

Geometry-Aware Planar Embedding of Treelike Structures

Ping Hu¹, Saeed Boorboor¹, Joseph Marino¹, and Arie E. Kaufman¹, *Fellow, IEEE*

Abstract—The growing complexity of spatial and structural information in 3D data makes data inspection and visualization a challenging task. We describe a method to create a planar embedding of 3D treelike structures using their skeleton representations. Our method maintains the original geometry, without overlaps, to the best extent possible, allowing exploration of the topology within a single view. We present a novel camera view generation method which maximizes the visible geometric attributes (segment shape and relative placement between segments). Camera views are created for individual segments and are used to determine local bending angles at each node by projecting them to 2D. The final embedding is generated by minimizing an energy function (the weights of which are user adjustable) based on branch length and the 2D angles, while avoiding intersections. The user can also interactively modify segment placement within the 2D embedding, and the overall embedding will update accordingly. A global to local interactive exploration is provided using hierarchical camera views that are created for subtrees within the structure. We evaluate our method both qualitatively and quantitatively and demonstrate our results by constructing planar visualizations of line data (traced neurons) and volume data (CT vascular and bronchial data).

Index Terms—Geometry-based techniques, camera view generation, planar embedding, biomedical visualization

1 INTRODUCTION

MANY critical biological systems, such as the vascular, respiratory, and nervous systems, consist of structures that express a treelike topology. Recent advancements in image acquisition technology coupled with techniques for 3D reconstruction and visualization [1] have enabled scientists to examine the attributes of these intricate structures at a finer resolution, such as their geometry, branching pattern, curvature, and flow capacity. However, 3D exploration becomes challenging as the branching morphology of an object grows in complexity. This is primarily due to data density clutter, visual occlusions, and at times, the inability to seamlessly manipulate camera orientations within the visualization application during inspection. Due to these challenges, domain scientists often prefer a simplistic schematic representation [2], [3]. This has motivated us to develop a method for generating planar embeddings of treelike topologies, such that there are no visual occlusions and the 2D layout preserves the geometric attributes of the original structure.

Given the significance of treelike structures in life sciences, planar visualizations can be very beneficial in clinical settings. For instance, it can be incorporated in medical visualization

tools as a guidance map to facilitate 3D inspection. By presenting a single-view representation, physicians can track their observation coverage as well as get a sense of the current 3D view location with respect to the entire structure. Moreover, akin to the common practice of hand-drawing schematic representations, a planar illustration capturing anatomical and physiological attributes can be used for medical record keeping. Using these images, experts can effectively annotate and communicate key features to other collaborators or use it to juxtapose the data with other records to track and detect variations. However, to ensure the effectiveness of these representations, it is important that the planar embedding preserves the geometry of the original structure and, specifically for treelike objects, avoid misleading overlaps and branching distortions.

In this work, we present a method that utilizes the 3D skeleton of a treelike object to construct a planar embedding of its topology. To maintain geometric context, our method progressively divides a tree structure into smaller segments of single branches and computes local projection planes that best preserve the topology of the points in each segment as well as the topology of points that connect adjacent segments, when mapped onto a 2D view. Our approach is in contrast with existing geometry-coherent planar visualization methods [4], [5] that resolve embeddings based on a single, user-defined global projection plane. To this end, we introduce a novel viewpoint determining algorithm, based on particle swarm optimization (PSO) [6], [7], that automatically computes the best projection plane for a set of 3D edges (lines connecting two 3D points). PSO is a computational technique that solves an objective function by distributing a swarm of particles in space and determines a candidate solution based on a given measure of quality. For our algorithm, we formulate a PSO that minimizes loss in geometry information – specifically length, curvature, and

- The authors are with the Department of Computer Science, Stony Brook University, Stony Brook, NY 11794 USA.
E-mail: {pihu, sboorboor, jmarino, ari}@cs.stonybrook.edu.

Manuscript received 14 Sept. 2021; revised 4 Feb. 2022; accepted 19 Feb. 2022.
Date of publication 25 Feb. 2022; date of current version 30 May 2023.

This work was supported by NSF under Grants CNS1650499, OAC1919752, ICER1940302, and IIS2107224.

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by Stony Brook University Human Subjects Committee (IRB) under Application No. 1173920.

(Corresponding author: Arie E. Kaufman.)

Recommended for acceptance by J. Tierny.

Digital Object Identifier no. 10.1109/TVCG.2022.3153871

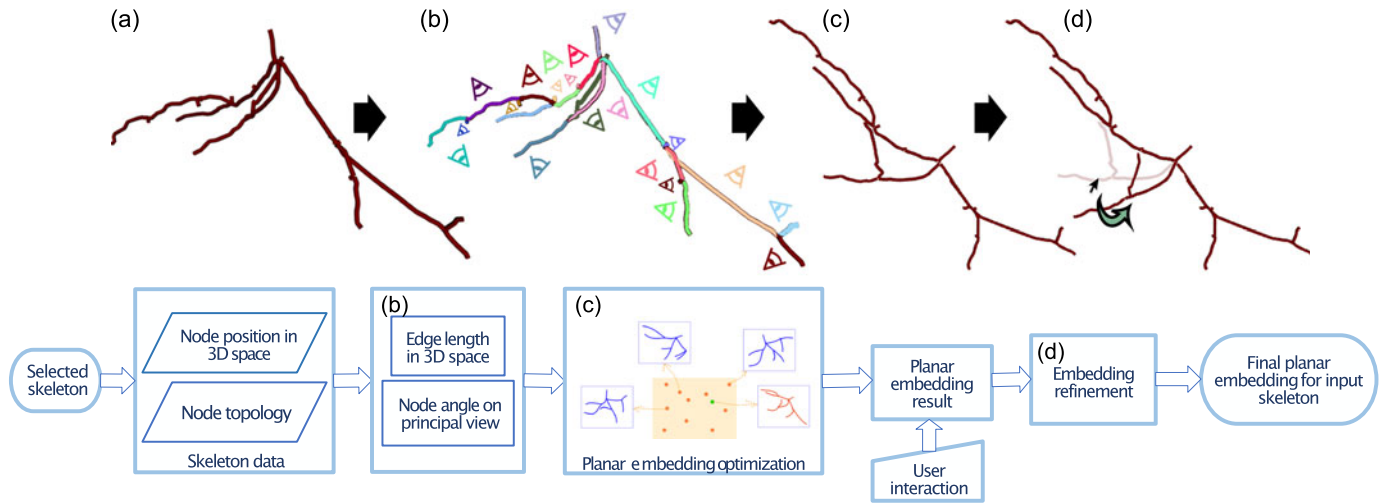


Fig. 1. Overview of our planar embedding algorithm. (a) The original 3D skeleton of a treelike object. (b) The skeleton is divided into branching segments. Local camera views are determined for each segment which best preserve its local geometric attributes when projected to a 2D plane. (c) Planar embedding of the 2D skeleton using the projected attributes from the camera views and the radius of each node. (d) Fine-tuning of the result by the user optionally clicking and dragging nodes. Our algorithm checks again to solve for any intersection introduced by this interaction.

branching angle – when projected onto a 2D plane. Using 2D projections of skeleton branches that are reflective of the original 3D morphology, we compute the set of bending angles that will subsequently be used to generate a planar embedding of the treelike structure. Similar to the viewpoint algorithm, we introduce a GPU accelerated energy function, based on PSO, that generates an embedding with no intersections and with minimal variations from the original 3D edge lengths and bending angles. Realizing that dimension reduction cannot possibly preserve all attributes, in our implementation we allow users to prioritize the preservation of length and angle attributes by adjusting their weights in the planar embedding energy function. Our planar embedding algorithm pipeline is shown in Fig. 1.

Moreover, we extend our viewpoint algorithm to facilitate 3D exploration of treelike objects. Essentially, visualizing 3D objects on a computer screen is analogous to 2D projection and it can be a cumbersome task to manually find a view that avoids visual occlusions. Particularly, for an object with open endings, such as a treelike structure, its shape and morphology are not only perceived by observing its continuous connected edges but also by the spatial relationship between indirectly connected segments. Our viewpoint selecting method handles this characteristic better than the existing methods for mesh-based objects. Finally, for an effective 3D exploration, our technique automatically generates a smooth camera path using a series of Bézier curves.

We summarize the contributions of our work as follows:

- A planar visualization method with interactive weight adjustment for treelike data to achieve an occlusion-free planar view while preserving the local geometry information (that is, the node angles and edge lengths) of each segment of the treelike structures, without requiring any initial viewpoint by the user.
- A planar embedding using automatically generated local viewpoints around the structure to preserve both the overall global shape and the local shape of individual branches.

- An interactive component where the user can manually reposition skeleton nodes and our method will update the final embedding.
- A hierarchical navigation of the original 3D structure based on our principal viewpoint collection.

2 RELATED WORK

The related works is divided in two areas: (1) planar visualization techniques and their applications, and (2) camera view generation.

2.1 Planar Visualization Techniques

Formal user studies have shown that using 2D data representations can be more effective in carrying out tasks that require spatial identification and precision, than 3D visualization [8], [9]. This evaluation is particularly important for clinical applications that involve inspection and diagnosis of medical data, and therefore, there has been a great effort in introducing methods for constructing their planar visualizations. In this section, we discuss techniques for the planar visualization of tubular objects expressing a treelike topology. For a complete review on varying biomedical data, we refer the reader to a survey by Kreiser *et al.* [2].

Broadly, planar embedding of tubular structures can be categorized into techniques that visualize the surface of the objects (interior or exterior), or its internal volume. Most often, surface flattening methods present a single non-occluded view by mapping the surface to a 2D plane in a linear arrangement [10], [11], [12], [13]. To visualize the surface of treelike structures, Lichtenberg and Lawonn [14] have presented a texture coordinate generation technique that can be applied to vascular contour parameterization, feature extraction, and structure segmentation. For a branching topology, Zhu *et al.* [15] have proposed partitioning the vessel anatomy into segments based on bifurcation points and subsequently map the straightened surface segments onto a plane. Moreover, Eulzer *et al.* [13] have developed a cutting and flattening method for vascular geometry, mapping the 3D surface onto 2D domain as a single patch. Though these methods

preserve local geometric properties, the overall context of the object's shape is lost.

To overcome this limitation, Won *et al.* [5] have developed a method that generates a schematic representation of branching structures by defining an optimization problem using simulated annealing. The object is first divided into bounding boxes, and an optimization function then solves for a spatial configuration of the boxes while avoiding overlaps and keeping its topology similar to an input projection view. Marino and Kaufman [4] have described an embedding approach that determines bending angles of a treelike structure by projecting its skeleton onto an input viewing plane. Starting from a modified radial layout, angular positions of the skeletal nodes are iteratively adjusted until the bending angles are recovered to the best possible extent, while avoiding intersections. We consider these two methods to be closest to our goal. While these approaches depend on an input global viewing direction, which may not truly represent the local morphology, our approach does not require a manual viewpoint selection to represent the global shape. In contrast, our method aims to preserve local morphology by projecting segments to automatically computed local planes. Particularly compared to [4], which uses a single viewpoint, we create a viewpoint for each branch so that both global and local shape are preserved to the greatest extent possible.

For visualizing the internal volume of vessels, curved planar reformation (CPR) is a commonly used technique and has been developed to map the lumen of a blood vessel on a 2D plane [16]. To solve for a tree topology, later works suggest applying CPR to individual branches and then compositing the results to generate a final visualization [17], [18], albeit being susceptible to occlusions. For removing occlusions, constraints on spatial relations have to be relaxed by stretching or straightening the vessels, at the cost of distorting geometric context. Many extensions to CPR have been presented that improve rendering of the interior volume and enhance visibility in the presence of other vessels or objects [19], [20], [21], [22]. Since CPRs also embed the volume surrounding the vessels, the different techniques trade off between overlapping vessels or losing isometry. Our work only embeds the segmented object and hence does not consider preserving the geometric context of the surrounding structures.

Particularly in biomedicine, tree structures often have a direct spatial correspondence and hence, to facilitate comprehension, developing applications for its visualization has been an ongoing area of research [23]. VesselMap [24] has been developed as an interactive visual analysis framework that presents a view-dependent 2D graph layout optimized for visualizing cerebral aneurysm and parent vasculature. NeuroLines [25] provides a visual interface to help neuroscientists analyze the connection between individual trees to identify appropriate attributes for neurite comparison. HemoVis [26] has been developed as an interactive visualization application for heart disease diagnosis that reconstructs a 2D tree diagram representation of coronary artery trees. While effective for comparison tasks, these diagrammatic representations are foreign to physicians, who are accustomed to the original shapes of the objects [27]. To encode a correspondence between the 3D structure and its 2D representation, Lichtenberg *et al.* [28] have proposed using a colored binary tree. Moreover, their work also visualizes the distance field on

the both views, based on the requirements of different application scenarios. We demonstrate that among other representations, our embedding result can be used as a visual interface for inspecting the topology of complex treelike objects.

2.2 Camera Viewpoint Selection

Given a scene content, a view is determined by the configuration of the camera. Through a good viewpoint, fundamental data can be perceived and significant objects can be recognized. Since viewpoint selection is a broad research topic, we focus on the relevant works adopting low level attributes of the view content and the viewpoint selection metrics. Bonaventura *et al.* [29] have summarized the pioneering viewpoint selection methods for polygonal data. Kamada and Kawai have proposed a basic rule for finding a good view, maximizing the projected lines on each object [30]. Inspired by information entropy theory [31], [32], many works have designed methods that formulate the problem in entropy space. Arbel *et al.* [33] measure object recognition in a monochrome view, which studied the view selection problem based on Shannon entropy maps. Vázquez *et al.* [34] have further modeled view selection as an object visibility maximization problem by transforming view finding into a visibility probability space. The attributes used are the projected area of each unit in the mesh-based objects. Stoev *et al.* [35] have designed a method for automatic camera positioning in time, considering projected area and scene depth. Podolak *et al.* [36] automatically choose a good viewpoint by minimizing the visible object symmetry.

View selection methods have also been investigated for volume data [37]. Bordoloi *et al.* [38] have developed a viewpoint selection method for static volume visualization and their method provides a set of representative views for the volume that partition the view space into several subsets based on view similarity. For dynamic volumes, Ji *et al.* [39] have formulated a utility function that combines voxel opacity, color, and curvature information. For terrain data visualization, Stove and Straber [35] have proposed considering maximum depth, in addition to projected area. Focusing on the camera views inside blood vessel models, Meuschke *et al.* [40] have developed an automatic viewpoint selection technique for exploring simulated aneurysm data. Based on their technique, Apilla *et al.* [41] have presented a camera path generation method to explore the model through optimal viewpoints showing the interesting regions during the cardiac cycle. Recently, methods for viewpoint generation are being developed that take perceptual factors into consideration [42], [43], [44]. Secord *et al.* [43] have proposed a perceptual model that combines attributes, projected model, and silhouette length of the polygonal objects to predict the users' preferences to the viewpoint of a variety of objects.

Our method shares some similarity with Bordoloi *et al.* [38] as it locates the best views to maximize the visible geometry attributes reflected on the skeletal data. However, we generate a series of hierarchical views that discover the details in the treelike structure based on subtree levels.

3 ALGORITHM

Unless visualized in a virtual reality setting, inspecting an object in 3D can be a cumbersome task as it involves

manipulating camera orientations to obtain a view with minimal occlusions and for it to be recognizable with respect to the entire structure. In this work, we address these challenges by introducing two visualization methods. First, a technique for planar embedding of treelike objects that preserves the global shape and local branching morphology of the structure, while avoiding intersections. Second, an automatically generated parallax-aware camera sequence that facilitates an effective 3D navigation of treelike objects. In elucidating shape preservation, we adopt a description similar to [45], where the overall shape of a treelike object is the combination of each branch segment's local shape, characterized by the set of node angles in the segment, along with the relative placement of the segment's siblings. Additionally, we define geometric attributes as branch length, thickness, and branching and bending angle. Briefly, our embedding algorithm takes as an input a 3D skeleton model of the object and follows three main steps: (1) divide the skeleton into segments and determine local viewpoints for each segment such that the projected points preserves the 3D geometric attributes, (2) generate a planar embedding without intersections, and (3) optionally refine the embedding result based on user interaction. An overview of these steps is illustrated in Fig. 1.

We assume that the skeleton model contains a set of nodes located at points in the structure where there is a change in morphology (such as bends or branching points). Moreover, each node should have exactly one parent, any number of children, and optionally, a scalar value to represent the radius of the edge (representing line width), that connects the node to its parent. We define the following terms when referring to the skeleton model:

- A *root* node can have several children but has no parent.
- A node with multiple children creating a junction in the tree is called a *branching* node.
- A node edge length, l , refers to the length of the edge connecting the node with its parent.
- A sequence of nodes with a single child and the edges connecting these nodes are considered a *segment*.
- A *node angle*, θ , is defined as the angle in the clockwise direction between the edges that connect the node to its parent (node-parent edge) and the parent to the node's grandparent (parent-grandparent edge).

In addition to our embedding technique, we have developed a camera sequence algorithm that generates an optimal 3D navigation path for each subtree of the object, based on its geometry information and spatial relationship with other neighboring branches. By generating the set of optimal local camera poses, we generate a smooth camera path using Bézier curves.

3.1 Hierarchical View Finding

The goal of finding *good* viewpoints naturally requires defining a set of metrics that can measure the preservation of geometric attributes. For treelike structures, based on existing literature [45], [46], [47], we have determined the metrics to be length, curvature, and angle, for geometry information; and radius, branching frequency, and adjacency, for morphology information. To this end, we present an algorithm to quantitatively optimize these metrics to

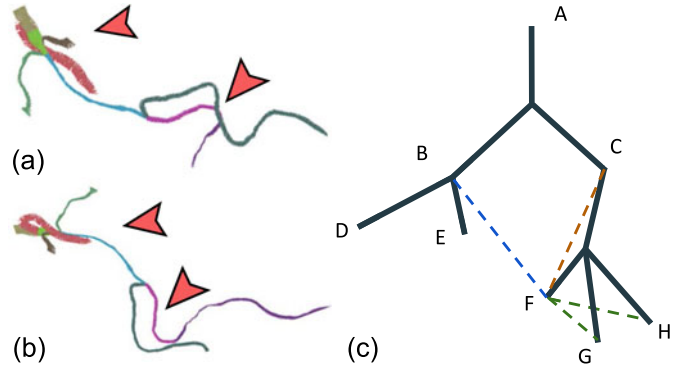


Fig. 2. Global best view generated using (a) the method in [37] and (b) our model. The red arrows show that our *enhanced* tree edges, represented by the dashed lines in (c), result in a more accurate projection of each node's relative placement. The green, red, and blue dashed lines represent three types of imaginary edges for node F.

find good camera viewpoints and subsequently, a method for fast camera path generation using these viewpoints.

3.1.1 Camera View Metric

Feixas *et al.* [48] formulate the measure of geometric attributes in a camera viewpoint as:

$$I(v, O) = \sum_{o \in O} p(o|v) \log \frac{p(o|v)}{p(o)}, \quad (1)$$

where $I(v, O)$ represents the geometry attributes carried by an object collection O in a camera view v , $p(o)$ is the probability that a corresponding object $o \in O$ is visible and conveys information in a scene, and $p(o|v)$ is the probability of an object o being visible in v . Essentially, this equation computes the sum of attribute information expressed for every object in a certain camera view.

While this model is widely used for viewpoint selection in mesh and volume data (see Section 2.2), it does not always yield a satisfactory camera view for treelike skeleton data. This is because treelike skeletons have open endings whose geometry information cannot be fully reflected by preserving only its original length ratio. To address this limitation, we have introduced a method that not only considers original 3D edges, but also a set of additional *imaginary* edges as shown in 2 (c), namely, the connection between each node and (i) its siblings, (ii) its grandparent, and (iii) its parent's siblings. For simplicity, we refer to the skeleton and the imaginary edges as an *enhanced tree*.

For an enhanced tree, we model the geometry attribute of an edge as the ratio between the edge length l and the total length of all edges. By extension, given a camera view, the projected attributes coming from an edge is the ratio of its projected length over the sum of projected lengths in the enhanced tree. The effect of views computed using the additional edges is shown in Figs. 2a and 2b.

3.1.2 Camera View Hierarchy

We present a camera view hierarchy algorithm that facilitates shape understanding by presenting level-of-detail (LoD) information using a computed best global view and a series of local view for fine structures. For each LoD hierarchy, our method determines the best camera position $v_h \in$

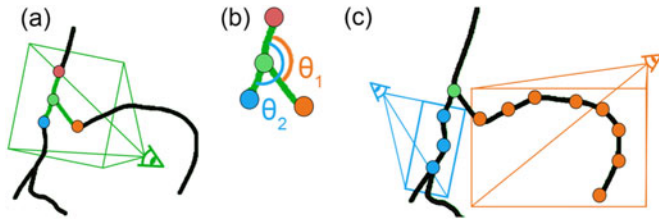


Fig. 3. The process of dividing a skeleton tree into segments to compute its planar angles. (a) A projection plane for a segment consisting of a branching node (green node), its parent (red node), and children (blue and orange nodes) is calculated first. (b) The branching angle for each child node is the angle between the child node, its parent, and its grandparent. (c) For each segment, bending angles are calculated by determining a projection plane that preserves the geometric attributes of the entire segment.

\mathbb{R}^3 . To this end, we formulate a hierarchical view finding optimization such that the camera viewpoint candidates lie on a spherical surface. The radius of this spherical search space is proportional to the subtree expansion radius with its center positioned at the weighted center of all subtree nodes, c^T . The distance D between the camera and the tree center is defined as $D = \beta R^T$, where R^T is the radius of the principal view with a weight factor β . Based on our experiments, we suggest β to be 1.5. Furthermore, the camera look-at direction is the ray from the camera center to c^T and the up direction perpendicular to the look-at direction. Thus, to find the optimal camera viewpoint v_h for a subtree, T_h , in the hierarchy level h of the tree, we define the following objective function:

$$v_h = \arg \min_{v_h} E_v, \quad E_v = I(v_h, T_h) \quad (2)$$

The hierarchical viewpoint set, H , for a tree structure is denoted as $H = \{\{v_h\} \mid \{v_{h=0}\} \cup \{v_{h=1}\} \cup \dots \cup \{v_{h=m}\}\}$, where $1, 2, \dots, m$ represent the hierarchy levels. The number of hierarchical viewpoints in each tree level is equal to the subtree number in that level.

3.2 Skeleton Planar Embedding

To maintain the 3D geometry of a structure in a single planar view, we first determine a method to best preserve its edge lengths and the node angles. To achieve this, we design a scheme for projecting the nodes onto a principal plane using our viewpoint selection method described in Section 3.1.2.

3.2.1 Node Angle

We categorize two sets of node connectivity that determine the target angles for planar embedding: chain and branching. A chain condition is the set of nodes in a segment where each node has a single child. A branching condition is when a node has multiple children. Given a subtree with a root node that has multiple children, a chain representing a branch of the subtree is collected starting from the root along each node that has a single child, as shown in Fig. 3a. A branching set contains a node, its parent and children, as show in Fig. 3b. To calculate target angles, we first project the relevant set of nodes onto a principal plane and then compute, for each node, the angle formed between its projected node-parent-grandparent edges, in a counterclockwise order. Given a set of nodes, the principal plane is the

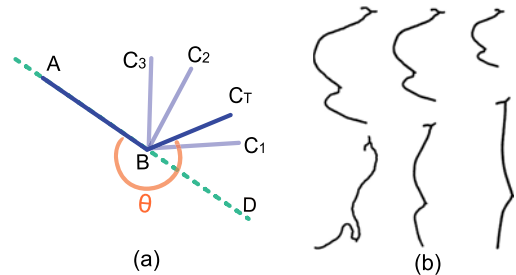


Fig. 4. (a) An example of the impact of r_a on θ_e with the node C 's target angle θ ($\angle ABC_T$ in counterclockwise order) as an reflex angle. If $r_a > 0$, node C is positioned at the range between BD and BC_T , e.g., C_1 . If $r_a = 0$, node C is at C_T . If $r_a < 0$, C is positioned at the range between BC_T and BA , e.g., C_2 or C_3 . Thanks to our constrain, C will not be on the C_T 's opposite side against the dash line. (b) Segment shapes with different parameters. The top left structure is the original skeleton. The top row shows the changed shape with varying length ratio r_l (left to right: 0.0, 0.2 and 0.5, respectively) and a constant r_a ($r_a = 0$). The bottom row shows the result of varying r_a (left to right: -0.02, 0.2 and 0.5, respectively) and $r_l = 0$.

plane passing through the subtree center and perpendicular to the look-at direction of the optimal camera view calculated using the method introduced in Section 3.1.

3.2.2 Embedding Optimization

Using the target angles and edge length, we formulate the planar embedding problem as a UV coordinate optimization procedure. In other words, we aim to find a position for each node on a planar view that minimizes loss in target angles and edge lengths. However, to ascertain geometry preservation of the tree structure, we cannot directly apply a constant minimization factor to all edges and nodes. For instance, applying a certain shrinking factor to a longer edge can account for a heavier loss in geometry than an edge with a smaller length. To address this, we minimize the percentage loss of edge length and node angle, denoted as r_l and r_a respectively, in the candidate 2D embedding. Given r_l and r_a , an edge length $l_e = (1 + r_l)l$, and a node angle θ_e is

$$\theta_e = \begin{cases} \theta + (\pi - \theta)r_a & \text{if } \theta \leq \pi \text{ and } r_a \geq 0 \\ (1 + r_a)\theta & \text{if } \theta \leq \pi \text{ and } r_a < 0 \\ \theta - (\theta - \pi)r_a & \text{if } \theta > \pi \text{ and } r_a \geq 0 \\ \theta - (2\pi - \theta)r_a & \text{if } \theta > \pi \text{ and } r_a < 0 \end{cases} \quad (3)$$

where θ is the target node angle calculated in Section 3.2.1. We constrain $r_l \in (0, 2.0]$ and $r_a \in [-1.0, 1.0]$ in the following optimization step. Fig. 4a shows an example of the θ_e based on different r_a when $\theta \in (\pi, 2\pi)$. By applying Eq. (3), the relative bending direction of the nodes will be maintained although the angle value may be changed, as shown in Fig 4b.

We optimize a single r_l and r_a for each segment using:

$$R_{li}^*, R_{ai}^* = \arg \min_{R_{li}, R_{ai}} E_p \\ E_p = \sum_i^M w_l \cdot r_{li}^2 + w_a \cdot r_{ai}^2 + w_X \cdot X(r_{li}, r_{ai}) \quad (4)$$

where w_l and w_a are the weights for the length and angle loss ratio, respectively, M is the total segment amount, w_X is the penalty weight for intersecting edges, and $X(r_{li}, r_{ai})$ are the corresponding length and angle responsible for the intersection in the current placement. We report our parameter

values in Section 5. The resulting node placement of a simple tree structure with different r_l and r_a settings are shown in Fig. 4b.

Since the search space for this optimization is not convex nor differentiable, we apply PSO [6] to find a local optimal embedding solution. PSO is a robust approach that controls parameters without depending on a search space gradient. In our algorithm, the skeleton placement is determined by the combination of the $\{r_l\}$ and $\{r_a\}$ ratios, denoted as R_l and R_a respectively. Specifically, each swarm particle P_i represents a candidate solution for the placement. R_l and R_a are initialized randomly. The energy E_p of this embedding is calculated using Eq. (4) that quantifies the goodness of a candidate nodes placement. A lower energy corresponds to less geometry loss and fewer intersections. In each iteration, a particle is updated based on its moving velocity U_i , current ratio set $\{R_{li}, R_{ai}\}$, historically best ratio set P_{ig} , and the ever best particle P_g amongst all particles. This is formulated as:

$$P_i^{c+1} = \omega_g \cdot (P_g^c - P_i^c) + \omega_p \cdot (P_{gi}^c - P_i^c) + \omega_{inert} \cdot (U_i^c + \gamma^c) + P_i^c \quad (5)$$

where ω_g , ω_p , ω_{inert} represent the effect of the global best ratio set, the best ratio set within the particle history, and the inertial velocity of the particle itself in the searching space, respectively, c refers to the iteration count, and γ is a randomized velocity value to increase the search diversity and accelerate convergence. U_i^c is calculated as the difference of the corresponding ratios between the (c) th and $(c-1)$ th iteration. The optimization is terminated when the geometry loss is zero or when the maximal loop limit is reached. Algorithm 1 provides a pseudocode of our method.

Algorithm 1. Planar Embedding Using PSO

Result: Optimal ratio set $\{R_l^*, R_a^*\}$
Input: Randomize R_l, R_a, U_i for each P_i ;
 $P_0 \leftarrow \{r_{lm} = 0, r_{am} = 0\}, P_1 \leftarrow \{Q_l, Q_a\}$;
 $P_g \leftarrow P_0, P_{ig} \leftarrow P_i, E_{pi} \leftarrow \infty, c \leftarrow 1, c_{max} \leftarrow 100$;
1: **while** $E_p > 0$ and $c \leq c_{max}$ **do**
2: For each P_i :
3: generate node placement based on its R_l, R_a ;
4: Update E_{pi} using $E_p(P_i)$ defined in Equation (4);
5: $P_{ig} = \min\{P_{i0}, P_{i1}, \dots, P_{ic}\}$;
6: End for;
7: $P_g = \arg \min_{P_i} E_p(P_i)$;
8: For each P_i :
9: $\{R_l, R_a\}$ is updated using Equation (5);
10: End for;
11: $c = c + 1$;

To encourage non-intersecting node placement candidates and to accelerate the optimization convergence, we propose setting w_X to have a value much larger than the other weights. In evaluating our experiments, we determined this weight to be 1.5 times the sum of the maximal edge length loss and the maximal angle loss. Since both the possible maximal angle and edge length loss equals the total number of edges in the tree, a node placement with a single edge crossing will have a higher energy than a placement without any edge crossing, in our implementation. Moreover, to ensure at

least one non-intersecting solution, we additionally generate a radial layout of the tree, using the method in [4], and calculate its $\{R_l, R_a\}$ as Particle 1's initial values, $\{Q_l, Q_a\}$. Thus, in contrast to a general PSO solution, our method guarantees a final non-intersecting placement, no worse than the radial layout.

3.3 Interaction

Researchers have varying preferences with regards to the strictness in geometry attributes preservation. For example, a neuroscientist may want to investigate the distance between endpoints of a traced neuron skeleton and as a result, may want to maximize the weight of length preservation. Likewise, an anatomist may prefer increasing the weight of angle preservation to observe curvatures along an anatomy. To facilitate this, our technique supports the ability to interactively adjust the length and angle weights.

Furthermore, our implementation also supports the interactive refining or adjustment of the planar embedding result. Specifically, a user can specify any curve or curve segment and rotate it around an anchor node. Following the manual adjustment, our method checks to remove any newly introduced intersections. An example of this user adjustment is shown in Fig. 6. It is worth noting here that the optimization post manual intervention considers the ratios of the user-adjusted segment to be fixed and only recalculates the angle and length ratios of other segments.

3.4 Exploration in 3D Space

It is inevitable to lose some 3D information when reducing the dimensionality from 3D to a planar embedding. Particularly, information such as the relative position between subtrees and the expansion in 3D space may not be maintained in a 2D view. To fully uncover the geometry information carried in a 3D structure, our method generates an exploration path based on the hierarchical view set H in Section 3.1. To support the effective investigation of geometry information inherent in a 3D structure, we introduce a technique to generate an exploration path based on the hierarchical view set H , discussed in Section 3.1.

This exploration path is composed of multiple groups of camera motion sequences. Each group includes two motion sequences: a transition from between consecutive principal views and a camera dolly motion to stimulate parallax. The camera positions in the transition clip is generated using a quadratic Bézier curve connecting one principal view to the next and the camera orientation is interpolated using a smooth transition between the two views. For the dolly motion, the camera is always looking at the center of the subtree and moving along an arc of 90%. Fig. 5 demonstrates an example of the camera path. Moreover, please refer to our supplementary video, available online for further examples.

4 RESULTS

We demonstrate our work using examples from traced neuronal data (one olfactory and two hippocampal neurons from the DIADEM dataset [50]) and segmented vascular (cranial, lower limb, and aorta) and bronchial data. The skeletons for the neuron datasets are extracted using gold-

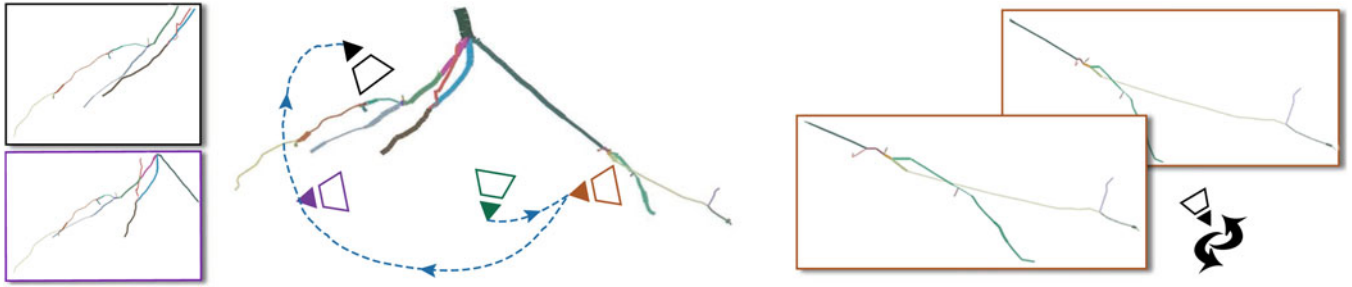


Fig. 5. The original structure in 3D space and four screen shots of our automatically generated 3D exploration views. The exploration starts with the best global view and then moves along a smooth Bézier path for each subtree's best local view. Screenshots of selected camera views are shown in the insets. We also generate a small camera motion to enhance parallax, as demonstrated using the two insets in the bottom right corner. A video of the exploration is available as supplementary material, which can be found on the Computer Society Digital Library at <http://doi.ieeecomputersociety.org/10.1109/TVCG.2022.3153871>.

standard neuron tracing algorithms, provided with the datasets, and the skeletons for the blood vessels are computed using [51]. Fig. 7 shows our planar embedding results for the five datasets, along with their 3D views and a comparison with embedding results generated using the state-of-the-art technique by Marino *et al.* [4]. It can be seen here that a single 2D projection of a 3D structure introduces distortions and false intersections. Whereas our embedding results can effectively preserve aspects of the original 3D shape using multiple local principal views.

4.1 Planar Visualization

The most simplistic visualization of an embedded structure is to render its projected skeleton as lines, similar to the rendered results in Fig. 10. Such a rendering provides a broad overview of the global and local morphology of a structure and can provide clear indications of specified locations along the skeleton.

A pseudo-surface rendering based on the radii values can be used to generate a 3D appearance of the embedded structure and also present a surface on which attributes can be mapped to color. Fig. 8 shows such a surface rendering of segmented blood vessels from a CT scan. The color map¹, from a shade of blue to yellow to red, represents the gradient of the radii difference of a segment with respect to the average radius of the entire structure, in order of increasing segment radius value.

A planar visualization for studying corresponding structural changes within a structure is possible by juxtaposing the projected structures in a single view. We demonstrate this idea in Fig. 9 using an isolated *healthy* olfactory neuron from a dense brain microscopy volume and its corresponding *diseased* state predicted using [49]. A planar embedding of the neuron's skeleton data (Fig. 9d) is first constructed using our method. Next, using a viewpoint perpendicular to the central axis of the skeleton, we render a bounded view volume using [53] and project it onto a single view along the embedding result. Finally, we juxtapose the two embedded volumes in a single view with an offset for clarity, as shown in Fig. 9e.

4.2 Camera Navigation Path

Exploring information in 3D space is the most natural way to fully investigate a 3D object. With this motivation, our

technique generates a camera path with dynamic focus on the different subtrees in the skeleton for users to automatically explore the data in 3D space. Fig. 5 demonstrates this concept for investigating a hippocampel neuron. This camera path follows a top-down approach, starting with the best global view of the structure and then hierarchically navigating each subtree. A spherical dolly motion around each subtree stimulates visual parallax to aid in understanding the geometry information along the view direction and a smooth transition between the two views gives the user a perception of the spatial relationship between different parts in the structure. We refer the reader to our supplementary video, available online for a more elaborate demonstration.

5 EVALUATION AND DISCUSSION

Our experiments were performed using a PC with an Intel Xeon Gold 6242 CPU @ 2.80 GHz, 64 GB RAM, and one NVIDIA Quadro RTX 6000 graphics card. The PSO algorithms were implemented in CUDA 11.2 such that each GPU thread is responsible for a solution candidate optimization. For our experiments, we set the following parameters: 40,960 swarm particles, $w_l = 2.0$, $w_a = 2.0$, $w_x = 0.2$, $w_g = 0.05$, $w_p = 0.05$, and $w_{inert} = 0.0375$. The performance of our technique is reported in Table 1 and compared to the method in [4], tested in the same environment.

5.1 Geometry Information Loss

We provide a quantitative and qualitative evaluation of our planar embedding results and camera path generation. Moreover, we compare our results with the skeleton embedding method in [4].

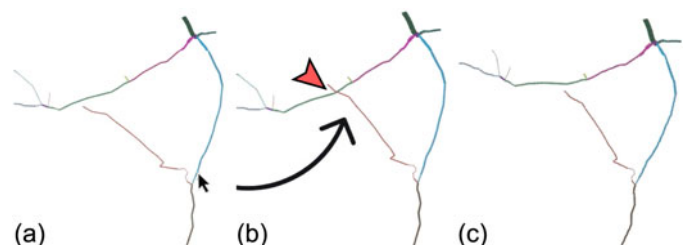


Fig. 6. Interactive branch adjustment. (a) The original embedding result. (b) The user has repositioned a branch, causing an intersection that is indicated by the red arrow. (c) The updated embedding result, with the intersection removed.

1. The diverging color map for three classes was designed using ColorBrewer [52] with the 'color blind safe' and 'print friendly' options.

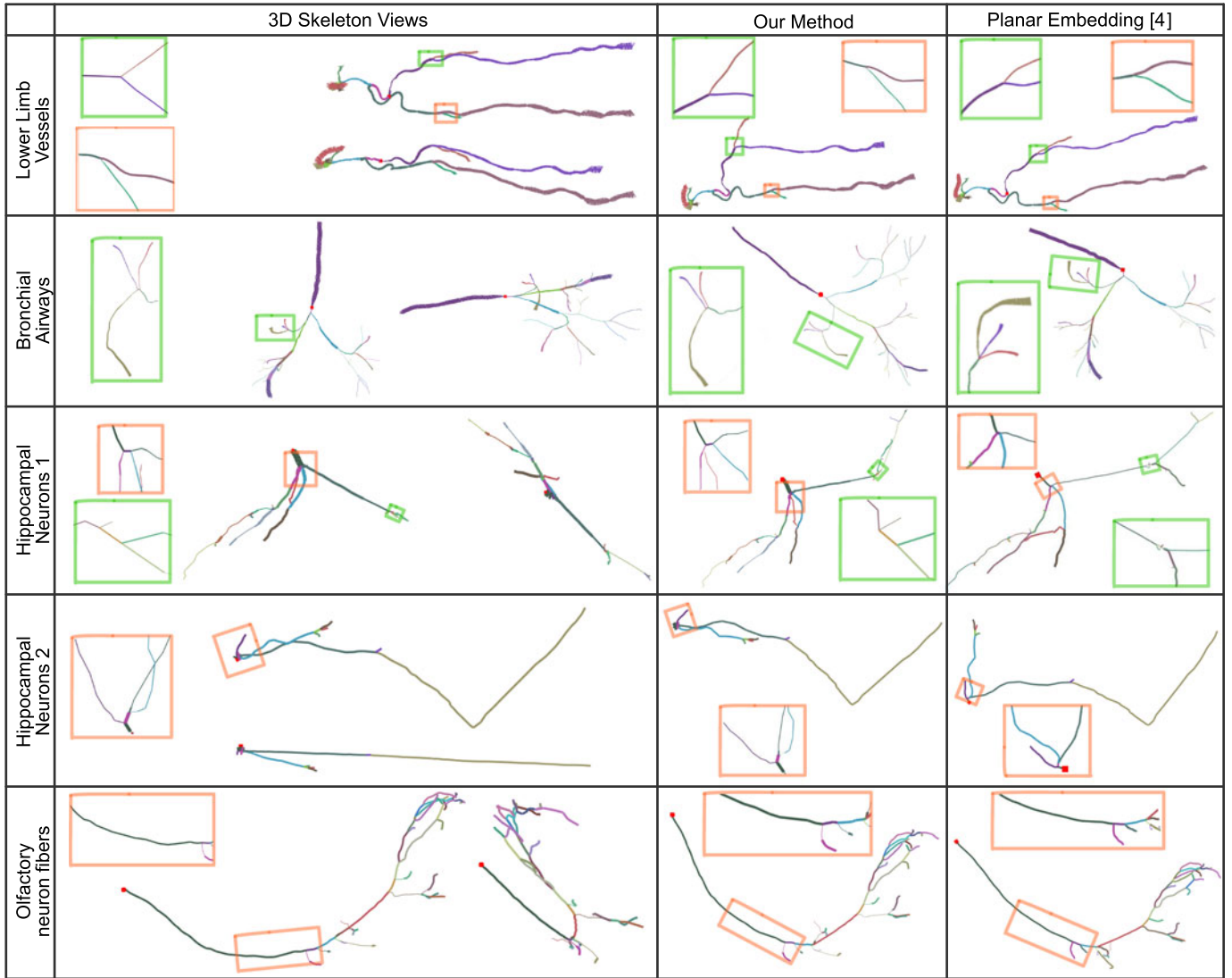


Fig. 7. Planar embeddings using our method and the state-of-the-art method [4]. For shape perception, the left column shows two 3D views of the skeleton with the top / left view being most representative chosen by a domain expert. We use colors to associate the respective segments in 3D and their planar representations. The thickness of the edges represents the node radii. The red square on each structure denotes the skeleton root node. For a closer comparison, the insets highlight the difference in geometry attributes preservation between our planar embedding result and the result using [4], compared to the original 3D view. Note that since [4] requires an input view, for a fair comparison, we generate the planar embedding from [4] using the expert-defined view.

5.1.1 Quantitative Analysis

Table 2 reports the geometry loss analysis using two metrics.

Metric 1. Projected Angle and Length: We evaluate the loss L_l and L_a by comparing our embedding with the target edge lengths and with the target node angles, respectively:

$$L_l = \sum_i^{N-1} \frac{|l_i - l_{ei}|}{l_i}, \quad L_a = \sum_i^N \frac{|\theta_i - \theta_{ei}|}{\theta_i}$$

where N is the number of nodes.

Metric 2. Angle and Length in 3D Space: Since the ultimate goal of planar embedding is to preserve the 3D geometry information, we also compute the edge length and angle loss compared to the original 3D structure. In the original structure, each node angle is the acute angle between the edge and its parent edge. The edge length and angle loss for the root node is 0. We show that our technique can maintain the segment length well and the overall angle loss is small.

5.1.2 Qualitative Analysis

As mentioned in Section 2.1, we consider [4], [5] as methods closest to our work and [4] as the state-of-the-art. The algorithm in these methods greatly depend on an appropriate input view, thus requiring users to manipulate 3D camera for an optimal input view. This is a challenging task that has motivated our work to require no user-defined input view. Furthermore, the embedding from these method only demonstrate limited geometry information in the original 3D skeleton. Fig. 7 shows that our method outperforms [4] regarding visualizing segment shapes, especially when the input structure expands in all three dimensions in space, e.g., the bronchial airways, the cranial blood vessels, and the dense neuron fibers in olfactory neurites.

Fig. 10 shows how a single global view can lead to misleading distortions. Specifically for complex treelike objects where branching structures have varying principal axes,

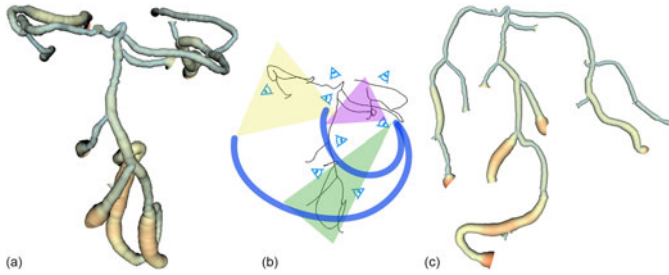


Fig. 8. Our planar embedding method for the cranial blood vessels. (a) A 3D mesh rendering of an original structure of cranial blood vessels. (b) Illustration of three camera view fields focusing on a certain subtree structure. The blue curves represent a navigation example which connects the three hierarchical views. (c) Planar embedding result using our energy function that preserves the global shape of the object along with its local morphology, while avoiding intersections. The color represents the blood vessel radius of the segment with respect to the average vessel radius of the entire structure.

reducing the image space dimensionality can project the structure in a way that can distort its local morphology. The central structure in Fig. 10a is a global projection of cranial blood vessels used in [4] to generate a planar embedding, and the corresponding insets show alternate projections from different 3D camera viewpoints. This global view gives a perception that the branch AC has a *curled* morphology. However, by observing the same branch from different viewpoints it can be seen that the AC has a much smaller curvature. In comparison, since our embedding result uses local projection views to maintain local morphology and spatial relation with neighboring nodes, we see that the embedded segment AC in our result, shown in Fig. 10c, does not have the misleading distortion.

5.2 Expert Feedback

We presented our technique to five experts (A1, A2, A3, A4, A5) from different research domains to comprehensively evaluate our method. Among the experts, A1 was a scientist in scientific visualization studying brain connectomics, A2 was a graduate student pursuing Doctor of Medicine who worked on medical visual analytics, A3 was a neuroscientist focusing on microscopy biology data including neurons, A4 was a doctor in nutritional science who heavily uses hierarchical visualization tools to organize datasets and conducts

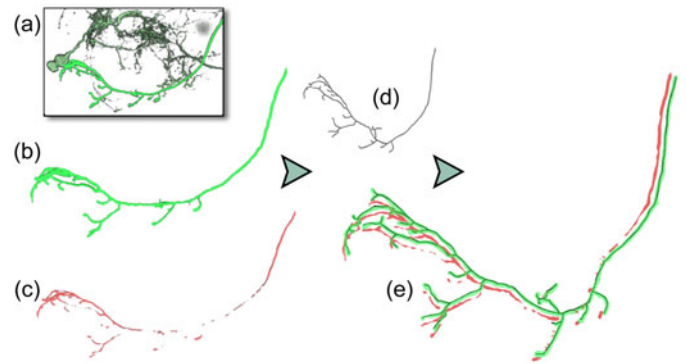


Fig. 9. Juxtaposition of olfactory neurons for neurodegeneration pattern analysis using our planar embedding method. (a) Brain microscopy volume. (b) Neurite with its extensions which was extracted from the microscopy data. (c) Corresponding predicted Alzheimer's disease structure [49]. (d) Planar embedding. (e) Juxtaposition of neurite (b) and prediction (c) using the planar embedding (d) with an offset for clarity.

data mining, and A5 was a scientist in single cell genomics, studying the evolutionary trajectories of plant cells.

Each expert participated in this evaluation separately. They were presented with the planar embedding of interested datasets, the interactive embedding GUI, and the automatic exploration in 3D space, followed by an interview to collect their feedback. A1 observed the neocortical axon structure and the bronchial airways; A2 observed the lower limb vessels, the upper aorta vessels, the olfactory neuron fibers, and the neocortical axon structure; A3 examined the neocortical axon structure; A4 observed the bronchial airways and the olfactory neuron fibers; and A5 studied one hippocampal neuron model and the olfactory neuron fibers.

The interview started with a description of treelike structure visualization. Next, the experts were asked three profile questions: "Do you have any experience of studying treelike structures datasets?"; "When studying your data, what are your focused attributes?"; and "Can you give a score (on a scale of 1 as no expertise, to 5 as authoritative understanding and experience) to rate your expertise on general data visualization and data visualization with treelike structures, respectively?". Experts' self ratings for the first and third questions are (5, 4, 5, 3, 4) and (4, 3, 5, 2, 2), respectively. For 3D navigation, experts chose the hierarchical views and

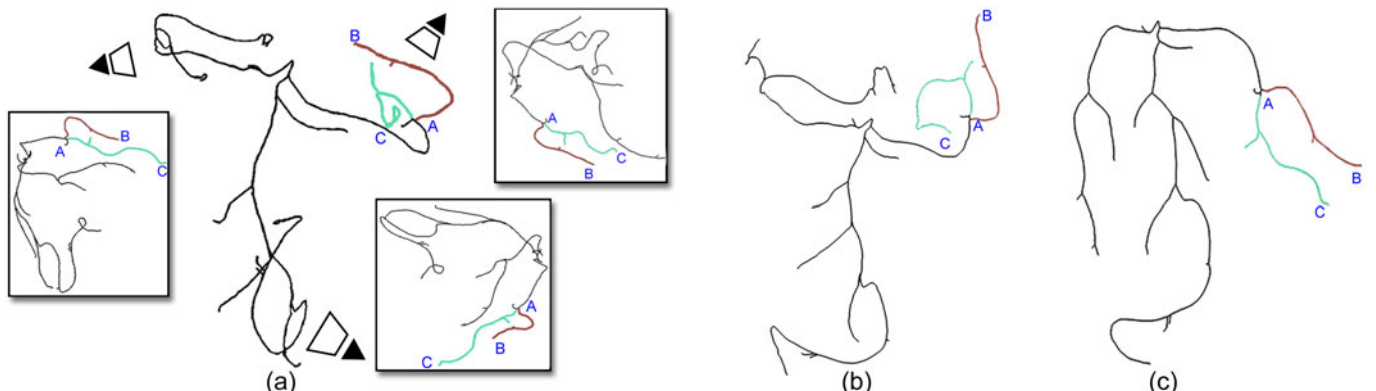


Fig. 10. Preservation of local morphology in our work compared to a global-view based technique [4]. (a) Different camera views of the structure, which may result in intersection and distortion of other branches, therefore making it difficult to find a global view that best captures the morphology of all branches. (b) Embedding result from [4] that uses the global projection, resulting in misleading distortions highlighted using the teal-colored branch AC. (c) Embedding using our method, which more accurately captures the local branching morphology.

TABLE 1
This Table Reports the Computation Time (in Seconds)
for Our Method and the Method in [4]

Dataset	Node number	Ours	[4]
Lower Limb Vessels	1538	212.5	11.5
Cranial Blood Vessels	1831	301.8	16.5
Aorta Vessels	2810	90.3	61.3
Bronchial	1031	119.9	4.5
Hippocampal Neurons 1	175	11.6	17.8
Hippocampal Neurons 2	101	5.1	12.1
Olfactory Neurons	235	19.2	13.9

We ran our algorithm 5 times and report an average time for each dataset. The first four datasets are densely sampled skeletons while the remaining are relatively sparse. Our method is sensitive to the number of nodes and node density, thus leading to an increased computation cost with increasing node density.

navigated themselves using a GUI slider to control the speed. We then asked “What features are helpful for your research?” and “What can be improved to better assist your research?”. Additionally, participants was asked to rate the overall practicality of our method and the two major functions, i.e., planar embedding with interactive edit, and suggested 3D exploration path. A1, A2, A3, and A4 rated the overall method and the embedding as 4. A5 rated the overall method as 5. For the itemized score, A1 rated the navigation as 5 and the others rated it as 4.

The feedback from the experts were positive. All experts appreciated the skeleton embedding with segment shape preservation. According to A2, in vascular studies, it is important to reveal the vessel’s branching pattern and connections since they are unique for each person. A3 noted that “This technique can be very useful in studying neuron sample’s structure”. A5 highly rated the planar embedding because “the method can be very useful for studying plant root system”. A1, A2, A4, and A5 favored the camera dolly motion around each hierarchical view in the path. A1 said, “This is helpful for understanding the neural connections coming from different areas in the brain”. A2 preferred our LoD function for 3D treelike structure. Both A1 and A3 emphasized that they would like to use the 3D exploration and 2D embedding visualization together because the combined modes can invoke different insights. A5 commented that this effective visualization tool simplifies the study of complex root structures. Regarding what can be improved, A1 commented that the sub-structures in 3D navigation view should be dynamically highlighted in the 2D embedding to enhance structural understanding. A2 suggested to

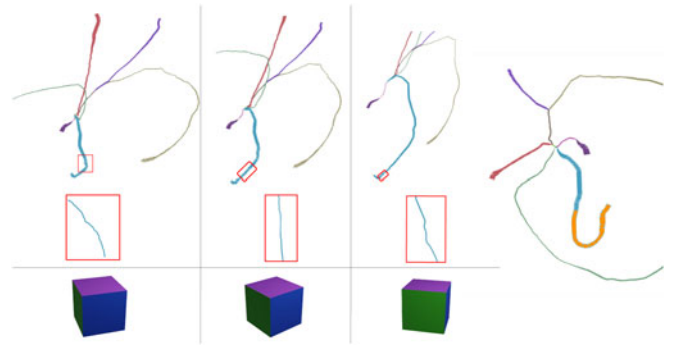


Fig. 11. Local shape preservation and an example of avoiding intersections. Left: details of the node angles are shown by zooming in. The cube below each subfigure shows the relative transformation between each view. Right: in the blue curve, the bending near the end is caused by the potential conflict with the green curve below. Our method adjusts the expansion scope of the blue curve based on the target of optimizing the sum of the geometry loss.

tune the distance between the camera and the center of the data and to adjust the visualized width of the segment in the skeleton structure for aesthetic purposes. A4 suggested indicating the existing manual edits on the embedding to help researchers be aware of the accumulated changes.

5.3 Discussion and Limitations

Our planar embedding method preserves segment length, angles, and high level shape, especially revealing detailed node angles that are easily overlooked by people in global view observation. Fig. 11 demonstrates an example of this. Although the blue curve in the upper aorta vessel structure appears to be quite straight in a global view, we can observe bending when zoomed into the subtree. However, since our method avoids intersection, some large bending angles may be introduced during optimization, when a curve’s expansion conflicts with other curves. For example, we notice a bending near the of the blue curve in Fig. 7. This is caused by the conflict between the expansion scope of the blue curve during the green curve’s placement below it.

While we optimize an exhaustive list, more attributes that preserve treelike structures need to be explored. For instance, relative position between different segment pairs may be a useful geometry attribute in some research domains. Since each segment is projected onto the principal plane which is determined by the hierarchical view, the relative location of segments on a projection of an arbitrary view is not always consistent with the view of the principal plane.

TABLE 2
Angle Loss Comparison Between Our Method and the Planar Embedding Method in [4]

Dataset	Metric 1: Projected Angle				Metric 2: Angle in 3D			
	Max $L(a)$		Avg $L(a)$		Max $L(a)$		Avg $L(a)$	
	Ours	[4]	Ours	[4]	Ours	[4]	Ours	[4]
Lower Limb Vessels	0	1.560	0	0.788	0.140	5.818	0	0.016
Cranial Blood Vessels	0.164	4.597	0.001	0.663	0.770	2.546	0	0.021
Aorta Vessels	0.090	5.279	0.001	0	0.634	2.951	0.005	0.082
Bronchial	0.001	4.395	0	0.996	1.097	1.923	0	0.047
Hippocampal Neurons 1	2.901	1.785	0.017	0.658	1.847	3.091	0.005	0.013
Hippocampal Neurons 2	0.119	3.528	0.001	0	0.248	2.225	0	0.010
Neuron Olfactory	0.275	7.919	0.002	0	0.354	2.049	0	0.029

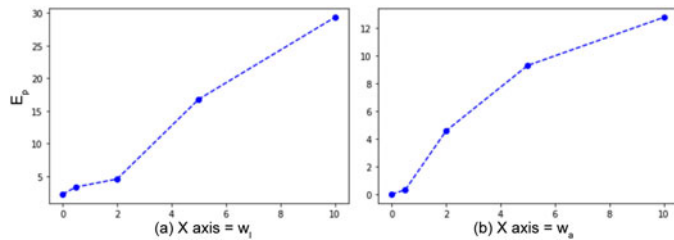


Fig. 12. The relationship between the embedding's geometry loss and the parameters w_l , w_a in the optimization. Left: $w_a = 2.0$. Right: $w_l = 2.0$

5.3.1 Parameter Analysis

The combination of w_l and w_a reflects the importance of length and angle preservation. With a fixed w_a , increasing w_l yields a smaller edge length loss. Similarly, fixed w_l and increasing w_a results in a smaller angle loss. Fig. 12 demonstrates geometry losses for w_l and w_a combinations, using Metric 1 on Hippocampal Neurons 1.

The parameter adjustment allows the users to assign varying w_l and w_a for different nodes based on their specific visualization needs. For instance, to impose strict length preservation closer to the root node, the nodes with the shorter depth can be assigned a higher w_l than other nodes in the tree.

5.3.2 Limitations

The PSO time complexity is $O(N^2)$. The performance bottleneck is the segment intersection checking task in each loop. This intersection checking is composed of a coarse and fine intersection checking. The coarse check facilitates faster computation when the nodes are sparsely placed. As shown in Table 1, when optimizing the planar embedding for structures with dense nodes, our time cost is more expensive than the method in [4].

One failure case is caused by extremely large radii of the tree segments. Since our embedding preserves the edge radius, the intersection checking stage filter out the conditions when wide edges overlap with each other. Therefore, it is guaranteed that when applying the surface mesh onto the skeleton, no occlusion will be introduced. In the cases with extremely large radius there may not even be a valid radial layout embedding, thus resulting in no solutions for the planar embedding.

5.3.3 Applications

It has been shown that 2D representation of medical data helps carry out tasks more effectively [26] and therefore, by preserving the shape of the original structure, a domain expert can more efficiently navigate the 3D structure to confirm the findings from the planar view. Preserving the shape also allows domain scientists to present the representation to collaborators who will be able to recognize the anatomy. In this paper, we used neurite analysis and blood vessel visualization as two examples to demonstrate the utility of our method.

Vascular Analysis. Though 3D rendering already provides a better view than naïvely scrolling through slices of the CT scan, its visualization efficiency depends on the user's ability to seamlessly manipulate camera views within the

application to study the entire structure and simultaneously make a mental note of features such as radii and abnormalities. In contrast, using a planar visualization as in Fig. 8c, the user is able to visualize all of the blood vessels and compare their geometric attribute (i.e., radius) within a single view. Moreover, at an application level, we have shown that our camera view technique can be used to generate viewpoints that results in minimal visual occlusion and maximizes the spatial understanding of the structure.

Neurite Analysis. Connectomics [54] is an emerging field for techniques that study complex neural connection maps. By mapping the brain connectivity, neuroscientists will be able to analyze how the human brain functions and its degeneration process as a result of cognitive decline or disease [55]. A common challenge in this domain is the immensity of volume data. Recent works [25], [56] have introduced methods to alternatively represent dense neuronal structures as abstracted diagrams, albeit in a linear graph-like arrangement. Though schematic diagrams help study connections and topological layouts, an examination of the shape of the structure is required to understand morphological changes during degeneration. Moreover, given the density of information within a microscopy volume and the intricate neurite morphology, neuroscientists independently observe regions of the brain and make general visual observations using population analysis. Our embedding and visualization method can allow for effective analysis of fragmentation and thinning patterns within a neuron by juxtaposing the *healthy* and *diseased* neurites, as shown in Fig. 9e, instead of separately visualizing them in 3D.

6 CONCLUSION AND FUTURE WORK

We have presented a new method of generating geometry-aware planar embeddings of treelike structures based on their skeletons. Unlike previous methods, our PSO implementation does not require an input view position, and it also accounts for the local morphology of the object. The user can interactively adjust branch locations as desired, and the embedding can be updated based on these modified positions. The local camera projection views generated to create the embedding can also be used for exploration of the original structure in 3D, providing optimal views of the various subtrees. We have demonstrated and evaluated our technique using a variety of real-world data, including blood vessels, bronchi, and traced neuron data. Currently, our work is limited to embedding the skeleton of the structure in 2D. While a pseudo-surface rendering based on the radius values for each node can give a sense of a solid structure, we plan to explore in the future how the actual surface or volume data can be incorporated into the embedding. We will also work with domain experts to further refine our results and incorporate our method into an intuitive user interface which addresses the needs specific to different fields.

ACKNOWLEDGMENTS

Bronchial CT scans courtesy of the EXACT'09 study, lower limb vessels courtesy of OsiriX, upper aorta courtesy of the Visible Korean Human Project, and cranial vessels courtesy of VolVis.

REFERENCES

- [1] B. Preim, A. Baer, D. Cunningham, T. Isenberg, and T. Ropinski, "A survey of perceptually motivated 3D visualization of medical image data," *Comput. Graph. Forum*, vol. 35, no. 3, pp. 501–525, 2016.
- [2] J. Kreiser, M. Meuschke, G. Mistelbauer, B. Preim, and T. Ropinski, "A survey of flattening-based medical visualization techniques," *Comput. Graph. Forum*, vol. 37, no. 3, pp. 597–624, 2018.
- [3] J.-H. Won, J. Rosenberg, G. D. Rubin, and S. Napel, "Uncluttered single-image visualization of the abdominal aortic vessel tree: Method and evaluation," *Med. Phys.*, vol. 36, no. 11, pp. 5245–5260, 2009.
- [4] J. Marino and A. Kaufman, "Planar visualization of treelike structures," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 1, pp. 906–915, Jan. 2016.
- [5] J.-H. Won, Y. Jeon, J. K. Rosenberg, S. Yoon, G. D. Rubin, and S. Napel, "Uncluttered single-image visualization of vascular structures using GPU and integer programming," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 1, pp. 81–93, Jan. 2013.
- [6] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw.*, 1995, pp. 1942–1948.
- [7] J. J. Jamian, M. N. Abdullah, H. Mokhlis, M. W. Mustafa, and A. H. A. Bakar, "Global particle swarm optimization for high dimension numerical functions analysis," *J. Appl. Math.*, vol. 2014, no. 329193, 2014, Art. no. 14.
- [8] M. Tory, D. Sprague, F. Wu, W. Y. So, and T. Munzner, "Spatialization design: Comparing points and landscapes," *IEEE Trans. Vis. Comput. Graphics*, vol. 13, no. 6, pp. 1262–1269, Nov./Dec. 2007.
- [9] D. H. Laidlaw et al., "Comparing 2D vector field visualization methods: A user study," *IEEE Trans. Vis. Comput. Graphics*, vol. 11, no. 1, pp. 59–70, Jan./Feb. 2005.
- [10] A. V. Bartroli, R. Wegenkittl, A. König, and E. Groller, "Nonlinear virtual colon unfolding," in *Proc. Int. Conf. IEEE Vis.*, 2001, pp. 411–579.
- [11] W. Hong, X. Gu, F. Qiu, M. Jin, and A. Kaufman, "Conformal virtual colon flattening," in *Proc. ACM Symp. Solid Phys. Model.*, 2006, pp. 85–93.
- [12] Z. Liang, C. Duan, D. Gu, and H. Lu, "Extracting the inner and outer borders of bladder wall and flattening the extracted wall for MR cystography," *World Congress on Medical Physics and Biomedical Engineering*, Berlin, Germany: Springer, 2009, pp. 322–325.
- [13] P. Eulzer, K. Richter, M. Meuschke, A. Hundertmark, and K. Lawonn, "Automatic cutting and flattening of carotid artery geometries," *Eurographics Workshop Vis. Comput. Biol. Med.*, S. Oeltze-Jafra, N. N. Smit, B. Sommer, K. Nieselt, and T. Schultz, Eds., Eurographics Assoc., pp. 79–89, 2021, doi: [10.2312/vcbm.20211347](https://doi.org/10.2312/vcbm.20211347).
- [14] N. Lichtenberg and K. Lawonn, "Parameterization, feature extraction and binary encoding for the visualization of tree-like structures," *Comput. Graph. Forum*, vol. 39, no. 1, pp. 497–510, 2020.
- [15] L. Zhu, S. Haker, and A. Tannenbaum, "Flattening maps for the visualization of multibranching vessels," *IEEE Trans. Med. Imag.*, vol. 24, no. 2, pp. 191–198, Feb. 2005.
- [16] A. Kanitsar, D. Fleischmann, R. Wegenkittl, P. Felkel, and E. Groller, "CPR: Curved planar reformation," in *Proc. Int. Conf. IEEE Vis.*, 2002, pp. 37–44.
- [17] A. Kanitsar, R. Wegenkittl, D. Fleischmann, and M. E. Groller, "Advanced curved planar reformation: Flattening of vascular structures," in *Proc. Int. Conf. IEEE Vis.*, 2003, pp. 43–50.
- [18] A. Kanitsar, D. Fleischmann, R. Wegenkittl, and M. E. Gröller, "Diagnostic relevant visualization of vascular structures," *Scientific Visualization: The Visual Extraction of Knowledge From Data*, Berlin, Germany: Springer, 2006, pp. 207–228.
- [19] N. Lee and M. Rasch, "Tangential curved planar reformation for topological and orientation invariant visualization of vascular trees," in *Proc. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2006, pp. 1073–1076.
- [20] T. Auzinger et al., "Vessel visualization using curved surface reformation," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2858–2867, Dec. 2013.
- [21] G. Mistelbauer et al., "Centerline reformations of complex vascular structures," in *Proc. IEEE Pacific Vis. Symp.*, 2012, pp. 233–240.
- [22] G. Mistelbauer et al., "Vessel visualization using curvilinear feature aggregation," *Comput. Graph. Forum*, vol. 32, pp. 231–240, 2013.
- [23] H. K. Hahn, B. Preim, D. Selle, and H.-O. Peitgen, "Visualization and interaction techniques for the exploration of vascular structures," in *Proc. Int. Conf. IEEE Vis.*, 2001, pp. 395–578.
- [24] J. Tao et al., "VesselMap: A web interface to explore multivariate vascular data," *Comput. Graph.*, vol. 59, pp. 79–92, 2016.
- [25] A. K. Al-Awami et al., "NeuroLines: A subway map metaphor for visualizing nanoscale neuronal connectivity," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 12, pp. 2369–2378, Dec. 2014.
- [26] M. Borkin et al., "Evaluation of artery visualizations for heart disease diagnosis," *IEEE Trans. Vis. Comput. Graphics*, vol. 17, no. 12, pp. 2479–2488, Dec. 2011.
- [27] P. Angelelli and H. Hauser, "Straightening tubular flow for side-by-side visualization," *IEEE Trans. Vis. Comput. Graphics*, vol. 17, no. 12, pp. 2063–2070, Dec. 2011.
- [28] N. Lichtenberg, B. Krayer, C. Hansen, S. Müller, and K. Lawonn, "Distance field visualization and 2D abstraction of vessel tree structures with on-the-fly parameterization," in *Proc. Workshop Vis. Comput. Biol. Med.*, 2019, pp. 265–278.
- [29] X. Bonaventura, M. Feixas, M. Sbert, L. Chuang, and C. Wallraven, "A survey of viewpoint selection methods for polygonal models," *Entropy*, vol. 20, no. 5, 2018, Art. no. 370.
- [30] T. Kamada and S. Kawai, "A simple method for computing general position in displaying three-dimensional objects," *Comput. Vis., Graph., Image Process.*, vol. 41, no. 1, pp. 43–56, 1988.
- [31] M. Feixas, E. Del Acebo, P. Bekaert, and M. Sbert, "An information theory framework for the analysis of scene complexity," *Comput. Graph. Forum*, vol. 18, no. 3, pp. 95–106, 1999.
- [32] M. Feixas, "An information-theory framework for the study of the complexity of visibility and radiosity in a scene," Doctoral Dissertation, Dept. Lang. Syst. Inform., Polytech. Univ. Catalonia, Barcelona, Spain, 2002.
- [33] T. Arbel and F. P. Ferrie, "Viewpoint selection by navigation through entropy maps," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1999, pp. 248–254.
- [34] P.-P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich, "Viewpoint selection using viewpoint entropy," in *Proc. 6th Int. Fall Workshop Vis. Model. Vis.*, 2001, pp. 273–280.
- [35] S. L. Stoev and W. Straßer, "A case study on automatic camera placement and motion for visualizing historical data," in *Proc. Int. Conf. IEEE Vis.*, 2002, pp. 545–548.
- [36] J. Podolak, P. Shilane, A. Golovinskiy, S. Rusinkiewicz, and T. Funkhouser, "A planar-reflective symmetry transform for 3D shapes," *ACM Trans. Graph.*, vol. 25, pp. 549–559, 2006.
- [37] S. Takahashi, I. Fujishiro, Y. Takeshima, and T. Nishita, "A feature-driven approach to locating optimal viewpoints for volume visualization," in *Proc. Int. Conf. IEEE Vis.*, 2005, pp. 495–502.
- [38] U. D. Bordoloi and H.-W. Shen, "View selection for volume rendering," in *Proc. Int. Conf. IEEE Vis.*, 2005, pp. 487–494.
- [39] G. Ji and H.-W. Shen, "Dynamic view selection for time-varying volumes," *IEEE Trans. Vis. Comput. Graphics*, vol. 12, no. 5, pp. 1109–1116, Sep./Oct. 2006.
- [40] M. Meuschke, W. Engelke, O. Beuing, B. Preim, and K. Lawonn, "Automatic viewpoint selection for exploration of time-dependent cerebral aneurysm data," *Bildverarbeitung fuer die Medizin*. Berlin, Germany: Springer, 2017, pp. 352–357.
- [41] V. Apilla, B. Behrendt, K. Lawonn, B. Preim, and M. Meuschke, "Automatic animations to analyze blood flow data," in *Proc. Euro Graph. Workshop Vis. Comput. Biol. Med.*, 2021, pp. 101–105.
- [42] H. Ono, B. J. Rogers, M. Ohmi, and M. E. Ono, "Dynamic occlusion and motion parallax in depth perception," *Perception*, vol. 17, no. 2, pp. 255–266, 1988.
- [43] A. Secord, J. Lu, A. Finkelstein, M. Singh, and A. Nealen, "Perceptual models of viewpoint preference," *ACM Trans. Graph.*, vol. 30, no. 5, pp. 1–12, 2011.
- [44] P. Hu, Q. Sun, P. Didyk, L.-Y. Wei, and A. E. Kaufman, "Reducing simulator sickness with perceptual camera control," *ACM Trans. Graph.*, vol. 38, no. 6, pp. 1–12, Nov. 2019.
- [45] A. Feragen, F. Lauze, P. Lo, M. de Bruijne, and M. Nielsen, "Geometries on spaces of treelike shapes," in *Proc. Asian Conf. Comput. Vis.*, 2010, pp. 160–173.
- [46] S. Biasotti et al., "Skeletal structures," *Shape Analysis and Structuring*, Berlin, Germany: Springer, 2008, pp. 145–183.
- [47] A. Feragen, S. Hauberg, M. Nielsen, and F. Lauze, "Means in spaces of tree-like shapes," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 736–746.
- [48] M. Feixas, M. Sbert, and F. Gonzalez, "A unified information-theoretic framework for viewpoint selection and mesh saliency," *ACM Trans. Appl. Perception*, vol. 6, no. 1, pp. 1–23, Feb. 2009.
- [49] S. Boorboor, S. Mathew, M. Ananth, D. Talmage, L. W. Role, and A. E. Kaufman, "NeuRegenerate: A framework for visualizing neurodegeneration," *IEEE Trans. Vis. Comput. Graphics*, to be published, doi: [10.1109/TVCG.2021.3127132](https://doi.org/10.1109/TVCG.2021.3127132).

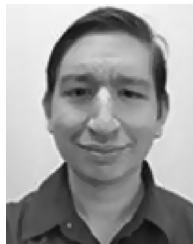
- [50] K. M. Brown *et al.*, "The DIADEM data sets: Representative light microscopy images of neuronal morphology to advance automation of digital reconstructions," *Neuroinformatics*, vol. 9, no. 2, pp. 143–157, 2011.
- [51] I. Bitter, A. E. Kaufman, and M. Sato, "Penalized-distance volumetric skeleton algorithm," *IEEE Trans. Vis. Comput. Graphics*, vol. 7, no. 3, pp. 195–206, Third Quarter 2001.
- [52] M. Harrower and C. A. Brewer, "ColorBrewer.org: An online tool for selecting colour schemes for maps," *Cartographic J.*, vol. 40, no. 1, pp. 27–37, 2003.
- [53] S. Boorboor, S. Jadhav, M. Ananth, D. Talmage, L. Role, and A. Kaufman, "Visualization of neuronal structures in wide-field microscopy brain images," *IEEE Trans. Vis. Comput. Graphics*, vol. 25, no. 1, pp. 1018–1028, Jan. 2019.
- [54] O. Sporns, G. Tononi, and R. Kötter, "The Human Connectome: A Structural Description of the Human Brain," *PLoS Comput. Biol.*, vol. 1, no. 4, 2005, Art. no. e42.
- [55] P. Ghahremani *et al.*, "NeuroConstruct: 3D reconstruction and visualization of neurites in optical microscopy brain images," *IEEE Trans. Vis. Comput. Graphics*, to be published, doi: [10.1109/TVCG.2021.3109460](https://doi.org/10.1109/TVCG.2021.3109460).
- [56] H. Mohammed *et al.*, "Abstractocyte: A visual tool for exploring nanoscale astroglial cells," *IEEE Trans. Vis. Comput. Graphics*, vol. 24, no. 1, pp. 853–861, Jan. 2018.



Ping Hu received the bachelor's degree in physics from Shandong University, China. She is currently working toward the PhD degree in computer science with Stony Brook University. Her research interests include computational camera control in scientific visualization and computer graphics.



Saeed Boorboor received the BSc degree (Hons.) in computer science from the School of Science and Engineering, Lahore University of Management Sciences, Pakistan. He is currently working toward the PhD degree in computer science with Stony Brook University. His research interests include scientific visualization, biomedical imaging, and computer graphics.



Joseph Marino received the BS degree in computer science and applied mathematics & statistics and the PhD degree in computer science from Stony Brook University in 2006 and 2012, respectively. He is currently a postdoctoral research fellow. His research interests include visualization and computer graphics for medical imaging, including visualization and analysis for pancreatic and prostate cancer screening, and enhancements for virtual colonoscopy. He was the recipient of the 2016 Long Island Technology Hall of Fame Patent Award (2016).



Arie E. Kaufman (Fellow, IEEE) received the PhD degree in computer science from Ben-Gurion University, Israel, in 1977. He is currently a distinguished professor of computer science, the director of the Center of Visual Computing, and the chief scientist of the Center of Excellence in Wireless and Information Technology, Stony Brook University. During 1999–2017, he was the chair of Computer Science Department. He has conducted research of more than 40 years in visualization and graphics and their applications, and authored or coauthored more than 350 refereed papers. During 1995–1998, he was the founding editor-in-chief of IEEE TVCG. He is an ACM fellow and National Academy of Inventors fellow. He was the recipient of the IEEE Visualization Career Award (2005), and inducted into IEEE Visualization Academy (2019) and Long Island Technology Hall of Fame (2013).

▷ **For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.**