

Vessel Visualization using Curved Surface Reformation

Thomas Auzinger*, Gabriel Mistelbauer*, Ivan Baclija, Rüdiger Schernthaner, Arnold Köchl, Michael Wimmer, Meister Eduard Gröller, *Member, IEEE Computer Society*, and Stefan Bruckner, *Member, IEEE Computer Society*

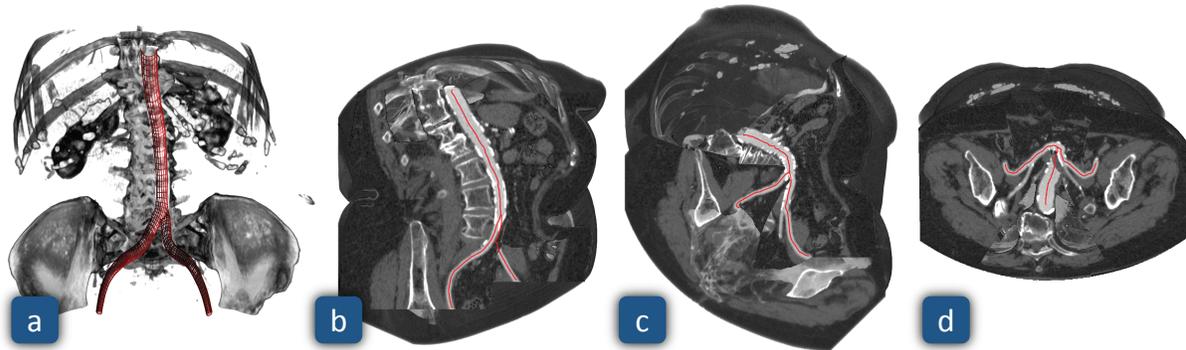


Fig. 1. Bifurcation of the human abdominal aorta. (a) shows a 3D direct volume rendering of the data set together with the surface of the vessel tree (red wireframe). The remaining three images present our approach for visualizing the vessels' interior by generating a view-dependent cut surface through the vasculature. Our method features a smooth cut through surrounding tissue (b), correct vessel lumen portrayal even at grazing angles (c) and the automatic placement of cutaways to allow inspections of large parts of the vasculature in a single view (d). The visibility of the vessel tree is correctly resolved for every view direction.

Abstract— Visualizations of vascular structures are frequently used in radiological investigations to detect and analyze vascular diseases. Obstructions of the blood flow through a vessel are one of the main interests of physicians, and several methods have been proposed to aid the visual assessment of calcifications on vessel walls. Curved Planar Reformation (CPR) is a wide-spread method that is designed for peripheral arteries which exhibit one dominant direction. To analyze the lumen of arbitrarily oriented vessels, Centerline Reformation (CR) has been proposed. Both methods project the vascular structures into 2D image space in order to reconstruct the vessel lumen. In this paper, we propose Curved Surface Reformation (CSR), a technique that computes the vessel lumen fully in 3D. This offers high-quality interactive visualizations of vessel lumina and does not suffer from problems of earlier methods such as ambiguous visibility cues or premature discretization of centerline data. Our method maintains exact visibility information until the final query of the 3D lumina data. We also present feedback from several domain experts.

Index Terms—Reformation, Volume Rendering, Surface Approximation.

1 INTRODUCTION

Vessel diseases, such as arteriosclerosis or pulmonary embolisms, are becoming an increasing concern in our aging society. Thus, the analysis of blood vessels for their diagnosis and treatment are important research fields of radiology. Angiographic interventions, such as stenting, balloon dilatation, or bypass surgery, need to be planned with care and precision, due to their major impact on the patient's health state. In order to optimally decide on the therapeutic procedure, special diagnostic methods are required. They assess the health state of vessels and

answer clinically relevant questions such as, e.g., if blood is partially or entirely hindered from flowing through a vessel by calcifications or soft plaque.

The gold standard for angiographic imaging is Digital Subtraction Angiography (DSA), a non-tomographic procedure. However, with the capabilities of modern medical scanners, less invasive, tomographic methods, like Computed Tomography Angiography (CTA) or Magnetic Resonance Angiography (MRA), are becoming increasingly popular. Both methods require the injection of a contrast agent into the patient's vasculature before the actual scan, to enhance the contrast in the obtained data set.

Confronted with a tomographic data set, special visualization techniques are required to assess possible vessel pathologies. Among these methods are Curved Planar Reformation (CPR) and Centerline Reformation (CR), which create a curved cut through a vessel along its central axis (*centerline*) to visualize its interior (*lumen*). Through cut surface rotation around the centerline of the vessel, they enable the identification of relevant blood flow obstructions or pathologies at the vessel walls. CPR has been initially designed for the visual analysis of peripheral arterial occlusive diseases, where the vessels are predominately oriented along the human lower extremities. Visualizing spatially arbitrarily oriented vessels leads to artifacts with this method in a majority of cases. CR addresses this limitation by employing wave-front propagation for the lumen visualization. However, both methods operate in 2D image space, thus suffering from a premature discretization of the vessel geometry, like decreased precision of centerline location and loss of visibility information.

*These authors contributed equally to this work.

- Thomas Auzinger, Gabriel Mistelbauer, Michael Wimmer and Meister Eduard Gröller are with Vienna University of Technology, Austria.
E-mail: {thomas.auzinger, gmistelbauer, wimmer, groeller}@cg.tuwien.ac.at.
- Stefan Bruckner is with University of Bergen, Norway.
E-mail: stefan.bruckner@uib.no.
- Ivan Baclija is with Kaiser-Franz-Josef Hospital Vienna, Austria.
E-mail: ivan.baclija@wienkav.at.
- Rüdiger Schernthaner is with Medical University of Vienna, Austria.
E-mail: ruediger.schernthaner@meduniwien.ac.at.
- Arnold Köchl is with Landeskrankenhaus Tulln, Austria.
E-mail: arnold.koechl@tulln.lknoe.at.

Manuscript received 31 March 2013; accepted 1 August 2013; posted online 13 October 2013; mailed on 4 October 2013.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

Table 1. Aspects and improvements of Curved Surface Reformation (CSR) with respect to CPR and CR. Our method retains all positive properties of the most relevant state-of-the-art techniques and extends several aspects significantly.

	CPR	CR	CSR
Lumen Rendering	Pixel-Based	Pixel-Based	Ray Casting
Vessel Orientation	Restricted ¹	Unrestricted	Unrestricted
View Direction	Restricted ²	Restricted ²	Unrestricted
Visibility	None	Fair ³	Good
Interactive	Yes	Fair	Yes

¹vessels should run along one major direction

²rotation around one axis

³trade-off between visibility of lumen and surrounding tissue

With Curved Surface Reformation (CSR) we propose a method designed to rectify these shortcomings (cf. Table 1). The main contributions of our work are:

- A method to generate the cut surface along a vessel centerline fully in 3D, thus avoiding limitations of previous 2D methods.
- The correct handling of occlusions of different parts of the vasculature by employing a cost function on the cut surfaces.
- An automatic generation of cutaways on the cut surfaces to reveal as much of the vessel tree as possible.
- A Level of Detail (LOD) approach to smoothly extend the cut surface into the surrounding tissue.
- The ability to freely rotate, translate and zoom into the vasculature at interactive frame rates.
- An outline of the corresponding rendering pipeline and its implementation as well as feedback by domain experts.

The remainder of the paper is structured as follows. In Section 2 related work is presented. Section 3 describes our proposed method in detail from both theoretical and practical points of view. In Section 4 results are demonstrated and discussed. An evaluation is given in Section 5 and the paper is concluded in Section 6.

2 RELATED WORK

The method proposed in this work extends several aspects of CPR and CR, which are shortly described in this section with other related work.

Curved Planar Reformation. Planar cuts through a data set are intended to provide more insight into otherwise obscured internal structures. Kanitsar et al. [16] describe Curved Planar Reformation (CPR) as a curved cut through a data set along the centerline of a single vessel. They present three different types of CPR (projected, stretched, and straightened) with different geometric properties. Multipath Curved Planar Reformation (mpCPR), proposed by Kanitsar et al. [17] and evaluated by Roos et al. [32], extends CPR to multiple vessels. Spiral CPR, also described by Kanitsar et al. [17], shows the interior of a vessel by flattening it along a spiral. This approach preserves isometry, but lacks spatial context. Kanitsar et al. [19] propose untangled CPR to avoid occlusions when projecting the vessel tree by using spatial deformations. Mistelbauer et al. [29] propose Centerline Reformation (CR) as a technique to visualize the lumen of arbitrarily oriented vessels. However, there is a trade-off between the visibility of the vessels and the displayed size of the surrounding tissue. Mistelbauer et al. [28] combine CPRs of several view directions into a single static image by aggregating certain features circularly along vessels. The vessels are shown straightened and rotation operations are unnecessary. Lampe et al. [22] introduce a more general curve-centric reformation that allows transforming the space around a curve. Their method produces visualizations similar to planar reformations used for virtual endoscopy. Lee and Rasch [23] introduce a tangential curved

planar reformation by sweeping the Frenet frame along the projected centerline of a vessel. This creates artifact-free visualizations of the vessel lumen, but not of the surrounding tissue. Saroul et al. [34] propose a technique for flattening free-form surfaces of anatomical medical structures that minimizes distortion along a user-specified curve or direction [33]. The surface is spanned between boundary curves, defined by users interactively placing markers within the 3D volume. CSR creates a view-dependent cut surface of the vasculature without any user interaction.

Context Visualizations. Straka et al. [36] propose the *VesselGlyph* to establish focus-and-context rendering for CPR and mpCPR. It provides the possibility to place vessels within a context visualization like Maximum Intensity Projection (MIP) or Direct Volume Rendering (DVR). Viola et al. [37] discuss a method for retaining context while refocusing on different structures of interest in a volume data set. This is also supported by our technique. DeCarlo et al. [7] present *suggestive contours* as the non-photorealistic rendering of lines in order to provide visual cues of a shape.

Cutaways. Diepstraten et al. [8] present a set of rules for rendering interactive cutaways of polygonal data. Burns and Finkelstein [3] describe view-dependent cutaways for polygonal scenes. Li et al. [24] present a system for creating cutaways of complex objects in real-time. They apply visual styles from conventional illustrations such as various cutting types (box, tube, or window cuts), visual cues (edge shadows and shading) and inset cuts that cut deeper layers of an object with a smaller extent. Sigg et al. [35] propose a degree-of-interest function for interactively specifying important features instead of manually placing them. In an iterative user-centric optimization process, cutaway primitives (cuboids, spheres, or cylinders) are optimally placed in order to allow a clear view of the desired object in the context of polygonal meshes and volumetric data. McInerney and Crawford [27] present a paint roller as an interaction metaphor for specifying the shape of the cutaway manually, while retaining ribbons of the occluding object. Burns et al. [4] use an importance function for generating volumetric contextual cutaways for medical visualization. They obtain a smooth transition between the occluder and the important object. Lidal et al. [25] give design guidelines with the aim to enhance shape and depth of the focused object in relation to its context. McGuffin et al. [26] use volume deformations and a set of interactive widgets to offer users the possibility to virtually browse through different layers of medical volume data. CSR will automatically create cutaways (cf. Figure 1d) to reveal occluded vessels by means of a cost function.

Volume Clipping. Konrad-Verse et al. [21] describe a virtual resection tool for organs employing user-placed meshes. After the user initially paints a stroke on the desired organ, a virtual cutting plane is computed via a principal component analysis. Additionally, the plane can be deformed in order to include or exclude objects from the cutting. Birkeland et al. [1] describe an illustrative clipping surface by means of physical properties of elastic membranes. It can serve as an illustrative slice with local DVR confined to its vicinity. User adjustment is enabled by introducing additional force fields. To avoid cutting near objects, the surface adapts to salient features of the underlying data. Weiskopf et al. [39] describe interactive clipping of convex and concave objects from a volume, while accounting for proper volume shading. Wang et al. [38] propose a method for volumetric sculpting. They apply various operations on the volume at voxel level. Zhang et al. [40] present a real-time clipping strategy for visualizing data inside a volume. They compare several classifications for volume rendering with ray casting. For further information considering physically based deformable models, we refer to the survey by Nealen et al. [30]. In our approach, the volumetric data is cut by a curved surface.

Curve Simplification. In our technique, we simplify centerline curves to generate a coarse representation of the vessel tree to infer the large-scale behavior of the cut surfaces. As an old problem from computational geometry, it was already investigated by Imai and Iri [13, 14] for polygonal curves in the late 80s. The most commonly used algorithm was developed by Ramer [31] and Douglas and Peucker [9]. Recent developments consider an improved distance metric [11] or utilize Bézier curves [5] for simplification.

3 CURVED SURFACE REFORMATION

In the field of radiology, the diagnostic visualization of vascular structures plays a vital part in the investigation of certain vessel pathologies. Among occlusive diseases, stenoses are of particular interest. They can be caused by calcifications or soft plaque on vessel walls. The manual analysis of the raw medical imaging data of vessels is very time consuming, and the use of medical visualization can improve the efficiency dramatically. Visualization methods usually require additional information on the imaging data, such as the locations of the vessel centerlines. These are modeled by radiological assistants in their daily clinical routine, in which they are assisted by semi-automatic detection methods such as the technique of Kanitsar et al. [18]. In the context of our visualization technique, centerline data is treated as input that was generated in a preprocessing step. Specialized visualization algorithms are then used for inspection of regions of interest and detection of pathologies.

General purpose methods can produce misleading visualizations. An example, when using MIP, is that vessels, which have concentric calcifications on their walls, incorrectly appear as completely occluded. To remedy this, planar reformations create a cut through the tissues or, as it is the case for vessel reformation, a cut along the vessels to visualize their interior. These approaches provide a good overview of the vessel lumen and enable the specialist delivering a qualified judgment on the severity of pathologies.

Different vessel reformation methods make various assumptions on the vessel geometry. CPR and its extension for multiple vessels, mpCPR [17], have been designed for the inspection of peripheral arterial occlusive diseases, where vessels strongly follow one principal direction. For display, the vessel is aligned with the up-axis \mathbf{u} , and the view direction is taken orthogonal to it. The cut surface is then spanned by the centerline and the direction that is orthogonal to both \mathbf{u} and the view direction. This entails two major restrictions: the view direction can only rotate around the up-axis, and the cut surface degenerates if the vessel direction becomes orthogonal to \mathbf{u} . The latter case leads to excessive vessel thinning as shown in Figure 7a. A possible workaround to alleviate this effect is the stretching or straightening of the vessels. However, this interferes with the spatial perception of the displayed vessels, which is usually not desired by domain experts. Furthermore, the initial projection of the centerline to 2D discards the visibility information, and overlapping vessels cannot be kept apart.

CR [29] removes some of these drawbacks by using wave-front propagation on the projected vessel data. Although this enables the depiction of the lumen of arbitrarily oriented vessels, it still operates in 2D image space. This leads to a significant loss of visibility information due to the necessary projection. The cut surface is created by spreading the vessel locations into their surroundings. The visibility then has to be reconstructed by walking along the projected vessel tree. This leads not only to performance issues, but also requires the use of thresholds. They effectively determine the method’s ability to either display the surrounding tissue or to correctly resolve the visibility. Figure 7b and Figure 8b provide an illustration of this trade-off.

To address these issues, we propose Curved Surface Reformation (CSR) to perform the lumen rendering entirely in 3D space. This new method allows the determination of the correct visibility of the various cut surfaces along the vessel tree. It also enables the extension of these surfaces into the surrounding tissue in a well-defined manner. Furthermore, we offer the possibility to use common interactions such as zooming, panning, or rotating around arbitrary axes in 3D for the first time in the field of vessel reformations. Our method builds on ray casting and is efficiently implementable on parallel hardware. We provide a CUDA implementation that runs at interactive frame rates.

In Section 3.1 the theory of our method is developed in a continuous setting. Its application on discrete input data is outlined in Section 3.2. Sections 3.3 and 3.4 discuss the resulting rendering pipeline (cf. Figure 4) and its implementation.

3.1 Theory

To render the lumen of a vessel, we employ ray casting of a surface given by a cut through the vessel along its centerline. As we treat arbitrarily oriented vessels and do not require a fixed camera position, the cut surface is generated dynamically, with its geometry depending on both the view direction and the orientation of the corresponding vessel.

In this section, we assume as input to our method a set of differentiable 3D curves, referred to as *vessel segments*, which constitute the centerlines of the relevant part of the vasculature for the visualization task. While our method does not require connected segments and is able to handle arbitrary topologies of the segment graph, the typical use case is a connected set of vessel segments with possible branchings at their endpoints. We will refer to this as the *vessel tree*. Information on the radius of the vessel along the centerline is optional and can be used in the visibility computations (cf. Section 3.1.3). As the geometrical concepts can be presented much clearer in a continuous setting, we take this route and describe the adaptations to the actual, discrete input data in Section 3.2.

3.1.1 Surface Generation

Given the centerline curve $\mathbf{c}(t) : [t_0, t_1] \rightarrow \mathbb{R}^3$ of a vessel segment and a view direction \mathbf{v} , the *local (cut) surface* M is given by

$$M(s, t) = \mathbf{c}(t) + s\mathbf{v} \times \mathbf{c}'(t) \quad \text{with } t \in [t_0, t_1] \text{ and } s \in \mathbb{R}. \quad (1)$$

This definition is motivated by the requirements of having the local surface contain the centerline and be as perpendicular to the view direction as possible. CPR and mpCPR, in comparison, use a fixed frame in the form of $\mathbf{c}(t) + s\mathbf{v} \times \mathbf{u}$, where \mathbf{u} denotes the up-axis, and the view vector is constrained by $\mathbf{v} \cdot \mathbf{u} = 0$.

To show a vessel cross section at the endpoints, we add two half-planes that are perpendicular to the view direction, i.e., for the start $\mathbf{c}(t_0)$ and endpoint $\mathbf{c}(t_1)$, the half-planes are given by $\mathbf{c}(t_0) + s\mathbf{v} \times \mathbf{c}'(t_0) + t\mathbf{v} \times (\mathbf{v} \times \mathbf{c}'(t_0))$ and $\mathbf{c}(t_1) + s\mathbf{v} \times \mathbf{c}'(t_1) - t\mathbf{v} \times (\mathbf{v} \times \mathbf{c}'(t_1))$ for $s \in \mathbb{R}$ and $t \in \mathbb{R}_+$. In the special case of an alignment of the centerline and the view direction ($\mathbf{v} \times \mathbf{c}'(t) = \mathbf{0}$), a half-plane with a boundary perpendicular to the last non-aligned centerline direction has to be added. If the centerline is parallel to the view direction at one of its endpoints, we add a plane perpendicular to the view direction in this point. This ensures that a projection of the resulting local surface covers the whole view plane at least once and that the vessel cross section is visible in case of a collinearity with the view direction.

In a sufficiently small neighborhood of the centerline, this construction provides the correct cut surface for the corresponding vessel segment. For a finite extent of the vessel, however, this naive approach exhibits various deficiencies. Our proposed method on how to rectify these issues to enable a full 3D reformation is presented in the following sections.

3.1.2 Main Challenges

The local surface M , as given in Equation 1, depends on both the view direction \mathbf{v} and the local geometry of the centerline, which is specified by its tangent vector $\mathbf{c}'(t)$. While this is appropriate for a small neighborhood of the centerline, this definition shows two types of undesired behavior when moving away from it. As the vessel lumen near the centerline is the main point of interest, it should be visible whenever it is not directly occluded either by the same centerline or different vessel segments. This behavior is not exhibited by M , as only the front-most parts of the whole vessel tree are visible. The second problem is that small local details or noise are amplified at the outer regions of the surface, which leads to self occlusions and depth discontinuities when viewed along the view direction \mathbf{v} (cf. Figure 3). A solution to the first problem is presented in Section 3.1.3 and to the second one in Section 3.1.4.

3.1.3 Cost Function

To enable inspection of as many vessels as possible in the same view, it is beneficial to reveal the local surface of all those vessel segments that

are not located behind other vascular structures in the current view. Hence, we create cutaways on the local surfaces of vessel segments closer to the viewer based on two simple principles:

1. The closer a point on a centerline is located to the viewer, the more relevant is the associated part of the local surface.
2. The closer a point on a local surface is located to its centerline, the more relevant it is.

With the use of spatial cost functions for all vessel segments, it is possible to mimic this behavior. For each centerline c_i of the vessel tree, we define a cost function

$$C_i(\mathbf{x}) = C_{\parallel}(\mathbf{x}) + \lambda C_{i,\perp}(\mathbf{x}), \quad (2)$$

where λ denotes the weighting between the component parallel to the view vector \mathbf{v} , i.e., $C_{\parallel}(\mathbf{x}) = \frac{\mathbf{x} \cdot \mathbf{v}}{\|\mathbf{v}\|}$, and the one normal to it. The latter is given as the distance $dist_{v,\perp}(\mathbf{x}, c_i)$ between \mathbf{x} and the image of curve c_i when both are projected onto a plane P normal to \mathbf{v} . The intuitive explanation of this procedure is that the parallel cost component penalizes the depth of surfaces while the normal component penalizes the distance from the respective centerline.

By evaluating the cost function on all local surfaces and projecting these values onto a plane P normal to \mathbf{v} , we obtain several cost values per point on P . Actually, each local surface contributes one or more cost values to each point on P , as they cover the full extent of P due to the nature of their construction. See Figure 2c for a local surface that contributes six cost values for each point on P in the center region of the trefoil knot. Another illustration can be found in Figure 6d. At each point we now select the smallest cost value and mark the corresponding surface location as visible. As a result, for each point on the view plane the visible local surface is selected and the global visibility is resolved (cf. Figure 2). Ambiguities that arise from several identical cost values at the same point of P can be neglected almost always as these are part of one-dimensional boundaries where the visibility changes between different centerlines. In very rare cases it is possible that whole local surface regions contribute the same cost values and that artifacts similar to z-fighting arise. As this behavior only occurs for specific view directions (we conjecture that these form a one-dimensional subset of the two-dimensional space of all view directions), we feel safe to ignore them as we did not encounter them in any of our experiments.

The parameter λ in the cost function encodes the preference for broader or thinner surrounding surfaces around vessels. A large λ heavily penalizes the projected distance, and thus the cutaways of occluded vessels reach close to the visible centerline. The contrary case of a small λ leads to a more liberal occlusion of vessels behind the current centerline.

A minor modification can be applied to the cost function to guarantee the preservation of the full extent of vessels, such that cutaways do not cut into the vessel lumen of a visible centerline. This can occur in the case of two centerlines with a small depth difference, as the cost function indicates a preference for selecting the occluded vessel very near to the centerline of the occluding vessel. To avoid this situation, the normal component of the cost function has to be modified to assign no distance cost to regions around the centerline within a certain vessel radius. This can be considered as placing a strip with possibly varying vessel radius around each centerline, in which the normal cost component of the associated cost function is zero. As a consequence, a vessel behind another centerline will always have a higher cost as both the depth and distance component contribute to its occlusion. Note that for very dense vascular structures, a large number of cutaways is generated, and the local surface of most visible vessels will not exceed the extent of the vessel itself. In this case, our method would resemble dense line rendering and due to conceptual similarities, only slight modifications would ensure convergence to the illustrative method of Everts et al. [10].

Our procedure resolves the visibility of the local surfaces of the vessel tree and thus remedies the first problem with the initial definition of the local surfaces. In the following section, the second challenge as described in Section 3.1.2 is tackled.

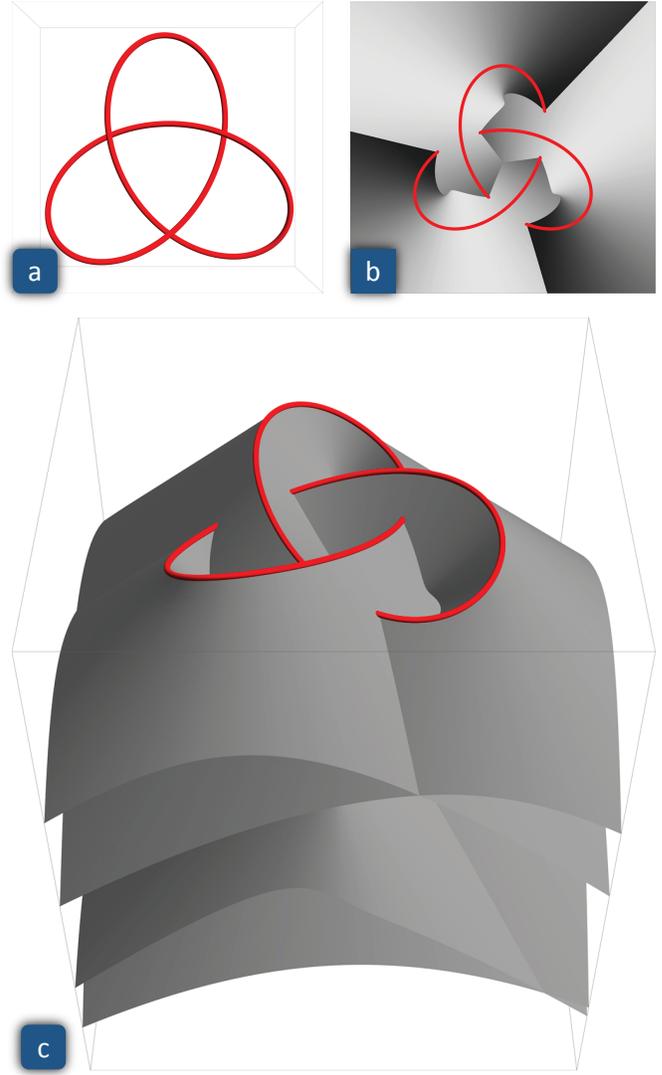


Fig. 2. Illustration of the visibility resolution using a cost function. (a) shows a synthetic vessel input in the form of a trefoil knot. We use a cost function as given by Equation 2 defined on the generated local surface to determine its visible points, which are characterized by the lowest cost value along the view direction. The cost function for a frontal view of (a) is visualized in (c) by encoding the *cost value* as the depth of the local surface. Note that each vertical view ray intersects the local surface several times, and only the topmost intersection points constitute the visible surface. The final result in form of a depth image is given in (b), where the grayscale encodes the depth of the local surface.

3.1.4 Centerline Simplification

Although the local surface M , as defined in Equation 1, is geometrically appropriate in a neighborhood of its respective centerline, globally it exhibits an undesired behavior (cf. Figure 3). This is caused by the use of centerline derivations in the definition of M , as small geometrical details along the vessel get amplified with increasing distance from the centerline. Generally, this leads to a multitude of undesired self-occlusions of the local surface when viewed along the view direction \mathbf{v} .

To combat this behavior, we propose a Level of Detail (LOD) approach based on centerline simplification. While small centerline details have to be accounted for to enable a faithful representation of the vessel geometry close to it, only the large-scale characteristics of the centerline should govern the surface behavior in the surrounding volume. It would be possible to develop a theory of a continuous sim-

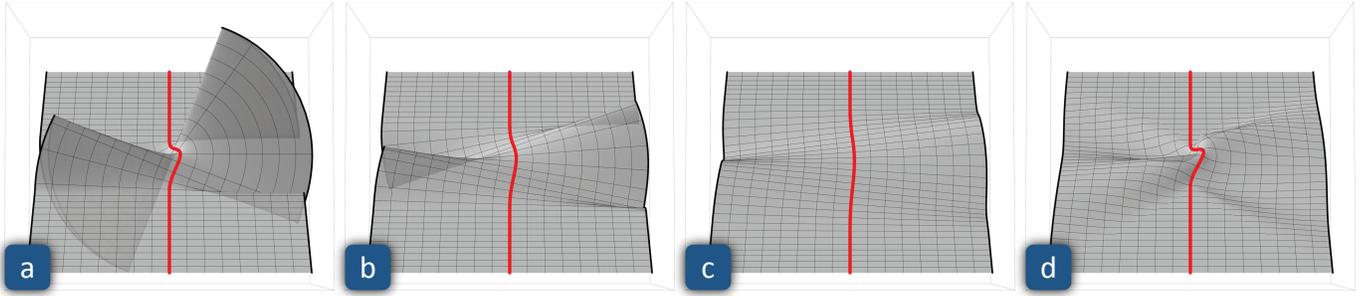


Fig. 3. Illustration of the instability of the local surface with respect to small perturbations of the centerline (red). (a) exhibits severe distortions by a small scale detail of the centerline leading to multiple self occlusions of the local surface. These can be drastically reduced by progressive smoothing of the centerline as shown in (b) and (c). Our proposed Level of Detail (LOD) method uses coarser and smoother representations of the centerline for increasingly distant parts of the surface. This eliminates the undesired self occlusions but preserves the detailed vessel geometry close to the centerline (d).

plification of the centerline, yielding infinitely many LODs. However, we employ just a finite number of discrete levels to perform the simplification, which is justified by the discrete nature of the actual input data and the fact that the continuous formulation would not aid further understanding.

For a fixed number $(N + 1)$ of LODs, each centerline is simplified to obtain N increasingly smoothed versions of it, where the initial curve is preserved in level 0. We specify the actual simplification method for discrete input data in Section 3.2.2.

The visibility between the local surfaces of the smoothed versions of a centerline is again resolved with the help of the cost function by evaluating it on the local surfaces of relevant LODs. Depending on the projected distance d between the centerline of a vessel segment and a point on the view plane, we determine the two most relevant LODs k_0 and k_1 by using a selection function $f(d)$ to obtain

$$k_0 = \min(N, \lfloor f(d) \rfloor) \quad \text{and} \quad k_1 = \min(N, \lfloor f(d) + 1 \rfloor).$$

To concentrate the initial LODs around the centerlines we use a sublinear selection function; in our implementation of the form $f(d) \propto \sqrt{dN}$.

As described in Section 3.1.3, we project the cost values of the two surfaces onto the view plane and, for each LOD separately, we chose the surface locations \mathbf{i}_0 and \mathbf{i}_1 with minimal cost for each point on the view plane. The final *cut surface* location \mathbf{i} is then given as a linear blend of the two LODs, i.e.,

$$\mathbf{i} = (k_1 - f(d))\mathbf{i}_0 + (f(d) - k_0)\mathbf{i}_1.$$

This gives a smooth progression between the LODs and a gradual simplification of the local surface as a function of distance d . Note that beyond the last LOD, no blending is performed, since $k_0 = k_1 = N$. A depiction of the result of this method is given in Figure 6e.

This concludes the theory section for our proposed method. Before describing the actual rendering pipeline, we outline the necessary adaptation of the aforementioned methods to handle discrete input.

3.2 Discrete Geometry

The theory in Section 3.1 was developed in a continuous setting, which does not reflect actual computation hardware. This section covers the required discretization steps necessary for numerical processing.

3.2.1 Surface Generation

Assuming that a vessel centerline is given as a piecewise linear curve, the corresponding local surface consists of planar pieces, namely line stripes, connecting triangles and half-planes at the endpoints. In this section, we give a short description on how to generate such a local surface.

For a line segment of the vessel tree given by the points \mathbf{l}_0 and \mathbf{l}_1 , a line stripe is generated by extruding the line segment in direction $\mathbf{v} \times (\mathbf{l}_1 - \mathbf{l}_0)$ (cf. Figure 6a). As this direction is orthogonal to the

view direction \mathbf{v} , the stripe is located between the depths $d_0 = \mathbf{v} \cdot \mathbf{l}_0$ and $d_1 = \mathbf{v} \cdot \mathbf{l}_1$ along the view direction.

If two consecutive line segments L_0 and L_1 , given by the three points \mathbf{l}_0 , \mathbf{l}_1 and \mathbf{l}_2 , are not collinear when projected onto the view plane, a gap would arise in the projected local surface in case only line stripes were used. To avoid this, we use a triangle that originates in the connection point \mathbf{l}_1 . The triangle edges extend along the boundaries B_0 and B_1 of the line stripes given by $B_0 = \{\mathbf{l}_1 + s\mathbf{v} \times (\mathbf{l}_1 - \mathbf{l}_0)\}$ and $B_1 = \{\mathbf{l}_1 + s\mathbf{v} \times (\mathbf{l}_2 - \mathbf{l}_1)\}$. This ensures that the triangle fills the gap between B_0 and B_1 . As such a gap only arises on one side (when projected onto the view plane), we choose either $s \in \mathbb{R}_+$ or $s \in \mathbb{R}_-$.

At the start and end point \mathbf{l}_0 (resp. \mathbf{l}_n) of a vessel segment, we add half-planes H_0 (resp. H_n) to complete the local surface. These are defined as $H_0 = \{\mathbf{l}_0 + s\mathbf{v} \times (\mathbf{l}_1 - \mathbf{l}_0) + t\mathbf{v} \times (\mathbf{v} \times (\mathbf{l}_1 - \mathbf{l}_0))\}$ and $H_n = \{\mathbf{l}_n + s\mathbf{v} \times (\mathbf{l}_n - \mathbf{l}_{n-1}) - t\mathbf{v} \times (\mathbf{v} \times (\mathbf{l}_n - \mathbf{l}_{n-1}))\}$ with $s \in \mathbb{R}$ and $t \in \mathbb{R}_+$.

If a line segment L is collinear with the view direction, we add an additional half-plane using the definition of H_n above, but substituting the points \mathbf{l}_{n-1} and \mathbf{l}_n by the endpoints of the last non-collinear line segment before L along the centerline. If the first or last line segment of a centerline is collinear, we create an additional plane normal to \mathbf{v} that contains the corresponding endpoint.

These constructions ensure that the piecewise planar surface of each centerline covers the whole view plane when projected onto it as illustrated in Figure 6b. As we employ ray casting to evaluate the cost function on this surface, we can use simple geometrical intersection tests between the rays and the unbounded polygons. The cost function of Section 3.1.3 can be directly used for this purpose.

3.2.2 Centerline Simplification

The LODs of the vessel tree are computed in a preprocessing step. We employ progressive filtering with spline functions to generate coarser representations of the individual vessel segments. The centerline of a given LOD is approximated by a cubic B-spline and resampled to generate the next LOD. The performance of this simple approach is sufficient, and the quality gains by employing more sophisticated methods such as the curve simplification algorithm by Chuon et al. [5] are negligible. Due to the absence of fine details in higher LODs, it is possible to faithfully represent the vessel segments with less line segments. In our implementation, we pursue an aggressive pruning as we drop half of the points for each successive LOD. For a sufficiently dense initial sampling of centerlines as in our input, we did not observe any artifacts resulting from this optimization, which drastically improves runtime.

As described in Section 3.1.4, our LOD approach uses the selection function $f(d)$ depending on the projected distance d of the ray from the centerline. This value can be easily determined by computing the distance between the ray and the line segments of the initial centerline, i.e., the unsmoothed LOD 0.

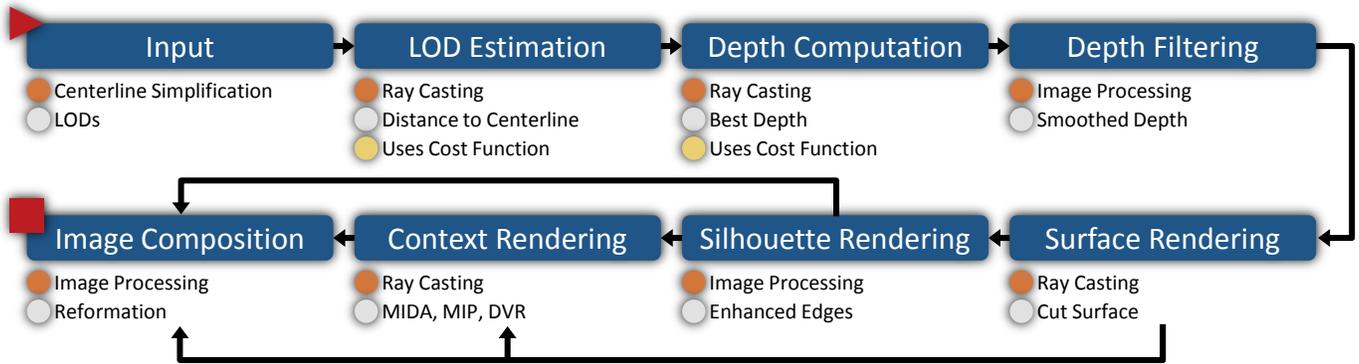


Fig. 4. Rendering pipeline of Curved Surface Reformation (CSR). The bullet denotes the method and the bullet the output of each stage, whereas the bullet indicates the stages where the cost function is used.

3.3 Rendering

This section presents the rendering pipeline of CSR as outlined in Figure 4. It takes a vessel tree, consisting of line segments, and a volume representation of the surrounding tissue as input and outputs an image of the cut surfaces along the vessel centerlines. As we employ ray casting at several stages of the pipeline, various buffers in the size of the final output image are used to store intermediate values. A sequence of such intermediate results is given in Figure 6. In the exceedingly rare case of z-fighting artifacts (cf. Section 3.1.3), we prevent their aggravation with temporal flickering by maintaining a strict rendering order of the input data. As a preprocessing step, the LODs of the vessel tree are computed as described in Section 3.2.2.

3.3.1 LOD Estimation

This stage generates a distance buffer that stores the projected distance of each image location to the nearest centerline of the vessel tree. This data is used as the argument d for the LOD selection function $f(d)$ as described in Section 3.1.4.

3.3.2 Depth Computation

Using the distance information of the previous stage, we determine the two relevant LODs using the method of Section 3.1.4. By using ray casting, we collect the cost values for all intersection points with local surfaces along the ray. This is done separately for each pair of LODs, and we retain the intersection points with the minimal cost for each pixel. The final intersection point is a weighted blend between the respective LOD intersections to ensure smooth progressions from one LOD to the next. This stage outputs the depth of the final intersection point to a depth buffer.

3.3.3 Depth Filtering

To remove aliasing artifacts that arise from undersampling of a high-frequency depth signal caused by fine geometric details of the vasculature, the depth image is convolved with a smoothing filter. This is necessary when sampling cut surfaces, which extend from centerlines that are nearly collinear with the view direction, since their planar parts shrink to sub-pixel sizes. We use a Gaussian filter kernel with standard deviation σ for this task and set its full width at half maximum (FWHM) in frequency space to half the Nyquist frequency of the output image, i.e., $\sigma = 2\sqrt{2\ln 2}/\pi$ pixel. The discrete version is cut off at 3σ , which gives a kernel size of 7×7 . The output makes up the depth values of the desired cut surface (cf. Figure 6c).

If the filter size is significantly increased, the depth discontinuities between the visible cut surface regions of different parts of the vessel tree are smoothed. In the final rendering, this leads to steep 'slopes' between the regions. We do not utilize this fact in our method and leave it just as an observation.

3.3.4 Surface Rendering

A simple lookup into the volumetric intensity data set, which surrounds the vessel centerlines, is performed with the depth values of the cut surface. We use trilinear interpolation for this task and obtain the intensity values of the visible vasculature's lumen and of the surrounding tissue. Afterwards, we apply a user-specified windowing function to map these values to grayscale colors and opacities.

3.3.5 Silhouette Rendering

After determining the depth of the cut surface and the subsequent filtering step, depth discontinuities still remain. These are, however, intentional, and constitute the switch from the surrounding of one part of the vessel tree to another one. Since these important spatial cues might not be easily perceivable, we improve the three-dimensional visual perception of the cut surfaces by highlighting these discontinuities. This is achieved by modulating a user-specified silhouette color (yellow in our case) with the response of a customized filter function.

Large depth discontinuities are revealed by applying a Sobel edge-detection filter. However, small discontinuities between spatially close vessel segments from different parts of the vasculature are not captured. To identify these, every point of the projected cut surface is marked with a unique identifier of the corresponding vessel segment (cf. Figure 5a). Small depth discontinuities can only occur along the borders of these influence zones of the vessel segments while branching locations remain unaffected, since they are continuous in depth.

Thus, we amplify the response of the Sobel filter at these borders. We observed that a simple multiplication by a constant factor already yields adequate results and improves the depth perception of vessel occlusions near branching locations. A final smoothing of the combined filter response provides anti-aliasing of the discontinuity lines and gives our final color modulation (cf. Figure 5b). This allows for a better perception of depth discontinuities.

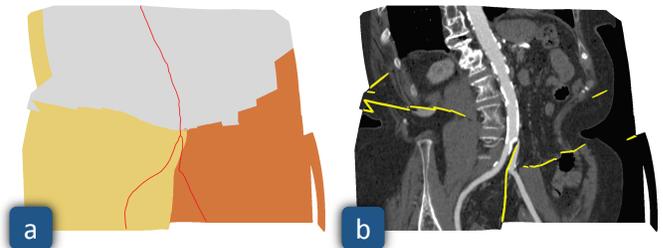


Fig. 5. Silhouette rendering. (a) shows the color coded influence zones of the three vessel segments. Small depth discontinuities at the zones' borders and large discontinuities anywhere are presented as yellow silhouettes in (b) as an overlay of the cut surface.

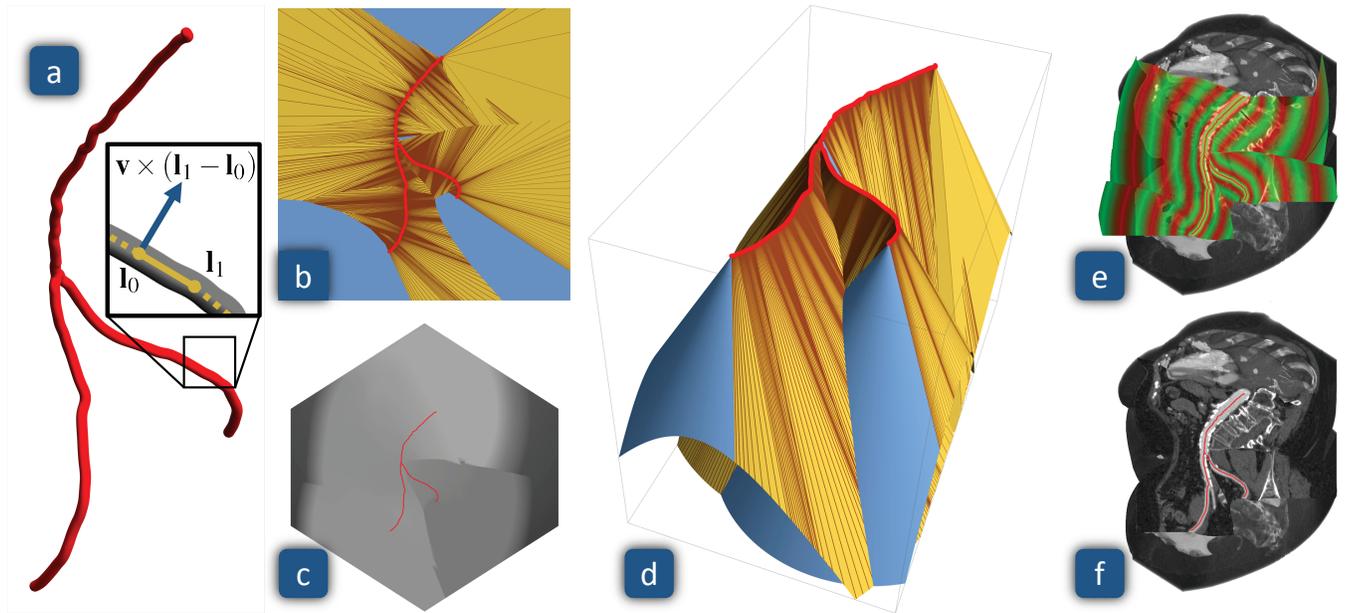


Fig. 6. Illustration of our rendering pipeline. (a) depicts an input vessel tree consisting of 800 line segments. A detailed view of a single line segment illustrates the variables of Section 3.2.1 with the view direction piercing orthogonally through the paper. The geometry of the local surface of the unsmoothed centerlines is shown in (b) where the planar parts are colored according to their type – dark brown for line stripes, light brown for triangles and blue for the half-planes at the endpoints. The final depth buffer after the depth computation and filtering stages is depicted in (c) and the corresponding cost function is illustrated in (d). (e) gives an overview of the LODs which are alternately colored in red and green. It also shows the part of the final output (f) that is covered by the cut surface whereas the other parts are generated by the context rendering stage.

3.3.6 Context Rendering

Due to the fact that the cut surface displays only parts of the volume, the perception of the overview can suffer. To remedy this, we optionally add a final context visualization step to the CSR workflow. Straka et al. [36] propose a focus+context approach by embedding the vessel lumen (focus) into a volume rendering (context). We render a context behind the regions of cut surfaces at locations where they lie outside the volume. When rotating, the context visualization becomes apparent, and the domain expert can clearly perceive the spatial location of the region of interest within the data set. Figure 6f shows such a context visualization, using Maximum Intensity Difference Accumulation (MIDA) as proposed by Bruckner and Gröller [2]. Any other volume rendering technique could serve as context visualization as well.

The context is displayed, whenever a ray hits the volume and the cut surface lies outside the volume at its pixel location or when the windowing function makes the cut surface fully transparent. In the first case, the rays traverse the whole volume, whereas in the second case, the rays start at the depth of the cut surface (cf. Figures 1b-d, 6f, 7d). The advantage of this context visualization is that it provides additional spatial cues when rotating while still depicting the occlusion-free cut surface.

3.4 Implementation

Our technique was implemented as part of the *AngioVis* framework, which is in clinical use at the Vienna General Hospital and the Kaiser-Franz-Josef Hospital. Our extension is not yet in clinical use. The LODs of the vessel tree centerlines are generated on the CPU as a pre-processing step at startup and regenerated after changes to the vessel tree. The remaining stages of Section 3.3 are implemented in CUDA and use OpenGL just for computing the entry and exit positions of the rays. This allows the whole pipeline to make use of the parallel computing capabilities of current GPUs which leads to an interactive performance of our implementation.

We analyzed the computing times of all individual stages of the CSR rendering pipeline. The outcome is summarized in Table 2 with the timings given in milliseconds. The measurements have been taken

Table 2. Performance timings in milliseconds per frame of CSR for the data sets presented in the results. The most time consuming operations are the LOD estimation, as every line segment has to be evaluated during this stage, and the context rendering. The depth computation is less expensive and all other operations have negligible temporal cost.

	Figure 1	Figure 7	Figure 8
Data Size	$512^2 \times 256$	$512^2 \times 1305$	$512^2 \times 575$
LOD Estimation	97.8 ms	272.6 ms	133.0 ms
Depth Computation	13.5 ms	105.6 ms	39.0 ms
Depth Filtering	1.7 ms	1.7 ms	1.7 ms
Surface Rendering	0.9 ms	1.0 ms	1.1 ms
Silhouette Rendering	7.6 ms	7.7 ms	7.7 ms
Context Rendering	96.9 ms	270.7 ms	131.8 ms
Frame Time	218.4 ms	659.3 ms	314.3 ms

on an Intel Core i7 with 3.07 GHz with 12 GB system memory and a GeForce 680GTX with 4 GB video memory. The LOD estimation is the most costly step, as each ray is checked with every line segment of the most detailed LOD of the vessel tree in order to compute an accurate distance. The performance of this step could be increased by gradually computing the distance from the coarsest to the finest LOD and updating only relevant distances in each step. The depth computation is less expensive, since only two coarse LODs are tested for most of the rays. The following operations consume nearly constant, insignificant amounts of time. Context rendering takes a considerable amount of time due to the ray marching of MIDA. Nevertheless, even our unoptimized implementation of CSR is capable of interactively rendering the cut surface together with the context visualization. Ray casting could also be substituted by rasterization of the local surface primitives similar to the method of Hoff et al. [12]. A performance gain is not guaranteed, however, as the cost function methodology cannot rely on the hierarchical depth-buffering capabilities of graphics hardware.

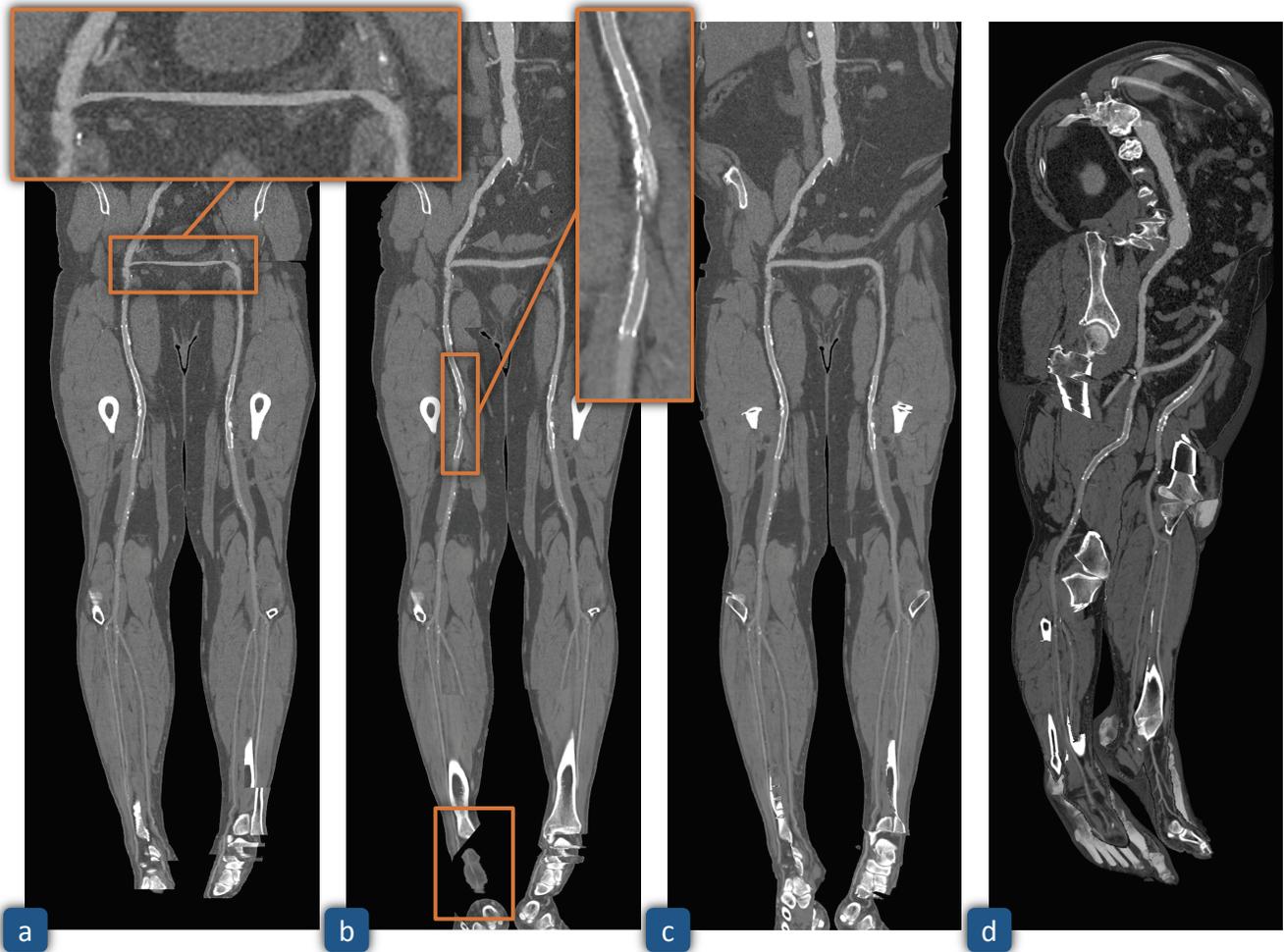


Fig. 7. A CTA data set of the vasculature of the human lower extremities with a cross-over bypass. (a) shows an mpCPR of the vasculature with an incorrectly narrowed lumen of the bypass (cf. zoom-in). (b) displays a CR which accounts for the correct lumen of the bypass, but shows artifacts resulting from the trade-off between vessel visibility and the displayed size of the surrounding tissue (cf. zoom-ins). The vessel radius is offset by 150 pixels and the arc-length threshold equals to 300 pixels. (c) presents CSR showing the correct lumen of the bypass and accounting for proper overall vessel visibility. (d) demonstrates a top-right view of the vasculature using CSR together with MIDA as context visualization.

4 RESULTS AND DISCUSSION

In this section, several results that were obtained with our method are presented. The weighting $\lambda = 10$ in the cost function (cf. Equation 2) is used for all subsequent results. First, we present a small data set to illustrate the capabilities of our technique. With the second and third data set, we compare against existing methods, such as mpCPR and CR, in order to highlight our improvements.

Figure 1 presents a CTA data set of a human abdominal aorta at its first bifurcation displayed for several view directions. A DVR visualization is shown in Figure 1a together with the vessel tree rendered as wireframe model in red. Figures 1b-d give results of our method together with MIDA as context visualization. Visibility is correctly resolved in Figure 1b, as the left vessel branch is partially hidden due to its larger distance from the viewer. The cut surface is smooth even far from the vessel due to our LOD approach and exhibits intended jumps in depth when changing from the influence zone of one branch to another one. Figure 1c shows a view from bottom right, which depicts the vessel lumen and surrounding tissue. The reason for the smoothness is again the continuous transition between different LODs of the vessel tree. A view from the bottom is presented in Figure 1d. Here, a cutaway is generated to show the lumen of the aorta, which is located farther away from the viewer than its two branches.

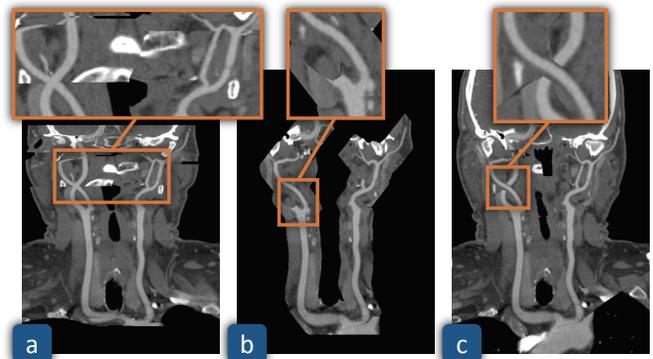


Fig. 8. Cervical vessels of a human head CTA data set. (a) shows an mpCPR of the vasculature, which does not account for correct visibility (cf. zoom-in). (b) displays the lumina of the vessels using CR with a small vessel radius offset (50 pixels) and arc-length threshold (150 pixels) exhibiting a visibility artifact caused by the trade-off. (c) presents CSR generating a smooth cut surface through the vasculature and surrounding tissue while preserving the visibility (cf. zoom-in).

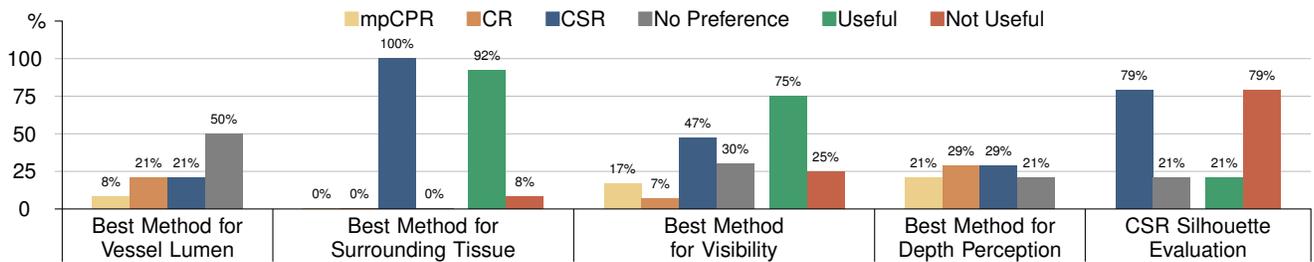


Fig. 9. Evaluation of CSR in comparison to mpCPR and CR according to five criteria. The vertical axis show the percentage (%) of the participants, where 100% equals to 12 domain experts. The preference of the participants was evaluated in the first three criteria, while in the last two, the number of correct answers has been counted. While no technique has clear advantages for inspecting the vessel lumen in a standard medical case, our technique compares favorably concerning the depiction of surrounding tissue and vessel visibility. The depth perception was roughly equal for all techniques and silhouettes provided no additional benefits.

Figure 7 shows a CTA data set of the vasculature of the human lower extremities with a cross-over bypass below the bifurcation of the abdominal aorta. We compare mpCPR (7a) with CR (7b) and CSR (7c). Although mpCPR provides a continuous cut surface through the vasculature and its surrounding tissue, it cannot visualize the lumen of the bypass adequately. The lumen is incorrectly narrowed, since the bypass runs horizontally, a situation that cannot be properly handled by mpCPR. This issue is addressed by CR, but a trade-off between the display of the surrounding tissue and the vessel visibility has to be made. This leads to artifacts, as pointed out in the zoom-ins, coming from the fact that the information of the left vessel overwrites information of the right one. CSR avoids these issues by creating the lumina entirely in 3D. Also the visibility of the denser vasculature in the lower legs is correctly preserved. Figure 7d shows a rotated view of the lumina of the vasculature using CSR. In order to enhance spatial perception, the cut surface is embedded into a context visualization.

A CTA data set of the human cervical vessels, including the carotid arteries, is shown in Figure 8. With mpCPR (8a), visibility is not preserved and the lumen of horizontally running vessels is not rendered correctly (cf. zoom-in). The surrounding tissue is, however, shown. With CR (8b), a trade-off between the display of the surrounding tissue and the visibility of the vessels has to be done. This is the reason why one vessel is cut off, as the vessel lies behind surrounding tissue (cf. zoom-in). Correcting the visibility would further limit the size of visible surrounding tissue. CSR (8c) preserves the visibility as well as provides a smooth cut through the surrounding tissue.

5 EVALUATION

We consulted 12 radiologists from three different hospitals for an evaluation of CSR in comparison to mpCPR and CR with a questionnaire. For a general assessment, the data set of Figure 1 was used, while the visibility was investigated on cervical vessels (cf. Figure 8). The questionnaire consisted of 21 questions from five criteria (cf. horizontal axis of Figure 9). Additionally, the usefulness of displaying the surrounding tissue, the visibility and the silhouettes were investigated, while the other two criteria are considered useful per se. The number of votes for the usefulness of the corresponding criterion is depicted as green bar, while the red bar shows votes doubting its usefulness.

There is no clear preference for a reformation technique for inspecting the **vessel lumen** in a standard medical case. mpCPR is rated lowest, but most participants see no clear difference as indicated by the large gray bar. This question should test if our method is equivalent to currently used clinical methods for standard cases.

All participants preferred CSR for investigating the **surrounding tissue**, and the domain experts regard this criterion as very useful (green bar). One radiologist specifically mentioned that the surroundings are beneficial for detecting other pathologies. This supports the LOD approach for smoothly extending the cut surfaces into the surrounding tissue.

For assessing the **visibility** of overlapping vessels, we use a cervical vessel data set (cf. Figure 8). The majority of participants rated CSR

highest. The other categories in order of popularity were no preference, mpCPR, and CR. Additionally, most domain experts considered a correct visibility as very useful (green bar). In fact, a smooth cut through the surrounding tissue while preserving the visibility of overlapping vessels was one of the motivations for our technique.

Regarding the **depth perception** of different vessels, CR and CSR are rated on the same level, mpCPR is rated worse. **Silhouettes** were not regarded as useful by most participants. We assume, however, that they could prove helpful after an accommodation period or for a complex vasculature, as specifically mentioned by one domain expert. A larger user study would be required to evaluate this effect as well as our context rendering.

Concerning the spatial orientation, domain experts pointed out that they use their anatomical knowledge to orient themselves and navigate within the 3D data sets and images. One radiologist specifically mentioned that visualizing the lumen of arbitrarily oriented vessels and having the possibility of unrestricted rotations, has great potential. In fact, this is part of the rationale for developing this method.

The feedback from domain experts indicates that CSR compares favorably to mpCPR and CR. Moreover, it enhances the visualization of surrounding tissue and the visibility of multiple overlapping vessels.

6 CONCLUSION AND FUTURE WORK

We proposed CSR, a novel reformation technique that generates a cut surface through the lumen of vessels entirely in 3D. It does not require adjustment of any parameters by the user, while preserving correct visibility of vessels and their surrounding tissue. Moreover, arbitrary rotations are supported, giving the user complete freedom of inspecting the vessel lumina from every desired view direction. As our proposed technique has been implemented on graphics hardware, it offers interactive visual analysis of vessel pathologies.

We see several avenues for **future work** based on our method. As it allows the inspection of a complex vasculature in 3D, a given CSR view could be linked with a detailed view of a single vessel that contains aggregated information, such as based on curvicircular feature aggregation [28]. The application to different curve-like anatomical structures such as nerves might be interesting. While we only evaluate our method's merits for vessel reformation, it is a general approach to generate view-dependent 2D surfaces along 1D curves that produces high-quality cuts through the surrounding 3D volume. The visualization of cut surfaces along curve-like features in scalar [6] and vector data [15, 20] in various scientific and technical domains are possible application scenarios.

ACKNOWLEDGMENTS

The work presented in this paper has been supported by the Austrian Science Fund (FWF) grants no. TRP 67-N23 (Knowledge Assisted Sparse Interaction for Peripheral CT-Angiography – KASI) and no. P23700-N23 (Modern Functional Analysis in Computer Graphics – MOFA). The data sets are courtesy of the Kaiser-Franz-Josef Hospital and the General Hospital of Vienna.

REFERENCES

- [1] Å. Birkeland, S. Bruckner, A. Brambilla, and I. Viola. Illustrative membrane clipping. *Computer Graphics Forum*, 31(3):905–914, 2012.
- [2] S. Bruckner and M. E. Gröller. Instant volume visualization using maximum intensity difference accumulation. *Computer Graphics Forum*, 28(3):775–782, 2009.
- [3] M. Burns and A. Finkelstein. Adaptive cutaways for comprehensible rendering of polygonal scenes. *ACM Trans. Graph.*, 27(5):154:1–154:7, 2008.
- [4] M. Burns, M. Haidacher, W. Wein, I. Viola, and M. E. Gröller. Feature emphasis and contextual cutaways for multimodal medical visualization. In *Proceedings of Eurographics / IEEE VGTC Symposium on Visualization (EuroVis 2007)*, pages 275–282, 2007.
- [5] C. Chuon, S. Guha, P. Janecek, and N. D. C. Song. Simplipoly: Curvature-based polygonal curve simplification. *Int. J. Comput. Geometry Appl.*, 21(4):417–429, 2011.
- [6] C. Correa, P. Lindstrom, and P.-T. Bremer. Topological spines: A structure-preserving visual representation of scalar fields. *IEEE Trans. Vis. Comput. Graphics*, 17(12):1842–1851, 2011.
- [7] D. DeCarlo, A. Finkelstein, S. Rusinkiewicz, and A. Santella. Suggestive contours for conveying shape. *ACM Trans. Graph.*, 22(3):848–855, 2003.
- [8] J. Diepstraten, D. Weiskopf, and T. Ertl. Interactive cutaway illustrations. *Computer Graphics Forum*, 22(3):523–532, 2003.
- [9] D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica*, 10(2):112–122, 1973.
- [10] M. H. Everts, H. Bekker, J. B. Roerdink, and T. Isenberg. Depth-dependent halos: Illustrative rendering of dense line data. *IEEE Trans. Vis. Comput. Graphics*, 15(6):1299–1306, 2009.
- [11] J. Gudmundsson, G. Narasimhan, and M. Smid. Distance-preserving approximations of polygonal paths. *Computational Geometry*, 36(3):183–196, 2007.
- [12] K. E. Hoff, III, J. Keyser, M. Lin, D. Manocha, and T. Culver. Fast computation of generalized Voronoi diagrams using graphics hardware. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques, SIGGRAPH '99*, pages 277–286. ACM Press/Addison-Wesley Publishing Co., 1999.
- [13] H. Imai and M. Iri. Computational-geometric methods for polygonal approximations of a curve. *Computer Vision, Graphics, and Image Processing*, 36(1):31–41, 1986.
- [14] H. Imai and M. Iri. *Computational Morphology*, chapter Polygonal approximations of a curve – formulations and algorithms, pages 87–95. North-Holland, 1988.
- [15] M. Jankun-Kelly, M. Jiang, D. Thompson, and R. Machiraju. Vortex visualization for practical engineering applications. *IEEE Trans. Vis. Comput. Graphics*, 12(5):957–964, 2006.
- [16] A. Kanitsar, D. Fleischmann, R. Wegenkittl, P. Felkel, and M. E. Gröller. CPR - curved planar reformation. In *Proceedings of IEEE Visualization*, pages 37–44, 2002.
- [17] A. Kanitsar, D. Fleischmann, R. Wegenkittl, and M. E. Gröller. Diagnostic relevant visualization of vascular structures. In G.-P. Bonneau, T. Ertl, and G. Nielson, editors, *Scientific Visualization: The Visual Extraction of Knowledge from Data*, Mathematics and Visualization, pages 207–228. Springer Berlin Heidelberg, 2006.
- [18] A. Kanitsar, D. Fleischmann, R. Wegenkittl, D. Sandner, P. Felkel, and M. E. Gröller. Computed tomography angiography: a case study of peripheral vessel investigation. In *Proceedings of IEEE Visualization*, pages 477–593, 2001.
- [19] A. Kanitsar, R. Wegenkittl, D. Fleischmann, and E. Gröller. Advanced curved planar reformation: Flattening of vascular structures. In *Proceedings of IEEE Visualization*, pages 43–50, 2003.
- [20] D. N. Kenwright and R. Haimes. Automatic vortex core detection. *IEEE Comput. Graph. Appl.*, 18(4):70–74, 1998.
- [21] O. Konrad-Verse, B. Preim, and A. Littmann. Virtual resection with a deformable cutting plane. In *Proceedings of Simulation und Visualisierung 2004*, pages 203–214, 2004.
- [22] O. D. Lampe, C. Correa, K.-L. Ma, and H. Hauser. Curve-centric volume reformation for comparative visualization. *IEEE Trans. Vis. Comput. Graphics*, 15(6):1235–1242, 2009.
- [23] N. Lee and M. Rasch. Tangential curved planar reformation for topological and orientation invariant visualization of vascular trees. In *Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE*, pages 1073–1076, 2006.
- [24] W. Li, L. Ritter, M. Agrawala, B. Curless, and D. Salesin. Interactive cutaway illustrations of complex 3D models. *ACM Trans. Graph.*, 26(3):31:1–31:11, 2007.
- [25] E. M. Lidal, H. Hauser, and I. Viola. Design principles for cutaway visualization of geological models. In *Proceedings of the 28th Spring Conference on Computer Graphics, SCCG '12*, pages 47–54. ACM, 2012.
- [26] M. J. McGuffin, L. Tancau, and R. Balakrishnan. Using deformations for browsing volumetric data. In *Proceedings of IEEE Visualization*, pages 401–408, 2003.
- [27] T. McInerney and P. Crawford. Ribbonview: interactive context-preserving cutaways of anatomical surface meshes. In *Proceedings of the 6th international conference on Advances in visual computing - Volume Part II, ISVC'10*, pages 533–544, 2010.
- [28] G. Mistelbauer, A. Morar, A. Varchola, R. Scherthaner, I. Baclija, A. Köchl, A. Kanitsar, S. Bruckner, and M. E. Gröller. Vessel visualization using curvilinear feature aggregation. *Computer Graphics Forum*, 32(3):231–240, 2013.
- [29] G. Mistelbauer, A. Varchola, H. Bouzari, J. Starinsky, A. Köchl, R. Scherthaner, D. Fleischmann, M. E. Gröller, and M. Srámek. Centerline reformations of complex vascular structures. In *Pacific Visualization Symposium (PacificVis), 2012 IEEE*, pages 233–240, 2012.
- [30] A. Nealen, M. Müller, R. Keiser, E. Boxerman, and M. Carlson. Physically based deformable models in computer graphics. *Computer Graphics Forum*, 25(4):809–836, 2006.
- [31] U. Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1(3):244–256, 1972.
- [32] J. E. Roos, D. Fleischmann, A. Köchl, T. Rakshe, M. Straka, A. Napoli, A. Kanitsar, M. Srámek, and E. Gröller. Multi-path curved planar reformation (mpCPR) of the peripheral arterial tree in CT angiography (CTA). *Radiology*, 244(1):281–290, 2007.
- [33] L. Saroul, O. Figueiredo, and R.-D. Hersch. Distance preserving flattening of surface sections. *IEEE Trans. Vis. Comput. Graphics*, 12(1):26–35, 2006.
- [34] L. Saroul, S. Gerlach, and R. D. Hersch. Exploring curved anatomic structures with surface sections. In *Proceedings of IEEE Visualization*, pages 27–34, 2003.
- [35] S. Sigg, R. Fuchs, R. Carnecky, and R. Peikert. Intelligent cutaway illustrations. In *Pacific Visualization Symposium (PacificVis), 2012 IEEE*, pages 185–192, 2012.
- [36] M. Straka, A. Köchl, M. Cervenansky, M. Srámek, D. Fleischmann, A. L. Cruz, and E. Gröller. The VesselGlyph: Focus & Context Visualization in CT-Angiography. In *Proceedings of IEEE Visualization*, pages 385–392, 2004.
- [37] I. Viola, M. Feixas, M. Sbert, and M. E. Gröller. Importance-driven focus of attention. *IEEE Trans. Vis. Comput. Graphics*, 12(5):933–940, 2006.
- [38] S. W. Wang and A. E. Kaufman. Volume sculpting. In *Proceedings of the 1995 Symposium on Interactive 3D Graphics*, pages 151–156, 1995.
- [39] D. Weiskopf, K. Engel, and T. Ertl. Interactive clipping techniques for texture-based volume visualization and volume shading. *IEEE Trans. Vis. Comput. Graphics*, 9(3):298–312, 2003.
- [40] Q. Zhang, R. Eagleson, and T. Peters. Rapid scalar value classification and volume clipping for interactive 3D medical image visualization. *The Visual Computer*, 27(1):3–19, 2011.