeletons for virtual navigation," in *Proc. IEEE Vis.*, Oct. 2001, pp. 239–246.
[14] M. S. Hassouna and A. A. Farag, "Variational curve skeletons using gradient vector flow," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2257–2274, Dec. 2009.
[15] R. Cárdenes, H. Bogunovic, and A. F. Frangi, "Fast 3D centerline computation for tubular structures by front collapsing and fast marching," in *Proc. Int. Conf. Image Process. (ICIP)*, Hong Kong, Sep. 2010, pp. 4109–4112.
[16] P. Y. Teng, A. M. Bagci, and N. Alperin, "Automated prescription of an optimal imaging plane for measurement of cerebral blood flow by phase contrast magnetic resonance imaging," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 9, pp. 2566–2573, Sep. 2011.

[17] J. Starzynski, Z. Krawczyk, B. Chaber, and R. Szmurlo, "Morphing algorithm for building individualized 3D skeleton model from CT data," *Appl. Math. Comput.*, vol. 267, pp. 655–663, Sep. 2015.
[18] M. Livesu, F. Guggeri, and R. Scateni, "Reconstructing the curve-skeletons of 3D shapes using the visual hull," *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 11, pp. 1891–1901, Nov. 2012.
[19] L. Wade and R. E. Parent, "Automated generation of control skeletons for use in animation," *Vis. Comput.*, vol. 18, no. 2, pp. 97–110, 2002.
[20] A. Sharf, T. Lewiner, A. Shamir, and L. Kobbelt, "On-the-fly curve-skeleton computation for 3D shapes," *Comput. Graph. Forum*, vol. 26, no. 3, pp. 323–328, 2007.
[21] V. Kurlin, "A one-dimensional homologically persistent skeleton of an unstructured point cloud in any metric space," *Comput. Graph. Forum*, vol. 34, no. 5, pp. 253–262, 2015.
[22] Y.-S. Wang and T.-Y. Lee, "Curve-skeleton extraction using iterative least squares optimization," *IEEE Trans. Vis. Comput. Graphics*, vol. 14, no. 4, pp. 926–936, Jul. 2008.
[23] M. S. Hassouna and A. A. Farag, "Robust centerline extraction framework using level sets," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 458–465.
[24] O. K.-C. Au, C.-L. Tai, H.-K. Chu, D. Cohen-Or, and T.-Y. Lee, "Skeleton extraction by mesh contraction," *ACM Trans. Graph.*, vol. 27, no. 3, p. 44, 2008.
[25] V. Pascucci, G. Scorzelli, P. T. Bremer, and A. Mascarenhas, "Robust on-line computation of Reeb graphs: Simplicity and speed," *ACM Trans. Graph.*, vol. 26, no. 3, pp. 58–71, 2007.
[26] P. Li, B. Wang, F. Sun, X. Guo, C. Zhang, and W. Wang, "Q-MAT: Computing medial axis transform by quadratic error minimization," *ACM Trans. Graph.*, vol. 35, no. 1, 2015, Art. no. 8.
[27] H. C. Thomas, E. L. Charles, and L. R. Ronald, *Introduction of Algorithms*. New York, NY, USA: McGraw-Hill, 2001.
[28] V. de Silva and J. B. Tenenbaum, "Global versus local methods in nonlinear dimensionality reduction," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 705–712.
[29] J. Lu, Y. Diaz-Mercado, M. Egerstedt, H. Zhou, and S.-N. Chow, "Shortest paths through 3-dimensional cluttered environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2014, pp. 6579–6585.
[30] Y. Zhou and Z. Huang, "Decomposing polygon meshes by means of critical points," in *Proc. 10th Int. Multimedia Modelling Conf.*, Jan. 2004, pp. 187–195.
[31] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 8, pp. 841–847, Aug. 1991.
[32] Y. Zheng, C.-L. Tai, and O. K.-C. Au, "Dot scissor: A single-click interface for mesh segmentation," *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 8, pp. 1304–1312, Aug. 2012.
[33] J. Wu, Q. Hu, and X. Ma, "Comparative study of surface modeling methods for vascular structures," *Comput. Med. Imag. Graph.*, vol. 37, no. 1, pp. 4–14, 2013.
[34] S. Li *et al.*, "A catheterization-training simulator based on a fast multigrid solver," *IEEE Comput. Graph. Appl.*, vol. 32, no. 6, pp. 56–70, Nov. 2012.
[35] Y. Zhou, K. Yin, H. Huang, H. Zhang, M. Gong, and D. Cohen-Or, "Generalized cylinder decomposition," *ACM Trans. Graph.*, vol. 34, no. 6, p. 171, 2015.



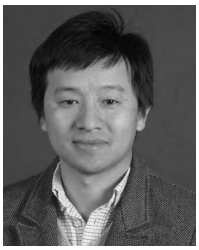
MINGQIANG WEI received the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong, in 2014. He is currently an Associate Professor with the Nanjing University of Aeronautics and Astronautics, China. His research interests include computer graphics and data mining.

QIONG WANG received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong, Hong Kong. She is currently an Associate Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. Her research interests include human-computer interaction and computer-assisted surgery.

YICHEN LI received the master's degree in educational engineering from Nanjing Normal University, China, in 2012. He is currently a Senior Engineering with DMS Tech. Co., Ltd., Beijing, China. His research interests include computer graphics and computer vision.

WAI-MAN PANG received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong in 2008. He is currently an Associate Professor with the School of Computing and Information Science, Caritas Institute of Higher Education, Hong Kong. His research interests include non-photorealistic rendering and healthcare related simulations.

LUMING LIANG received the Ph.D. degree from the Department of Electrical Engineering and Computer Science, Colorado School of Mines, Golden, CO, USA, in 2014. He is currently a Software Engineer with Uber. His research interests include finding shape correspondences and image analysis.



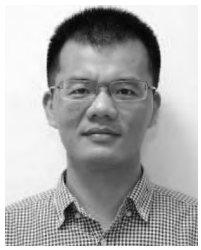
JUN WANG received the Ph.D. degree in computer-aided design from the Nanjing University of Aeronautics and Astronautics (NUAA) in 2007. He is currently a Professor with NUAA. His current research interests include geometry processing and geometric modeling.

KELVIN KIAN LOONG WONG received the Ph.D. degree in electrical and electronic engineering from The University of Adelaide, Adelaide, SA, Australia, in 2009. He is currently with the School of Medicine, Western Sydney University, Sydney, NSW, Australia. His research interests include virtual surgery and medical image analysis.

DEREK ABBOTT (M'85–SM'99–F'05) was born in London, U.K., in 1960. He received the B.Sc. degree (Hons.) in physics from Loughborough University, Leicestershire, U.K., in 1982, and the Ph.D. degree in electrical and electronic engineering from The University of Adelaide, Adelaide, SA, Australia, in 1995, under the supervision of K. Eshraghian and B. R. Davis. From 1978 to 1986, he was a Research Engineer with the GEC Hirst Research Centre, London. From 1986 to 1987, he was a VLSI Design Engineer with Austek Microsystems, Australia. Since 1987, he has been with The University of Adelaide, where he is currently a Full Professor with the School of Electrical and Electronic Engineering. He coedited *Quantum Aspects of Life* (London, U.K.: Imperial College Press, 2008), co-authored *Stochastic Resonance* (Cambridge, U.K.: Cambridge University Press, 2012), and co-authored *Terahertz Imaging for Biomedical Applications* (New York, NY, USA: Springer-Verlag, 2012). He holds over 800 publications/patents and has been an invited speaker at over 100 institutions. His research interests include the area of multidisciplinary physics, electronic engineering applied to complex systems, stochastics, game theory, photonics, biomedical engineering, and computational neuroscience.

Prof. Abbott is a fellow of the Institute of Physics. He received the number of awards, including the South Australian Tall Poppy Award for Science in 2004, the Premier's SA Great Award in Science and Technology for outstanding contributions to South Australia in 2004, and an Australian Research Council Future Fellowship in 2012. With his colleagues, he received the IEEE SENSORS Journal Best Paper Award in 2014. He received the David Dewhurst Medal for biomedical engineering in 2015. He has served as an Editor and/or Guest Editor for a number of journals, including the IEEE JOURNAL OF SOLID-STATE CIRCUITS, the *Journal of Optics B*, *Microelectronics Journal*, *Chaos*, *Smart Structures and Materials*, *Fluctuation and Noise Letters*, *PLOS One*, the PROCEEDINGS OF THE IEEE, the IEEE PHOTONICS JOURNAL. He is currently on the Editorial Boards of *Scientific Reports* (Nature), *Royal Society Open Science*, the IEEE Access, and *Frontiers in Physics*.

JING QIN received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong in 2009. He is currently an Assistant Professor with The Hong Kong Polytechnic University from 2016. His research interests include virtual reality and computer graphics.



JIANHUANG WU received the Ph.D. degree from the Shenyang Institute of Automation, Chinese Academy of Sciences, in 2007. He is currently a Professor with the Research Laboratory for Imaging and Digital Surgery, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. His research interests include virtual reality and medical visualization.