Constructing High-Resolution Microvascular Models

David Mayerich[1], Jaerock Kwon[1], Yoonsuck Choe[1], Louise Abbott[2], John Keyser[1]

[1]Department of Computer Science, Texas A&M University, USA
[2]Department of Veterinary Integrative Biosciences, Texas A&M University, USA

Abstract— The anatomical structure of the brain microvascular system plays an important role in understanding the function of chemical transport within the brain. However, there have been no imaging techniques that provide the high-resolution and high-throughput data acquisition required to image complete microvascular networks. In this paper, we use a new high-throughput technique to image the entire mouse brain vascular system at a sample rate high enough to resolve all components of the vascular system. We describe the imaging and segmentation methods used to construct a structural model of complex microvascular networks. This model is then used to extract high-resolution anatomical statistics while providing a framework for further study.

I. INTRODUCTION

Complete models of vascular structure are important for understanding several medical conditions. Brain microvasculature has been shown to play an important role in disorders and neurodegenerative diseases including Alzheimer's Disease, Multiple Sclerosis, and Parkinson's Disease [1]. Capillaries, the basic components of the microvascular system, perform important nutritional functions and may also affect the neural response [2]. However, very little is known about the structure of microvascular networks. This is due both to their small size and extraordinary complexity.

Microvascular networks have several properties that make them difficult to both image and model. The capillaries are $\approx 5\mu m$ in diameter, requiring high-resolution imaging for reconstruction. Despite their small diameter, connected components in a capillary network span several cubic millimeters of tissue. Creating complete data sets representing microvascular networks requires imaging entire organs at a microscopic resolution. Advanced imaging methods and segmentation techniques are required in order to cope with these large and complex data sets.

In this paper, we describe a framework for creating high-resolution microvascular models. First, we create a complete image of the mouse brain vascular system in high-resolution using new microscopy techniques. In Section III, we discuss tracking methods used to find the medial axis of capillaries that make up each network. In Section IV, we describe a method for combining topological information with an incomplete or damaged isosurface in order to create a model useful for statistical analysis and simulation. In Section V, we use this model to perform a high-resolution statistical analysis of several important features of the mouse microvascular system. This paper describes two major contributions. First of all, we

Corresponding author: David Mayerich, email: mayerichd@neo.tamu.edu.

create a whole-brain image of the mouse vascular system at the microscopic scale, as well as an image of the somatosensory cortex containing both vascular and cellular information (Sec. II). We also develop a technique for using mutual information from both the network skeleton and isosurface to create compute radii for high-resolution structural models of the microvascular network.

II. IMAGING

Several three-dimensional techniques currently exist for imaging the vascular system at the macroscopic level. Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) are commonly used for imaging vascular structure. Although these methods are non-invasive, the sampling resolution is insufficient for reconstructing a capillary network.

Micro-CT is often used for imaging capillary networks [3], however the resolution is generally limited to $8\mu m - 20\mu m$, while the average microvessel diameter in many mammals drops below $4\mu m$ [4], [5]. Methods such as synchrotron radiation Micro-CT (SRμCT) [6] have been used to reach resolutions of $1.4\mu m$, however the imaging time is prohibitive for entire organs such as the brain. Finally, CT vascular imaging cannot be used to image additional structures, such as cells, in the surrounding tissue.

Microscopy techniques, such as confocal microscopy [7], are sufficient for resolving the three-dimensional structure of capillary networks but are dependent on light penetration into the tissue and are therefore limited to the specimen surface. In order to reconstruct complete microvascular models, we require microscopy techniques capable of imaging tissue on a large scale in all three dimensions.

A. High-Throughput Microscopy

We create our data sets using a high-throughput technique known as Knife-Edge Scanning Microscopy (KESM) [8]. This imaging method overcomes several of the limitations inherent to standard confocal microscopy by using *physical serial sectioning*. Thin sections of tissue are concurrently cut and imaged through a light microscope using a high-speed camera (Fig. 1). This provides several benefits over standard three-dimensional microscopy techniques:

- KESM is not constrained to the specimen surface
- Resolution along the imaging axis is generally higher
- Imaging speed is significantly faster (in excess of 100MB/s)
- No registration or deconvolution is necessary

One of the major disadvantages of using physical sectioning is that the tissue is destroyed during the imaging process. Large volumes are therefore imaged in several separate columns and assembled into a three-dimensional mosiac [9] (Fig. 2). In addition, any information that we wish to associate with the microvascular network, such as cellular data, must be imaged in a single pass and stored in the same data set. These extra structures often reduce image contrast and compete with the capillary network during segmentation.
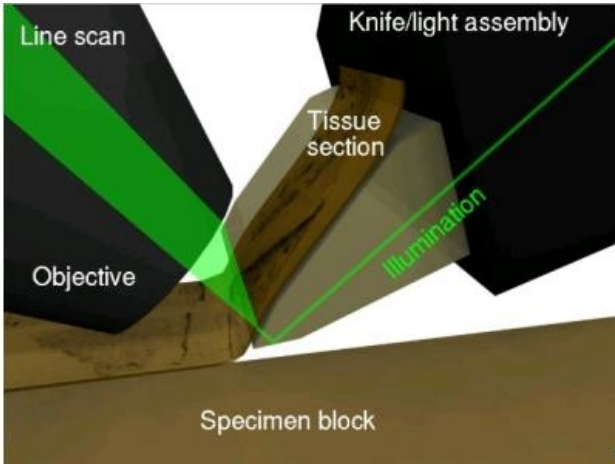

Fig. 1. Imaging and sectioning using Knife-Edge Scanning Microscopy [8]. Tissue is cut using a diamond knife and concurrently imaged with a high-speed line-scan camera. Cut sections are imaged aligned and in-focus, eliminating the need for alignment and deconvolution.
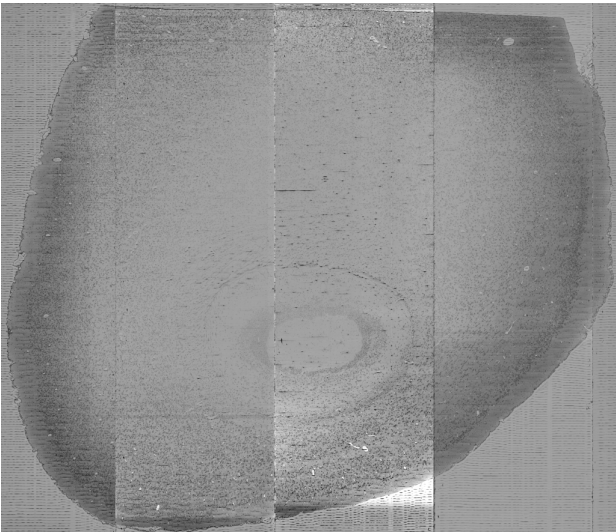

Fig. 2. Creating a mosaic from imaged KESM columns. Each column is initially imaged as an independent three-dimensional data set. The columns are then compiled together into a single data set.

B. Imaging Methods

We created two large volumetric data sets based on different staining procedures for brain microvasculature. We first created a high-contrast whole-brain data set by perfusing India ink through the mouse circulatory system. This is a well-known stain that dyes the microvessels black (Fig. 3). Although this method is high-contrast, it does not provide any information about the tissue surrounding the microvascular system.

As stated previously (Section II-A), physical sectioning destroys the tissue as it is imaged. Therefore, in order to gather as much information as possible during a single imaging pass, we create an additional data set by imaging tissue stained with Nissl [10]. This stain labels all cells and extracellular tissue in the brain while the vasculature remains unstained (Fig. 3). This provides a means of associating cellular information with the vascular structure, although the contrast between the unstained capillaries and the surrounding tissue is greatly reduced.

The whole-brain data set took approximately two weeks to image and requires 2TB of uncompressed storage. We limited the Nissl-stained data set to a large region of somatosensory cortex. This region is important because it allows us to see the change in vasculature and cellular density between white and grey matter. For the whole-brain data set, we focus on regions of high vasculature, including the spinal cord and cerebellum (Fig. 5a and b). For the Nissl stained brain, we examine the somatosensory cortex and underlying white matter (Fig. 5c).
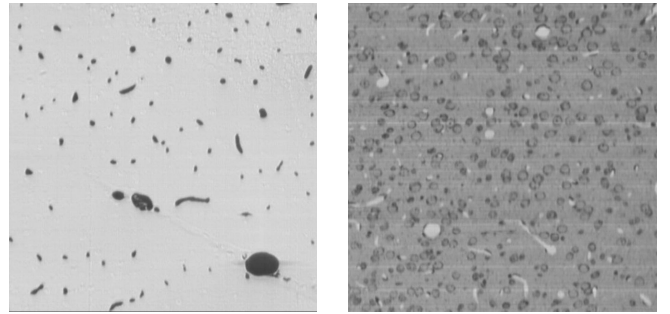

Fig. 3. Cropped samples of tissue stained with India ink (left) and Nissl (right). Vascular filaments are unstained when the tissue is prepared with Nissl, and stained black with an India ink perfusion.

C. Image Processing

Both random noise and lighting artifacts are present in KESM images. This is due to both lighting conditions and mechanical vibrations. Lighting artifacts are removed using a median-based destriping algorithm [11]. This technique is fast and preserves features in the data set, but does not eliminate noise. Since the diameters of microvessels are near the resolving power of the microscope, noise and changes in contrast result in frequent gaps and artifacts in the network isosurface (Fig. 4). In addition, standard image processing techniques such as median filtering and blurring [12] can further damage the network isosurface. Therefore, we elect not to use blurring for noise removal, which may cause data loss and introduce further breaks in the network. Instead we rely on segmentation algorithms that are robust in the presence of noise.

III. EXTRACTING NETWORK STRUCTURE

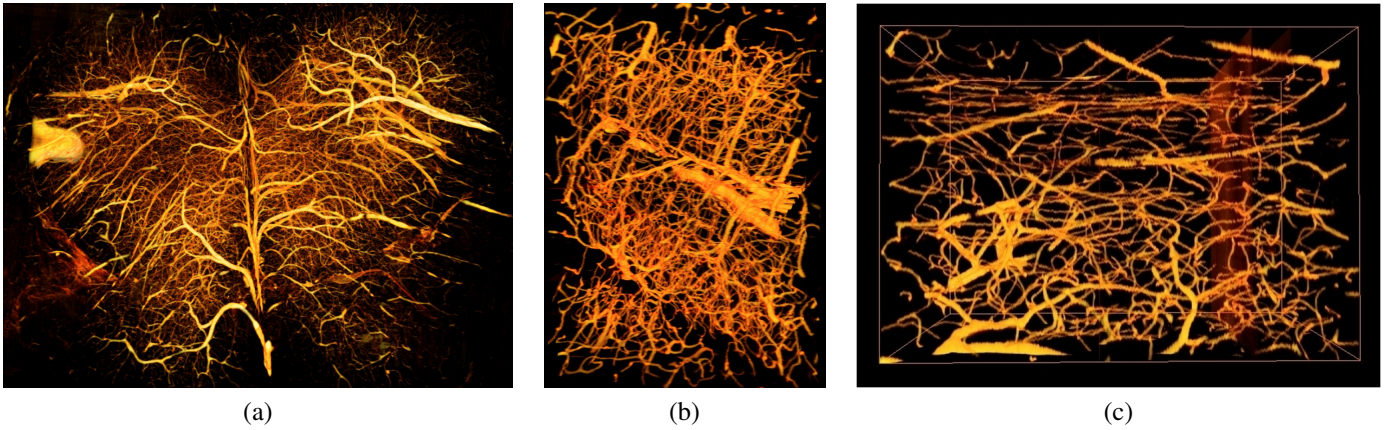Given a three-dimensional data set representing a tissue sample, we create a graph describing the structure of the

Fig. 5. Maximum intensity projection of the mouse spinal cord (left, 1500 sections) and close-up views of a capillary network in the cerebellum (center, 512x512x512) and the neocortex (right, 512x512x512).
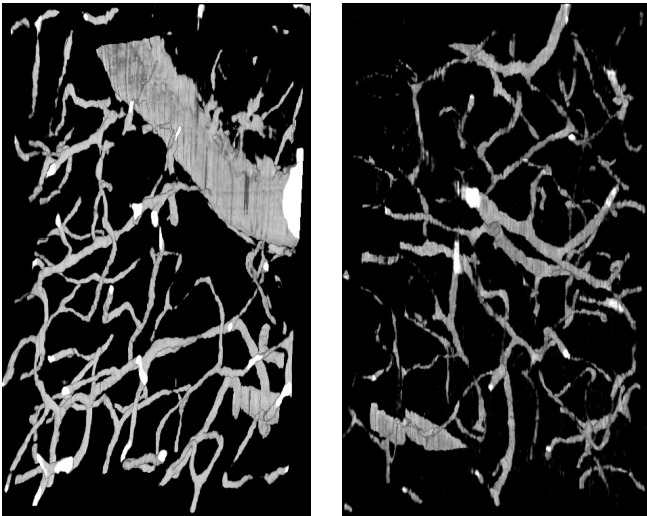

Fig. 4. Cropped isosurfaces from India ink (left) and Nissl (right). Because of noise and low contrast, the network isosurfaces contain frequent gaps, making it difficult to accurately reconstruct network connectivity and topology.

embedded filament network. We do this by finding the internal medial axis of each filament, including points where the filaments are connected.

A. Previous Work

There are several segmentation tools available for tracking macroscopic vascular structures, such as those found in MRI and CT data sets. An overview of these methods is presented by Kirbas et al. [13]. These techniques often rely on centerline extraction from an isosurface [14], region growing [15], or template-matching [12]. In addition, filtering techniques can be used to enhance the quality of segmentation for linear and curvilinear objects [16]. As mentioned previously (Section II-C), the isosurface can be damaged due to noise while features in the surrounding tissue can produce over-segmentation when using thresholding. Although region growing methods are effective for finding the filament surface, they generally require some initial segmentation in order to converge to a meaningful result. Although we have found template matching methods to be robust in the presence of noise, these techniques involve

testing various template sizes and orientations with each voxel in a data set. This is computationally expensive since the template must be tested in several positions, orientations, and sizes.

B. Segmentation

Vector tracking methods [17], [18], [19], [20] rely on template matching local to the region around a filament. The degree to which the filament cross section matches the provided template is used to estimate the filament trajectory. This estimated trajectory is then used to reduce the number of template sizes and orientations tested.

Given a point that lies on a filament, vector tracking algorithms predict the trajectory of the filament by sampling a region around this initial point. We then step along the filament in the direction of the estimated trajectory, thereby traversing the filament axis. Vector tracking generally uses three-dimensional template matching and is therefore robust in the presence of noise and broken filaments. Unlike standard template matching, we use a heuristic to limit sampling to a region near the filament. Since microvasculature occupies a very small volume of the data set (Section V), the number of samples required is greatly reduced. In this section, we describe our vector tracking methods, while further details can be found in our previous work [21].

1) Heuristics: Given a point p_i on the filament, we predict the next point p_{i+1} along the filament axis by sampling the region around p_i using a template. We look for the optimal transformation matrix T that minimizes the heuristic function

$$h(T) = \int_x \int_y \int_z |\Phi(T\mathbf{x}) - \gamma(\mathbf{x})| d\mathbf{x} \qquad (1)$$

where Φ is the volumetric data set, γ is a template function, and the point $\mathbf{x} = [x, y, z]$ is a point on the template. The matrix T is composed of affine components that describe the position, orientation, and size of the template:

$$T = Tr \times R \times S \qquad (2)$$

In this equation, Tr, R, and S are affine transformations respectively representing a sampled position, orientation, and scale of the template γ.

2) Tracking: We find the minimum value of the heuristic function $h(T)$ by sampling a discrete set of transformations. We construct Tr based on the initial position p_i:

$$Tr = \begin{pmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad (3)$$

We then construct a set of rotation matrices $\mathbf{R} = [R_0, R_1...R_N]$ that orient the template along an associated set of sample directions $\mathbf{r} = [r_0, r_1...r_N]$. We use a template that is rotationally invariant along the direction vector r_i. Therefore, the associated orientation matrix R_i can be composed using any two other vectors orthogonal to r_i. Likewise, we construct a series of scale matrices $S_i = s_i I$ where $s_i = [s_0, s_1...s_M]$ and I is the identity matrix. By sampling combinations of R and S, we select the samples that minimize the cost function $h(T)$. The position of the template is updated by taking a step along the estimated filament trajectory based on the selected orientation vector r_i, which produced the minimum value of h.

We use a volumetric cylinder as a template since it is both rotationally invariant along r_i and accurately represents the structure of most capillaries. The orientation and size of the template is adjusted as it is moved down each filament, in order to provide an accurate match (Fig. 6). Although the template size can be used as an estimate of the filament radius, it is highly dependent on how well defined the surface is relative to the background. As each step along a filament is taken, the current position p_i is tested against previously traced filaments in order to detect intersections. Further details on efficient filament tracking methods can be found in Al-Kofahi et al. [18] as well as our previous work on hardware-accelerated techniques [22].
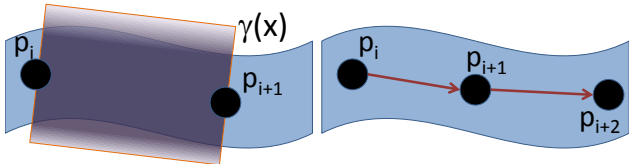

Fig. 6.   (left) We track the filament axis by matching a template function γ to the filament. (right) After computing the optimal template orientation (Eq. 1), we take successive steps along the filament axis.

3) Seed Points: Filament tracking requires us to create initial seed points on each filament from which to initiate tracking. We place seed points using the method proposed by Al-Kofahi et al. [18]. We project a region onto a two-dimensional plane using a maximum intensity projection for Nissl or a minimum intensity projection for India ink. Seed points are then placed on the plane using a conservative threshold and projected back into the three-dimensional region. Note that we are not concerned with over-seeding since branch detection will remove excess seed points.

IV. Refinement

Based on the filament skeleton found using vector tracking, we construct a graph G representing the structure of the filament network. Nodes connecting two edges in G represent samples along a single capillary. Branch points within the capillary network are represented by nodes in G with more than two edges (Fig. 7a).

Filament radius can then be estimated based on the optimal size of the template estimated during each tracking step, or by segmenting the filament cross-section using methods such as active contours [3]. When basing the radius estimate on the template size, we are limited to the discrete number of sample sizes used during tracking. In addition, these radii are dependent on the estimated trajectory of the filament in both cases. Therefore, small errors in filament trajectory can result in a cross-section that is not orthogonal to the true filament trajectory. Another option is to evolve a level set surface outward from the skeleton [23], however both active contours and level sets require several parameters to optimize fitting the resulting curves or surfaces.

Our approach instead relies on the network isosurface. Since our sampling resolution is sufficient to resolve the smallest vessels, there is little fear of misclassifying large vascular segments due to undersampling. As mentioned previously, however, the network isosurface contains structural flaws that make it difficult to determine network topology and connectivity. However, the isosurface does contain useful information about the surface structure and diameter of each filament in the network. In this section we discuss how we refine the network with morphological information from the microvascular isosurface, providing radius information for anatomical studies.

Given our original data set represented as the scalar volume function Φ, we manually select an isovalue that most accurately represents the microvascular surface Γ embedded in Φ. When computing an isosurface from image data, noise and artifacts cause misclassifications of the volume in one of two ways:

- (A) External regions are incorrectly classified as interior regions (false positives). This results in over-segmentation, causing erroneous surfaces to be created in the tissue surrounding the filament network.
- (B) Interior regions are incorrectly classified as exterior regions (false negatives). These errors result in unusually thin filaments or gaps in the network.

We compute the capillary radii for our vascular model by using mutual information between the extracted skeleton (Sec. III) and the network isosurface. By mapping isosurface information onto the network skeleton, we can find many of the erroneous cases produced by misclassifications and edit them out of our final model.

A. Mapping

We first create a mapping between the network skeleton and the data set. By overlaying the graph G, representing the network skeleton, with the scalar volume function Φ, we create a direct mapping $G \Rightarrow \Phi$ where any point on a node or edge of G represents a three-dimensional position within the data set Φ (Fig. 7b). We then construct an implicit signed distance function F_{sdf} based on the data set such that $\Phi \Rightarrow F_{sdf}$ and

therefore $G \Rightarrow F_{sdf}$. We use the standard definition of a signed distance function:

$$F_{sdf} = \begin{cases} d_\Phi(x, \Gamma) & \text{if } x \text{ is outside } \Gamma \text{ and} \\ -d_\Phi(x, \Gamma) & \text{if } x \text{ is inside } \Gamma. \end{cases} \quad (4)$$

where x is a point on G and $d_\Phi(x, \Gamma)$ is the shortest distance between x and the isosurface Γ.
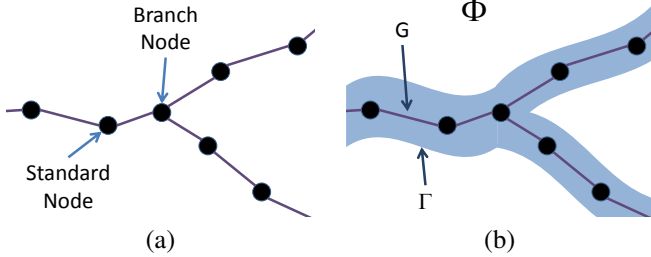
Fig. 7. (left) The traced graph G contains standard and branch points indicating positions on the filament axis. (right) We create a mapping from G to Φ by overlaying the traced graph onto the volumetric function.

Note that, if the function Φ is noise-free and all points in the graph G lie exactly on the filament axis, the radius r at any point x on G can be found by looking up the associated point in F_{sdf}:

$$r = -F_{sdf}(x) \quad (5)$$

We will first discuss efficient methods for creating the signed distance function F_{sdf} and then describe how G is refined in order to find the radius of each point in the network.

B. Computing a Distance Field

We compute an implicit signed distance function by solving the Eikonal equation $|\nabla F| = 1$ on a discrete grid. There are several efficient methods available, including Fast Sweeping [24] and Fast Marching [25]. Fast Sweeping provides an $O(n)$ solution, but requires that every voxel be evaluated. Fast Marching can be done in $O(n \log n)$ time but allows us to march outward from a surface and stop computation when necessary. We note that it is only necessary to solve the negative, or internal, portion of F_{sdf} since the radius information exists inside the surface Γ specified by the zero level-set of F_{sdf}. Since the volume inside Γ is usually much smaller than the volume of the data set (see Sect. V), we use a single pass of Fast Marching to evaluate the distance field only inside the network isosurface.

We first initialize F_{sdf}. Grid points next to the surface Γ are initialized with the distance from the surface while all other grid points are set to some large positive value. We then march inward computing the distance function for all values inside Γ (Fig 8). A detailed description is given by Osher and Fedkiw [26].

C. Radius Computation

We create an initial estimate of the filament radius by resampling F_{sdf} at points that lie on G. By limiting sampling to points on or near nodes and edges in G, we greatly reduce noise due to misclassifications of type A (false positives). This
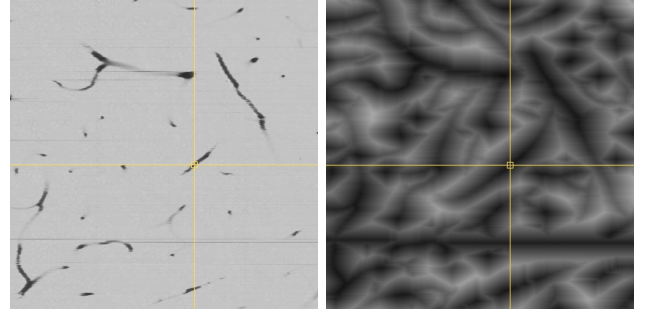
Fig. 8. Orthogonal section from the cortical data set (left) and equivalent section from the function F_{sdf} (right). For clarity, F_{sdf} has been fully evaluated and scaled to $[0...255]$.

is because false positives, by definition, occur outside of the network while the distance values in F_{sdf} are always based on the closest point on Γ (Fig. 9a).

Misclassifications of type B (false negatives) give erroneous results by causing interior regions to be labeled as exterior. This causes breaks in the filament or makes the isosurface unnaturally narrow along a portion of a capillary (Fig. 9b). When these regions are sampled, r is either very small or negative. These misclassifications are resolved by propagating known values from neighboring nodes in G.

Since the graph G is based on a heuristic estimate of the network embedded in Φ, it is unlikely that all points on G lie exactly on the filament axis. Therefore, sampling exactly on G leads to under-estimating the capillary radius since the corresponding point in F_{sdf} may lie closer to the filament surface. Note that the actual filament axis lies on the point inside the filament that is furthest from the filament isosurface. In F_{sdf}, this surface is represented by the zero level-set. Therefore, when sampling F_{sdf} using a point $\mathbf{x}$ on G, we instead sample a region in F_{sdf} that lies within ϵ of $\mathbf{x}$.

We set the radius in G equal to the maximum value of r found within ϵ of $\mathbf{x}$. We set ϵ to $5\mu m$, which is close to the known radius of microvessels. Since microvessels are sparsely arranged relative to their diameter, it is unlikely for sampling to cross into other capillaries using this value for ϵ.
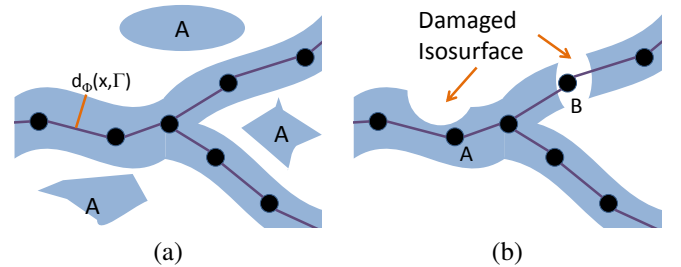
Fig. 9. We use the graph G to interpolate through errors in the isosurface. (left) False positives cause erroneous regions to be formed outside the isosurface. These regions are avoided by only sampling F_{sdf} near the graph. (right) Points in G with very small (A) or negative (B) values of F_{sdf} indicate isosurface damage. The radius for these points are computed by propagating valid information about F_{sdf} from neighboring points.

V. Results and Discussion

We first imaged an entire mouse brain stained with India ink at a resolution of $0.6\mu m \times 0.7\mu m \times 1.0\mu m$. Images were processed to remove lighting artifacts (Sect. II) and stored as 512x512x512 raw image files. We then specified several brain regions of interest for analysis. We particularly focused on the cerebellar cortex and spinal cord (Fig. 5a and b). We also imaged a large region of mouse cortex stained with Nissl, processing and storing it in the same manner (Fig. 5c).

A. Computational Anatomy

We used vector tracking (Sect. III) to perform automated segmentation of the network. We then collected several anatomical statistics on the structure of the network, such as the number of segments and branch points per unit volume of tissue (Table I). We also focus on obtaining local anatomical information such as the direction of travel of capillaries (Fig. 10). Statistics and anatomical information are of interest to anatomists since very little information is available on the three-dimensional structure of microvasculature. Although we are unaware of any studies performed at this scale and resolution, our results are similar to morphometric studies that have been performed using samples of $SR\mu CT$-collected data for the mouse cortex [6] and confocal microscopy images of human brain [5].
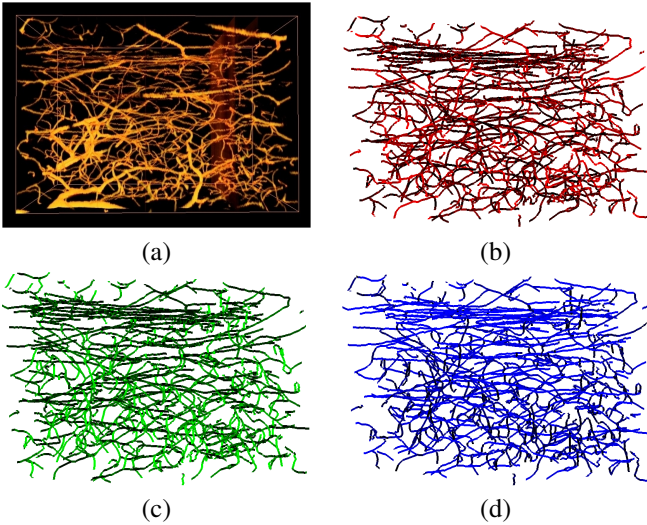

Fig. 10. Vascular trajectory in the neocortex. The tracked vascular network from a single block (Fig. 11) is shown with filaments colored based on their direction of travel (red = sagitally, green = horizontally, blue = coronally). The highly oriented blue filaments are running coronally through white matter.

B. Modeling

Using the structural information available from tracked filaments, we refine the network based on the filament isosurface. This involves manually selecting an isovalue that accurately describes the vascular surface (Fig. 11a). Although the surface is noisy and contains misclassifications, we map the isosurface information onto the network skeleton. Using our refinement techniques (Sec. IV), we create a structural model of the vascular network that describes filament radius as well as

position and connectivity. We then use the refined model to compute the microvascular volume and surface per unit volume of tissue (Table I).
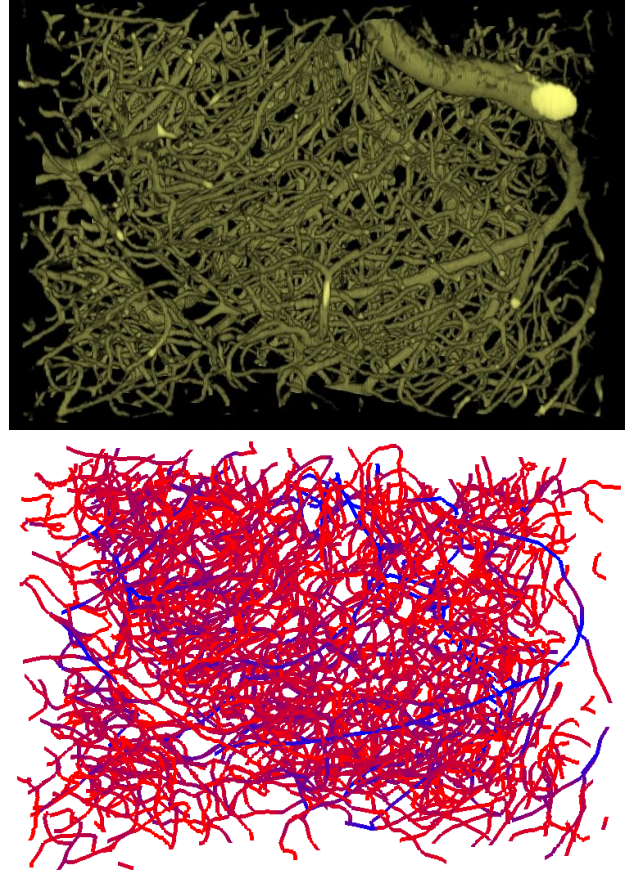

Fig. 11. Cerebellar cortex. Vascular isosurface (left) used for refining radius measurements. Traced vascular network (right) colored based on radius (red $< 2\mu m$, blue $> 5\mu m$).

C. Evaluation

We have limited means for comparison and evaluation due to a lack of similar high-resolution data sets. We instead focus on the evaluation of vector tracking as a means of segmentation. Although our algorithm performs well under a visual evaluation, the complexity of large microscopy data sets makes errors difficult to find visually.

Because of the nature of microvascular networks, all segments must be connected in order to allow blood flow. We consider any traced capillaries that terminate without branching to be errors. These terminations, accounting for $\approx 3.2\%$ of the network, could be due either to errors in tracking or inadequate staining. However, changes in brain microvasculature are known to occur and some portion of these may be developing or degenerating microvessels.

D. Discussion

In this paper we describe a framework that provides high-resolution information about the structure of microvasculature spanning large three-dimensional regions of tissue. Our analysis focuses on mouse brain microvasculature, however similar

TABLE I

Region	Segments	Length (mm)	Branches	Surface (mm^2)	Volume (mm^3)	Volume(% of total)
Neocortex	11459.7	758.5	9100.0	10.40	0.0140	1.4%
Cerebellum	34911.3	1676.4	19034.4	20.0	0.0252	2.5%
Spinal Cord	36791.7	1927.6	26449.1	22.2	0.0236	2.4%

staining methods have been used on other tissue samples from other species.

Further work can be done by extending these models to pathological tissue. This would allow high-resolution analysis of blood-flow patterns in different disease models. Although frameworks for fluid simulation in microvessels have been developed [27], these methods have not been extended to larger scales due to a lack of high-resolution structural information. Finally, very little research has been done in the area of simulating cellular and vascular relationships, which can be observed in Nissl-stained tissue. Modulation is known to occur between neurons and microvasculature. Simulation of these relationships would require detailed cellular morphology as well as microvascular anatomy.

REFERENCES

[1] Berislav Zlokovic, "The blood-brain barrier in health and chronic neurodegenerative disorders," *Neuron*, vol. 57, pp. 178–201, 2008.

[2] C. I. Moore and R. Cao, "The hemo-neural hypothesis: On the role of blood flow in information processing," *Journal of Neurophysiology*, vol. 99, pp. 2035–2047, 2007.

[3] Jack Lee, Patricia Beighley, Erik Ritman, and Nicholas Smith, "Automatic segmentation of 3d micro-ct coronary vascular images," *Medical image analysis*, vol. 11, pp. 630–647, Dec. 2007.

[4] Daniel C. Morris, Kenneth Davies, Zhenggang Zhang, and Michael Chopp, "Measurement of cerebral microvessel diameters after embolic stroke in rat using quantitative laser scanning confocal microscopy," *Brain Research*, vol. 876, pp. 31–36, Sept. 2000.

[5] F. Lauwers, F. Cassot, V. Lauwers-Cances, P. Puwanarajah, and H. Duvernoy, "Morphometry of the human cerebral cortex microcirculation: General characteristics and space-related profiles," *Neuroimage*, vol. 39, pp. 936–948, 2008.

[6] Stefan Heinzer, Thomas Krucker, Marco Stampanoni, Rafael Abela, Eric P. Meyer, Alexandra Schuler, Philipp Schneider, and Ralph Muller, "Hierarchical microimaging for multiscale analysis of large vascular networks," *NeuroImage*, vol. 32, pp. 626–636, Aug. 2006.

[7] J. B. Pauley, Ed., *Handbook of Biological Confocal Microscopy*, Plenum Press, New York, 1995.

[8] David Mayerich, Louise C. Abbott, and Bruce H. McCormick, "Knifeedge scanning microscopy for imaging and reconstruction of three-dimensional anatomical structures of the mouse brain," *Journal of Microscopy*, vol. 231, pp. 134–143, June 2008.

[9] Jaerock Kwon, David Mayerich, Yoonsuck Choe, and Bruce H. McCormick, "Automated lateral sectioning for knife-edge scanning microscopy," in *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, 2008.

[10] Louise C. Abbott and Constantino Sotelo, "Ultrastructural analysis of catecholaminergic innervation in weaver and normal mouse cerebellar cortices," *The Journal of Comparative Neurology*, vol. 426, pp. 316–329, 2000.

[11] D. Mayerich, B. H. McCormick, and J. Keyser, "Noise and artifact removal in knife-edge scanning microscopy," *Proceedings of the 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 556–559, 2007.

[12] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Prentice Hall, 2nd edition, 2002.

[13] C. Kirbas and F. Quek, "A review of vessel extraction techniques and algorithms," *ACM Computing Surveys*, vol. 36, pp. 81–121, 2004.

[14] T. Tozaki, Y. Kawata, N. Niki, H. Ohmatsu, K. Eguchi, and N. Moriyama, "Three-dimensional analysis of lung areas using thin slice ct images," in *Proceedings of SPIE*, 1996, vol. 2709, pp. 1–11.

[15] D. Nain, A. Yezzi, and G. Turk, "Vessel segmentation using a shape driven flow," *Medical Imaging Computing and Computer-Assisted Intervention*, 2004.

[16] Y. Sato, S. Nakajima, N. Shiraga, H. Atsumi, S. Yoshida, T. Koller, G. Gerig, and R. Kikinis, "Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images," *Medical Image Analysis*, vol. 2, pp. 143–168, 1998.

[17] Ali Can, Hong Shen, James N. Turner, Howard L. Tanenbaum, and Badrinath Roysam, "Rapid automated tracing and feature extraction from retinal fundus images using direct exploratory algorithms," *IEEE Transactions on Information Technology in Biomedicine*, vol. 3, pp. 125–138, 1999.

[18] Khalid Al-Kofahi, Sharie Lasek, Donald Szarowski, Christopther Pace, George Nagy, James Turner, and Badrinath Roysam, "Rapid automated three-dimensional tracing of neurons from confocal image stacks," *IEEE Transactions on Information Technology in Biomedicine*, vol. 6, pp. 171–186, 2002.

[19] Karl Krissian, Gregorie Malandain, Nicholas Ayache, Rgis Vaillant, and Yves Trousset, "Model-based detection of tubular structures in 3d images," *Computer Vision and Image Understanding*, vol. 80, pp. 130–171, 2000.

[20] S.. Worz and K.. Rohr, "Segmentation and quantification of human vessels using a 3-d cylindrical intensity model," *Image Processing, IEEE Transactions on*, vol. 16, pp. 1994–2004, 2007.

[21] David Mayerich and John Keyser, "Filament tracking and encoding for complex biological networks," *Proceedings of the ACM Solid and Physical Modeling Symposium*, pp. 353–358, 2008.

[22] David M. Mayerich, Zeki Melek, and John Keyser, *Fast Filament Tracking Using Graphics Hardware*, Technical report (submitted for review), Texas A&M University, 2007.

[23] L.M. Lorigo, O.D. Faugeras, W.E. Grimson, R. Keriven, R. Kikinis, A. Nabavi, and C.F. Westin, "Curves: curve evolution for vessel segmentation," *Medical Image Analysis*, vol. 5, pp. 195–206, 2001.

[24] H. Zhao, "A fast sweeping method for eikonal equations," *Mathematics of Computation*, vol. 74, pp. 603–627, 2004.

[25] J. A. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, Cambridge University Press, 1999.

[26] Stanley J. Osher and Ronald P. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer, 2002.

[27] A. R. Pries, T. W. Secomb, P. Gaehtgens, and J. F. Gross, "Blood flow in microvascular networks. experiments and simulation," *Circulation Research*, vol. 67, pp. 826–834, 1990.