

Electronic Thesis and Dissertation Repository

1-31-2019 12:00 PM

Vessel Tree Reconstruction with Divergence Prior

Zhongwen Zhang

The University of Western Ontario

Supervisor

Yuri Boykov

The University of Western Ontario

Graduate Program in Computer Science

A thesis submitted in partial fulfillment of the requirements for the degree in Master of Science

© Zhongwen Zhang 2019

Follow this and additional works at: <https://ir.lib.uwo.ca/etd>



Part of the [Artificial Intelligence and Robotics Commons](#), and the [Other Computer Sciences Commons](#)

Recommended Citation

Zhang, Zhongwen, "Vessel Tree Reconstruction with Divergence Prior" (2019). *Electronic Thesis and Dissertation Repository*. 6008.

<https://ir.lib.uwo.ca/etd/6008>

This Dissertation/Thesis is brought to you for free and open access by Scholarship@Western. It has been accepted for inclusion in Electronic Thesis and Dissertation Repository by an authorized administrator of Scholarship@Western. For more information, please contact wlsadmin@uwo.ca.

Abstract

Accurate structure analysis of high-resolution 3D biomedical images of vessels is a challenging issue and in demand for medical diagnosis nowadays. Previous curvature regularization based methods [10, 32] give promising results. However, their mathematical models are not designed for bifurcations and generate significant artifacts in such areas. To address the issue, we propose a new geometric regularization principle for reconstructing vector fields based on prior knowledge about their divergence. In our work, we focus on vector fields modeling blood flow pattern that should be divergent in arteries and convergent in veins. We show that this previously ignored regularization constraint can significantly improve the quality of vessel tree reconstruction particularly around bifurcations where non-zero divergence is concentrated. Our divergence prior is critical for resolving (binary) sign ambiguity in flow orientations produced by standard vessel filters, *e.g.* Frangi. Our vessel tree centerline reconstruction combines divergence constraints with robust curvature regularization. Our unsupervised method can reconstruct complete vessel trees with near-capillary details on both synthetic and real 3D volumes. Also, our method reduces angular reconstruction errors at bifurcations by a factor of two.

Keywords: Computer vision, divergence constraint, curvature regularization, vessel tree reconstruction

Acknowledgements

First of all, I would like to thank my supervisor, Professor Yuri Boykov, for his support. His enthusiasm for the research and pursuit of perfection motivated me a lot. He always patiently gives me advice and guidance when I am lost during the research. Thanks to him, I now understood how graph cuts and regularization-based methods work. I am also very grateful to him for accepting me as a PhD student so that I can continue learning from him and gradually achieve my dream.

I would also like to thank Maria Drangova (Robarts Research Institute, London, Ontario) for providing high-resolution microscopy CT volumes with cardiac vessels. I used TRW-s code by Vladimir Kolmogorov (ITS, Vienna, Austria) for efficient minimization of binary orientation variables. Marc Moreno Maza (Western University, London, Ontario) shared his expertise in high-performance computing allowing our efficient implementation of trust region. This research would not be possible without support by the Canadian government including Discovery and RTI programs by NSERC.

I really appreciate the help from other members of our Computer Vision Group. Special thanks to Dmitrii Marin and Egor Chesakov, who helped me a lot during my graduate study and their contributions to this work.

Last but not least, I would like to thank my parents who always give me support and love to choose whatever I like. Also, special thanks to my dear friends Hao Jiang, Jingyi Ren and Yifang Liu, who always stand by me and give me big support to keep going on.

Contents

Acknowledgements	i
Abstract	ii
List of Figures	v
1 Introduction	1
1.1 Biomedical Vascular Imaging	1
1.2 Vessel Detection Overview	3
1.2.1 Vessel Representation: Centerline or Segment	4
1.2.2 Towards Whole Tree Centerline	4
1.3 Contributions	6
1.4 Thesis Structure	8
2 Background on Vessel-tree Reconstruction	9
2.1 Vessel Enhancement Filter	9
2.2 Curvature Regularization for Centerline	11
2.3 Minimum Spanning Tree with Arc Length	13
3 Oriented Curvature and Divergence Constraints	16
3.1 Oriented Curvature Constraint	16
3.2 Curvature and Orientation Ambiguity	17
3.3 Divergence Constraint	19
3.4 Oriented Centerline Estimation	22
3.4.1 Binary Optimization via TRW-s	24
3.4.2 Trust Region Optimization for Tangent Estimation	24
4 Evaluation	29
4.1 Synthetic Vessel Volume	29
4.2 Real Vessel Data	34

5 Conclusion and Future Work	41
Bibliography	42
Curriculum Vitae	46

List of Figures

1.1	Visualization (Maximum Intensity Projection) of the raw volumetric data obtained from a mouse heart by <i>micro computer tomography</i> (micro-CT). The data is provided by Maria Drangova from the Robarts Research Institute in London, Canada.	2
1.2	Vessel centerline extracted by different methods. (a) minimum spanning tree (MST) built on the result of minimizing (2.12) [10]. (b) MST built on the output of skeleton-based algorithm [4].	5
1.3	[Vessel-tree divergence] Vessels are the blood flow <i>pathlines</i> and could be assigned orientations. Green arrows represent vessels with blood flow orientation. To estimate orientations, we penalize negative (or positive) “vessel divergence”, which we define as the divergence of oriented <u>unit</u> tangents of vessels/pathlines. Such <i>unit tangent flow</i> divergence is positive (red) or negative (blue) at bifurcations, see (a-c). Note that standard curvature models [38, 32] and oriented curvature models given in (3.2) either can not distinguish (b) from (a) and (c) or may even prefer (b) depending on specific combinations of bifurcation angles. For example, compare vessel direction disambiguation based on curvature and the divergence prior in Figure 3.4 (a) and (b).	7
2.1	Centerline definition and curvature model of [38]. (a) illustrates an example of noisy measurements along with the centerline C . (b) illustrates the curvature model. Given two points p and q on the red curve (centerline) and two tangents l_p and l_q at these points, the integrals of curvature are approximated by ((2.9)-(2.11))	12
2.2	Triangle artifacts at bifurcation. Optimization of energy (2.12) ignoring tangent orientations often leads to a strong local minima as in (a) and (b). The line segments are the estimated tangents of the centerline.	13
2.3	Illustration of the arc. The red curve is the arc from point P to Q , which is tangential with l_p at point P . Point O is the center of the circle to which the arc belongs. θ is the osculation angle.	14

3.1	Illustrative examples of three interacting tangents with unoriented curvature (a) as in energy (2.12) and two alternative oriented configurations (b) and (c) with oriented curvature as in energy (3.1). The green line denotes pairwise interaction with low curvature value. Note, that unoriented curvature (2.9–2.11) always chooses the smallest angle for calculation. The red line shows “inactive” pairwise interaction, <i>i.e.</i> interaction where curvature in (3.2) reaches the high saturation threshold.	17
3.2	Elimination of triangle artifact at bifurcation. New curvature term (3.2) takes into account tangent orientation resolving the artifact.	18
3.3	Robustness of curvature (3.2). The pairs of tangent vectors that has angle greater than $\text{acos } \tau$ are not considered as belonging to the same vessel. A constant penalty is assigned to such pairs. This “turns off” smoothness enforcement at bifurcations.	19
3.4	Disambiguating flow directions in Frangi output [16]. Both examples use fixed (unoriented) vessel tangents $\{l_p\}$ produced by the filter and compute (oriented) vectors $\bar{l}_p = x_p l_p$ (3.3) by optimizing binary sign variables $\{x_p\}$ using energies (3.4) in (a) and (3.7) in (b). The circles indicate divergent (red) or convergent (blue) bifurcations similarly to the diagrams in Figure 1.3. The extra divergence constraint in (3.6) enforces consistent flow pattern (b).	20
3.5	Centerline estimation for the data in Figure 3.4. Instead of showing tangent orientations estimated at the first iteration as in Figure 3.4, we now show the final result at convergence for minimizing energy (3.1) in (a) and energy (3.6) in (b). Blue circle shows bifurcation reconstruction artifacts due to wrong estimation of vessel orientations in Figure 3.4(a).	21
3.6	Divergence of a sparse vector field $\{\bar{l}_p p \in \Omega\}$. Assuming that the corresponding “extrapolated” dense vector field is constant inside Voronoi cells (a), it is easy to estimate (non-zero) divergence $\nabla \bar{l}_{pq}$ (3.5) concentrated in a narrow region f_{pq}^ϵ around each facet (b) using the divergence theorem.	22
3.7	Representative example of decrease in energy (3.6) for block-coordinate descent iterating optimization of (3.7) and (3.8). For initialization, we use raw undirected tangents $\{l_p\}$ generated by Frangi filter [16]. Then, we iteratively reestimate binary sign variables $\{x_p\}$ and unoriented tangents $\{l_p\}$	23
3.8	Illustration of the parameterization of point p . \tilde{p} is the noisy observation of the point p . Since p is assumed to be the closest point on tangent l_p from \tilde{p} , p is calculated by the orthogonal projection of \tilde{p} onto the line l_p	25

4.1	An example of one volume synthetic data. The white lines inside vessels denote ground truth of centerline.	30
4.2	Experimental flow diagram. We mainly compares regularization-based methods.	31
4.3	Comparison of our method (OriAbsCurv and OriQuaCurv) with the unoriented quadratic curvature (QuaCurv) [32], non-maximum suppression (NMS) based on Frangi outputs, SegmentTubes (Aylward <i>et al.</i> [3]) and medial axis extraction (Bouix <i>et al.</i> [4]) at three different noise levels. The four letters on yellow dots denote different seed point lists. <i>a</i> : using root and all leaf points; <i>b</i> : using 50% of the mixture of all bifurcation and leaf points and root; <i>c</i> : using middle points of all branch segments; <i>d</i> : using all bifurcation and leaf points and root.	32
4.4	Comparison only at bifurcation point.	33
4.5	Illustration of the angle at bifurcation point. (a) shows the ground truth where the flow directions are well-defined. The angle θ is used as the ground truth angle of bifurcation. In (b), the blue vectors represent the oriented tangents estimated by minimizing the energy (3.6). The connections <i>BO</i> , <i>BP</i> and <i>BQ</i> are given by MST. The absolute difference of angle θ and ϕ is the angle error.	35
4.6	Angle error comparison.	36
4.7	Examples of the results around bifurcations with regularization methods. White line is the ground truth tree. A tree extracted from NMS output directly (without regularization) is shown in the first row. Solution of (2.12) [32] is shown in second row. Our model (3.6) is shown in third row. Our model (3.6) with absolute curvature is shown in the last row.	37
4.8	Visualization (MIP) of the volumetric data after applying Frangi filter.	38
4.9	Vessel tree reconstruction from real data in Figure 4.8 based on our method for estimating centerline tangents using prior knowledge about vessel divergence. The final tree structure is extracted by MST on K-nearest-neighbour (KNN) weighted graph with edge weights w_{pq} defined as the average <i>arc-length</i> between neighbors <i>p</i> and <i>q</i> for two circles containing <i>p</i> and <i>q</i> and tangential to either l_p or l_q . Three different colors (red, blue, yellow) denote three main branches.	39
4.10	Qualitative comparison results on real data. Red represents the result of [32], see (2.12). Blue curve is the centerline obtained based on our directed vessel estimation model (3.6) with divergence prior and absolute curvature regularization. The yellow circles highlight improvements at bifurcations due to correct estimation of the flow direction.	40

Chapter 1

Introduction

3D imaging plays more important role in medical diagnosis. Doctors usually make diagnostic decisions based on the analysis of organ structure obtained from Computed Tomography (CT), Magnetic Resonance Imaging (MRI), etc. Modern imaging techniques bring microscopic results and, hence, many advanced analysis methods arise as well. However, some tasks, like near-capillary vessel structure analysis, still remain challenging. Figure 1.1 shows an example of a mouse cardiac CT image resolving vessels at near-capillary level. As a consequence of the emergence of high resolution imaging techniques, new robust analysis algorithms are very required.

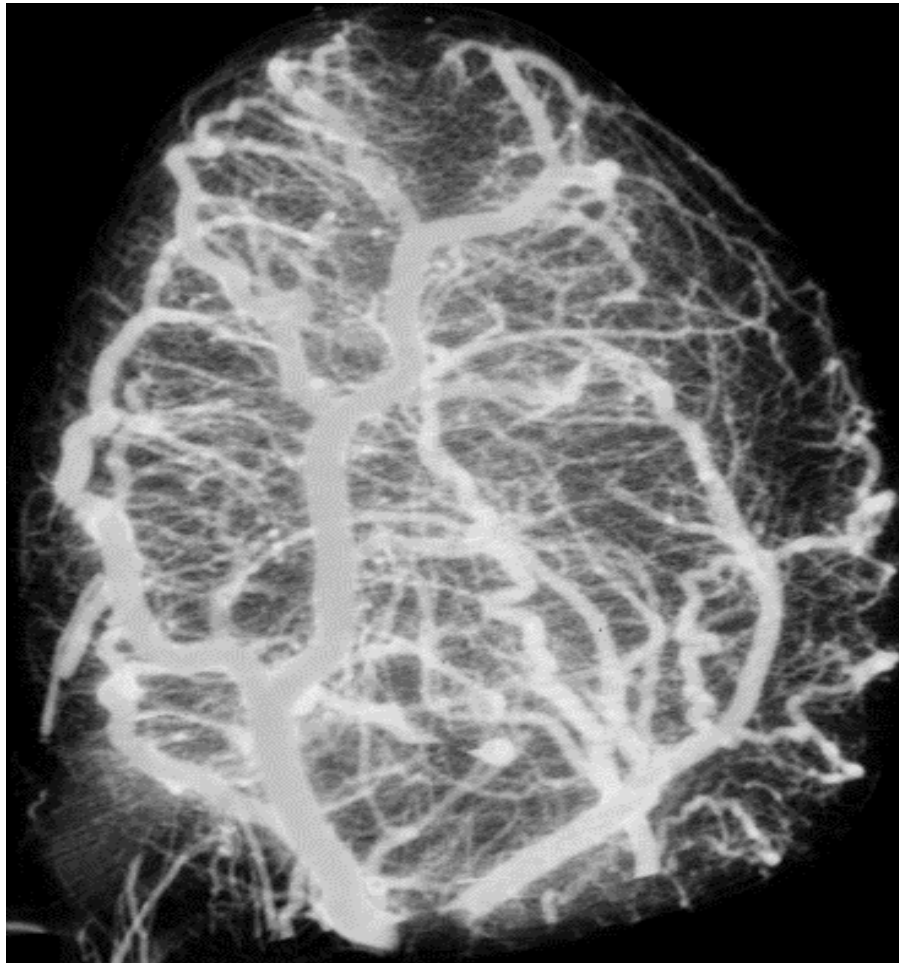
In Section 1.1, we introduce several techniques for biomedical vascular imaging. In Section 1.2, we give an overview of existing vessel detection algorithms and their limitations. Our motivation and contribution are described in Section 1.3. The outline of thesis is given in Section 1.4.

1.1 Biomedical Vascular Imaging

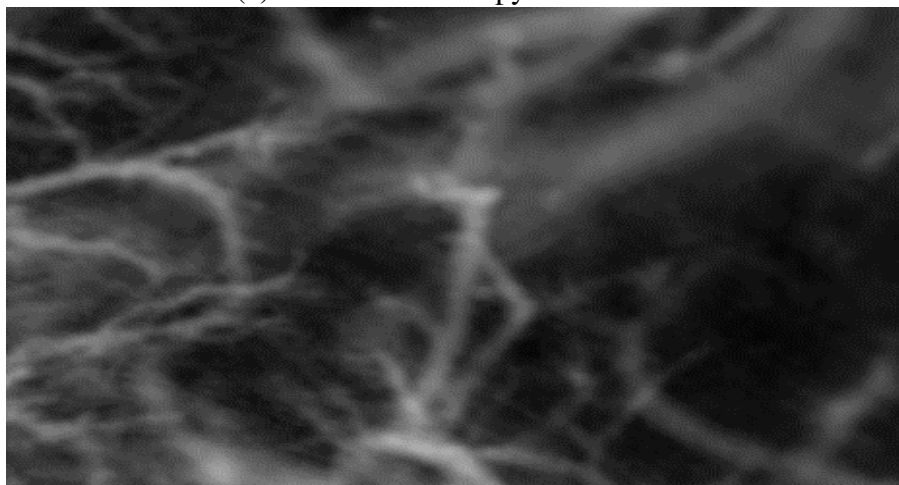
There are mainly three modalities for vascular imaging. First is X-ray which has been used for many years to image large vessel structure. X-ray is the oldest imaging technique and, however, has an obvious shortcoming that the 3D structures and organs overlap in 2D projection image. Due to the fast development of computation speed and algorithm researchers invented cross-sectional imaging which allows much clearer imaging of complicated structures inside a body.

Second, MRI uses strong magnetic fields and radio waves of a specific form to produce 3D images inside body. Advanced MRI techniques, such as Diffusion Tensor Imaging (DTI), can provide us additional information about the region of the interest inside a body.

Third, CT can generate 3D volume data from 2D X-ray cross-sectional images taken from different positions. Modern CT imaging techniques can produce fairly high-resolution im-



(a) cardiac microscopy CT volume



(b) zoom-in

Figure 1.1: Visualization (Maximum Intensity Projection) of the raw volumetric data obtained from a mouse heart by *micro computer tomography* (micro-CT). The data is provided by Maria Drangova from the Robarts Research Institute in London, Canada.

ages of certain parts of our body, like coronary vessels. Current Micro-computed Tomography (micro-CT) is able to provide ultra high resolution (voxel size $\leq 20\mu\text{m}^3$)

Medical diagnosis usually needs advanced analysis of the organ structure from medical images, such as vascular tree topology, angle between bifurcating branches and the diameter of vessels. However, existing methods, such as optical measurement, are extremely time-consuming and need a large amount of manual work [22]. Consequently, there is a need for an automatic and efficient framework for vessel detection and analysis. The method has to be robust with respect to the wide range of vessel scales and complicated topology of the vascular tree [10].

Generally, the newer micro-CT already has higher signal-to-noise ratios (SNR). However, people are using it to solve more challenging image analysis problems, such as analyzing blood vessels at near-capillary level. This analysis was particularly impossible before, since even the micro-CT has low SNR at near capillary level. As seen in the Figure 1.1(b), it is hard to distinguish the thin near-capillary vessels from the background by intensity alone.

1.2 Vessel Detection Overview

There is plenty of prior work on estimation of vessels in computer vision and biomedical imaging communities [34]. Typically, pixel-level detection of tubular structures is based on multi-scale eigen analysis of intensity Hessians developed by Frangi *et al.* [16], which will be reviewed in Section 2.1, and other research groups [14]. At any given point (pixel/voxel) such vessel enhancement filters output a *tubularness/vesselness measure* and estimates of the vessel's scale and orientation, which describes the flow direction up to a sign. While such local analysis of Hessians is very useful, simple thresholding of points with large-enough vesselness measure is often unreliable as a method for computing the vessel tree structure. While thresholding works well for detecting relatively large vessels, detection of smaller vessels is complicated by noise, partial voluming, and outliers (e.g. ring artifacts caused by an improperly calibrated CT-scan [20]). More importantly, standard tubular filters exhibit signal loss at vessel bifurcations as those do not look like tubes.

Regularization methods can address the vessel continuation problems due to noise, outliers, and signal loss at thinner parts and bifurcations. D. Marin *et al.*[32] and E. Chesakov [10] employ the curvature regularization for vessel centerline extraction on 3D images resolving near-capillary vessels. Based on the analysis of their drawbacks, we adapt the curvature regularizer for a robust version and propose a new regularization prior based on knowledge of the flow pattern divergence. Particularly, this new prior is critical for disambiguating flow directions, which provide important cues about the vessel tree structure. Followed subsections

outline existing regularization methods for vessel reconstruction and motivate our approach.

1.2.1 Vessel Representation: Centerline or Segment

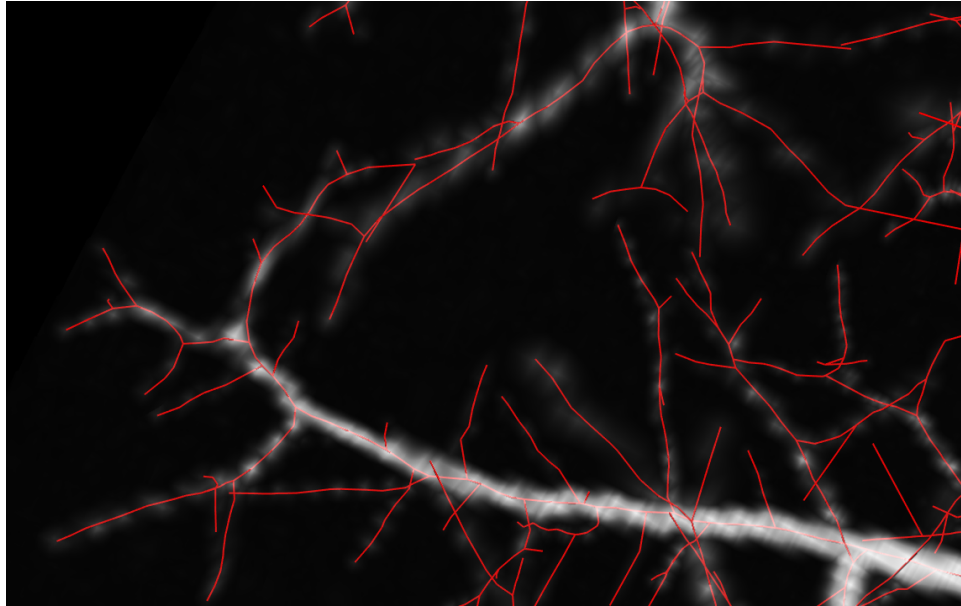
Two common approaches to representing vessels in reconstruction methods are volumetric binary mask and centerline. A volumetric mask is typical for techniques directly computing vessel segmentation, *i.e.* binary labeling of pixels/voxels. In contrast, centerline is a 1D abstraction of the vessel. But, if combined with information about vessel radii, it is easy to obtain a volumetric mask or segmentation from the vessel's centerline, e.g. using MAT [43]. Vice versa, centerline could be estimated from the vessel's binary mask using skeletonization algorithms. Bouix *et al.* [4] adapted a skeletonization algorithm for centerline extraction on tubular structures. The skeletonization algorithm exploits properties of the average outward flux of the gradient vector field of a Euclidean distance function from the boundary of the structure. However, this method relies on the quality of the segmentation output (or boundary detection output) and it only utilizes local information to estimate the medial curve, which results in very unsmoothed centerline, see Figure 1.2(b). Note that, in Figure 1.2, I showed results after applying minimum spanning tree which will be explained in Section 2.3.

In the context of regularization-based methods for vessel reconstruction, centerline representation offers significant advantages since powerful higher-order regularizers are easier to apply to 1D structures. For example, centerline's curvature can be regularized [32], while conceptually comparable regularization for vessel segmentation requires optimization of Gaussian or minimum curvature of the vessel's surface, with no known practical algorithms. In general, curvature remains a challenging regularization criteria for surfaces [41, 44, 19, 40, 37]. Alternatively, some vessel segmentation methods use simpler first-order regularizers producing minimal surfaces. While tractable, such regularizers impose a wrong prior for surfaces of thin structures due to their bias to compact blob shapes (a.k.a. shrinking bias).

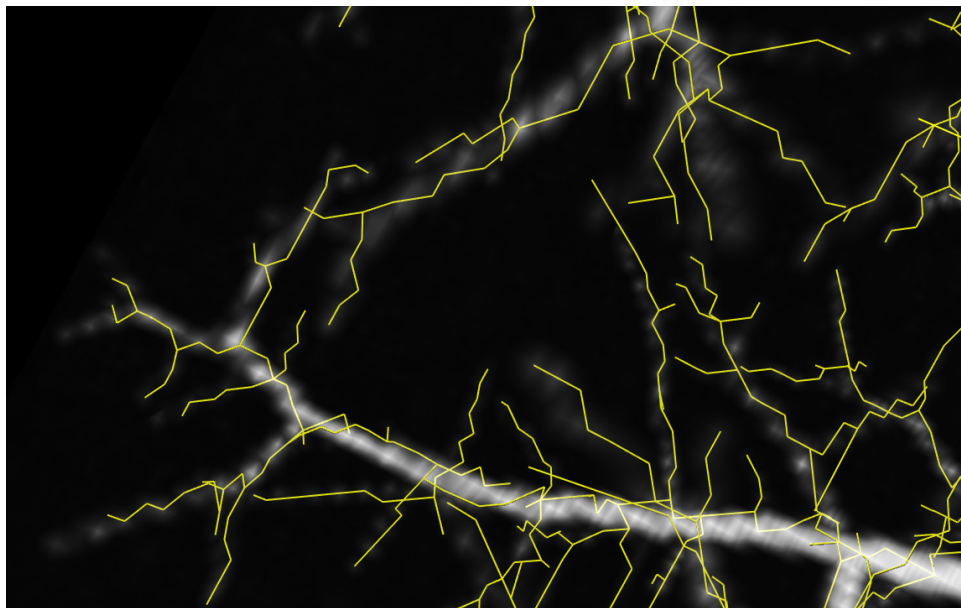
1.2.2 Towards Whole Tree Centerline

Many vessel reconstruction methods directly compute centerlines of different types that can be informally defined as simplified (e.g. regularized) 1D representation of the blood flow *pathlines*. For example, the A/B shortest path methods require a user to specify two end points of a vessel and apply the Dijkstra [12] to find an optimal pathline on a graph with edge weights based on vesselness measure.

Interactive A/B methods are not practical for large vessel tree reconstruction problems. While it is OK to ask a user to identify the tree *root*, manual identification of all the end points (*leaves*) is infeasible. There are *tracing* techniques [3] designed to trace vessel tree from a given



(a)



(b)

Figure 1.2: Vessel centerline extracted by different methods. (a) minimum spanning tree (MST) built on the result of minimizing (2.12) [10]. (b) MST built on the output of skeleton-based algorithm [4].

root based on vesselness measures and some local continuation heuristics. Our evaluations on synthetic data with ground truth show that local tracing methods do not work well for large trees with many thin vessels even if we use the ground truth to provide all tree leaves as extra seeds in addition to the root.

Our goal is unsupervised reconstruction of the whole vessel tree centerline. We optimize a global objective function for a field of centerline tangents. Such objectives can combine vesselness measures as unary potentials with different regularization constraints addressing centerline completion. Related prior work using centerline curvature regularization is reviewed in Section 2.2.

1.3 Contributions

This work addresses an important limitation of vessel tree reconstruction methods due to sign ambiguity in vessel orientation produced by local vesselness filters, e.g. Frangi. This orientation is described by the smallest eigenvector of the local intensity Hessian, but its sign is ambiguous. Thus, the actual flow directions are not known, even though they are an important reconstruction cue particularly at bifurcations. This binary direction ambiguity can be resolved only by looking at the global configuration of vessel orientations (tangents) allowing to determine a consistent flow pattern.

We propose a divergence prior for disambiguating the global flow pattern over the vessel tree, see Figure 1.3. This prior can be imposed as a regularization constraint for a vector field of oriented unit tangents for vessel pathlines. We penalize negative (or positive) divergence for such unit *tangent flow* to enforce a consistent flow pattern¹. The summary of our contributions:

- Prior knowledge about divergence is generally useful for vector field inference. We propose a way to evaluate divergence for sparsely sampled vector fields via pairwise potentials. This makes divergence constraints amenable to a wide range of optimization methods for discrete or continuous hidden variables.
- As an important application, we show that known divergence can disambiguate vessel directions produced by standard vessel filters, e.g. Frangi [16]. This requires estimation of binary “sign” variables. The constraint penalizing positive (or negative) divergence is non-submodular, but it is well optimized by TRW-s [25].
- To estimate vessel tree centerline, the divergence constraint can be combined with robust *oriented* curvature regularization for pathline tangents. Additional options include outliers/detection variables [32] and/or tree structure completion techniques, e.g. using Minimum Spanning Tree (MST).

¹This divergence constraint is specific to unit tangent flow. Note that divergence for consistent blood flow velocities is zero even at bifurcations assuming *incompressible* blood.

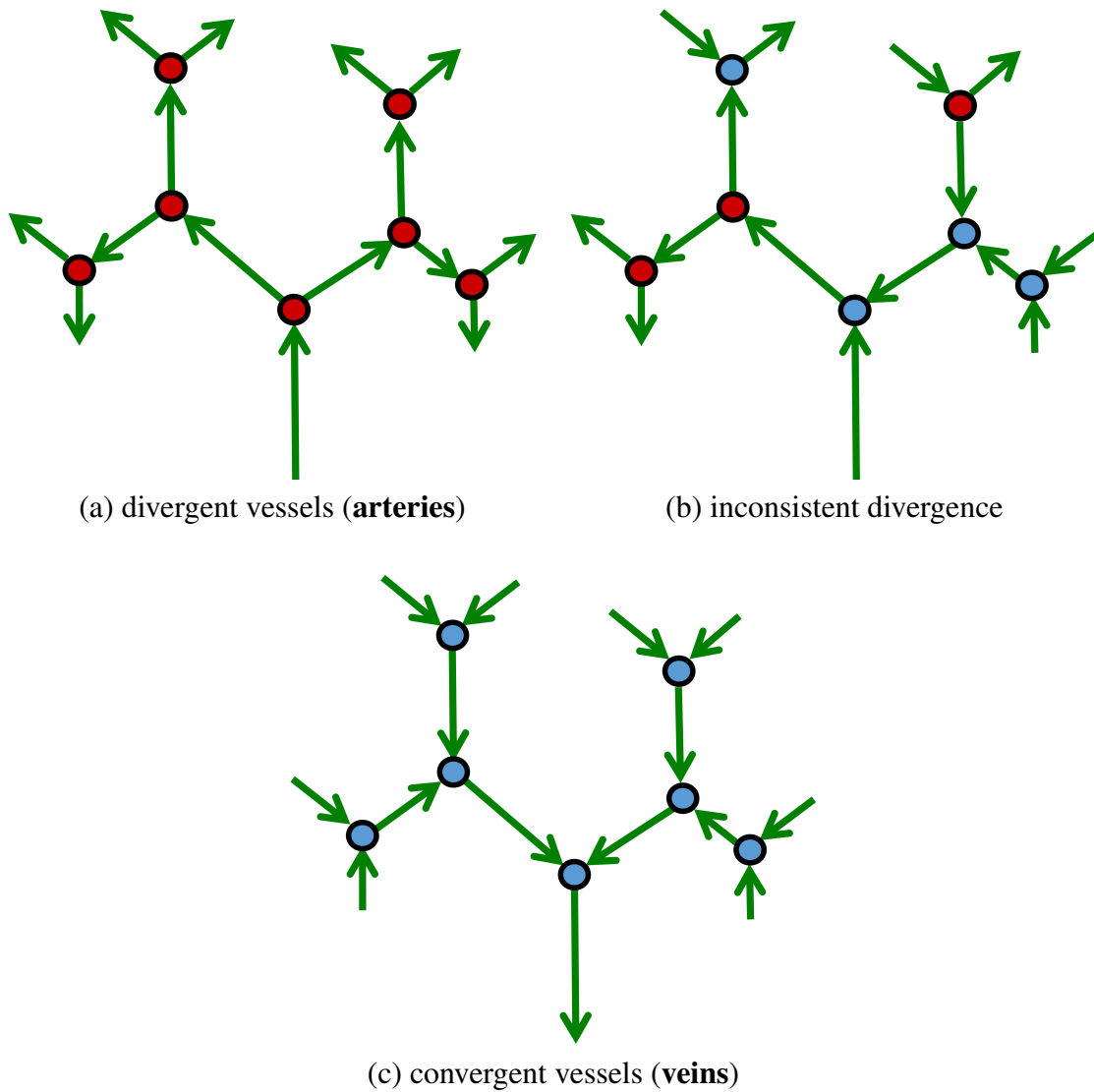


Figure 1.3: [**Vessel-tree divergence**] Vessels are the blood flow *pathlines* and could be assigned orientations. Green arrows represent vessels with blood flow orientation. To estimate orientations, we penalize negative (or positive) “vessel divergence”, which we define as the divergence of oriented unit tangents of vessels/pathlines. Such *unit tangent flow* divergence is positive (red) or negative (blue) at bifurcations, see (a-c). Note that standard curvature models [38, 32] and oriented curvature models given in (3.2) either can not distinguish (b) from (a) and (c) or may even prefer (b) depending on specific combinations of bifurcation angles. For example, compare vessel direction disambiguation based on curvature and the divergence prior in Figure 3.4 (a) and (b).

- We provide extensive quantitative validation on synthetic vessel data, as well as qualitative results on real high-resolution volumes.

1.4 Thesis Structure

This thesis is organized as follows. In Chapter 2 we introduce the basic elements in our vessel centerline extraction method. In Chapter 3 we introduce the concept of oriented curvature and divergence constraints as well as the optimization algorithms. In Chapter 4 we evaluate our proposed method on both synthetic data and real data. Chapter 5 concludes our work and gives feasible directions for future work.

Chapter 2

Background on Vessel-tree Reconstruction

This chapter outlines several basic elements in our approach. In Section 2.1, we describe the mechanism of the vessel enhancement filter developed by Frangi *et al.* [16]. The output of this filter¹ is thresholded and then used as the input for the curvature regularization framework, as described in Section 2.2. The Minimum Spanning Tree (MST) [27] is finally used for extracting tree topology and we describe an MST variant based on arc length in Section 2.3.

2.1 Vessel Enhancement Filter

This section describes the algorithm for multiscale vessel enhancement filtering developed by Frangi *et al.* [16]. The basic idea behind the algorithm is using multiscale eigen analysis of the intensity Hessian matrix.

The Hessian matrix consists of second-order derivatives which are sensitive to noise. Hence, we usually use, say, a Gaussian kernel filter to smooth image before computing the Hessian matrix. Now, consider such a smoothed image and a function giving intensity value $I(p, s)$ for each pixel p at scale s defined by the Gaussian kernel.

In order to analyze the local behavior of the image, we usually consider the Taylor expansion up to second-order in the neighborhood of p :

$$I(p + \delta p, s) \approx I(p, s) + \delta p^T \nabla I_{p,s} + \delta p^T H_{p,s} \delta p \quad (2.1)$$

where $\nabla I_{p,s}$ and $H_{p,s}$ are gradient vector and Hessian matrix evaluated at p , respectively.

To calculate these differential operators in a well-posed fashion, we resort to the concept of *linear scale space theory* [15, 23]. In this framework, the differentiation is defined as the

¹Non-maximum suppression is usually applied on the output to reduce data size.

convolution with the derivative of the Gaussian function. For instance²,

$$\frac{\partial}{\partial x}I(p, s) = I(p, s) * \frac{\partial}{\partial x}G(p, s), \quad (2.2)$$

where $p = (x, y, z)$ is the location of pixel p , G represents the three-dimensional Gaussian kernel and $*$ is the convolution symbol. Here,

$$\frac{\partial}{\partial x}G(p, s) = \frac{1}{\sqrt{(2\pi s^2)^3}} \exp\left(\frac{\|p\|^2}{2s^2}\right) \cdot \frac{-x}{s^2}. \quad (2.3)$$

Then, the Hessian matrix in Equation (2.1) is given by:

$$H_{p,s} = \begin{bmatrix} \frac{\partial^2}{\partial x^2}I(p, s) & \frac{\partial^2}{\partial x\partial y}I(p, s) & \frac{\partial^2}{\partial x\partial z}I(p, s) \\ \frac{\partial^2}{\partial y\partial x}I(p, s) & \frac{\partial^2}{\partial y^2}I(p, s) & \frac{\partial^2}{\partial y\partial z}I(p, s) \\ \frac{\partial^2}{\partial z\partial x}I(p, s) & \frac{\partial^2}{\partial z\partial y}I(p, s) & \frac{\partial^2}{\partial z^2}I(p, s) \end{bmatrix}. \quad (2.4)$$

Let $\lambda_{s,k}$ denote the eigenvalue and $\mu_{s,k}$ denote the eigenvector, then:

$$H_{p,s}\mu_{s,k} = \lambda_{s,k}\mu_{s,k}, \quad k = 1, 2, 3, \quad (2.5)$$

Without loss of generality the eigenvalues are ordered by: $|\lambda_{s,1}| \leq |\lambda_{s,2}| \leq |\lambda_{s,3}|$. The eigenvector $\mu_{s,1}$ indicates the direction of minimum intensity variation, i.e. along the vessel. The $\mu_{s,2}$ and $\mu_{s,3}$ form a base for the orthogonal plane.

The three following local structures can be identified by analyzing eigenvalues: a plate-like structure ($|\lambda_{s,1}| \approx 0, |\lambda_{s,1}| \approx |\lambda_{s,2}|, |\lambda_{s,2}| \ll |\lambda_{s,3}|$), a curvilinear structure ($|\lambda_{s,1}| \approx 0, |\lambda_{s,1}| \ll |\lambda_{s,2}|, |\lambda_{s,2}| \approx |\lambda_{s,3}|$) and a blob-like structure ($|\lambda_{s,1}| \approx |\lambda_{s,2}| \approx |\lambda_{s,3}|$) [16][10]. Furthermore, Frangi *et al.* [16] propose a *vesselness measure* which indicates the probability of a pixel belonging to the vessel. For each pixel p , the *vesselness measure* at scale s is defined as:

$$V(p, s) = \begin{cases} 0, & \text{if } \lambda_{s,2} > 0 \text{ or } \lambda_{s,3} > 0 \\ \exp\left(-\frac{R_{A,s}^2}{2\alpha^2}\right)(1 - \exp\left(-\frac{R_{B,s}^2}{2\beta^2}\right))(1 - \exp\left(-\frac{M_s^2}{2\gamma^2}\right)) & \text{otherwise} \end{cases} \quad (2.6)$$

where:

$$\begin{aligned} R_{A,s} &= \frac{|\lambda_{s,1}|}{\sqrt{|\lambda_{s,2}||\lambda_{s,3}|}} \\ R_{B,s} &= \frac{|\lambda_{s,2}|}{|\lambda_{s,3}|} \\ M_s &= \sqrt{|\lambda_{s,1}|^2 + |\lambda_{s,2}|^2 + |\lambda_{s,3}|^2}. \end{aligned} \quad (2.7)$$

$R_{A,s}$ distinguishes between a blob-like ($R_{A,s} \approx 1$) and a curvilinear or a plate-like ($R_{A,s} \approx$

²For simplicity, we omit the parameter for defining the normalized derivative introduced by Lindeberg [31].

0) structure. $R_{B,s}$ distinguishes between a curvilinear ($R_{B,s} \approx 1$) and a plate-like ($R_{B,s} \approx 0$) structures. The Frobenius norm M_s has a maximal response at a high contrast region and low response at the background. Hence the *vesselness measure* in Equation (2.6) is high if the pixel belongs to vessels (characterized by small $|\lambda_{s,1}|$ and large $|\lambda_{s,2}|, |\lambda_{s,3}|$) and low otherwise. α , β and γ are hyperparameters that control the sensitivity of the *vesselness measure* to $R_{A,s}$, $R_{B,s}$ and M_s , respectively.

Eventually, we compute the *vesselness measure* at each pixel p by:

$$V(p) = \max_{s_{min} \leq s \leq s_{max}} V(p, s) \quad (2.8)$$

where the scale range $[s_{min}, s_{max}]$ must cover that of the expected vessel radii. The s with a highest response of $V(p, s)$ approximates the vessel radius.

2.2 Curvature Regularization for Centerline

In this section, we review related prior work using curvature regularization to extract centerline. The output of Frangi filter is noisy and simple thresholding of points with large-enough *vesselness measure* value is often unreliable as a method for vascular tree reconstruction. The presence of near-capillary vessels exacerbates the effect of such thresholding. To address this issue, the curvature, a second-order smoothness term, is used to regularize the model parameters defining the centerline, see Figure 2.1(a). In general, curvature has been studied for image segmentation [41, 44, 42, 6, 19, 40, 37, 32], for stereo or multi-view-reconstruction [30, 39, 47], connectivity measures in analysis of diffusion MRI [35], for tubular structures extraction [32], for *inpainting* [2, 8] and edge completion [17, 46, 1].

Olsson *et al.* [38] propose curvature approximation for surface fitting regularization. Their framework exploits tangential approximation of surfaces and assumes that the data points are noisy readings of the surface. The method estimates local surface patches, which are parameterized by a tangent plane. It is assumed that the distance from the data point to its tangent plane is a surface norm. That implicitly defines the point of tangency.

Assume there is a smooth curve, see Figure 2.1(b). Points p and q on the curve and tangents l_p and l_q at these points are given. Then the integrals of curvature $\kappa(\cdot)$ is estimated by:

$$\int_p^q |\kappa(s)| ds \approx \frac{\|p - l_q\|}{\|p - q\|} \quad (2.9)$$

and

$$\int_p^q |\kappa(s)|^2 ds \approx \frac{\|p - l_q\|^2}{\|p - q\|^3}, \quad (2.10)$$

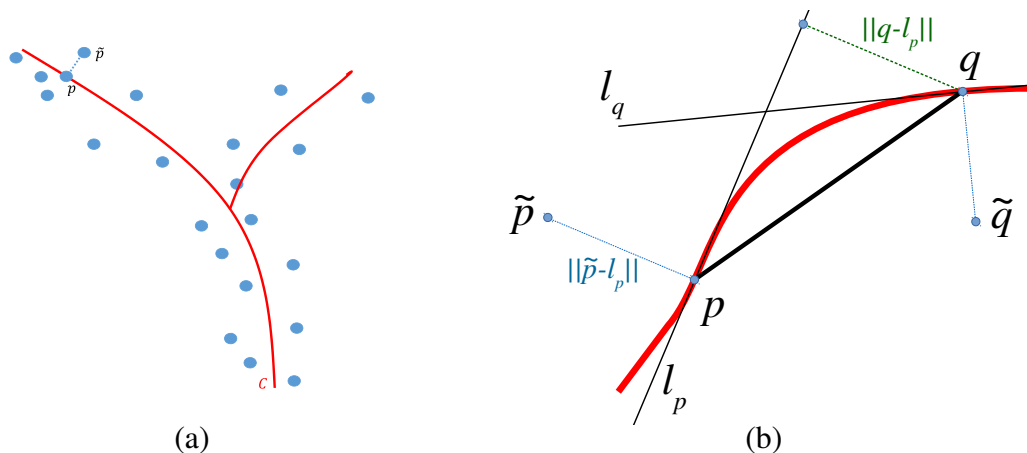


Figure 2.1: Centerline definition and curvature model of [38]. (a) illustrates an example of noisy measurements along with the centerline C . (b) illustrates the curvature model. Given two points p and q on the red curve (centerline) and two tangents l_p and l_q at these points, the integrals of curvature are approximated by ((2.9)-(2.11))

where $\|p - l_q\|$ is the distance between point p and the tangent line at point q represented by vector l_q . Olsson *et al.* [38] explores properties of these approximations and argues

$$\kappa_{pq}(l_p, l_q) := \frac{1}{2} \frac{\|p - l_q\|^2 + \|q - l_p\|^2}{\|p - q\|^2} \quad (2.11)$$

is a better regularizer, where we used a variant of the symmetric version of integral in (2.10).

Marin *et al.* [32] generalized this surface fitting problems to detection problems where majority of the data points, *e.g.* image pixels, do not belong to a thin structure. In order to do that, they introduced binary variables in their energy indicating if a data point belongs to the thin structure. One of their applications is vessel detection. The proposed vessel-tree extraction system includes vessel enhancement filtering, non-maximum suppression for data reduction, tangent approximation of vessel centerline and minimum spanning tree for tree topology extraction. Assuming that detection variables are computed, the tangent approximation of vessels' centerline is found by minimizing energy:

$$E_u(l) = \sum_p \|\tilde{p} - l_p\|^2 + \gamma \sum_{(p,q) \in \mathcal{N}} \kappa_{pq}(l_p, l_q), \quad (2.12)$$

where summations are over detected vessel points, \tilde{p} is the noisy measurement, l_p is the tangent vector at point p , the denoised point location p is constrained to be the closest point on tangent line at p , and $\mathcal{N} \subset \Omega^2$ is the neighborhood system. The curvature term in the energy makes the tangents "collapse" onto a one-dimensional centerline as shown in Figure 2.2(a)(b). However, the same figures also show artifacts around bifurcations where undesired triangular structure

forms indicating unoriented tangent model limitation. This phenomenon is the main issue we solve in this thesis and will be addressed in next chapter.

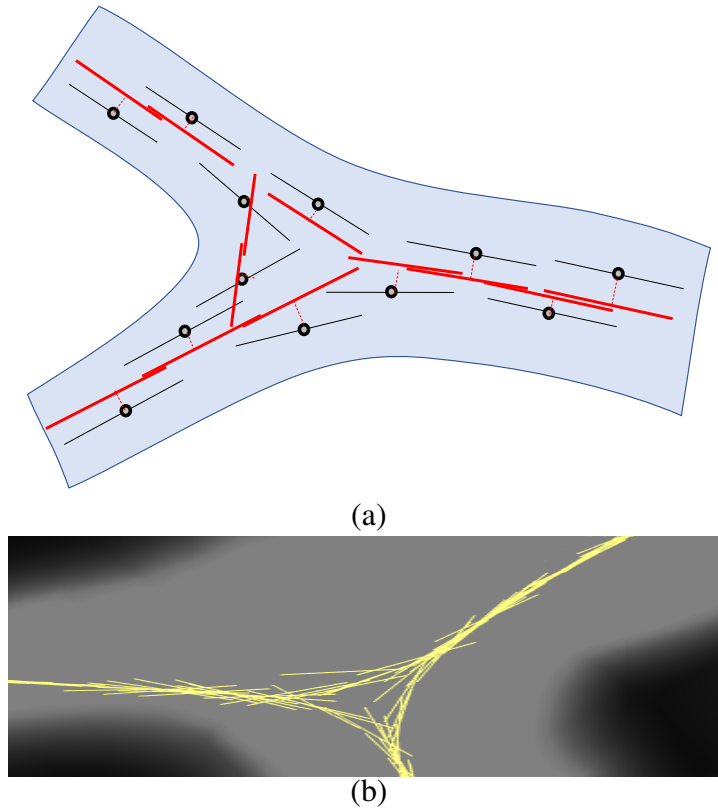


Figure 2.2: Triangle artifacts at bifurcation. Optimization of energy (2.12) ignoring tangent orientations often leads to a strong local minima as in (a) and (b). The line segments are the estimated tangents of the centerline.

2.3 Minimum Spanning Tree with Arc Length

The outputs of minimizing energy (2.12) are simply locations of the points along with tangent vectors. In order to extract the tree topology, we employ the Minimum Spanning Tree algorithm developed by Kruskal [27].

Normally, given a set of point cloud and a graph built on it, the weight of each edge is computed by calculating the Euclidean distance between two end points of such an edge. In our case, we consider the anisotropy of vessel structure and exploit the tangent at each point to compute the arc length as edge weight. Since two tangents at each pair of points may not be coplanar, we use the average of two arc lengths as the edge weight. For simplicity, we only show the two-dimensional case here, see Figure 2.3. Such edge weight has a good property where, considering two fixed points, the edge weight will be smaller if two tangents at ends of

the edge become collinear. Thereby, it is more robust to use the MST algorithm with such a edge weight for vessel tree extraction.

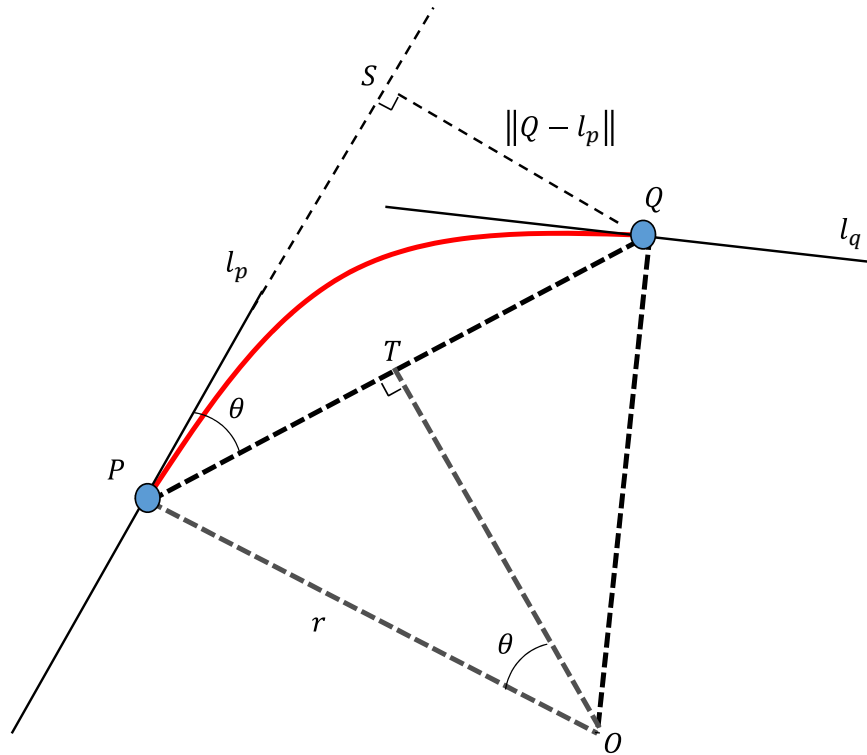


Figure 2.3: Illustration of the arc. The red curve is the arc from point P to Q , which is tangential with l_p at point P . Point O is the center of the circle to which the arc belongs. θ is the osculation angle.

In practice, there is one issue regarding the calculation of the arc length. Computing the arc length involves the cross-product of l_p and l_q , which becomes quite numerically unstable when the two tangents are almost collinear. Hence, we adopt a stable approach for approximating the arc length.

Considering two points P and Q and the tangents l_p and l_q at these points, see Figure 2.3, we introduce an arc that is tangential with l_p and l_q ³. In $\triangle OPT$, we have

$$\sin\theta = \frac{\frac{\|P-Q\|}{2}}{r}. \quad (2.13)$$

In $\triangle PQS$, we also have

$$\sin\theta = \frac{\|Q - l_p\|}{\|P - Q\|}. \quad (2.14)$$

³Since this is a two-dimensional case, the arc must be tangential to both tangents.

Combining (2.13) and (2.14) yields

$$r = \frac{\|P - Q\|^2}{2 * \|Q - l_p\|}. \quad (2.15)$$

The arc length \widehat{PQ} is calculated by $r \cdot 2\theta$ and substituting (2.14) and (2.15) into it yields:

$$\widehat{PQ} = \frac{\|P - Q\|^2}{\|Q - l_p\|} \cdot \arcsin\left(\frac{\|Q - l_p\|}{\|P - Q\|}\right). \quad (2.16)$$

Then, we exploit the Taylor expansion of $\arcsin(x)$ up to the second term to approximate the arc length:

$$\widehat{PQ} \approx \|P - Q\| + \frac{\|Q - l_p\|^2}{6\|P - Q\|}. \quad (2.17)$$

Chapter 3

Oriented Curvature and Divergence Constraints

In this chapter we focus on analysis of failure cases, *e.g.* the triangle artifact as shown in Figure 2.2, and improvement of the regularization stage for tangent approximation. In particular we will show the drawbacks of the curvature models (2.9-2.11) in the context of vessel tree extraction and propose a solution, as described in the following sections. In Section 3.1 and 3.2, we present an oriented version of the curvature model and its limitations. Then we introduce an important constraint (prior) using concept of divergence in Section 3.3. Finally, we will review the optimization algorithms addressing the mixed integer non-linear problem in Section 3.4.

3.1 Oriented Curvature Constraint

Previous works [38, 40, 32] ignored orientations of tangent vectors $l_{p \in \Omega}$. Equations (2.9)-(2.12) do not depend on the orientations of l . In practice, the orientations of vectors are arbitrarily defined. Ignoring the orientation in energy (2.12) results in significant "triangle" artifacts around bifurcations, see Figure 2.2. Consider an illustrative example in Figure 3.1(a). Each of the three tangents interacts with the other two. The prior knowledge about blood flow pattern dictates that among those three tangents there should be one incoming and one outgoing. Introduction of orientation allows us to distinguish between the incoming and outgoing tangents and subsequently inactivate one of the interactions, see Figure 3.1(b), resulting in disappearance of these artifacts, see Figure 3.2.

In order to introduce oriented curvature, we introduce a new vector field \vec{l}_p , which we call *oriented*. Then, we introduce energy $E_o(\vec{l})$ by replacing curvature term in energy (2.12) with a

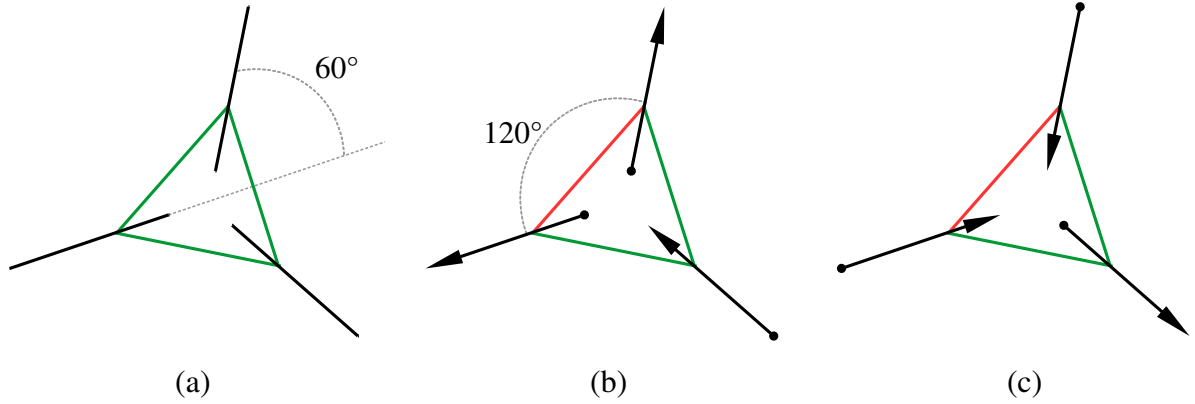


Figure 3.1: Illustrative examples of three interacting tangents with unoriented curvature (a) as in energy (2.12) and two alternative oriented configurations (b) and (c) with oriented curvature as in energy (3.1). The green line denotes pairwise interaction with low curvature value. Note, that unoriented curvature (2.9–2.11) always chooses the smallest angle for calculation. The red line shows “inactive” pairwise interaction, *i.e.* interaction where curvature in (3.2) reaches the high saturation threshold.

new oriented curvature as following:

$$E_o(\bar{l}) = \sum_p \|\tilde{p} - \bar{l}_p\|^2 + \gamma \sum_{(p,q) \in \mathcal{N}} \bar{\kappa}_{pq}(\bar{l}_p, \bar{l}_q) \quad (3.1)$$

where

$$\bar{\kappa}_{pq}(\bar{l}_p, \bar{l}_q) := \begin{cases} \kappa_{pq}(\bar{l}_p, \bar{l}_q) & \langle \bar{l}_p, \bar{l}_q \rangle \leq \tau \\ 1 & \text{otherwise} \end{cases} \quad (3.2)$$

and $\langle \bar{l}_p, \bar{l}_q \rangle$ is the dot product of \bar{l}_p and \bar{l}_q and $\tau \leq 0$ is a positive threshold discussed in Figure 3.3.

The connection between oriented field \bar{l} and l is

$$\bar{l}_p = x_p \cdot l_p \quad (3.3)$$

where binary variables $x_p \in \{-1, 1\}$ flip or preserve the arbitrary defined orientations of l_p .

3.2 Curvature and Orientation Ambiguity

Introduction of oriented curvature resolves triangle artifacts, see Figure 3.2. However, the orientations are not known in advance. For example, Frangi filter [16] defines a tangent as a unit eigenvector of a special matrix. The unit eigenvectors are defined up to orientation, which is chosen arbitrarily. One may propose to treat energy (3.1) as a function of tangent orientations

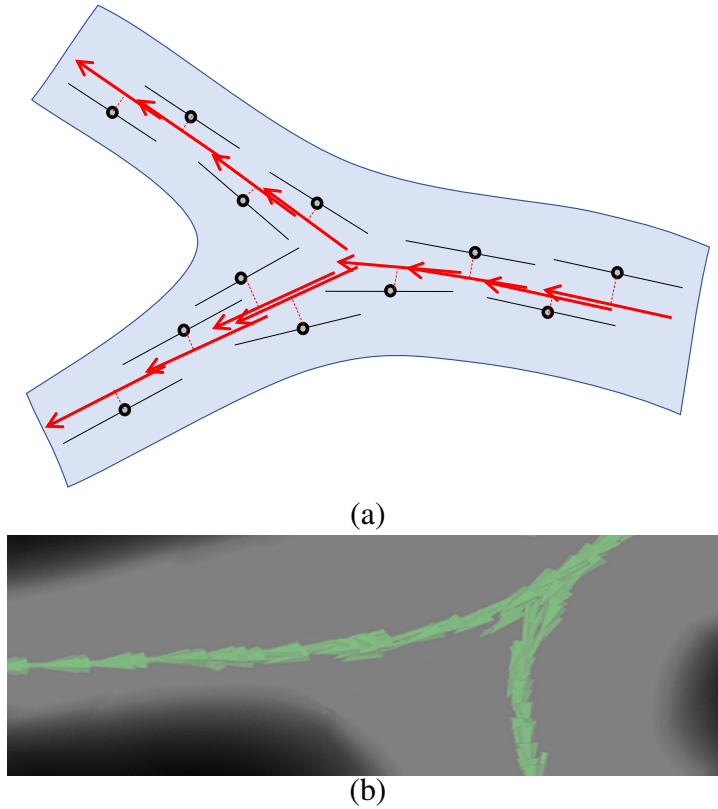


Figure 3.2: Elimination of triangle artifact at bifurcation. New curvature term (3.2) takes into account tangent orientation resolving the artifact.

x via relationship (3.3) as follows:

$$E_o(x) := E_o(\{x_p \cdot l_p\}) \Big|_{l_p = \text{const}}. \quad (3.4)$$

However, energy (3.4) is under-constrained because it allows multiple equally good solutions, see Figure 3.1(b) and (c). The example in (b) shows a divergent pattern while (c) shows a convergent pattern suggesting artery/vein ambiguity. Unfortunately, energy (3.4) does not enforce consistent flow pattern convergent bifurcations as in Figure 1.3. Real data experiments confirm this conclusion, see Figure 3.4(a).

Thus, oriented curvature model (3.1) has a significant problem. While it can resolve "triangle artifacts" at bifurcations, see Figure 2.2, it will break the wrong sides of the triangles at many bifurcations where it estimates the flow pattern incorrectly and then give the incorrect estimation of centerline, see Figure 3.5. Below we introduce our divergence prior directly enforcing consistent flow pattern over the vessel tree.

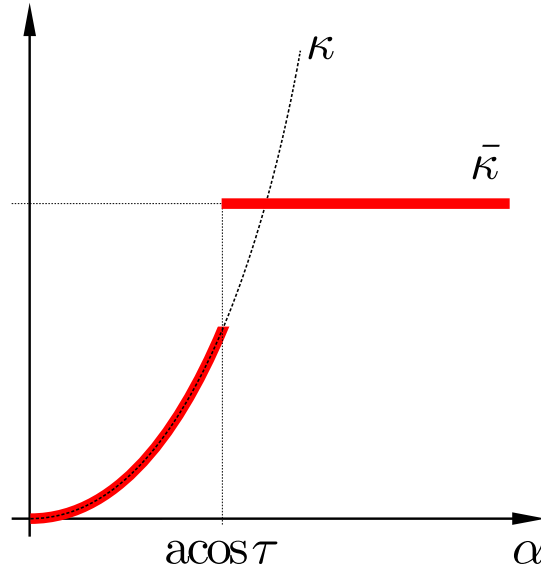


Figure 3.3: Robustness of curvature (3.2). The pairs of tangent vectors that has angle greater than $\text{acos } \tau$ are not considered as belonging to the same vessel. A constant penalty is assigned to such pairs. This “turns off” smoothness enforcement at bifurcations.

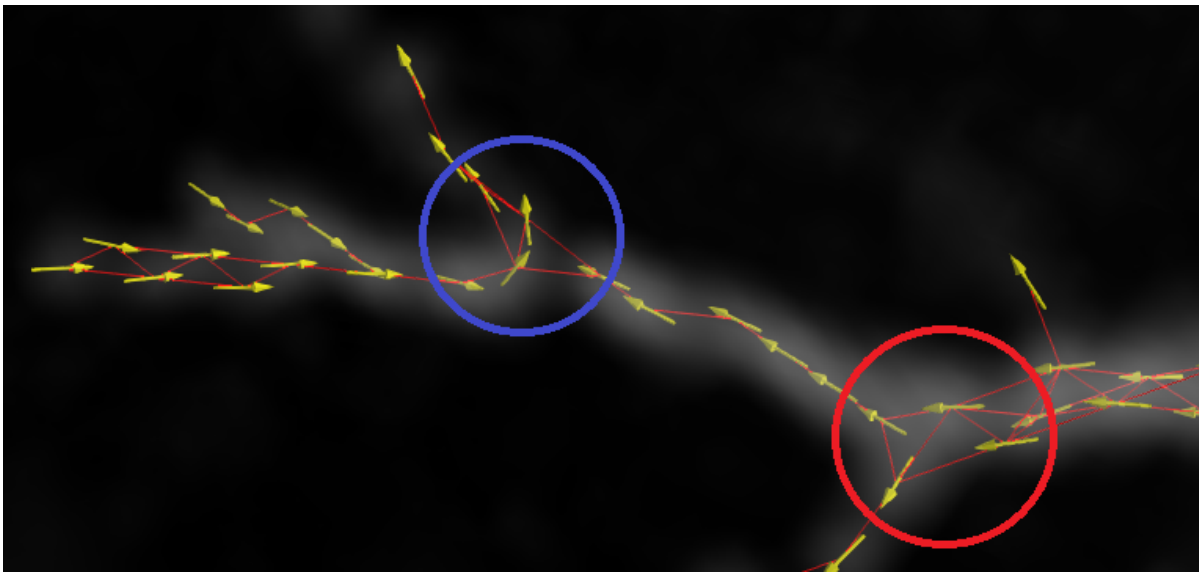
3.3 Divergence Constraint

The divergence constraint is imposed by estimating divergence. Figure 3.6 describe our (finite element) model for estimating divergence of a sparse vector field $\bar{l}_p|_p \in \Omega$ defined for a finite set of points $\Omega \subset \mathcal{R}^3$. We extrapolate the vector field over the whole domain \mathcal{R}^3 assuming constancy of the vectors on the interior of the Voronoi cells for $p \in \Omega$, see Figure 3.6(a). Thus vectors change only in the (narrow) region around the cell facets where all non-zero divergence is concentrated. To compute the integral of divergence in the area between two neighboring points $p, q \in \Omega$, see Figure 3.6(b), we estimate the flux of the extrapolated vector field over ϵ -thin box f_{pq}^ϵ around facet f_{pq}

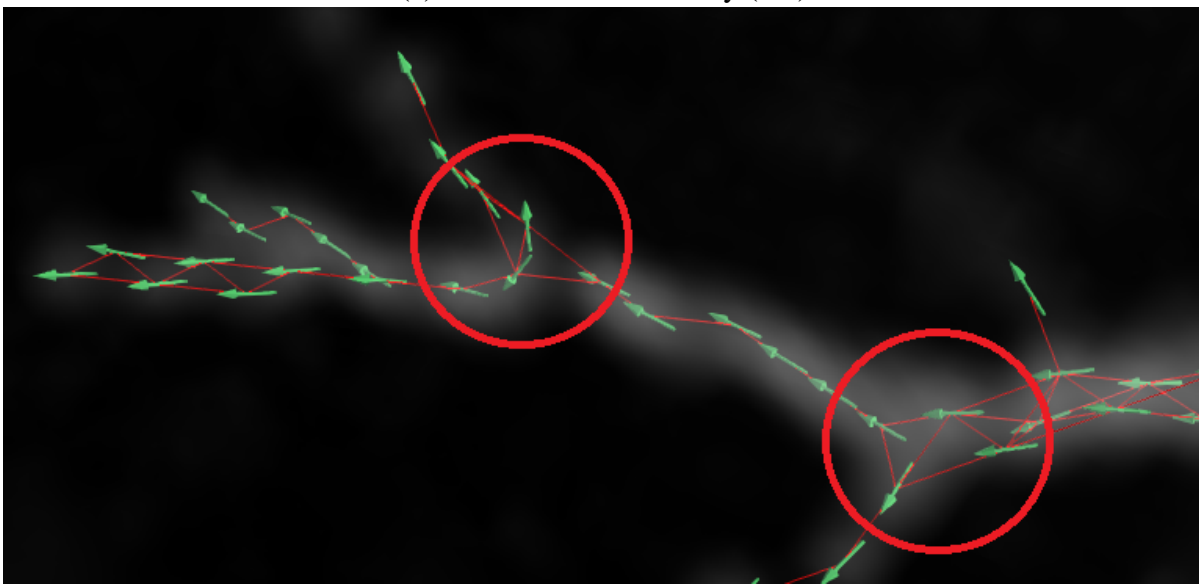
$$\int_{f_{pq}^\epsilon} \langle \bar{l}, n_s \rangle ds = \frac{\langle \bar{l}_q, pq \rangle - \langle \bar{l}_p, pq \rangle}{|pq|} \cdot |f_{pq}| + o(\epsilon),$$

where n_s is the outward unit normal of the box and $|f_{pq}|$ is the facet’s area. Then, the divergence theorem implies the following formula for the *integral* of divergence of the vector field inside box f_{pq}^ϵ :

$$\nabla \bar{l}_{pq} = \frac{\langle \bar{l}_q, pq \rangle - \langle \bar{l}_p, pq \rangle}{|pq|} \cdot |f_{pq}|, \quad (3.5)$$

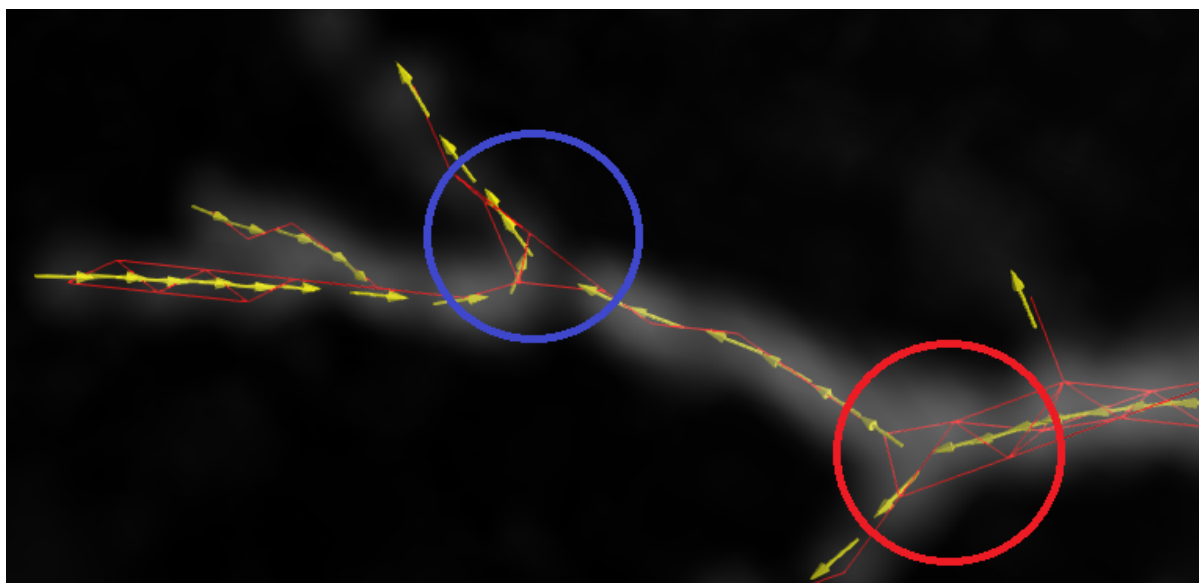


(a) oriented curvature only (3.4)

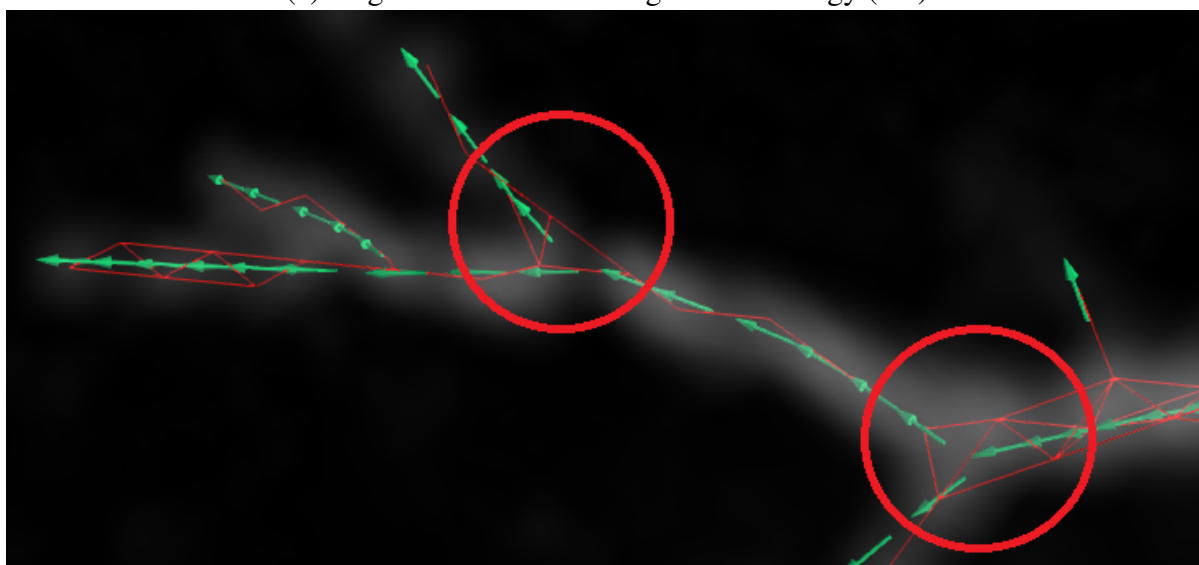


(b) with divergence prior (3.7)

Figure 3.4: Disambiguating flow directions in Frangi output [16]. Both examples use fixed (unoriented) vessel tangents $\{l_p\}$ produced by the filter and compute (oriented) vectors $\bar{l}_p = x_p l_p$ (3.3) by optimizing binary sign variables $\{x_p\}$ using energies (3.4) in (a) and (3.7) in (b). The circles indicate divergent (red) or convergent (blue) bifurcations similarly to the diagrams in Figure 1.3. The extra divergence constraint in (3.6) enforces consistent flow pattern (b).



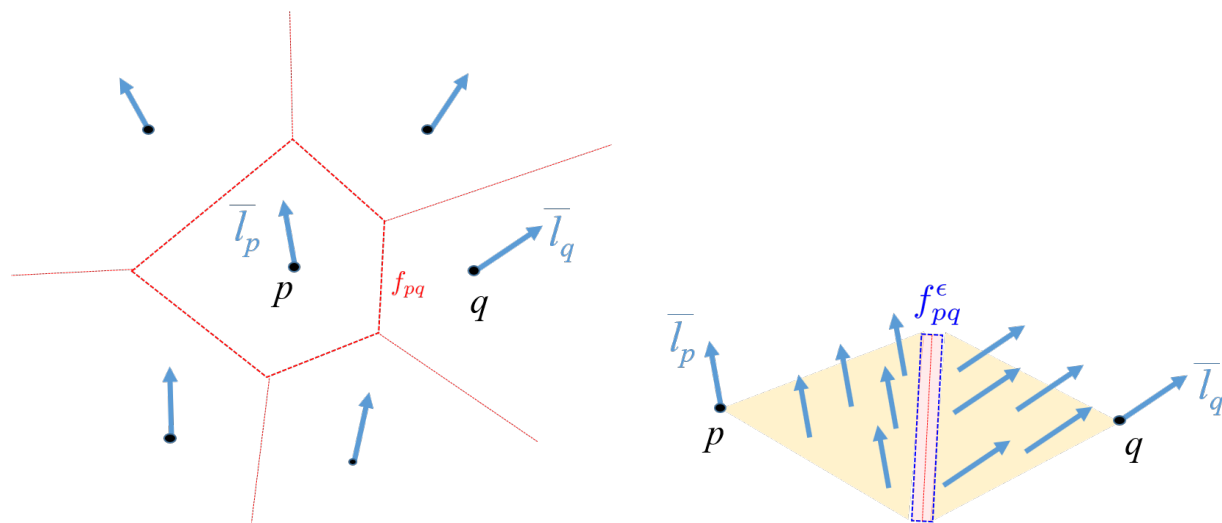
(a) tangent vectors at convergence for energy (3.1)



(b) tangent vectors at convergence for energy (3.6)

Figure 3.5: Centerline estimation for the data in Figure 3.4. Instead of showing tangent orientations estimated at the first iteration as in Figure 3.4, we now show the final result at convergence for minimizing energy (3.1) in (a) and energy (3.6) in (b). Blue circle shows bifurcation reconstruction artifacts due to wrong estimation of vessel orientations in Figure 3.4(a).

where we ignore only infinitesimally negligible $o(\epsilon)$ term.



(a) Voronoi cells for $p, q \in \Omega$ and facet f_{pq}

(b) ϵ -thin box f_{pq}^ϵ around facet f_{pq}

Figure 3.6: Divergence of a sparse vector field $\{\bar{l}_p | p \in \Omega\}$. Assuming that the corresponding “extrapolated” dense vector field is constant inside Voronoi cells (a), it is easy to estimate (non-zero) divergence $\nabla \bar{l}_{pq}$ (3.5) concentrated in a narrow region f_{pq}^ϵ around each facet (b) using the divergence theorem.

3.4 Oriented Centerline Estimation

Constraints for divergence $\nabla \bar{l}_{pq}$ in the regions between neighbors $p, q \in \mathcal{D}$ in Delaunay triangulation of Ω can be combined with $E_o(\bar{p})$ in (3.1) to obtain the following joint energy for estimating oriented centerline tangents \bar{l}_p :

$$E(\bar{l}) = E_o(\bar{l}) + \lambda \sum_{(p,q) \in \mathcal{D}} (\nabla \bar{l}_{pq})^-, \quad (3.6)$$

where the negative part operator $(\cdot)^-$ encourages a divergent flow pattern as in Figure 1.3(a). Alternatively, one can use $(\cdot)^+$ to encourage a convergent flow pattern as in Figure 1.3(c). This joint energy for oriented centerline estimation $E(\bar{l})$ combines Frangi measurements, centerline curvature regularity, and consistency of the flow pattern, see Figure 2.2(b). Note that specific value of facet size in (3.5) had a negligible effect in our centerline estimation tests as it only changes a relative weight of the divergence penalty at any given location. For simplicity, one may use $|f_{pq}| \approx \text{const}$ for all $p, q \in \mathcal{D}$.

Optimization of oriented centerline energy $E(\bar{l})$ in (3.6) over oriented tangents \bar{l}_p can be

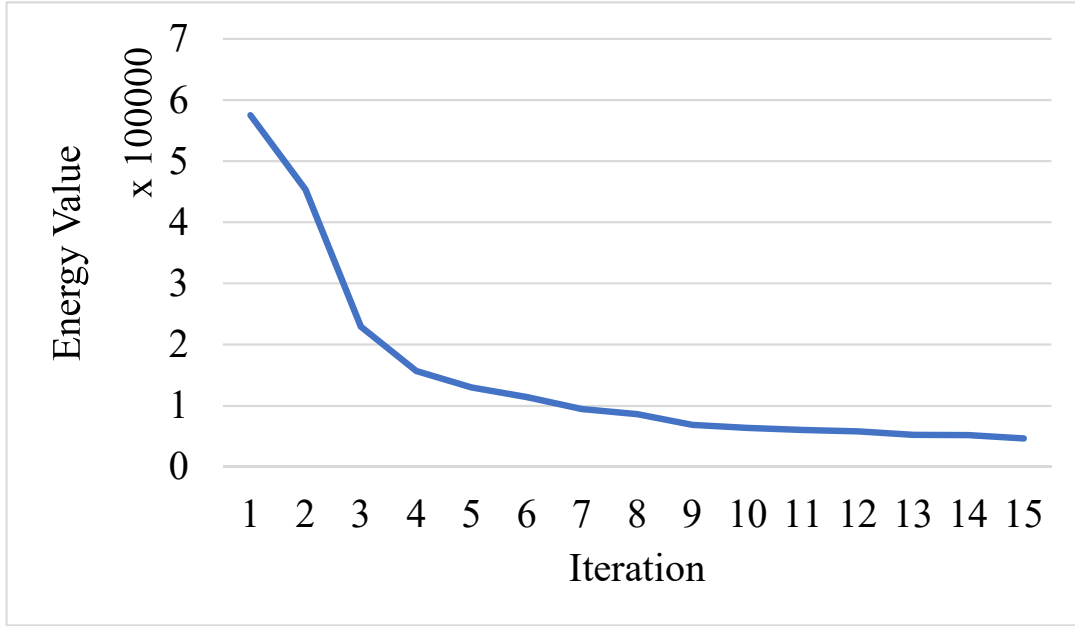


Figure 3.7: Representative example of decrease in energy (3.6) for block-coordinate descent iterating optimization of (3.7) and (3.8). For initialization, we use raw undirected tangents $\{l_p\}$ generated by Frangi filter [16]. Then, we iteratively reestimate binary sign variables $\{x_p\}$ and unoriented tangents $\{l_p\}$.

done via block-coordinate descent. As follows from definition (3.3):

$$E(\bar{l}) \equiv E(x_p \cdot l_p).$$

We iterate TRW-s [25] for optimizing non-submodular energy for binary "sign" disambiguation variables x_p

$$E(x) := E(x_p \cdot l_p) \Big|_{l_p = \text{const}} \quad (3.7)$$

and *trust region* [48, 32] for optimizing robust energy for aligning tangents into 1D centerline

$$E(l) := E(x_p \cdot l_p) \Big|_{x_p = \text{const}}. \quad (3.8)$$

Figure 3.7 shows a representative example illustrating convergence of energy (3.6) in a few iterations. In next subsections, more details regarding TRW-s and *trust region* are explained.

Note that the divergence constraint in joint energy (3.6) resolves the problem of under-constrained objective (3.1) discussed at the end of Section 3.1. Since the flow pattern consistency is enforced, optimization of (3.6) should lead to a consistent resolution of the triangle artifacts at bifurcation, see Figure 3.5(b). Our experimental results support this claim as shown

in next chapter.

3.4.1 Binary Optimization via TRW-s

Combinatorial optimization, especially binary optimization, is of fundamental importance in computer vision. For instance, binary segmentation can be viewed as assigning label 0 or 1 to each pixel to minimize an energy. When such energy is submodular, we can efficiently optimize it via graph cuts to get the optimal solution [5, 21, 24]. However, minimizing such energy is generally NP-hard, so researchers have also focused on approximate algorithms.

Linear programming(LP) relaxation is widely studied in the context of solving such binary optimization [9, 26, 45]. Wainwright *et al.* [45] studied such optimization problem in the context of tree-reweighted message passing (TRW) which is a certain form closely related to LP relaxation dual problem. However, the lower bound is not guaranteed to increase consistently and thus the convergence can not be guaranteed. Kolmogorov [25] then proposed a sequential version of TRW (TRW-s) and demonstrated that the lower bound does not decrease.

Here, we expand (3.4) as¹

$$E(x) = \sum_p \underbrace{\frac{1}{\sigma_p^2} \|\tilde{p} - x_p l_p\|^2}_{\theta_p(x_p)} + \sum_{(p,q) \in \mathcal{N}} \underbrace{\gamma \bar{K}_{pq}(x_p l_p, x_q l_q) + \lambda (\nabla \bar{l}_{pq})^-}_{\theta_{pq}(x_p, x_q)}. \quad (3.9)$$

We need to compute $\theta_p(1)$ and $\theta_p(-1)$ for each pixel p . As for each pair (p, q) in the neighborhood system, we compute $\theta_{pq}(-1, -1)$, $\theta_{pq}(-1, 1)$, $\theta_{pq}(1, -1)$ and $\theta_{pq}(1, 1)$ based on (3.2) and (3.5). Then, we use the TRW-s code provided by Kolmogorov to optimize (3.9).

3.4.2 Trust Region Optimization for Tangent Estimation

In this section, we talk about the trust region optimization for tangent estimation. Our problem is formulated as minimizing the energy (3.8) and we expand the energy as

$$E(l) = \sum_p \underbrace{\frac{1}{\sigma_p^2} \|\tilde{p} - l_p\|^2}_{D^2(l_p)} + \sum_{(p,q) \in \mathcal{N}} \underbrace{\gamma \bar{K}_{pq}(l_p, l_q) + \lambda (\nabla l_{pq})^-}_{V^2(l_p, l_q)}, \quad (3.10)$$

where σ_p denotes the radius of vessel and we leave out the fixed x_p here. We also add unary potential coefficient $\frac{1}{\sigma_p^2}$ which determines how far tangents l_p can move away from \tilde{p} . Since this data has high variability in vessel thickness, we can not use the same σ for each voxel [32].

¹We made modifications by adding unary coefficients, which will be explained in Subsection 3.4.2, and using the same neighborhood system for both curvature and divergence constraints.

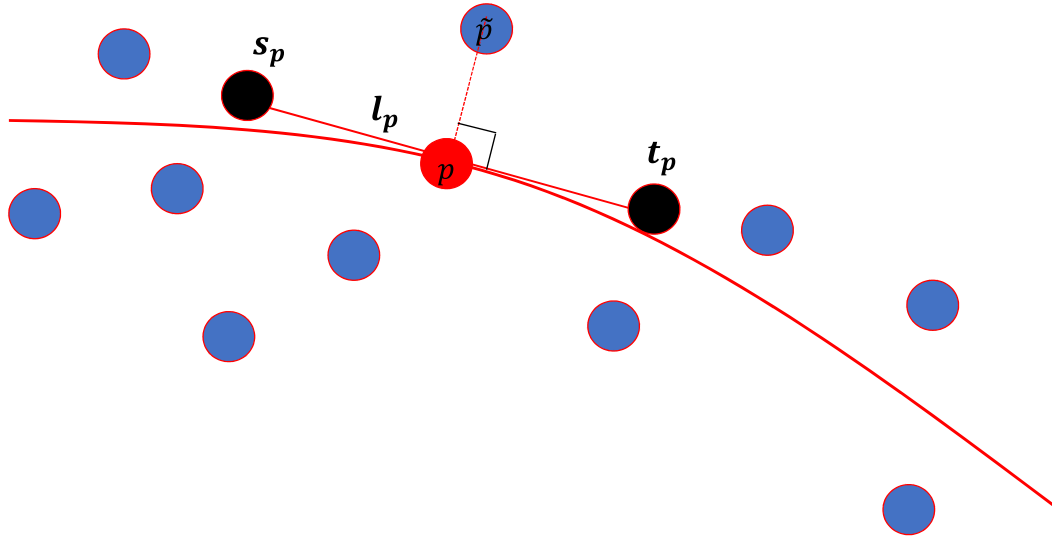


Figure 3.8: Illustration of the parameterization of point p . \tilde{p} is the noisy observation of the point p . Since p is assumed to be the closest point on tangent l_p from \tilde{p} , p is calculated by the orthogonal projection of \tilde{p} onto the line l_p .

We follow [10] for a detailed implementation. We parameterize the l_p as two variables s_p and t_p , which represent the two ends of the l_p respectively, see Figure 3.8. We can represent any point on the line through s_p and t_p by

$$s_p + \alpha(t_p - s_p),$$

where $\alpha \in R$ is a parameter determined by the position of such point. Since the denoised point location p is assumed to be the closest point on the tangent l_p from the noisy measurement \tilde{p} , we can calculate the α_p by:

$$\alpha_p = \frac{(\tilde{p} - s_p)(t_p - s_p)}{\|t_p - s_p\|^2}. \quad (3.11)$$

Hence, p is given by:

$$p = s_p + \alpha_p(t_p - s_p). \quad (3.12)$$

Then we reformulate (3.10) as an unconstrained non-linear least squares problem of the form:

$$\min_x F(x) = \|f(x)\|^2 = f^T(x)f(x), \quad (3.13)$$

where²

$$x = \begin{bmatrix} \vdots \\ s_p \\ t_p \\ \vdots \\ s_q \\ t_q \\ \vdots \end{bmatrix}, \quad f(x) = \begin{bmatrix} \vdots \\ D(l_p) \\ \vdots \\ D(l_q) \\ \vdots \\ V(l_p, l_q) \\ \vdots \end{bmatrix}. \quad (3.14)$$

In order to solve the problem of the form in (3.13), we adopt the Levenberg-Marquardt algorithm which is a trust region second-order continuous iterative optimization method [29, 33, 48].

Trust region is a strategy for simultaneously choosing the most reasonable direction and step size within a certain region that is trusted. To be specific, we first set a certain region size. Within such region, we use a so called model function $m(x)$ to approximate the objective function and minimize the model function. If the optimal solution of the model function can lead to a reasonable decrease of objective function, we will move on with the next iteration. Otherwise, the region size will contract and we will find a new approximated minimizer.

Next, we first review the Levenberg-Marquardt algorithm and then move forward to an inexact version. The Levenberg-Marquardt algorithm [29, 48] is an algorithm, which adopts the trust region framework, for solving non-linear least square problem. In the k th iteration, we approximate the objective function:

$$F(x) = f^T(x)f(x) \quad (3.15)$$

in Equation (3.13) with the model function

$$m_k(x) = \tilde{f}_k^T(x)\tilde{f}_k(x) \quad (3.16)$$

where

$$\tilde{f}_k(x) = f(x_k) + J(x_k)(x - x_k) \quad (3.17)$$

is the first order Taylor expansion at the neighbor of x_k and $J(x_k)$ is the Jacobian matrix. Sub-

²For simplicity and cleanness, here we still use l_p and l_q to represent $f(x)$.

stituting (3.17) into (3.16) and supposing $x - x_k = \delta x$, then we have:

$$m_k(x_k + \delta x) = f^T(x_k)f(x_k) + 2f(x_k)^T J(x_k)\delta x + \delta x^T J(x_k)^T J(x_k)\delta x. \quad (3.18)$$

Then we compute the gradient of $m_k(x_k + \delta x)$ with respect to δx :

$$\nabla m_k = 2J^T(x_k)f(x_k) + 2J^T(x_k)J(x_k)\delta x \quad (3.19)$$

Next we can set (3.19) to zero and solve the system of linear equations

$$J^T(x_k)J(x_k)\delta x = -J^T(x_k)f(x_k) \quad (3.20)$$

to obtain the solution δx_k^* , which is used for updating the variable x by

$$x_{k+1} = x_k + \delta x_k^*. \quad (3.21)$$

Notice that there is no constraint on δx . If this step does not lead to sufficient decrease of objective function, we will contract the size of trust region. One possible way is to limit the length of δx . However, Levenberg [29] proposed using a damped parameter $\lambda_k > 0$ to implicitly control the size of trust region in each iteration by solving:

$$\min_{\delta x} m_k(x_k + \delta x) + \lambda_k \|\delta x\|^2 \quad (3.22)$$

where the value of λ_k can be chosen by many strategies [29]. Thereby, the system of equations in (3.20) can be replaced by the following damped version:

$$(J^T(x_k)J(x_k) + \lambda_k I)\delta x = -J^T(x_k)f(x_k). \quad (3.23)$$

In order to avoid the computation of $J^T(x_k)J(x_k)$ in (3.23) which is prohibitive for a large size problem, and by observing that the Jacobian matrix is sparse, we adopted the inexact Levenberg-Marquardt algorithm [48]. The algorithm internally uses LSQR [7] to solve linear least squares problem given in (3.22) in each iteration. The advantage of inexact Levenberg-Marquardt algorithm is that, instead of requiring a large amount of computation on $J^T(x_k)J(x_k)$, it only needs the matrix-vector multiplications, which is much cheaper than the operations of matrix-matrix multiplication. Wright and Holt [48] have also shown that one can just apply a few iterations of LSQR on (3.22) to save big computational efforts.

In our work, we extended the implementation based on [10]. The main modification we made is the evaluation of Jacobian matrix and objective function value according to (3.10).

Jacobian matrix can be represented in the following symbolic form with respect to l_p and l_q as:

$$J(\dots l_p \dots l_q \dots) = \begin{bmatrix} \frac{\partial D(l_p)}{\partial l_p} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{\partial D(l_q)}{\partial l_q} \\ \vdots & & \vdots \\ \frac{\partial V(l_p, l_q)}{\partial l_p} & \dots & \frac{\partial V(l_p, l_q)}{\partial l_q} \end{bmatrix} \quad (3.24)$$

We adopt the *automatic differentiation* [36] to calculate the derivative values. There are two modes in automatic differentiation: forward mode and reverse mode. We use the forward mode which is accomplished by introduction of an algebra *dual numbers*³.

We show a simplified version of inexact Levenberg-Marquardt algorithm in Alg.1. Wright and Holt [48] proposed a ratio ρ to decide whether the step should be accepted by comparing it to some given threshold c_1 . Interested readers can refer to [10] for the update rule of λ_k .

<pre> Input : x_0 1 $k \leftarrow 0$ 2 evaluate $J(x_k)$ 3 while <i>convergence criterion is not satisfied</i> do 4 $\delta x \leftarrow$ LSQR minimizes $m_k(x_k + \delta x) + \lambda_k \ \delta x\ ^2$ 5 $\rho \leftarrow \frac{F(x_k) - F(x_k + \delta x)}{F(x_k) - m_k(x_k + \delta x) - \lambda_k \ \delta x\ ^2}$ 6 if $\rho < c_1$ then δx does not sufficiently decrease the objective function value and the size of trust region is contracted 7 update λ_k 8 else δx is accepted 9 $x_{k+1} \leftarrow x_k + \delta x$ 10 evaluate $J(x_{k+1})$ 11 update λ_k 12 $k \leftarrow k+1$ 13 end 14 end Result: x_k </pre>

Algorithm 1: Inexact LevenbergMarquardt Algorithm

³Refer to the link: https://en.wikipedia.org/wiki/Dual_number

Chapter 4

Evaluation

Our experiments were conducted on both synthetic data and real data from the micro-CT scans. We will show the main comparison results on synthetic data and few qualitative examples on real data. Our goal is to find a better validation method for comparing different vessel centerline extraction algorithms. In Section 4.1, we give both quantitative and qualitative results on synthetic data. In Section 4.2, we also demonstrate the effectiveness of our proposed method on real data.

4.1 Synthetic Vessel Volume

We used the modification¹ of a method generating synthetic 3D vessel tree data [18]. The generated data consists of CT-like volume and ground truth vessel centerline tree, see Figure 4.1 for an example. We generate 15 artificial volumes $100 \times 100 \times 100$ containing synthetic vascular trees with voxel intensities in the range 0 to 512. The size of each voxel is 0.046 mm. We use three different levels of additive Gaussian noise [28] with standard deviations 5, 10 and 15.

Evaluation setup. Our evaluation system follows [32], see Figure 4.2. We first apply the Frangi filter [16] with hyperparameters $\alpha = 0.5$, $\beta = 0.5$, $\gamma = 30$, $\sigma_{min} = 0.023$ mm and $\sigma_{max} = 0.1152$ mm. The filter computes a *tubularness measure* and estimates the tangent l_p at each voxel p . Then we threshold the tubularness measure to remove background pixels. Then we use non-maximum suppression² (NMS), resulting in voxel set Ω . We use a 26-connected neighborhood system N . Next, we optimize our new joint energy (3.6) to disambiguate tangent orientation and estimate centerline location, see Section 3.4. The hyperparameters are $\gamma = 3.80$ (see energy (3.1)), $\lambda = 18.06$ (see energy (3.6)), $\tau = \cos 70^\circ$ (see equation (3.2)), and the

¹The implementation of [18] contains bugs, which were fixed.

²The use of NMS is mainly for data reduction. Our method is able to work on thresholded data directly, see Figure 3.2(b).

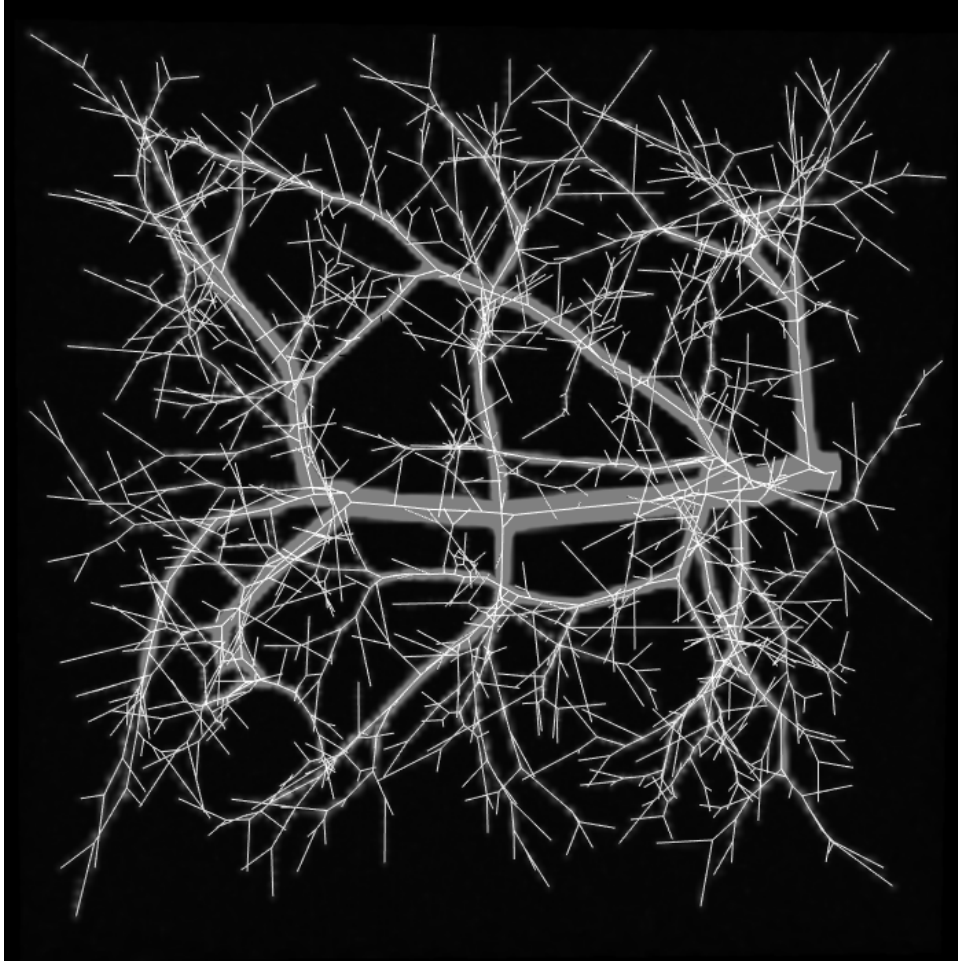


Figure 4.1: An example of one volume synthetic data. The white lines inside vessels denote ground truth of centerline.

maximum number of iterations is 1500 for both TRW-s and Levenberg-Marquardt. Finally, we extract oriented vessel tree centerline as the minimum spanning tree of the complete graph.

Energy (3.6) assumes a quadratic curvature term (2.11). However, if we replace it with (2.9), we can get an absolute curvature variant of our energy.

We evaluate different regularization methods including energy (2.12) (QuaCurv), energy (3.6) with either quadratic curvature (OriQuaCurv) or absolute curvature (OriAbsCurv) within the system outline above. We also compare to a tracing method [3] and medial axis [4].

We adopt *receiver operating characteristic* (ROC) curve methodology for evaluation of our methods and [4]. We compute *recall* and *fall-out* statistics of an extracted vessel tree for different levels of the threshold. The computed statistics define the ROC curve.

While ground truth is defined by locations at bifurcations and leaves of the tree, all evaluated methods yield densely sampled points on the tree. Therefore, we resample both ground

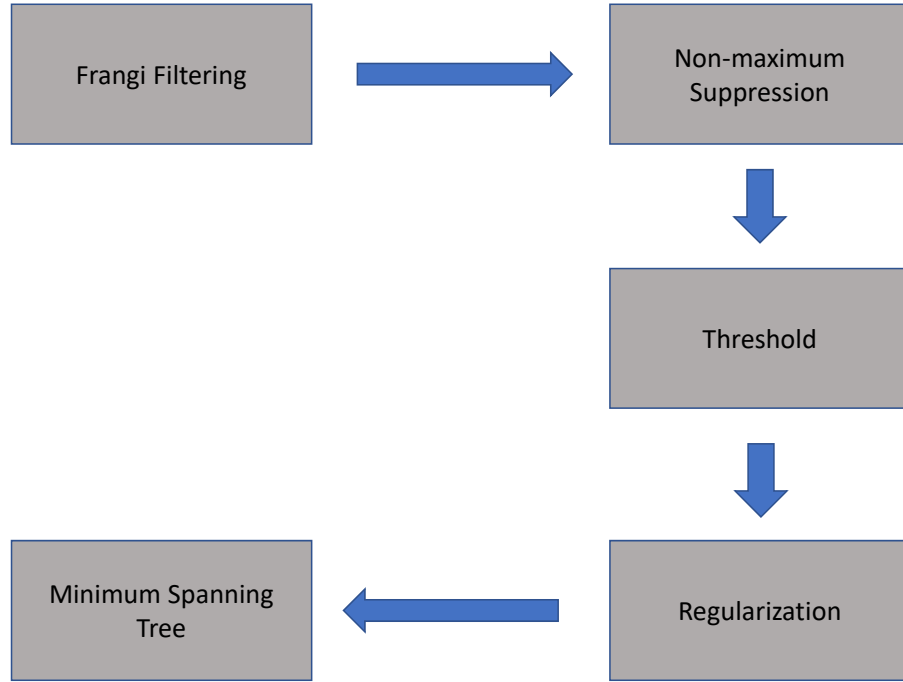


Figure 4.2: Experimental flow diagram. We mainly compares regularization-based methods.

truth and reconstructed tree with step size 0.0023 mm. For each point on one tree, we find the nearest point on the other tree and compute the Euclidean distance. If the distance is less than $\max(r, c)$ voxels, this pair of points is considered a *match*. Here r is the vessel radius at the corresponding point of the ground truth and $c = 0.7$ is a matching threshold measured in voxels. The recall is

$$\frac{N_{GTmatch}}{N_{GTtotal}}$$

where $N_{GTmatch}$ is the number of matched points in the ground truth and $N_{GTtotal}$ is the total number of points in the ground truth. The fall-out is

$$1 - \frac{N_{RTmatch}}{N_{RTtotal}}$$

where $N_{RTmatch}$ is the number of matched points in the reconstructed tree and $N_{RTtotal}$ is the total number of points in the reconstructed tree.

The tracing method of [3] requires a seed points list as an input. We generate four seed lists as described in Figure 4.3. The ROC curves in Figure 4.3 favor our method. Note that the left vertical parts of the ROC curves shift a bit right. This can be explained as follows. With higher threshold on Frangi outputs, fewer points remain and this results in the decrease

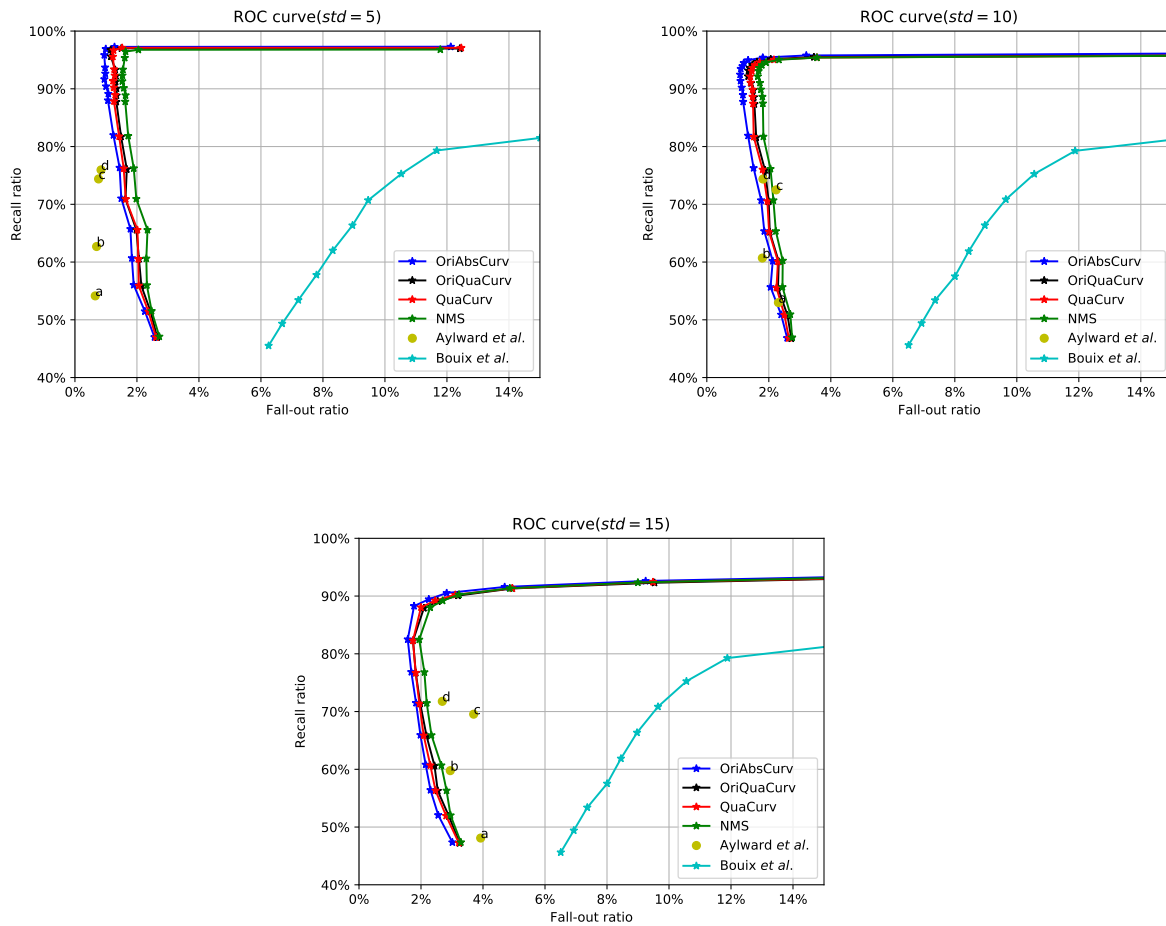


Figure 4.3: Comparison of our method (OriAbsCurv and OriQuaCurv) with the unoriented quadratic curvature (QuaCurv) [32], non-maximum suppression (NMS) based on Frangi outputs, SegmentTubes (Aylward *et al.* [3]) and medial axis extraction (Bouix *et al.* [4]) at three different noise levels. The four letters on yellow dots denote different seed point lists: *a*: using root and all leaf points; *b*: using 50% of the mixture of all bifurcation and leaf points and root; *c*: using middle points of all branch segments; *d*: using all bifurcation and leaf points and root.

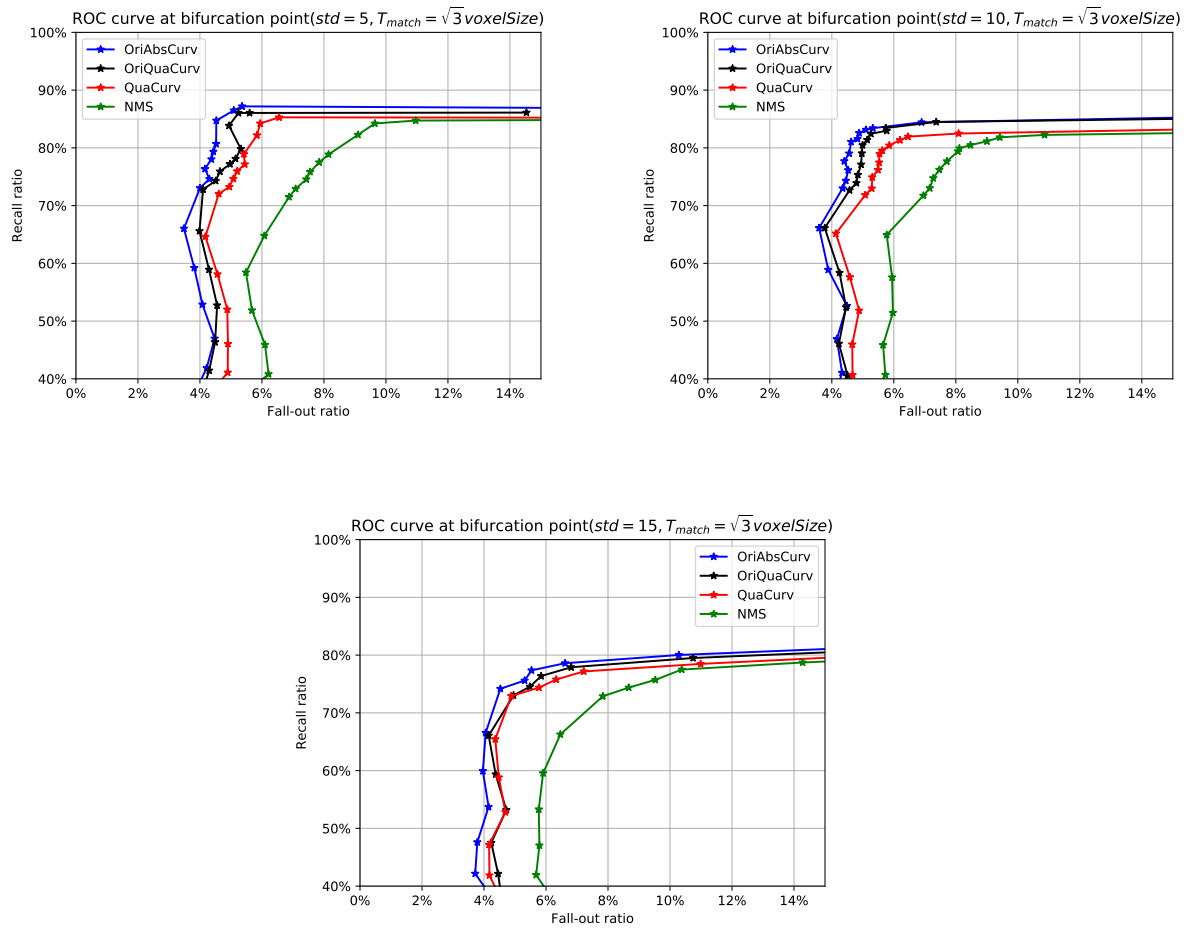


Figure 4.4: Comparison only at bifurcation point.

of “Recall ratio” on the vertical parts of the curves. However, in the final stage where we use MST to extract the tree, some improper connectivities of the tree show up because the graph becomes sparse due to fewer points. These connectivities result in an increase of the “Fall-out ratio”, which makes the curves shift right.

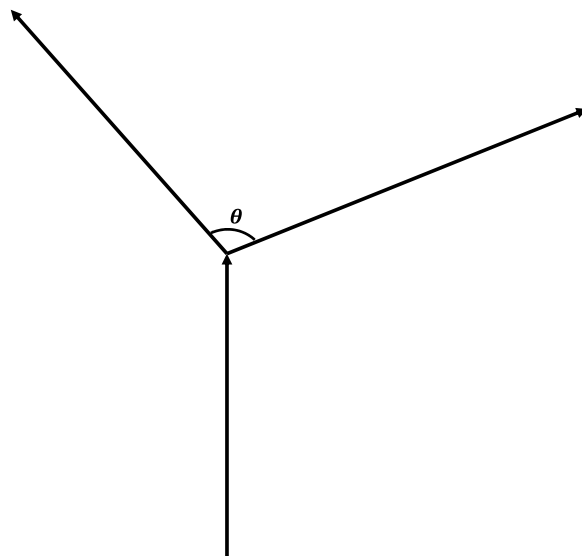
Since bifurcations are only a small fraction of the data, the improvements around bifurcations are largely unnoticed in these curves. Therefore, we compute the ROC curves for only bifurcation nodes. We use a bigger matching threshold $c = \sqrt{3}$ voxels. The results are shown in Figure 4.4 where the gap between methods is bigger. Our proposed method better reconstructs the vessel bifurcation which is very critical for diagnostic analysis.

In addition, we compute angle errors at bifurcations. In order to compute the angle error, we first define the ground-truth angle as the one between two outgoing branches at the bifurcation point, see Figure 4.5(a). This can be easily obtained from our ground-truth tree which provides flow orientation. Since our proposed method provides information of the orientation representing the flow direction³, we can define the corresponding angle at the reconstructed bifurcation point. As shown in Figure 4.5(b), we first calculate three inner products $\langle \vec{BO}, \vec{l}_O \rangle$, $\langle \vec{BP}, \vec{l}_P \rangle$ and $\langle \vec{BQ}, \vec{l}_Q \rangle$. In our case, the largest two values are $\langle \vec{BO}, \vec{l}_O \rangle$ and $\langle \vec{BP}, \vec{l}_P \rangle$ as these two are positive while the other one is negative. Then we have BO and BP as the two outgoing branches which define the angle at the reconstructed bifurcation. The results are shown in Figure 4.6 as well as a few representative examples in Figure 4.7.

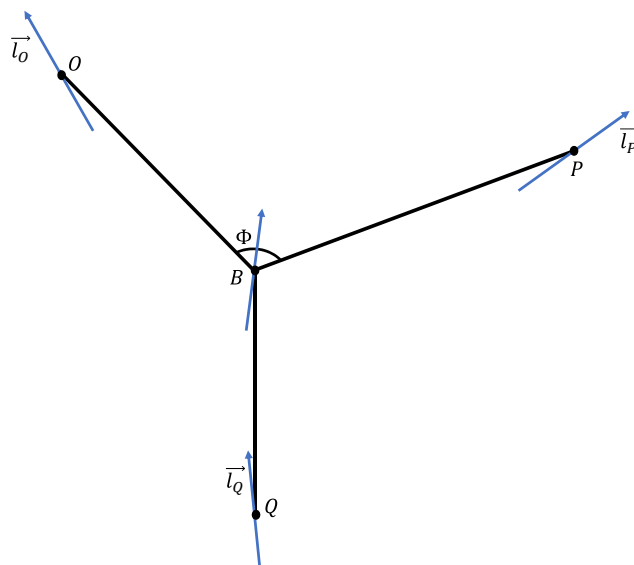
4.2 Real Vessel Data

We obtained qualitative experimental results using a real micro-CT scan of a mouse’s heart as shown in Figure 4.8. The size of the volume is $585 \times 525 \times 892$ voxels. Most of the vessels are thinner than the voxel size. Due to the size of the volume, the problem has higher computational cost than in Section 4.1, we used modified GPU-implementation of the Inexact Levenberg-Marquardt algorithm to handle the large volume size [10]. Figure 4.9 shows the reconstructed centerline. Figure 4.10 demonstrate a significant improvement of centerline estimation around bifurcations.

³We use the orientation provided by oriented model for unoriented one.



(a) Ground truth bifurcation



(b) Reconstructed bifurcation

Figure 4.5: Illustration of the angle at bifurcation point. (a) shows the ground truth where the flow directions are well-defined. The angle θ is used as the ground truth angle of bifurcation. In (b), the blue vectors represent the oriented tangents estimated by minimizing the energy (3.6). The connections BO , BP and BQ are given by MST. The absolute difference of angle θ and ϕ is the angle error.

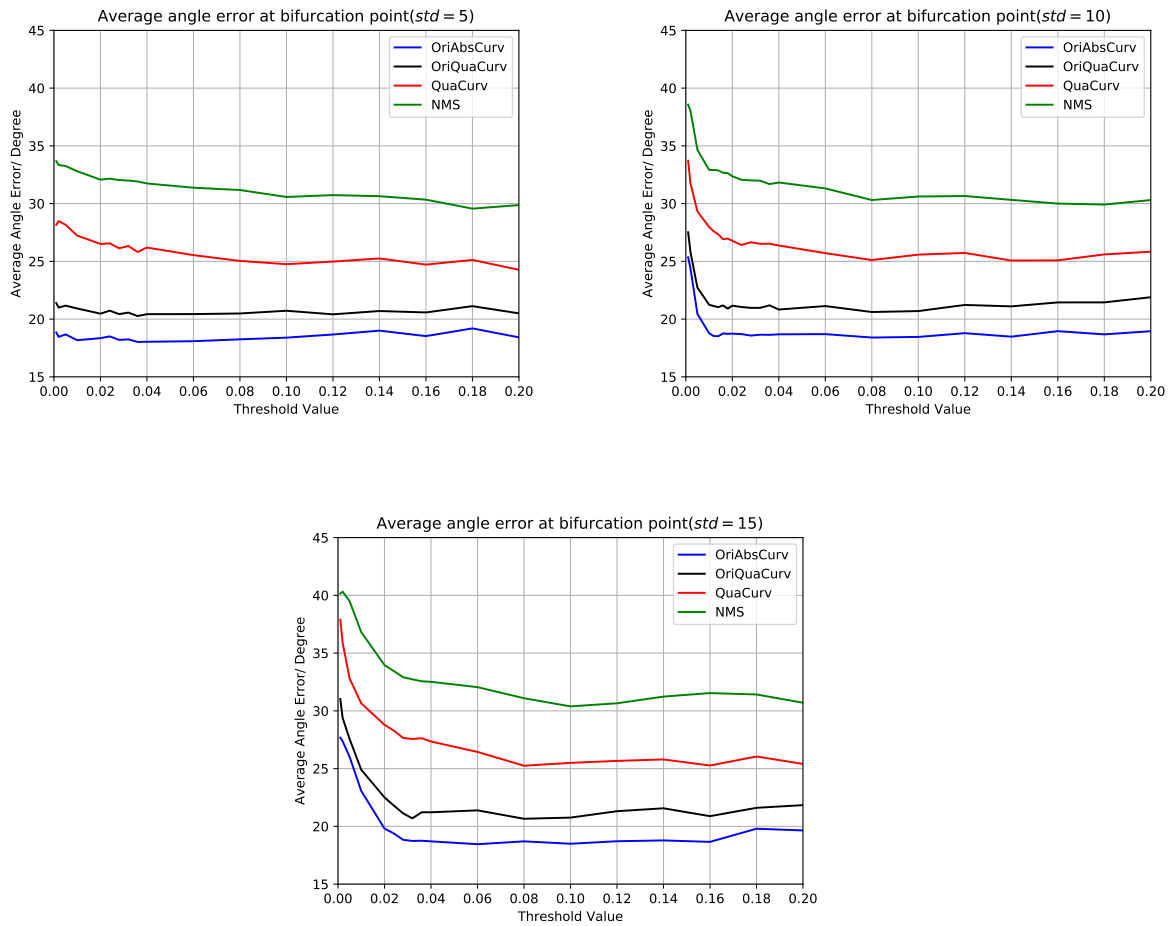


Figure 4.6: Angle error comparison.

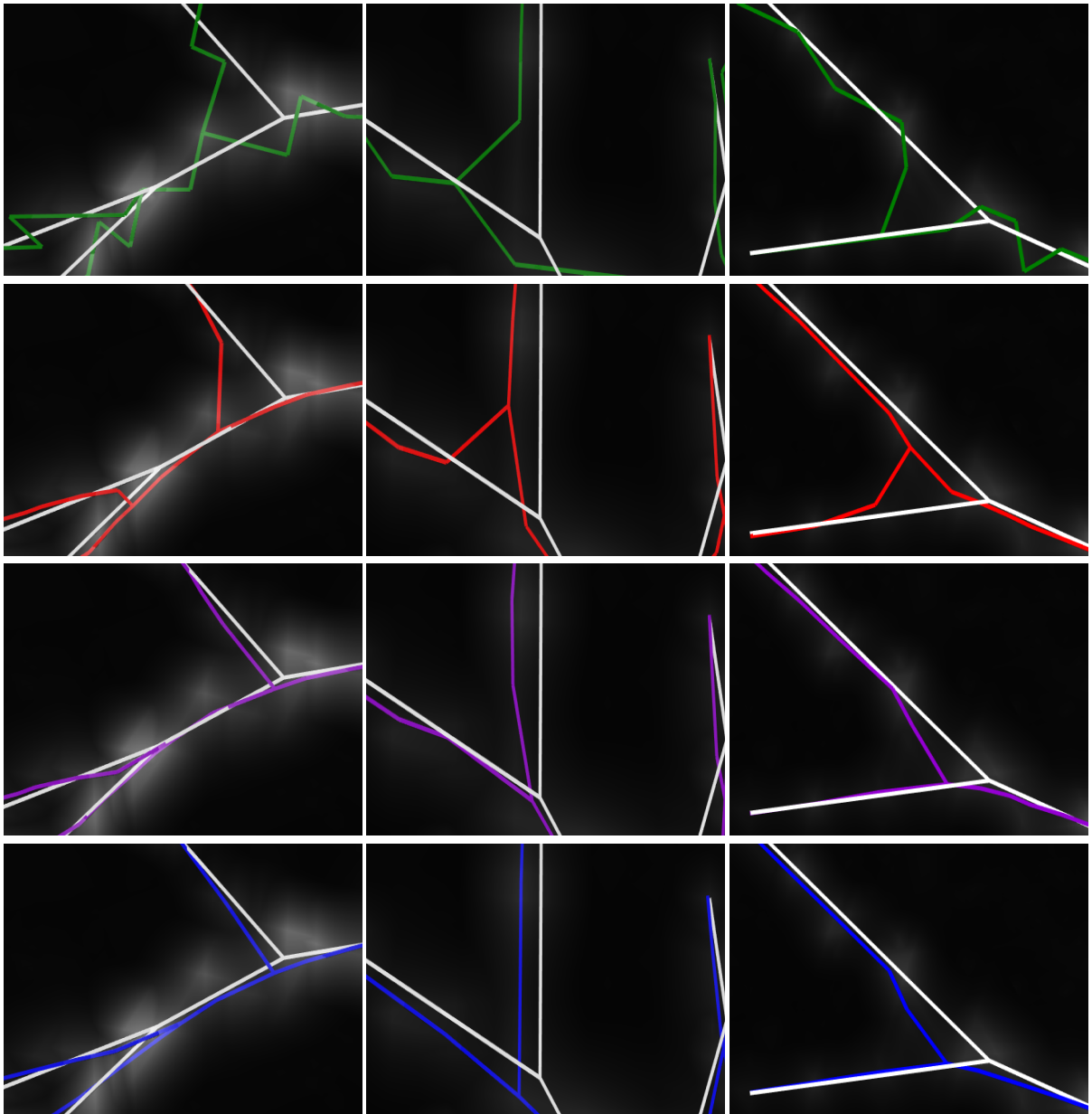


Figure 4.7: Examples of the results around bifurcations with regularization methods. White line is the ground truth tree. A tree extracted from NMS output directly (without regularization) is shown in the first row. Solution of (2.12) [32] is shown in second row. Our model (3.6) is shown in third row. Our model (3.6) with absolute curvature is shown in the last row.

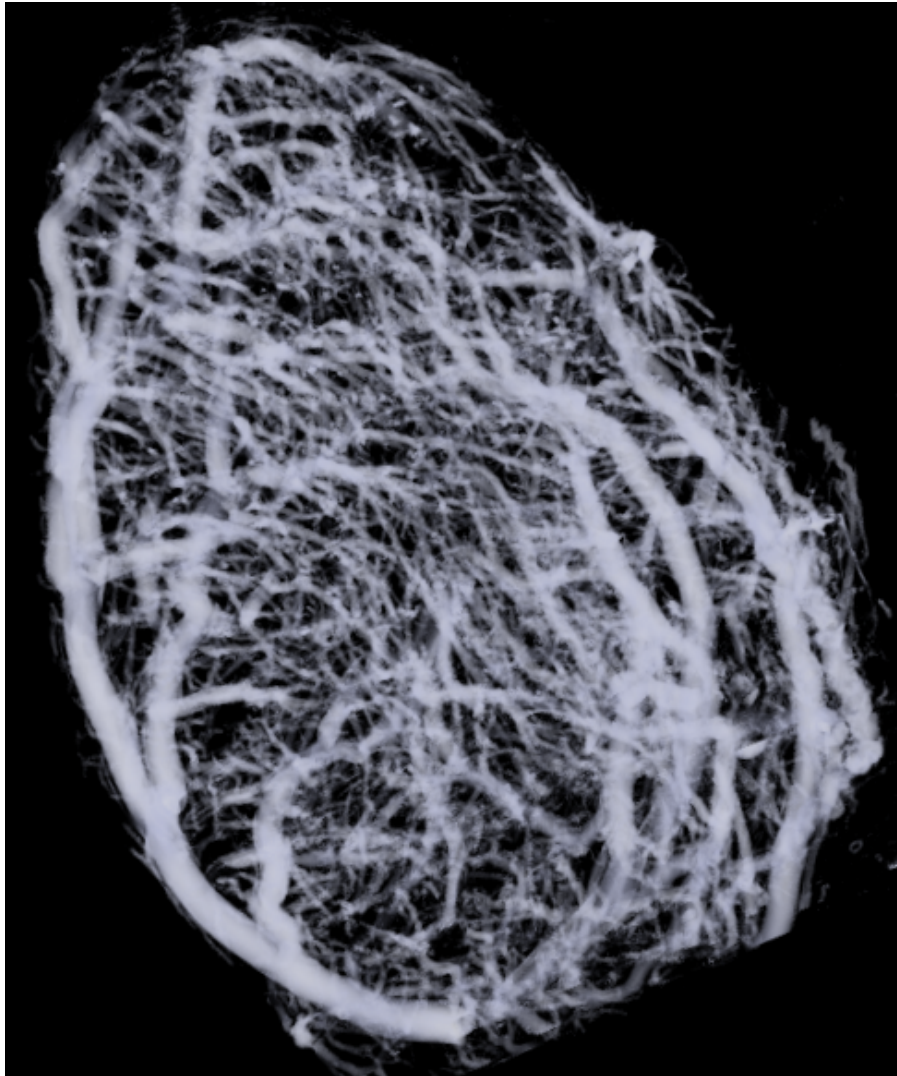


Figure 4.8: Visualization (MIP) of the volumetric data after applying Frangi filter.

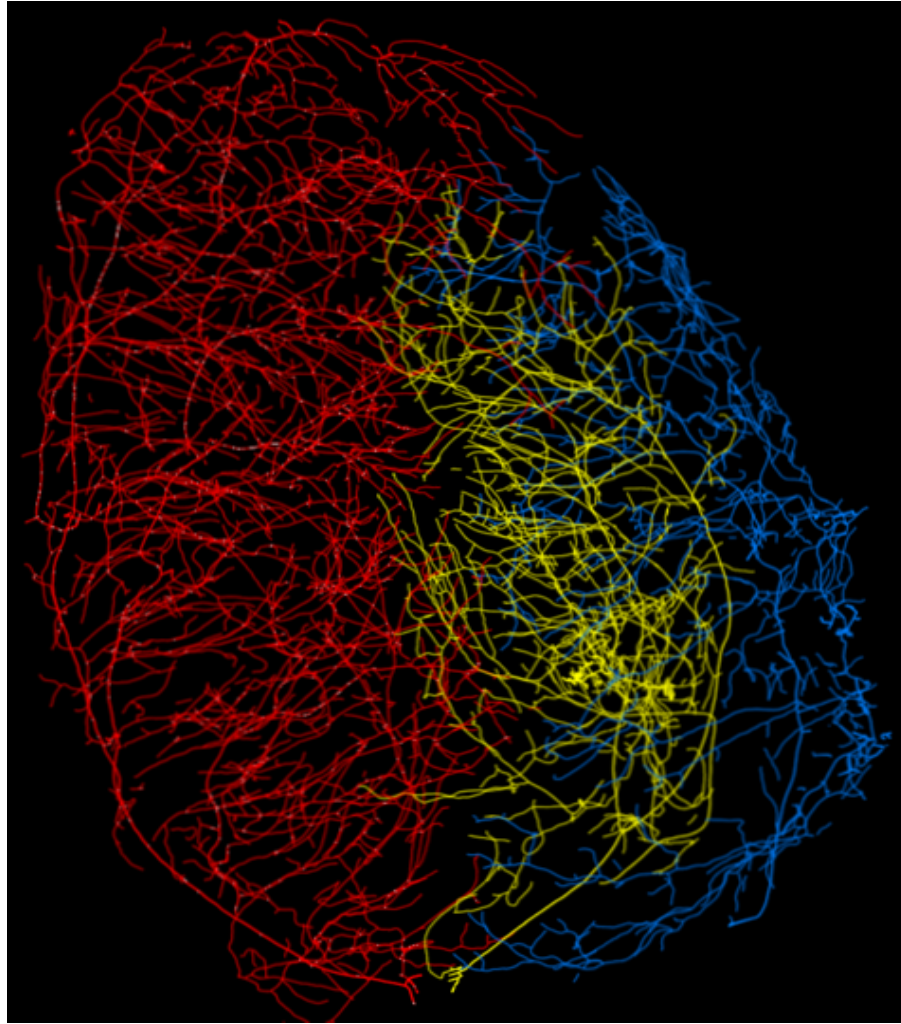


Figure 4.9: Vessel tree reconstruction from real data in Figure 4.8 based on our method for estimating centerline tangents using prior knowledge about vessel divergence. The final tree structure is extracted by MST on K-nearest-neighbour (KNN) weighted graph with edge weights w_{pq} defined as the average *arc-length* between neighbors p and q for two circles containing p and q and tangential to either l_p or l_q . Three different colors (red, blue, yellow) denote three main branches.

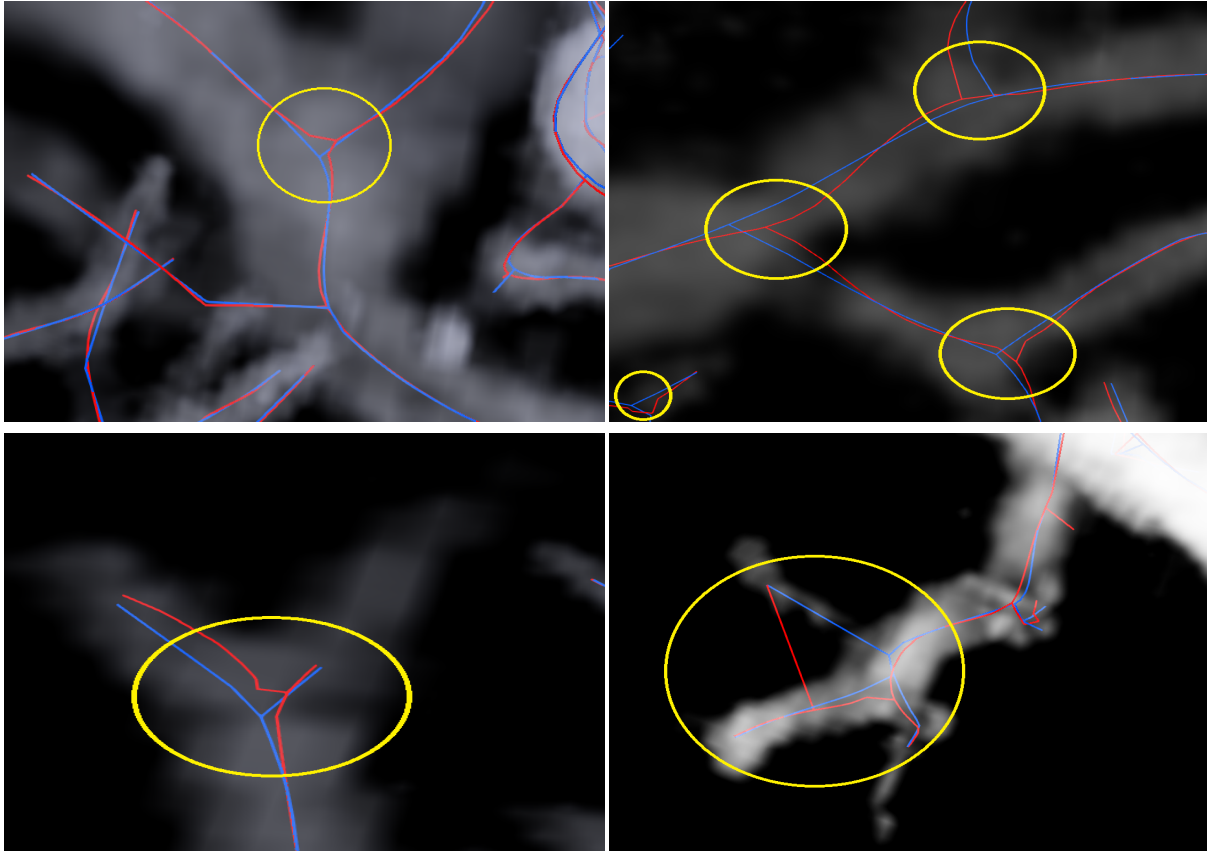


Figure 4.10: Qualitative comparison results on real data. Red represents the result of [32], see (2.12). Blue curve is the centerline obtained based on our directed vessel estimation model (3.6) with divergence prior and absolute curvature regularization. The yellow circles highlight improvements at bifurcations due to correct estimation of the flow direction.

Chapter 5

Conclusion and Future Work

We propose a divergence prior for vector field reconstruction problems. In the context of vessel tree estimation, we use the divergent vessel prior to estimate vessel directions disambiguating orientations produced by the Frangi filter. Our method significantly improves the accuracy of reconstruction at bifurcations, reducing the corresponding angle estimation errors by about 50%.

There are interesting extensions for our work on estimating vessel orientations. For example, such orientations can be directly used for extracting vessel tree topology or connectivity. Instead of using standard MST on undirected graphs, *e.g.* as in [32], we can now use Chu-Liu-Edmonds algorithm [11, 13] to compute a minimum spanning *arborescence* (a.k.a. directed rooted tree) on a directed weighted graph where a weight of any edge (p, q) estimates the length of a possible direct “vessel” connection specifically from p to q . Such a weight can estimate the *arc length* from p to q along a unique circle such that it contains p and q .

Bibliography

- [1] Tao D Alter and Ronen Basri. Extracting salient curves from images: An analysis of the saliency network. *IJCV*, 27(1):51–69, 1998.
- [2] Luis Alvarez, Pierre-Louis Lions, and Jean-Michel Morel. Image selective smoothing and edge detection by nonlinear diffusion. ii. *SIAM Journal on numerical analysis*, 29(3):845–866, 1992.
- [3] Stephen R Aylward and Elizabeth Bullitt. Initialization, noise, singularities, and scale in height ridge traversal for tubular object centerline extraction. *IEEE transactions on medical imaging*, 21(2):61–75, 2002.
- [4] Sylvain Bouix, Kaleem Siddiqi, and Allen Tannenbaum. Flux driven automatic centerline extraction. *Medical image analysis*, 9(3):209–221, 2005.
- [5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11), November 2001.
- [6] Kristian Bredies, Thomas Pock, and Benedikt Wirth. Convex relaxation of a class of vertex penalizing functionals. *Journal of Mathematical Imaging and Vision*, 47(3):278–302, 2013.
- [7] Paige. Christopher C and Saunders. Michael A. Lsqr: An algorithm for sparse linear equations and sparse least squares. *ACM transactions on mathematical software*, 8(1):43–71, 1982.
- [8] Tony F Chan and Jianhong Shen. Nontexture inpainting by curvature-driven diffusions. *Journal of Visual Communication and Image Representation*, 12(4):436–449, 2001.
- [9] C. Chekuri, S. Khanna, J. Naor, and L. Zosin. Approximation algorithms for the metric labeling problem via a new linear programming formulation. 2000.

- [10] Egor Chesakov. Vascular tree structure: Fast curvature regularization and validation. *Electronic Thesis and Dissertation Repository. The University of Western Ontario*, (3396), 2015. Master of Science thesis.
- [11] Y. J. Chu and T. H. Liu. On the shortest arborescence of a directed graph. *Science Sinica*, 14:1396–1400, 1965.
- [12] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numer. Math.*, 1(1):269–271, December 1959.
- [13] Jack Edmonds. Optimum branchings. *J. Res. Nat. Bur. Standards*, 71B(4), October–December 1967.
- [14] Andinet Enquobahrie, Luis Ibanez, Elizabeth Bullitt, and Stephen Aylward. Vessel enhancing diffusion filter. *The Insight Journal*, 1:1–14, 2007.
- [15] L. M. J. Florack et al. Scale and the differential structure of images. *Imag. and Vis. Comp.*, 10(6):376–388, July/August 1992.
- [16] Alejandro F Frangi, Wiro J Niessen, Koen L Vincken, and Max A Viergever. Multiscale vessel enhancement filtering. In *MICCAI'98*, pages 130–137. Springer, 1998.
- [17] Gideon Guy and Gérard Medioni. Inferring global perceptual contours from local features. In *CVPR*, 1993.
- [18] Ghassan Hamarneh and Preet Jassi. Vascusynth: simulating vascular trees for generating volumetric image data with ground-truth segmentation and tree analysis. *Computerized medical imaging and graphics*, 34(8):605–616, 2010.
- [19] Stefan Heber, Rene Ranftl, and Thomas Pock. Approximate envelope minimization for curvature regularity. In *ECCV*, 2012.
- [20] J. Hsieh. Computed tomography: principles, design, artifacts, and recent advances. *SPIE*, 2009.
- [21] Hiroshi Ishikawa. Exact optimization for markov random fields with convex priors. *PAMI*, 25(10):1333–1336, October 2003.
- [22] Timothy L Kline, Mair Zamir, and Erik L Ritman. Accuracy of microvascular measurements obtained from micro-ct images. *Annals of biomedical engineering*, 38(9):2851–2864, 2010.

- [23] J. J. Koenderink. The structure of images. *Biol. Cybern.*, 50:363–370, 1984.
- [24] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *PAMI*, 26(2):147–159, February 2004.
- [25] Vladimir Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *PAMI*, 28(10):1568–1583, 2006.
- [26] A. Koster, C. P. M. van Hoesel, and A. W. J. Kolen. The partial constraint satisfaction problem: Facets and lifting theorems. *Operation Research Letters*, 23(3–5):89–97, 1998.
- [27] Joseph B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. 7(1):48–50, February 1956.
- [28] G Lehmann. Noise simulation. *The Insight Journal*, January-June 2010.
- [29] K Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics* 2, pages 164–168, 1944.
- [30] Gang Li and Steven W Zucker. Differential geometric inference in surface stereo. *PAMI*, 32(1):72–86, 2010.
- [31] T. Lindeberg. Edge detection and ridge detection with automatic scale selection. pages 465–470, June 1996.
- [32] Dmitrii Marin, Yuchen Zhong, Maria Drangova, and Yuri Boykov. Thin structure estimation with curvature regularization. In *International Conference on Computer Vision (ICCV)*, Santiago, Chile, December 2015.
- [33] D. W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial Applied Mathematics*, 11(2):431–441, 1963.
- [34] Sara Moccia, Elena De Momi, Sara El Hadji, and Leonardo Mattos. Blood vessel segmentation algorithms — review of methods, datasets and evaluation metrics. *Computer Methods and Programs in Biomedicine*, 158:71–91, 2018.
- [35] Parya MomayyezSiahkal and Kaleem Siddiqi. 3d stochastic completion fields for mapping connectivity in diffusion mri. *PAMI*, 35(4):983–995, 2013.
- [36] Richard D. Neidinger. Introduction to automatic differentiation and matlab object-oriented programming. *SIAM Review*, 52(3):545–563, 2010.

- [37] C. Nieuwenhuis, E. Toeppe, L. Gorelick, O. Veksler, and Y. Boykov. Efficient squared curvature. In *CVPR*, Columbus, Ohio, 2014.
- [38] Carl Olsson and Yuri Boykov. Curvature-based regularization for surface approximation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1576–1583. IEEE, 2012.
- [39] Carl Olsson, Johannes Ulén, and Yuri Boykov. In defense of 3d-label stereo. In *CVPR*, pages 1730–1737. IEEE, 2013.
- [40] Carl Olsson, Johannes Ulén, Yuri Boykov, and Vladimir Kolmogorov. Partial enumeration and curvature regularization. In *ICCV*, pages 2936–2943. IEEE, 2013.
- [41] T. Schoenemann, F. Kahl, and D. Cremers. Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In *ICCV*, Kyoto, 2009.
- [42] Thomas Schoenemann, Fredrik Kahl, Simon Masnou, and Daniel Cremers. A linear framework for region-based image segmentation and inpainting involving curvature penalization. *IJCV*, 2012.
- [43] Kaleem Siddiqi and Stephen Pizer. *Medial representations: mathematics, algorithms and applications*, volume 37. Springer Science & Business Media, 2008.
- [44] Petter Strandmark and Fredrik Kahl. Curvature regularization for curves and surfaces in a global optimization framework. In *EMMCVPR*, pages 205–218. Springer, 2011.
- [45] M.J. Wainwright, T.S. Jaakkola, and A.S. Willsky. Map estimation via agreement on (hyper)trees: Message-passing and linear-programming approaches. *IEEE Transactions on Information Theory*, 51(11):3697–3717, November 2005.
- [46] Lance R Williams and David W Jacobs. Stochastic completion fields: A neural model of illusory contour shape and salience. *Neural Computation*, 9(4):837–858, 1997.
- [47] Oliver Woodford, Philip Torr, Ian Reid, and Andrew Fitzgibbon. Global stereo reconstruction under second-order smoothness priors. *PAMI*, 31(12):2115–2128, 2009.
- [48] SJ Wright and John Norman Holt. An inexact levenberg-marquardt method for large sparse nonlinear least squares. *The Journal of the Australian Mathematical Society. Series B. Applied Mathematics*, 26(04):387–403, 1985.

Curriculum Vitae

Name: Zhongwen Zhang

**Post-Secondary
Education and
Degrees:** Zhejiang University
China
2012 - 2016 B.E.

University of Western Ontario
London, ON
2017 - 2018 M.S.

**Honours and
Awards:** Western Graduate Research Scholarship
2017-2018

**Related Work
Experience:** Teaching Assistant
The University of Western Ontario
2017 - 2018

Research Assistant
The University of Western Ontario
2017 - 2018