

# Untangling Vascular Trees for Surgery and Interventional Radiology

Guillaume Houry<sup>1</sup>[0000-0002-1918-0545], Tom Boeken<sup>1,2</sup>[0000-0003-2990-5180],  
Stéphanie Allasonnière<sup>1</sup>[0000-0002-5692-4945], and  
Jean Feydy<sup>1</sup>[0000-0001-6049-563X]

<sup>1</sup> Inria, Université Paris Cité, Inserm, HeKA, F-75015 Paris, France

<sup>2</sup> Department of Vascular and Oncological Interventional Radiology,  
Hôpital Européen Georges Pompidou, AP-HP, 75015 Paris, France  
`jean.feydy@inria.fr`

**Abstract.** The diffusion of minimally invasive, endovascular interventions motivates the development of visualization methods for complex vascular networks. We propose a planar representation of blood vessel trees which preserves the properties that are most relevant to catheter navigation: topology, length and curvature. Taking as input a three-dimensional digital angiography, our algorithm produces a faithful two-dimensional map of the patient’s vessels within a few seconds. To this end, we propose optimized implementations of standard morphological filters and a new recursive embedding algorithm that preserves the global orientation of the vascular network. We showcase our method on peroperative images of the brain, pelvic and knee artery networks. On the clinical side, our method simplifies the choice of devices prior to and during the intervention. This lowers the risk of failure during navigation or device deployment and may help to reduce the gap between expert and common intervention centers. From a research perspective, our method simulates the cadaveric display of artery trees from anatomical dissections. This opens the door to large population studies on the branching patterns and tortuosity of fine human blood vessels. Our code is released under the permissive MIT license as part of the scikit-shapes Python library<sup>3</sup>.

**Keywords:** Vascular trees · Visualization · Computational anatomy · Computer-assisted intervention · Interventional radiology.

## 1 Introduction and Related Works

**Endovascular Interventions.** Unfolding curved anatomical structures for visualization on two-dimensional displays is a fundamental task in medical imaging, with applications ranging from the detection of rib fractures [21] to the

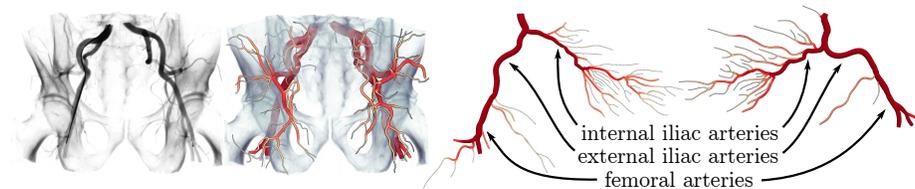
<sup>3</sup> A demonstration of our code is available in the following folder:  
`github.com/scikit-shapes/scikit-shapes/tree/main/arteries-atlas.`

planning of tumor ablations [13]. Meanwhile, the emergence of endovascular interventions has sparked increasing interest in the detailed mapping of complex arterial networks, understood as modern highways for surgeons, cardiologists and interventional radiologists [1,17,22,24]. Our work lies at the intersection of these developments: we focus on generating planar representations of intricate vascular structures to facilitate catheter navigation during interventions. Specifically, we aim to embed vascular networks in 2D while preserving key geometric features for effective endovascular guidance: the vessel lengths and curvatures.

**Vessel Tree Analysis and Visualization.** The creation of blood vessel maps is an active field of research; we refer to [3] for a recent overview. Our work places a strong emphasis on vessel lengths and curvatures, features that are critical for catheter navigation but often overlooked in existing methods. This focus presents unique challenges: while efficient methods can embed spatial graphs in the plane without intersections [13,19], doing so while preserving the length and tortuosity of curves is not straightforward.

Several techniques have been proposed to produce geometry-accurate planar representations of vascular networks. Curved planar reformations [10,11,25] generate 2D slices from volumetric images via non-linear projection along vessel centerlines, preserving local geometry and surrounding context. However, they require complex decluttering and cannot guarantee intersection-free layouts. Alternatively, conformal mappings [26] flatten the blood vessels while preserving boundary properties. Yet, these approaches neglect the global vascular structure, making it unsuited to our purpose. A last approach, arguably the closest to our objective, consists in building the embedding by joining and untangling multiple local linear projections of the vessel tree [7,14], offering a good balance between geometric fidelity and global readability.

Unfortunately, these techniques are not fully adapted to our problem. They usually neglect the branching angles at the vessel junctions, which are of critical importance in endovascular navigation. More importantly, current methods do not scale well with large vascular systems: for instance, [25] requires 12 minutes to process a tree with 72 branches, while [7,14] needs several minutes for networks with a few thousand nodes. In contrast, the vascular trees we analyze



**Fig. 1.** *Left:* Our method takes as input a three-dimensional angiogram, such as this scan of the pelvic region. *Middle:* First, we apply fast morphological filters to segment the blood vessels. *Right:* Then, our novel tree embedding algorithm creates a planar vessel map that is optimized for the planning of endovascular interventions.

are significantly more complex, featuring dense branching patterns and intricate topologies comprising over 10,000 nodes and more than 500 branches.

**Contributions.** To tackle this challenge, we propose an algorithm that produces clear visualizations of complex vascular trees while preserving vessel curvature as accurately as possible. Our method generates qualitatively satisfying layouts in a few seconds, even in our most challenging cases. The resulting embeddings contain no intersections, exactly preserve vessel lengths, and maintain the correct angles at the majority of branching points. Finally, we provide a portable code that implements our full pipeline as part of the scikit-shapes Python library ([github.com/scikit-shapes/scikit-shapes/tree/main/arteries-atlas](https://github.com/scikit-shapes/scikit-shapes/tree/main/arteries-atlas)).

## 2 Methods

Our method takes as input an acyclic vascular network understood as a tree  $\mathcal{T} = (V, E)$  whose root node  $v_{\text{root}}$  corresponds to the blood source. We identify each vertex  $v \in V$  with its vector of 3D coordinates  $v \in \mathbb{R}^3$ . We also require an estimate of the local vessel radius at every vertex.

Because most vertices have exactly two neighbors, we can decompose the tree into a set of contiguous branches  $B = (b_1, \dots, b_N)$  that are topologically equivalent to segments. This induces a coarse-grained directed tree structure  $\mathcal{T}_{\text{coarse}} = (J, B)$  where  $J$  is the subset of vertices with a degree different from two – namely, the root, junctions, and leaves. These vertices are connected by the branches in  $B$ . Our method consists of two main steps:

1. A recursive planar embedding of the branches of  $\mathcal{T}_{\text{coarse}}$  without intersections that prioritizes the preservation of the angles at junctions;
2. A refinement step using a force-directed scheme to enhance the final layout.

This section provides a detailed overview of our method. We begin by describing the data and preprocessing steps, followed by a formal definition of the planar embedding problem using an angular reparametrization of vertex positions. We then use this angular representation to define a target curvature at each tree node and conclude with a description of the two steps of our embedding algorithm.

**Data and Preprocessing.** We work with three-dimensional angiograms acquired via 3D X-ray imaging – more precisely, a Philips Azurion system following a Cone Beam CT (CBCT) protocol. Voxel values are expressed in Hounsfield units and blood vessels are highlighted using a contrast agent. Motivated by applications in stroke surgery, our primary focus is on cerebral vascular networks. The injection protocol selectively highlights one hemisphere of the brain arterial network, allowing us to assume a tree-like topology. To showcase the generalizability of our method, we also apply it to arterial networks around the pelvis and knee (see figs. 1 and 8). This study was approved by two institutional review boards (Registre général des traitements de l’Assistance Publique–Hôpitaux no.

20220128085623 and CERAR Institutional Review Board 00010254-2022-025) in accordance with the Declaration of Helsinki.

As an initial processing step, we convert the 3D images into graphs using efficient implementations of classical image processing techniques: hysteresis thresholding [2], signed distance transform, Frangi filtering [4], and skeletonization [18]. To correct topological artifacts such as small cycles or multiple disconnected components, we compute the minimum spanning tree of the graph [12,23]. We illustrate this learning-free pipeline in fig. 2. Although not state-of-the-art, this simple method produces satisfying segmentations for our high-resolution data. Leveraging PyTorch [20] and the Taichi library [8], our code is able to process images of size  $500 \times 500 \times 500$  within seconds.

**Angular Parametrization.** The planar embedding task consists in associating a 2D position  $\bar{v} \in \mathbb{R}^2$  to each vertex  $v \in \mathbb{R}^3$  of the three-dimensional vascular tree  $\mathcal{T}$ . In order to guarantee the preservation of vessel lengths, we propose to use an intermediate representation via real-valued angular coordinates  $\theta(v)$ . Placing the root node at the origin so that  $\bar{v}_{\text{root}} = (0, 0)$ , we compute the node positions  $\bar{v} \in \mathbb{R}^2$  from the values of  $\theta(v)$  recursively with:

$$\text{for all edge } (v \rightarrow v') \in E, \quad \bar{v}' = \bar{v} + \|v - v'\|_{\mathbb{R}^3} \cdot (\cos \theta(v'), \sin \theta(v')). \quad (1)$$

**Computing Signed Curvatures on the Vessel Tree.** Since endovascular navigation is strongly influenced by the angles at vessel junctions and the accumulation of torque along the catheter, it is important to reflect the original 3D vessel curvatures in the 2D embedding. To this end, we impose a constraint on bending angles: for any three consecutive nodes  $v \rightarrow v' \rightarrow v''$ , the absolute difference  $|\theta(v) - \theta(v')|$  should match the angle between the 3D segments  $\vec{vv'}$  and  $\vec{v'v''}$ . To capture the correct global shape of the vessel, we must also assign the appropriate sign to this difference, indicating a left or right bend in the 2D plane. Following [6], we use parallel transport to propagate a reference normal vector  $\vec{n}(v)$  along the tree  $\mathcal{T}$  and define the bending direction using the sign of the dot product  $\vec{n}(v) \cdot \vec{vv'}$  (see fig. 3). This allows us to define a signed angular curvature  $\kappa(v)$  at each vertex. Preserving the curvature of blood vessels in the planar visualization is then equivalent to making sure that:

$$\text{for all edge } (v \rightarrow v') \in E, \quad \theta(v') \simeq \theta(v) + \kappa(v'). \quad (2)$$

**Intersection-free Planar Layout.** In general, the constraint in eq. (2) cannot be satisfied simultaneously for all vertices without causing edge intersections. As a first step that we illustrate in fig. 4, we compute an embedding that guarantees no intersections and attempts to preserve angle variations around branching points. To this end, we work with the coarse-grained structure  $\mathcal{T}_{\text{coarse}}$  and recursively tessellate the embedding plane in polygonal regions  $U_v$  that contain the embedding locations of all vertices downstream of point  $v$ .

First, we associate the upper half-plane  $U_{v_{\text{root}}} = \mathbb{R} \times \mathbb{R}_{\geq 0}$  to the root node embedded at the origin point  $\bar{v}_{\text{root}} = (0, 0)$ . Then, working recursively at every junction node  $v$ , we embed downstream branches  $b_1, \dots, b_k$  by integrating eq. (2). This exactly preserves vessel curvatures until a collision is detected, as highlighted in red in fig. 4. Such collisions can occur either with previously embedded parts of branches  $b_1, \dots, b_k$  or with the boundary of the region  $U_v$ . When a collision is encountered, we embed the remaining portion of the affected branch as a straight line. The domain  $U_v$  is then subdivided into  $k$  non-overlapping sectors  $U_{v'_1}, \dots, U_{v'_k}$  associated with the endpoints of the branches  $b_1, \dots, b_k$ . These sectors define feasible regions for the subsequent recursive calls at the child nodes  $v'_1, \dots, v'_k$  in  $\mathcal{T}_{\text{coarse}}$ . We show a typical output of this recursive embedding method on the left panel of fig. 5.

**Force-directed Layout Refinement.** This initial embedding straightens distal vessels. To enhance clarity, we denote by  $\theta_0(v)$  the angles produced by our recursive embedding algorithm and iteratively apply the following update:

$$\theta_{t+1}(v) \leftarrow \theta_t(v) + \lambda_{\text{rep}} \cdot F_{\text{rep}}(v) + \lambda_{\text{bend}} \cdot F_{\text{bend}}(v) + \lambda_{\text{struct}} \cdot F_{\text{struct}}(v) ,$$

where  $F_{\text{rep}}$  is a repulsive force between nearby vertices that prevents overlaps,  $F_{\text{bend}}$  bends the angles towards their true curvature and  $F_{\text{struct}}$  maintains the global structure of the initial embedding. We use PyTorch to implement the following functions at every node  $v$  with parent  $p$  and vessel radius  $r$ :

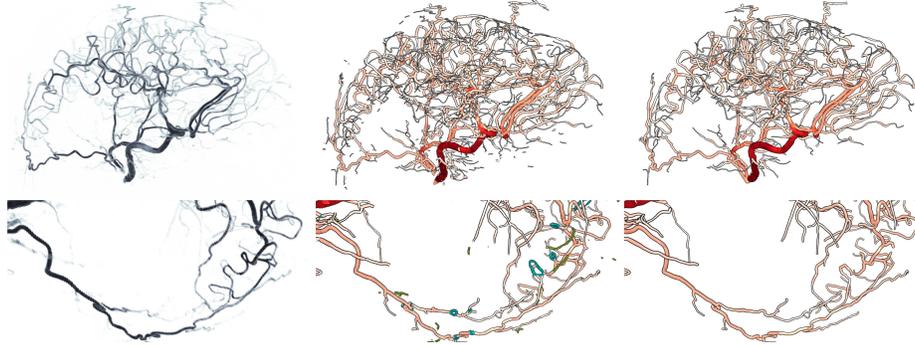
$$F_{\text{rep}}(v) = -\frac{\partial}{\partial \theta_t(v)} \sum_{v' \neq v} \left[ \left( \frac{1}{\|\bar{v} - \bar{v}'\|_{\mathbb{R}^2}^2} + \frac{\alpha}{\|\bar{v} - \bar{v}'\|_{\mathbb{R}^2}} \right) \cdot \exp\left(-\frac{1}{\sigma} \|\bar{v} - \bar{v}'\|_{\mathbb{R}^2}\right) \right] ,$$

$$F_{\text{bend}}(v) = r^\mu \cdot [\kappa(v) - (\theta_t(v) - \theta_t(p))] , \quad F_{\text{struct}}(v) = \theta_0(v) - \theta_t(v) .$$

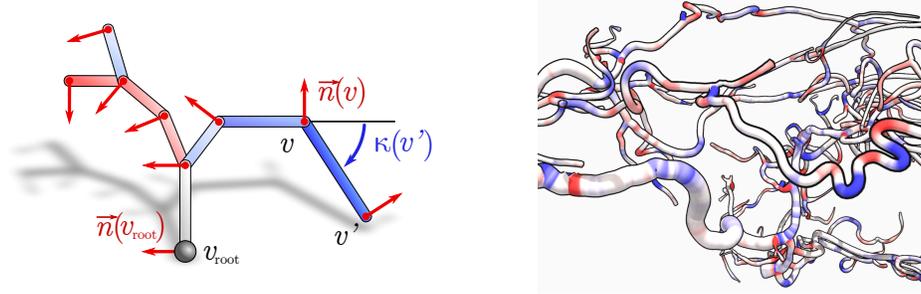
$F_{\text{rep}}$  is the sum of two forces operating at different scales: the first prevents vessel crossings while the second spreads the branches to enhance their readability. The exponential factor limits the repulsion range for numerical stability. The term  $r^\mu$  in  $F_{\text{bend}}$  prioritizes the curvature of the largest vessels over the finest ones. A typical result is shown on the right panel of fig. 5 with  $\lambda_{\text{rep}} = 25$ ,  $\lambda_{\text{bend}} = 0.4$ ,  $\lambda_{\text{struct}} = 5 \cdot 10^{-5}$ ,  $\alpha = 0.1$ ,  $\sigma = 200$  and  $\mu = 1$ . These parameters influence the embedding style and should be tuned according to the anatomical region's characteristics, such as average vessel length and density of ramifications.

### 3 Results and Conclusion

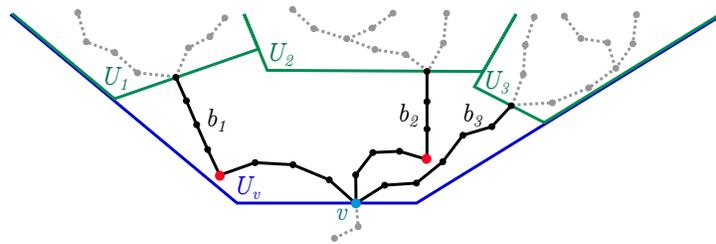
**Surgical Planning and Anatomical Comparisons.** As illustrated in fig. 6, our method turns peroperative three-dimensional angiograms of the brain arterial network into legible planar representations. Unlike preexisting approaches, our layout preserves junction angles, vessel lengths and curvatures. This is especially relevant for interventional radiology; at a glance, a physician can now



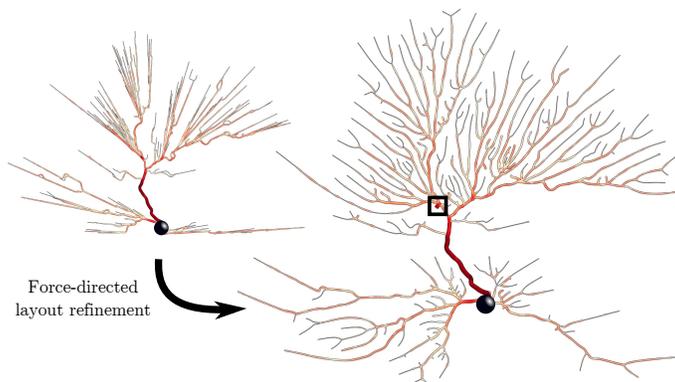
**Fig. 2.** *Left:* Three-dimensional cerebral angiogram. *Middle:* The segmented graph. Shades of red encode the estimated vessels radius, blue edges correspond to cycles and green edges belong to small disconnected components. *Right:* The final vessel tree.



**Fig. 3.** *Left:* Flowing from the root node  $v_{\text{root}}$ , we define the point curvature  $\kappa(v)$  as the angle between two successive segments of the curve. We transport a reference normal vector  $\vec{n}(v)$  from  $v_{\text{root}}$  as in [6] and use the sign of the dot product  $\vec{n}(v) \cdot \vec{v}'$  to orient the curvature  $\kappa(v')$ . *Right:* We display the values of  $\kappa(v)$  on a typical example: red segments are associated to positive values, i.e. left turns in our planar layout; blue segments correspond to negative values, i.e. right turns.



**Fig. 4.** At every new junction  $v$ , we draw downstream branches  $b_1, \dots, b_k$  by integrating eq. (2) until colliding with the boundary of  $U_v$  or with any other branch (red nodes). We embed the rest of the branch as a line, subdivide the domain  $U_v$  in angular sectors  $U_1, \dots, U_k$  and proceed iteratively.

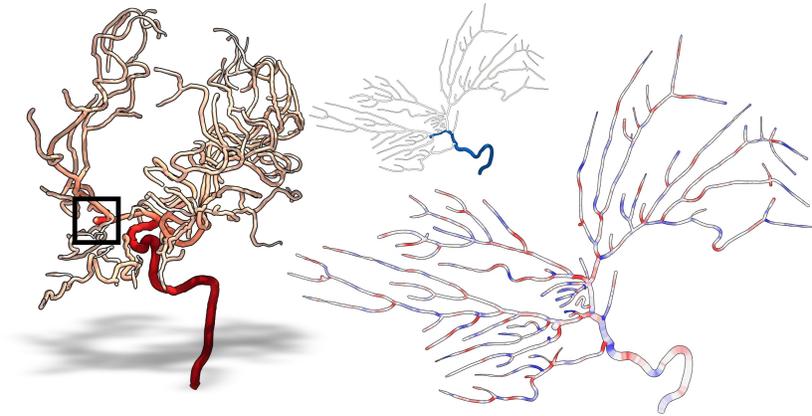


**Fig. 5.** *Left:* Result of the recursive embedding for the brain artery tree of fig. 2 (running time: 1 s). *Right:* Final embedding, after 300 steps of iterative refinement (running time: 2 s). The layout closely mimics pictures of genuine anatomical dissections [9], with the aneurysm clearly visible (black square). For the sake of clarity, we display both internal (top) and superficial (bottom) arteries, artificially tying them together around the root node (black disk).

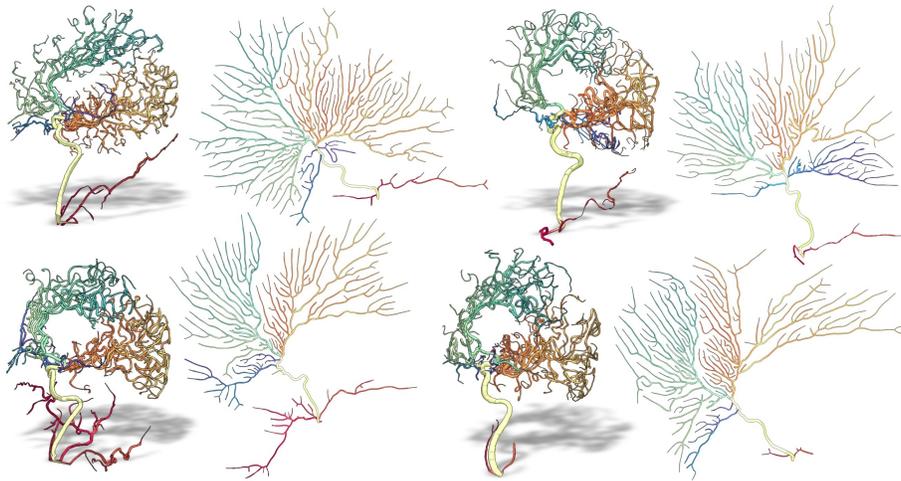
understand the shape of the path to be followed. This vessel map is thus ideally suited to guide the choice and thermo-forming of a catheter with a suitable tip and stiffness. Crucially, our implementation is released under the permissive MIT license and runs in less than ten seconds on a standard laptop CPU: this opens the door to its integration in the interventional workflow.

Our method also enables the visualization of detailed images of the cerebral vascular system that may contain very distal vessels. Due to their intricate geometries, the original three-dimensional volumes are difficult to read even for experts: our planar layouts solve this issue at a minimal computational cost. Our method is thus similar to Cerebrovis [19], but with a focus on geometry instead of pure topology. In fig. 7, we display the brain artery networks of four patients that underwent surgery for aneurysms: we believe that our vessel map will help physicians formulate novel hypotheses on, for example, risk factors for aneurysms or thromboses, which may then be validated by a morphometric study [5].

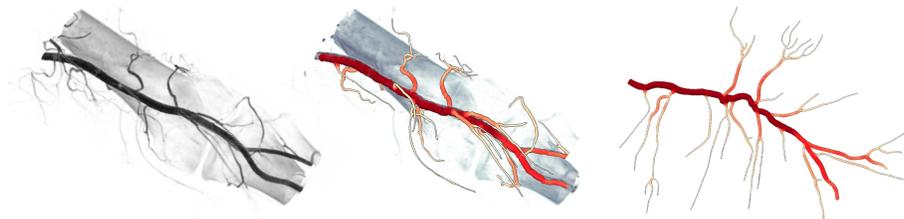
**Robustness and Quantitative Evaluation.** Although vessel segmentation is an active research topic [16], we found that our learning-free pipeline works surprisingly well on peroperative three-dimensional angiograms. Notably, applying a vesselness filter to the signed distance function instead of raw intensity values is enough to prevent the “fusion” of vessels that run parallel to each other such as the left and right anterior cerebral arteries. This is testament to the superior quality of data that is acquired in an interventional context, with high resolution devices and no motion artifacts. Although our work was first motivated by cerebral interventions, we stress that our method performs just as well on other anatomical regions such as the pelvis of fig. 1 and the knee of fig. 8.



**Fig. 6.** *Left:* Segmented brain artery tree in 3D, with an aneurysm (black square). *Top:* Path from the root node to the aneurysm. *Right:* Planar layout, colored by curvature.



**Fig. 7.** Untangling four cerebral vascular trees that include the left carotid artery (yellow), anterior cerebral artery (green) and middle cerebral artery (orange).



**Fig. 8.** *Left:* Three-dimensional angiography of the popliteal artery across the knee. *Middle:* Segmented artery tree, colored by the vessel radius. *Right:* Planar layout.

In the examples of figs. 5 to 8, our method preserves, within  $\pm 10^\circ$ , 75% to 90% of junction angles between vessels with radius larger than 2 mm. This proportion drops to between 45% and 55% for smaller vessels, which are necessarily compressed in the 2D embedding. Likewise, the curvature constraint of eq. (2) is satisfied up to  $2.5^\circ$  on average for large vessels and  $4.0^\circ$  for smaller vessels. This should be compared with the standard deviation of  $5.2^\circ$  for the bending angles  $\kappa(v')$ . These results highlight our method’s ability to preserve the geometry of major vessels while tolerating some loss of accuracy in finer vessels whose detailed shapes are less clinically relevant.

**Limitations and Future Works.** Going forward, we believe that our method could benefit from three major improvements. First, going beyond scalar radii to handle non-circular vessel sections would be desirable, especially in the context of interventions on aneurysms and thromboses. Second, we intend to combine our embedding algorithm with state-of-the-art segmentation networks to handle more common but challenging data, such as routinely acquired magnetic resonance images. Finally, extending our method to arterial networks that contain cycles (such as the circle of Willis) would be a significant breakthrough; this is a fundamental research problem, for which we may leverage recent insights from the computer graphics literature [15].

**Acknowledgments.** This study was funded by the France 2030 program managed by the Agence Nationale de la Recherche (ANR-23-IACL-0008, PR[AI]RIE-PSAI).

**Disclosure of Interests.** The authors have no competing interests to declare that are relevant to the content of this article.

## References

1. Acebes, C., Moustafa, A.H., Camara, O., Galdran, A.: The centerline-cross entropy loss for vessel-like structure segmentation: Better topology consistency without sacrificing accuracy. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 710–720. Springer (2024)
2. Canny, J.: A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence* **6**, 679–698 (1986)
3. Eulzer, P., Meuschke, M., Mistelbauer, G., Lawonn, K.: Vessel maps: A survey of map-like visualizations of the cardiovascular system. In: Computer Graphics Forum. vol. 41, pp. 645–673. Wiley Online Library (2022)
4. Frangi, A.F., Niessen, W.J., Vincken, K.L., Viergever, M.A.: Multiscale vessel enhancement filtering. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 130–137. Springer (1998)
5. Guo, X., Basu Bal, A., Needham, T., Srivastava, A.: Statistical shape analysis of brain arterial networks (ban). *The Annals of Applied Statistics* **16**(2), 1130–1150 (2022)
6. Hanson, A.J., Ma, H.: Parallel transport approach to curve framing. *Indiana University, Techreports-TR425* **11**, 3–7 (1995)

7. Hu, P., Boorboor, S., Marino, J., Kaufman, A.E.: Geometry-aware planar embedding of treelike structures. *IEEE transactions on visualization and computer graphics* **29**(7), 3182–3194 (2022)
8. Hu, Y., Li, T.M., Anderson, L., Ragan-Kelley, J., Durand, F.: Taichi: a language for high-performance computation on spatially sparse data structures. *ACM Transactions on Graphics (TOG)* **38**(6), 1–16 (2019)
9. Kahilogullari, G., Ugur, H.C., Comert, A., Tekdemir, I., Kanpolat, Y.: The branching pattern of the middle cerebral artery: is the intermediate trunk real or not? an anatomical study correlating with simple angiography. *Journal of neurosurgery* **116**(5), 1024–1034 (2012)
10. Kanitsar, A., Fleischmann, D., Wegenkittl, R., Felkel, P., Groller, E.: CPR – curved planar reformation. *IEEE* (2002)
11. Kanitsar, A., Wegenkittl, R., Fleischmann, D., Groller, M.E.: Advanced curved planar reformation: Flattening of vascular structures. *IEEE* (2003)
12. Kruskal, J.B.: On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society* **7**(1), 48–50 (1956)
13. Lichtenberg, N., Kraymer, B., Hansen, C., Müller, S., Lawonn, K.: Distance field visualization and 2d abstraction of vessel tree structures with on-the-fly parameterization. In: *VCBM*. pp. 265–278 (2019)
14. Marino, J., Kaufman, A.: Planar visualization of treelike structures. *IEEE Transactions on Visualization and Computer Graphics* **22**(1), 906–915 (2015)
15. Minarčík, J., Estep, S., Ni, W., Crane, K.: Minkowski penalties: Robust differentiable constraint enforcement for vector graphics. *ACM SIGGRAPH 2024 Conference Proceedings* **43**(4) (2024). <https://doi.org/10.1145/3641519.3657495>
16. Moccia, S., De Momi, E., El Hadji, S., Mattos, L.S.: Blood vessel segmentation algorithms — review of methods, datasets and evaluation metrics. *Computer methods and programs in biomedicine* **158**, 71–91 (2018)
17. Naeem, R., Hagerman, D., Svensson, L., Kahl, F.: Trexplorer: Recurrent DETR for topologically correct tree centerline tracking. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. pp. 744–754. Springer (2024)
18. Palágyi, K., Kuba, A.: A 3d 6-subiteration thinning algorithm for extracting medial lines. *Pattern Recognition Letters* **19**(7), 613–627 (1998)
19. Pandey, A., Shukla, H., Young, G.S., Qin, L., Zamani, A.A., Hsu, L., Huang, R., Dunne, C., Borkin, M.A.: Cerebrovis: Designing an abstract yet spatially contextualized cerebral artery network visualization. *IEEE transactions on visualization and computer graphics* **26**(1), 938–948 (2019)
20. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* **32** (2019)
21. Ringl, H., Lazar, M., Töpker, M., Woitek, R., Prosch, H., Asenbaum, U., Balassy, C., Toth, D., Weber, M., Hajdu, S., et al.: The ribs unfolded—a CT visualization algorithm for fast detection of rib fractures: effect on sensitivity and specificity in trauma patients. *European radiology* **25**, 1865–1874 (2015)
22. Sinha, A., Hamarneh, G.: Trind: Representing anatomical trees by denoising diffusion of implicit neural fields. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. pp. 344–354. Springer (2024)

23. Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, İ., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* **17**, 261–272 (2020). <https://doi.org/10.1038/s41592-019-0686-2>
24. Wittmann, B., Glandorf, L., Paetzold, J.C., Amiranashvili, T., Wälchli, T., Razansky, D., Menze, B.: Simulation-based segmentation of blood vessels in cerebral 3d octa images. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. pp. 645–655. Springer (2024)
25. Won, J.H., Rosenberg, J., Rubin, G.D., Napel, S.: Uncluttered single-image visualization of the abdominal aortic vessel tree: Method and evaluation. *Medical Physics* **36**(11), 5245–5260 (2009)
26. Zhu, L., Haker, S., Tannenbaum, A.R.: Conformal flattening maps for the visualization of vessels. In: *Medical Imaging 2002: Visualization, Image-Guided Procedures, and Display*. vol. 4681, pp. 742–748. SPIE (2002)