Vascular Segmentation in Magnetic Resonance Angiography

by

Wai Kong Law

A Thesis Submitted to
The Hong Kong University of Science and Technology
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy
in Computer Science and Engineering

July 2010, Hong Kong

Copyright © by Wai Kong Law 2010
Authorization

I hereby declare that I am the sole author of the thesis.

I authorize the Hong Kong University of Science and Technology to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize the Hong Kong University of Science and Technology to reproduce the thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

WAI KONG LAW
VASCULAR SEGMENTATION IN MAGNETIC RESONANCE ANGIOGRAPHY

by

WAI KONG LAW

This is to certify that I have examined the above Ph.D. thesis and have found that it is complete and satisfactory in all respects, and that any and all revisions required by the thesis examination committee have been made.

DR. ALBERT C. S. CHUNG, THESIS SUPERVISOR

PROF. MOUNIR HAMDI, HEAD OF DEPARTMENT

Department of Computer Science and Engineering

22 July 2010
ACKNOWLEDGMENTS

I would like to thank:

My supervisor, Dr. Albert C. S. CHUNG for supervising, supporting, enlightening and inspiring me to conduct research in the past few years;

Professor Pheng Ann HENG, Dr. Chiew-Lan TAI, Dr. Pedro SANDER and Dr. George Jie YUAN, who serve as my thesis examination committee members;

K.S. Lo Foundation for Financial Support;

All members of K.S. Lo Medical Image Analysis Laboratory, Wilbur C.K. Wong, Rui GAN, John J. WU, Billy S. LIAO, Vincent W. CAI, Vincent Y. YUAN, Tommy W.H. TANG, Ronald W.K. SO, Lisa Y.S. LUO and Joanna N. ZHU.
# TABLE OF CONTENTS

Title Page  
Authorization Page  
Signature Page  
Acknowledgments  
Table of Contents  
List of Figures  
List of Tables  
Abstract  

Chapter 1  Introduction  
  1.1 Background  
  1.2 Literature review  
    1.2.1 Probability models  
    1.2.2 Geometric models  
    1.2.3 Active contour models  

Chapter 2  Weighted Local Variance Based Edge Detection and Vascular Segmentation  
  2.1 Introduction  
  2.2 Methodology  
    2.2.1 Edge detection and weighted local variance  
    2.2.2 Properties  
    2.2.3 Computing edge normal orientation and edge strength  
    2.2.4 Implementation  
    2.2.5 Segmentation using active contour model  
  2.3 Experiments  
    2.3.1 Synthetic image volumes
4.2.8 The oriented flux symmetry based active contour model 103
4.2.9 Fourier expressions of the oriented flux measure and the oriented flux antisymmetry measure 105

4.3 Experimental Results 107
4.3.1 Synthetic data 108
4.3.2 Application example - blood vessel extraction 112
4.3.3 Active contour segmentation on synthetic data 113
4.3.4 Active contour segmentation on real vascular images 115

4.4 Perspectives 119

Chapter 5 Conclusion and future development 124

Appendix A Relationship between confidence value and the smoothed image gradient 128

Appendix B Incorporating WLV-EDGE in the Hessian matrix 130

References 131
LIST OF FIGURES

1.1 An example illustrating the difference between the boundaries shown without and with the subvoxel accuracy. The boundaries are shown along with the discrete image grid to illustrate the image resolution. From left to right, the original circular structure with an intensity range $[-1, 1]$; the intensity of the structure after being discretized on the image grid; the zero-intensity boundary of the discretized structure, the boundary is rendered without the subvoxel accuracy; the zero-intensity boundary of the discretized structure, the boundary is rendered with the subvoxel accuracy.

2.1 The plots of the values of $\sqrt{\text{WLV}_1(n(\theta))}$ (second column), $\sqrt{\text{WLV}_2(n(\theta))}$ (third column), and $\min(\sqrt{\text{WLV}_1(n(\theta))}, \sqrt{\text{WLV}_2(n(\theta))})$ (forth column) against different values of $\theta$ obtained from horizontal edges having different levels of intensity contrast, a corner and two edges of different values of curvature.

2.2 (a) An example of 24 discrete 2D orientation samples for calculation of confidence values. (b) An example of 281 discrete 3D orientation samples for calculation of confidence values.

2.3 Single synthetic tube. (a) The 50th slice of the image volume, $z = 50$. (b) The 12th slice of the image volume, $x = 12$. (c) From top to bottom: The intensity profile of the tube along the $z$-direction with $x = 12$ and $y = 10$; The square root of the trace of the structure tensor computed by $\text{ST}$; The $\text{GRADIENT}$ magnitude; The edge strength computed by $\text{WLV}$. (d) Along the $z$-direction with $x = 12$, it shows the slices of edge strengths detected by (from top to bottom) $\text{ST}; \text{GRADIENT}; \text{WLV}$.

2.4 Multiple synthetic tubes. (a) The 50th slice of the image volume ($181 \times 217 \times 181$ voxels) consists of 25 synthetic tubes of radii equal to 2, 3, 4, 5 and 6. (b) An example of the ground truth normal orientations on the tube surfaces.

2.5 (a) The 50th slice of the bias field $A$ obtained from the Brainweb [1]. (b) The 50th slice of the image volume after applying the bias field model $A$ without noise. (c) The 50th slice of the image volume after applying the bias field model $A$ and the additive Gaussian noise with the scale parameter equal to 0.05.

2.6 (a) The 50th slice of the bias field $B$ obtained from the Brainweb [1]. (b) The 50th slice of the image volume after applying the bias field model $B$ without noise. (c) The 50th slice of the image volume after applying the bias field model $B$ and the additive Gaussian noise with the scale parameter equal to 0.05.
2.7 (a) The 50th slice of the bias field $C$ obtained from the Brainweb [1]. (b) The 50th slice of the image volume after applying the bias field model $C$ without noise. (c) The 50th slice of the image volume after applying the bias field model $C$ and the additive Gaussian noise with the scale parameter equal to 0.05.

2.8 The third synthetic and numerical volume consists of 18 non-overlapping tori. The size of each torus is defined by two parameters $r$ and $R$ as shown in (d). (d) The description of the setting of a torus. $r$ specifies the shortest distance between the centerline of the torus (denoted as the dashed line inside the torus) and the surface of the torus. $R$ determines the radius of the torus centerline. $\hat{n}_1$ and $\hat{n}_2$ are two examples of the ground truth normal orientations of the torus surface. They are pointing outward from the torus centerline to the surface. Using the convention $\{r, R\}$, the sizes of all tori (in voxel) are $\{2, 12\}$, $\{2, 24\}$, $\{2, 36\}$, $\{2, 48\}$, $\{2, 60\}$, $\{2, 72\}$, $\{3, 18\}$, $\{3, 32\}$, $\{3, 46\}$, $\{3, 60\}$, $\{3, 76\}$, $\{4, 24\}$, $\{4, 48\}$, $\{4, 64\}$, $\{5, 48\}$, $\{5, 66\}$, $\{6, 54\}$ and $\{6, 72\}$. (a) A top-view of the torus iso-surfaces. (b) An elevation-view of the torus iso-surfaces. (c) A side-view of the torus iso-surfaces. In this subfigure, there are five layers. The top layer has two tori of $r = 6$, the next layer has two tori of $r = 5$, . . ., and the bottom layer has six tori of $r = 2$.

2.9 The MIP images of a TOF-MRA image volume (First row: axial projection; second row: coronal projection and sagittal projection). This TOF-MRA image volume has dimension $188 \times 168 \times 39$ voxels and voxel size $0.4\text{mm} \times 0.4\text{mm} \times 1.0\text{mm}$.

2.10 Segmentation results of WLV-FLUX.

2.11 (a) Segmentation results of FLUX. (b) The 14th and 15th axial slices; (c) The 11th and 12th axial slices of the TOF-MRA image volume and the contours are corresponding to the segmentation results, as shown in (a). The white arrows in (a) are corresponding to the positions pointed at by the white arrows in (b) and (c) where leakages occur.

2.12 The MIP images of a PC-MRA speed image volume (First row: axial projection; second row: coronal projection and sagittal projection). This image volume has dimension $130 \times 286 \times 52$ voxels and voxel size $0.4\text{mm} \times 0.4\text{mm} \times 1.0\text{mm}$.

2.13 Segmentation results of WLV-FLUX.

2.14 Segmentation results of FLUX. (b) The 26th and 27th axial slices; (c) The 40th and 41st axial slices of the PC-MRA image volume and the contours are corresponding to the segmentation results, as shown in (a). The white arrows in (b) and (c) indicate the low contrast vessels, which are missed by FLUX.
2.15 The MIP images of a PC-MRA speed image volume (First row: axial projection; second row: coronal projection and sagittal projection). This image volume has dimension 67 × 257 × 35 voxels and voxel size 0.4mm×0.4mm×0.8mm.

2.16 Segmentation results of **WLV-FLUX**.

2.17 Segmentation results of **FLUX**. (b) The 25th and 26th axial slices; (c) The 29th and 30th axial slices of the PC-MRA image volume and the contours are corresponding to the segmentation results, as shown in (a). The white arrows in (b) and (c) indicate the low contrast vessels, which are missed by **FLUX**.

2.18 The MIP images of a PC-MRA speed image volume (First row: axial projection; second row: coronal projection and sagittal projection). This image volume has dimension 104 × 252 × 64 voxels and voxel size 0.4mm×0.4mm×1.0mm.

2.19 Segmentation results of **WLV-FLUX**.

2.20 Segmentation results of **FLUX**

2.21 Segmentation results of **FLUX**. (a) The 17th and 18th coronal slices; (b) The 25th and 26th coronal slices of the PC-MRA image volume and the contours are corresponding to the segmentation results, as shown in Figure 2.20. The white arrows in (a) and (b) indicate the low contrast vessels, which are missed by **FLUX**.

3.1 An example of the function $\tilde{\Upsilon}([\mu_1, \mu_2, 0]^T; 12)$. (a) The plot of the coefficients for the plane $[\mu_1, \mu_2, 0]^T$ in the Fourier domain. (b) The plot of the coefficients for the line $[\mu_1, 0, 0]^T$ in the Fourier domain. The shaded regions are not covered by the Subband$_1$. (c) The zooming of (b).

3.2 The gray regions represent the coverages of different frequency subbands in the Fourier domain. (a) Subband$_1$; (b) Subband$_{1.5}$; (c) Subband$_2$.

3.3 An example, which shows that $\tilde{\Upsilon}^{-1}\{\tilde{\Upsilon}\}([x, 0, 0]^T; 12)$ is discretized with different frequency subbands. (a, b) $\tilde{\Upsilon}^{-1}\{\Upsilon_{\text{Subband}_1}\}([x, 0, 0]^T; 12)$, (b) is a zooming of (a); (c, d) $\tilde{\Upsilon}^{-1}\{\Upsilon_{\text{Subband}_{1.5}}\}([x, 0, 0]^T; 12)$, (d) is a zooming of (c).

3.4 Flow charts of the computation of the multiscale spherical flux. (a) The conventional spatial implementation. (b) The proposed implementation. $L$ represents a set of scales. $\sigma$ represents the scale parameter of the Gaussian function being applied on the image $I(\vec{x})$. 

x
3.5 The synthetic and numerical image volumes. (a) The isosurfaces of the synthetic and numerical tubes. (b) The slice of the tube image volume at \( z = 20 \). The tubes have radii 1, 2, 4, 6 and 8 voxels, and intensity values 0.6, 0.7, 0.8, 0.9 and 1. (c) The isosurfaces of the synthetic and numerical tori. (d) Top: A vertical cross section of a torus; bottom: the slice of the torus image volume at \( y = 90 \). The intensity values of the tori are 0.8 for the gray tori and 1 for the white tori. Expressing the sizes of tori in voxels using the representation (small radius, large radius), from top to bottom, the tori have sizes (1, 24), (1, 48), (1, 56), (2, 24), (2, 32), (2, 40), (2, 48), (2, 56), (4, 24), (4, 48), (4, 64), (6, 36), (6, 60), (8, 48) and (8, 60).

3.6 Top: The image slices of the noise corrupted synthetic and numerical tubes at \( z = 20 \); bottom: the image slices of the noise corrupted synthetic and numerical tori at \( y = 90 \). From left to right, the image volumes were corrupted by an additive zero mean Gaussian noise with standard deviations 0.01, 0.02, 0.03, 0.05 and 0.1, respectively.

3.7 Top: The isosurfaces of the initial levelset function (the first column) and the segmentation results of noise corrupted synthetic tubes (from the second column to the sixth column). Bottom: The isosurfaces of the initial levelset function (the first column) and the segmentation results of noise corrupted synthetic tori (the second column to the sixth column). From the second column to the sixth column, the segmentation results of the image volumes corrupted by an additive zero mean Gaussian noise with standard deviations 0.01, 0.02, 0.03, 0.05 and 0.1. The results are obtained by applying the flux maximizing geometric flows along with the proposed implementation.

3.8 The plot of the average running times for multiscale spherical flux computation using the conventional spatial implementation against different steps (or scales). The times listed are obtained by averaging the running times of computing multiscale spherical flux in 10 different noise corrupted image volumes (5 for tubes and 5 for tori), which were generated for the segmentation experiments on the synthetic and numerical image volumes, as shown in Figure 3.5.

3.9 The number of orientation samples taken for the conventional computation of spherical flux with various radii.

3.10 The maximum intensity projections of the five clinical PC-MRA image volumes used in the experiments. For each sub-figure, the top-left image is the axial view, the bottom image is the coronal view, and the top-right image is the sagittal view.

3.11 The axial views of the segmentation results of the five clinical PC-MRA image volumes based on FMGF using the proposed implementation. The corresponding maximum intensity projections are shown in Figures 3.10a-e.
3.12 The axial views of the segmentation results of the five clinical PC-MRA image volumes based on **FMGF** using the conventional implementation. The corresponding maximum intensity projections are shown in Figures 3.10a-e.

3.13 The axial views of the vessel extraction results of the five clinical PC-MRA image volumes based on the vesselness measure, proposed by Frangi *et al.* [2]. The corresponding maximum intensity projections are shown in Figures 3.10a-e.

4.1 (a, b, c) The values of $|\vec{W}_{12}(\cdot)|^T \hat{n}$. (d) The intensity scale of the images in (a-c).

4.2 Examples of evaluating OOF using multiple radii. (a, b) The slices of $z = 0$ (left) and $x = 0$ (right) of two synthetic image volumes consisting of synthetic tubes with a radius of 4 voxel length. C1 and C2 are the positions of the centers of the tubes. L1, R1 and L2, R2 are the positions of the boundaries of the tubes centered at C1 and C2 respectively. (b) The width of the separation between the closely located tubes is 2 voxel length. (c, d) The intensity profiles along the line $x = 0, z = 0$ of the synthetic image volumes shown in (a) and (b) respectively. (e, f) The trace of $Q_{x,\hat{\rho}}$ along the line $x = 0, z = 0$ of the image volumes shown in (a) and (b) respectively.

4.3 Illustrations of image gradients which form various patterns. The black arrows and grey solid lines represent image gradients and structure boundaries respectively. (a) Image gradients along a curvilinear structure cross-sectional plane. (b-e) Four examples showing image gradients located at the local spherical region boundaries (black dotted circles), projected along $\hat{\rho}$.

4.4 (a) An xy-plane which shows the cross-section of the synthetic tube with a 4 voxel radius. (b) An xz-plane of the synthetic tube. (c) The numbers represent the intensity of various parts of the image in (b). (d) An xz-plane of the synthetic tube corrupted by additive Gaussian noise with standard deviation 0.1. (e-g) The xz-planes which shows different measures. The black line in (g) showing the boundary where $\max_{\hat{\rho}} |s(\vec{x}; p(\vec{x}), \hat{\rho})|$ is maximal along the vertical directions from the tube center to the image background. (h-i) The profiles of different measures obtained along the lines shown as dotted lines in (c). (j-k) The values of $\Xi(\vec{x})$ and $\max_{\hat{\rho}} |a(\vec{x}; r, \hat{\rho})|$, which are obtained using one radius for each sub-figure.

4.5 The description of the tori. These tori have been used in the synthetic data experiments. The center of the tori in each layer is randomly selected from the positions of $(x = 35, y = 35), (x = 45, y = 35), (x = 35, y = 45)$ and $(x = 45, y = 45)$. The values of $d$ and $R$ are fixed to generate a torus image. In the experiments, there are 10 torus images generated by using 10 pairs of $\{d, R\}, \{2, 1\}, \{2, 2\}, \{2, 3\}, \{2, 4\}, \{2, 5\}, \{5, 1\}, \{5, 2\}, \{5, 3\}, \{5, 4\}$ and $\{5, 5\}$. 

xii
4.6 (a) A phase contrast magnetic resonance angiographic image volume with the size of $213 \times 143 \times 88$ voxels. (b) The vessel extraction results obtained by using the *optimally oriented flux* based method. (c) The vessel extraction results obtained by using the Hessian matrix based method.

4.7 The $80 \times 80 \times 80$ voxels synthetic image used in the synthetic data experiment. (a) the isosurface of the spiral with the isovalue of 0.5; (b) the 15th slice showing the bottom part of the noise corrupted spiral; (c) the 65th slice showing the top part of the noise corrupted spiral; (d) the initial level set function for the segmentation of the spiral.

4.8 The segmentation results of the noise corrupted synthetic spiral by using CURVES, FLUX and the proposed method.

4.9 The image volumes used in the real vascular image experiment. (a, b) The perspective maximum intensity projections, along the axial, the sagittal and the coronal directions of two intracranial PC-MRA volumes; (c) the axial perspective maximum intensity projection (left) and the 53th image slice (right) of an intracranial TOF-MRA volume; (d) The 182th (left) and 214th (right) slices of a cardiac CTA volume. The red circles indicate the aorta and the blue dots are the manually placed initial seed points.

4.10 (a, b, e, f) The segmentation results of the clinical cases shown in Figs. 4.9a, b, c and d respectively, by using CURVES. (c, d, g, h) The segmentation results of the clinical cases shown in Figs. 4.9a, b, c and d respectively, by using FLUX.

4.11 The segmentation results obtained by using the proposed method from the four angiographic images shown in Figs. 4.9a-d.

5.1 An illustration of the partial volume effect occurred on the a tubular structure. Left: A tubular structure shown along with the discrete image grid to illustrate the image resolution. Right: The intensity of the structure after being discretized on the image grid. The contrast between the discretized structure and the image background is varying along the structure.

xiii
LIST OF TABLES

2.1 Standard deviations of edge strengths computed by different methods after applying different bias field models \texttt{FIELDA, FIELDB} and \texttt{FIELDC} on the volume shown in Figure 2.4. The edge strengths computed from each method, \texttt{ST}, \texttt{GRADIENT} and \texttt{WLV-EDGE}, are normalized to have unit mean values.

2.2 Standard deviations of edge strengths computed by different methods after applying different bias field models \texttt{FIELDA, FIELDB} and \texttt{FIELDC} on the volume shown in Figure 2.8. The edge strengths computed from each method, \texttt{ST}, \texttt{GRADIENT} and \texttt{WLV-EDGE}, are normalized to have unit mean values.

2.3 Angular discrepancies (in radian) between ground truth normal orientation and the edge normal orientation estimated by \texttt{ST, GRADIENT} and \texttt{WLV-EDGE} on the surfaces of the synthetic tubes, as shown in Figure 2.4.

2.4 Angular discrepancies (in radian) between ground truth normal orientation and the edge normal orientation estimated by \texttt{ST, GRADIENT} and \texttt{WLV-EDGE} on the surfaces of the synthetic tubes, as shown in Figure 2.8.

3.1 The segmentation accuracies of the synthetic and numerical image volumes, as shown in Figure 3.5, using the proposed implementation based and the conventional spatial implementation based flux maximizing geometric flows. The values obtained by the conventional spatial implementation are enclosed by the brackets. The values for "True positive", "True negative", "False positive" and "False negative" are in voxels.

3.2 (Second and third columns): The total running times (in seconds) for multiscale spherical flux computation with 10 scales based on the proposed implementation and the conventional spatial implementation; (the last column): the mean absolute difference between the multiscale spherical flux values computed by the both implementations. The times listed are obtained by averaging the running times of computing multiscale spherical flux in 10 different noise corrupted image volumes (5 for tubes and 5 for tori), which were generated for the segmentation experiments on the synthetic and numerical image volumes, as shown in Figure 3.5. The values in the second, third and last columns are rounded to 4 significant digits.
3.3 The running times (in seconds) of different steps for multiscale spherical flux computation based on the proposed implementation (top) and the conventional spatial implementation (bottom). The values listed in the table are obtained by averaging the running times of computing multiscale spherical flux in 10 different noise corrupted image volumes (5 for tubes and 5 for tori), which were generated for the segmentation experiments on the synthetic and numerical image volumes, as shown in Figure 3.5. The values in the brackets are the percentages, which are given as \( \left( \frac{\text{PI}}{\text{CSI}} \times 100\% \right) \). The values are rounded to 4 significant digits, except the values enclosed by brackets.

3.4 The resolutions and voxel sizes of the image volumes shown in Figure 3.10.

3.5 (Second and third columns): The running times (in seconds) for multiscale spherical flux computation based on the proposed implementation and the conventional spatial implementation; (the fifth column): the mean absolute differences between the spherical flux values computed by the conventional spatial implementation and the proposed implementation; (the last column) The estimated DSC values based on the segmentation results obtained by using the proposed implementation and the conventional spatial implementation. The values in the second, third, fifth and last columns are rounded to 4 significant digits.

3.6 The running times (in seconds) of different steps for multiscale spherical flux computation based on the proposed implementation (top) and the conventional spatial implementation (bottom). The values in the brackets are the percentages, which are given as \( \left( \frac{\text{PI}}{\text{CSI}} \times 100\% \right) \). The values are rounded to 4 significant digits, except the values enclosed by brackets.

4.1 The analysis of the response magnitudes of various measures obtained at different positions \( \vec{x} \). In the second, the third and the fifth columns, \( \hat{\rho} \) is given as the direction on the structure cross-sectional plane, pointing from object centers to \( \vec{x} \). In the second to the fifth columns except the entries with *, \( r \) is given as the distance from \( \vec{x} \) to the nearest object boundary; in the entries with *, \( r \) is assumed to be a value smaller than the structure radius.

4.2 The performance of optimally oriented flux and the Hessian matrix obtained in the synthetic data experiments. The entries in the columns of ”Angular discrepancy” include two values, the mean and the standard deviation (the bracketed values) of the resultant values of Equation 4.27. The values in the columns of ”Response fluctuation” are the results based on Equation 4.28.

4.3 The changes of mean angular discrepancy and response fluctuation from the case of ”\( d = 5, \sigma_{\text{noise}} = 0.75 \)” to other three cases presented in Table 4.2.
VASCULAR SEGMENTATION IN MAGNETIC RESONANCE ANGIOGRAPHY

by

WAI KONG LAW

Department of Computer Science and Engineering
The Hong Kong University of Science and Technology

ABSTRACT

Clinical assessment of vasculatures is essential for the detection and treatment of vascular diseases which can be potentially fatal. To facilitate clinical assessment of blood vessels, there is a growing need of developing computer assisted vessel segmentation schemes based on magnetic resonance angiographic (MRA) images. A vast number of approaches have been proposed in the past decade for the segmentation of vascular structures in MRA images. These approaches were devised according to different assumptions on the shape of blood vessels and different underlying prior knowledge about the desired imaging modalities. The development of these approaches aims at delivering more accurate and robust segmentation results. Nonetheless, these approaches face different technical challenges that prohibit them from being widely employed in the clinical environment. The challenges include significant contrast variation of vessel boundaries in MRA images, the excessive computation time required by some algorithms and the complicated geometry of vascular structures. These challenges motivate us to propose three novel edge detection and vascular segmentation methods.

In the first proposed method, vessel segmentation is performed grounded on the edge detection responses given by the weighted local variance-based edge detector. This detector is robust against large intensity contrast changes and capable of re-
turning accurate detection responses on low contrast edges. Our second method is an efficient implementation of a well founded vessel detection approach. The proposed efficient implementation is a thousand times faster than the conventional implementation without segmentation performance deterioration. The third method is a curvilinear structure descriptor which is robust against the disturbance induced by closely located objects. Preliminary experimental results show that the proposed methods are very suitable for vascular segmentation in MRA images.
CHAPTER 1

INTRODUCTION

1.1 Background

Clinical assessment of vasculatures is essential for the treatments of vascular diseases. The assessment of vasculatures relies on inspecting images such as Magnetic Resonance Angiography (MRA). Vasculatures in these images are commonly separated from their backgrounds prior to clinical assessment. This is because the extracted vascular regions are straightforward to be analyzed, for instance, reconstruction of three dimensional vessel models for visual inspection, and quantitative measurements for surgical planning and medical image registration. As such, extracting and separating blood vessels from non-vessel structures in angiographic images is a vital procedure.

However, the manual extraction of blood vessels is a time consuming task. With advances in the computation power of model computers, computer aided or fully automated segmentation methods are now capable of offering satisfactory segmentation results in a relatively short period of time, compared to manual extraction. Therefore, developing new vessel segmentation techniques based on angiographic images is currently receiving significant attention from researchers.

Segmentation of vascular structures in angiographic images is a challenging task. Firstly, vessels intensity can fluctuate largely in some non-invasive MRA images, while the image background intensity can vary from region to region. It is impossible to obtain satisfactory segmentation results by conducting segmentation solely based on image voxel intensity. Detection of vasculatures in these images requires the use of some advanced image features in addition to the voxel intensity. Furthermore, the complicated vessel geometry and the disturbance introduced by adjacent non-vascular structures can adversely affect the accuracy of vessel segmentation results. On the other hand, the computation time required for an algorithm to deliver a vessel segmentation is critical in a clinical environment, which is particularly crucial in incorporating the segmentation algorithm in a practical diagnostic process.
These challenges motivate us to propose three novel edge detection and vascular segmentation methods. In the first proposed method, vessel segmentation is performed grounded on the edge detection responses given by the weighted local variance-based edge detector. This detector is robust against large intensity contrast changes and capable of returning accurate detection responses on low contrast edges. Our second method is an efficient implementation of a well-founded vessel detection approach. The proposed efficient implementation is a thousand times faster than conventional implementation, without hindering the vessel detection accuracy. The third method is a curvilinear structure descriptor which is robust against the disturbance induced by closely located objects. By the experiments based on both the synthetic data and clinical cases, the proposed methods have proven to be very suitable for vascular segmentation in MRA images.

1.2 Literature review

Various approaches exploit the nature of vascular image modalities and shapes of vessels to perform vessel segmentation. Based on their underlying theories, the methods discussed in this chapter are briefly categorized as,

- Probability models,
- Geometric models,
- Active contour models.

The methods in the category of "Probability models" mainly rely on modeling the distributions of the image intensity or image features. Since the noise level, noise models, image resolutions and vessel appearances vary largely in different image acquisition techniques, some of the methods covered in the category of "Probability models" are applicable in only a few image modalities.

"Geometric models" based approaches involve the use of linear operators to highlight tubular structures. This is due to the fact that vessels are normally elongated tubular structures with circular cross sections. The approaches in this category are widely utilized in the analysis of medical images because of their modality-independence and simple formulations.
The techniques in the third category, "active contour models" (or referred as "active surface models" or "deformable models" in other literature) are very popular for vascular segmentation. The active contour models are based on evolving contours to progressively segment target objects. The contour is evolved to perform the energy-functional gradient descent optimization, or according to the "force" which drives the contours to their desired locations. There is a large variability to tailor a functional or a force according to the image content to perform active contour based segmentation.

1.2.1 Probability models

The methods discussed in this category include the techniques describing image intensity or features using various probability models, such as the Gaussian mixture model [3], the Maxwell-Gaussian model [4], the Rician model [5] and the finite mixture model [6], and the methods making use of Markov random field models [4, 7]. In [3], Florin et al. employed a Gaussian mixture distribution to model the vessel intensity. The Gaussian mixture model copes with a large number of parameters to describe the local structure elongation and the local structure orientation to formulate feature vectors to segment cardiac vessels. A Monte-Carlo technique is applied to estimate the posterior probability distribution function of the feature vectors to update the segmented regions.

For some probability models [4–6], the choices of models are related to the characteristics of their target image modalities, which leads to modality-dependent methods. For example, Chung and Noble proposed to model the voxel intensity of phase contrast (PC) and time-of-flight (TOF) MRA images using the Rician distribution [5]. The image intensities of these two types of MRA images are magnitudes of velocity vectors of blood flows. The magnitudes of the flow-velocity vectors are computed by the modulus of the speed values acquired along three orthogonal axes. The speed values along each axis are constructed in the frequency domain. Since the frequency domain coefficients embody real and imaginary parts, the voxel intensity of PC-MRA and TOF-MRA images is composed of six values in total. Chung and Noble realized that if each of the six values is corrupted by the same level of the Gaussian noise, the Rician model is theoretically the best model to describe the intensity distributions in PC-MRA and TOF-MRA images. Based on this model, an optimal intensity threshold
value is calculated to generate the segmentation results.

Chung et al. introduced a vectorial measure "Local phase coherence" (LPC) in [4]. The LPC measure quantifies the coherence of the flow-velocity vectors present in PC-MRA images. It suppresses noise as noisy voxels do not have flow direction coherent to neighboring voxels. The LPC measure is also sensitive to the low intensity vessels which have small blood flows. The results of the LPC measure are finally by the Maxwell-Gaussian distribution. The segmentation results are given by the maximum a posteriori technique. The method described in [4] is applicable only to PC-MRA images as the Maxwell-Gaussian distribution is tailored to PC-MRA images.

Rui et al. employed a finite mixture model [6] in order to estimate the optimal intensity threshold value to segment vessels. The finite mixture model aims at describing the intensity distributions of vessels and background regions in maximum intensity projection (MIP) images. It models the intensity distributions in maximum intensity projection images and avoids the estimated intensity threshold values biased against the vascular regions which occupy a very small portion of an image volume.

Markov Random Field (MRF) models are capable of encoding the local interaction between neighboring voxels. Such interactions, for example, can be expressed as a term which measures the local structural similarity between neighboring voxels. Wong et al. introduced the use of the orientation smoothness prior in [7] to perform vessel segmentation. In [7], the orientation similarity is measured as dot products between the orientation vectors computed in adjacent voxel pairs by the orientation tensor [8, 9]. A small dot product value implies inconsistent orientations. The voxels with inconsistent orientations are assigned with small local conditional probabilities in the MRF framework. As a result, the method discourages the inclusion of voxels with having inconsistent orientations in segmented regions. Finally, the smoothness prior is incorporated in the iterated conditional modes in the Bayesian framework to obtain segmentation results.

The probability models are usually designated for a few image modalities. The usability of the modality-dependent techniques is limited. On the other hand, the segmentation methods using probability models involve iterative optimization algorithms to estimate the model parameters to fit the observed data. These algorithms are always computationally demanding, are slow even though they are implemented on modern
1.2.2 Geometric models

The geometric model based methods analyzed in this section rely on the assumption that vessels are mainly elongated tubular shapes with circular cross sections. These methods discover vasculatures by detecting the intensity changes across the vessel boundaries. These methods include those based on the first order intensity variation [10, 11], those based on the second order intensity variation [2, 12–15] and the frequency domain based techniques [7–9]. In these methods, the vessel detection responses are acquired by processing a set of filtering responses computed by applying edge-sensitive or tubular-structure-sensitive linear filters on vascular images. The detection responses are post-processed in order to obtain the vessel segmentation results.

First order intensity variation detectors

The first order intensity variation is commonly associated with intensity discontinuities occurring at object boundaries. Canny [16] gave a comprehensive analysis on edge detection based on the first order intensity variation statistics and proved in [16] that the first derivative of Gaussian filter is the optimal edge detector.

Using the Canny’s optimal edge detector, Koller et al. [10] formulated multiscale line filters to detect tubular structures, such as vessels. These filters are obtained by rotating and translating a number of pairs of the first derivative of Gaussian filters. Since the rotated and translated pairs of the first derivative of Gaussian filters are orientation-sensitive and scale-variant, each multiscale line filter can detect vascular structures in the given direction and scale. A set of detection responses are obtained by repetitively applying multiscale line filters on the vascular image using various combinations of orientations and scales. The vessel directions and widths are then estimated as the orientation and the scale that give the maximal detection responses. In three dimensional images, the authors of [10] managed to detect the vessel direction using the Hessian matrix in order to apply the multiscale line filters only along the detected vessel direction. Based on the multiscale line filtering responses, the vessels are finally segmented by thresholding the response values.
Similar to [10] but developed independently, Poli and Voli [11] proposed vessel detection filters by using a number of pairs of rotated and translated first derivatives of Gaussian filters. The vessel detection filters are also orientation-sensitive and scale-variant. Distinct from [10], Poli and Voli proposed to increase the vessel detection filter orientation-sensitivity by altering the scale parameters of the Gaussian kernels of the vessel detection filters. Their elaboration of the vessel detection filters also suggested the minimum number of discrete orientation samples used by the vessel detection filters to estimate vessel direction. Analogous to [10], segmented vasculatures are acquired by thresholding the filtering responses.

Second order intensity variation detectors

The Laplacian operator and the Hessian matrix are the most common operators used to measure the second order intensity variation. Since the second derivative operation implicitly amplifies noisy signals, low pass filtering is always performed prior to the retrieval of the image second order intensity variation statistics.

Lindeberg [15] conducted comprehensive research regarding the use of the Gaussian smoothing function along with the Laplacian operator to detect tubular structures. Based on the elegant scale-space theory, Lindeberg employed a set of scale factors used by the Gaussian smoothing filter to compute second order statistics to detect tubular structures in a multiscale manner.

In contrast to the isotropic Laplacian operation described in [15], Koller et al. [10] introduced the use of the Hessian matrix to aid the detection of vessels in three dimensional images. The entries of the Hessian matrix are obtained by carrying out directional second derivative operations on the image. This encodes the vessel directional information which is utilized to apply multiscale line filters along the detected vessel direction. Krissian et al. extended the work of [10] in [13]. They introduced a cylindrical template with a two dimensional Gaussian intensity profile in the cross section of the template to help in the detection of vessels. They also suggested combining various eigenvalues extracted from the Hessian matrix to provide additional information to deliver more accurate vessel segmentation results.

Grounded on the theory presented in Lindeberg’s work [15], Sato et al. [14] gave a thorough study on the use of the Hessian matrix in tubular structure extraction. They
showed that computing the Hessian matrix on a Gaussian smoothed image is closely related to applying the directional second derivative of Gaussian filters on the image. Their observation is that the eigenvalues of the Hessian matrix are equivalent to the results of convolving the image with the directional second derivative of a Gaussian filter along the corresponding eigenvectors. The directional second derivative of a Gaussian filter is a three dimensional function. It offers a differential effect which computes the second order intensity variation along the given direction. With an appropriate combinations of eigenvalues, it is capable of highlighting tubular structures, such as blood vessels. Furthermore, the work in [14] exploited various heuristic combinations of the eigenvalues extracted from the Hessian matrix. These combinations aim at eliminating the filtering responses produced by non-tubular (implicitly, non-vascular) structures. The binary vascular segmentation results are retrieved by thresholding the Hessian based detection responses.

Frangi et al. [2] had another approach to utilize the Hessian matrix based on the scale-space theory described in the work [15]. They introduced the notion of the vesselness measure in [2] to detect vascular structures. The vesselness measure makes use of various combinations of Hessian matrix eigenvalues. Compared to the study in [14], the vesselness measure has less heuristic parameters and employs the $L_2$-norm of the Hessian matrix to aid in the detection of vasculatures. The blood vessel segmentation results in [2] are in turn obtained by thresholding the vesselness detection results.

**Frequency domain operators**

*Orientation tensor* [8, 9] is a less popular operator for the detection of blood vessels. It offers the ability to take care of both the first and second order intensity variation. The computation of the orientation tensor involves applying six quadrature filters on the three dimensional image. These filters are constructed in the frequency domain and designed to be directional bandpass filters. In the spatial domain, these filters are complex-valued. The real parts of the filters are symmetric, which are sensitive to tubular structures analogous to the second order intensity statistics. The antisymmetric imaginary part, in contrast, is capable of discovering the first order intensity variation. The orientation tensor is employed in [7, 17] for the segmentation and enhancement of vascular images.
Discussion

The filters utilized to detect the first order and the second order intensity variations, and the frequency domain based operators have analytical Fourier expressions. Computationwise, the analytical Fourier expressions lead to the efficient implementation of filtering operations by employing frequency domain techniques. Due to the ease of implementations and fairly low computational demands, the approaches covered in this section are widely used in medical image analysis.

A shortcoming of the geometric models is that they rely on matching the filter template on vascular images to highlight vessels, which lacks flexibility to handle the vessels which deviate from the presumed tubular shapes. Such vascular structures commonly exist, for instance, crossing vessels, bifurcations and high curvature small vessels. It is common to observe false-negative cases by using geometric models to detect the aforementioned structures. A part of this issue was discussed in [14], Sato et al. suggested adjusting the heuristic parameters used in the functionals proposed in [14]. In addition, some geometric models involve computationally demanding operations for volumetric images, such as orientation sampling [18, 19] or formulation in the orientation domain [20]. Nevertheless, further studies is needed to overcome this weakness in geometric models.

1.2.3 Active contour models

The first active contour model was proposed by Kass et al. in [21] for general two dimensional image segmentation. Nowadays, active contour models are commonly used in the application of medical image segmentation [22–24]. An advantage of active contour models is that the active contour based segmentation results can be rendered with the subvoxel accuracy. Segmentation results with subvoxel accuracy could mimic the partial volume effects occurring at the curved surfaces of vessel boundaries (see Figure 1.1).

One of the most promising features of active contour models is the introduction of ”force” which drives the active contours to desired positions. The ”force” can be briefly classified as the internal force and the external force. The external force is exerted by image intensity dependent terms while the internal force merely depends on
Figure 1.1: An example illustrating the difference between the boundaries shown without and with the subvoxel accuracy. The boundaries are shown along with the discrete image grid to illustrate the image resolution. From left to right, the original circular structure with an intensity range $[-1, 1]$; the intensity of the structure after being discretized on the image grid; the zero-intensity boundary of the discretized structure, the boundary is rendered without the subvoxel accuracy; the zero-intensity boundary of the discretized structure, the boundary is rendered with the subvoxel accuracy.

the geometry of the evolving contour. The external force attracts the active contour to the position on intensity edges [21, 25, 26]. Meanwhile, the internal force usually acts as a regularization term to maintain the contour smoothness.

In state-of-the-art research, active contour models are also utilized as optimization methods to segment blood vessels [18, 27–37, 39]. As such, the active contour evolution procedures are regarded as iterative and gradient descent based optimizers. In which, the internal force is still effective as contour smoothness regularization in [18, 27–30, 35–37, 39], or in some techniques [31–34], different variants of the internal forces were proposed.

Since contour positions are queried frequently in contour evolution procedures, contour representation is crucial to ensure an efficient retrieval of point positions over active contours. As the contour originally proposed in [21], the contours are represented parametrically as a set of points. These points are moved and the contours are expressed as curves which bridge the points. This representation introduces a significant amount of computation cost to visualize the contours, especially in three dimensional cases. This has been unfavorable in recent research.

Instead, the levelset method [25, 40] is now a popular technique to cooperate with active contour models [41]. Using the levelset contour representation, the evolving active contour is modeled as the zero-level boundaries of a high dimensional levelset function. The advantage of utilizing the levelset representation is that it handles topological changes to contours naturally, such as merging or splitting of contours. Furthermore, its computational cost is greatly reduced by recent developments of discretization strategies of levelset functions, such as the narrow-band method [25] and the sparse
field levelset method [42].

**Intensity or gradient magnitude based features**

One crucial step in the design of active contour models is to specify the dynamics of active contours according to the image content. In [25], Malladi *et al.* added a balloon force to keep the active contour expanding. With the aid of an edge detector term, the expansion effect enacted on the active contours is suppressed at the positions having large gradient magnitudes. Consequently, evolving contours are halted over object boundaries.

McInerney and Terzopoulos introduced T-Snake [43]. It employs triangular meshes to represent active contours. The evolution of contours is driven by several internal forces along with the thresholded image data. Caselles devised the geodesic active contour model in [28]. This active contour model aims at finding the minimum distance curve in the Riemannian space with an image dependent metric. The geodesic active contour model [28] was extended in [29, 32–34, 44, 45] for segmentation of blood vessels.

**Gradient based vectorial features**

Some active contour methods make use of the directional information encoded in the image gradient to improve the accuracy of vessel segmentation results. Vasilevskiy and Siddiqi introduced *flux maximizing geometric flows* in [18]. The main goal of the evolution of active contours in [18] is to maximize the image gradient inward flux. They also introduce a multiscale approximation of the Laplacian operator in order to handle vascular structures of various sizes. With the help of directional information, the authors of [18] reported that their proposed technique is highly sensitive to low contrast small vessels.

Other research grounded on the vectorial image gradient feature is proposed by Xiang *et al.* in [35]. The main idea is to model the intensity edges as magnetic field emitting sources. The directions of the emitted magnetic fields are determined according to the intensity edge gradient directions. The active contours receive magnetic potential energy and are destined to reach positions where the received magnetic potential energy attains minimal.
Geometric features

Although the curvature regularization term in active contour models plays an important rule on the prevention of contour leakages, the enforcement of the smoothness regularization is sometimes prohibited in vessel segmentation. The reason is that the smoothness regularization possibly denies the active contours evolving into the thin vessels which are commonly associated with high curvature boundaries. To avoid missing small vessels while the smoothness constraint is still enforced, various methods made use of geometric features. These features allow contours to evolve into thin and high curvature vessels while reducing the chance of leakages in other positions.

Lorigo et al. presented the CURVES algorithm [34], which extended the gradient magnitude based geodesic active contour method [28] using the arbitrary codimension framework [33]. The codimension framework is grounded on the assumption that vessels are curvilinear structures. It treats the active contour as a one-dimensional line structure. As such, the smoothness regularization is applied only along the tangential directions of the apparently-one-dimensional contours. This regularization scheme is referred to as the minimal curvature regularization which allow contours to evolve into thin vessels while the smoothness regularization is still effective.

In [29], Yan and Kassim described a different approach to employing geometric features. They introduced the capillary action as a refinement to the geodesic active contour model [28]. The validity of applying capillary action force relies on the tubular shapes of vessels. The capillary action force is competed with the smoothness regularization term to pull the evolving contour into thin and tubular vascular structures.

Nain et al. described the use of the so called ball filter in [31] as an alternative to the contour smoothness constraint. The ball filter detects excessive widening of contours and discourages contour expansion when excessive widening occurs. It shows a promising effect on preventing contour leakages in vessels with blurry boundaries. However, the ball filter adds a considerable computation cost to the implementation of the contour evolution.
Hybrid features

With the success of the aforementioned methods, some works attempt to fuse various active contour models to return better segmentation results. In [46], Descoteaux et al. make use of the Frangi’s vesselness measure [2] to process the detection responses given by the multiscale flux [18]. The main goal of this combination is to extend the advantages of these two methods, the accurate detection response to tubular vascular structures of the vesselness measure; and the high sensitivity to low contrast vessels of the multiscale flux.

In [44, 45], Gazit et al. developed an edge measure similar to the gradient flux presented in [18] based on the edge detection theory described in [16] and [47]. This measure collaborates with the geodesic active contour method [28] and the minimal variance method [48] to segment blood vessels.

Other features

There are some methods that do not solely utilize the image intensity and the image gradient. Chan and Vese devised the functional minimal variance in their work active contour without edges in [48]. This approach is to minimize the intensity variance among the segmented structures and among the background regions. The dynamic of the active contours is independent of intensity edges. It could segment objects with very blurry or even no observable boundaries. It is later extended in [44, 45] which combines various functionals to perform segmentation of vascular structures.

Discussion

In the active contour based segmentation, correct contour initialization is vital to obtain desired segmentation results. Although the classical snake [21] requires the initial contour being placed closed to the target region boundaries, recently proposed active contour methods have eliminated this requirement. Some approaches [34, 48] allow parts of initial contours to be placed outside the vessels. The current trend of contour initialization strategies is to obtain contour seed points in highly confident regions [18, 35–37, 43, 46].
In active contour models, external force and the functional optimization mechanism can be employed individually or simultaneously, with an optional internal force to govern the dynamics of contours to segment blood vessels. As such, active contour models have a considerable variability to be advanced by inventing new external forces, functionals and internal forces, or making use of new combinations of them.
CHAPTER 2
WEIGHTED LOCAL VARIANCE BASED
EDGE DETECTION AND VASCULAR
SEGMENTATION

2.1 Introduction

Precise extraction of vessels requires accurate edge detection techniques. To extract blood vessels in the magnetic resonance angiograms, image gradient magnitude is widely used for observing the intensity differences between vessels and background regions. For instance, Malladi et al. [25] proposed to extract vessels by halting contours at positions where the values of $|\nabla G \ast I|$ are large. Caselles et al. [28] proposed and employed the geodesic active contour to extract blood vessels using a minimal distance curve based on the image gradient magnitude, $|\nabla G \ast I|$. McInerney and Terzopoulos introduced T-Snake [43], which was based on image gradient magnitude, and used the Laplacian operation to discover boundaries for tissue extraction in medical images.

Along the same research line, not only gradient magnitude, but also gradient direction has been used as a feature for the extraction of vessels. Xiang et al. proposed an elastic interaction model [35, 49]. The main concept is to locate vessel boundaries by minimizing an energy term associated with a magnetic field calculated from image gradient. Vasilevskiy and Siddiqi introduced [18] the flux maximizing geometric flows. The vessel boundaries were selected according to the zero-crossing boundaries of the magnetic flux, which was computed from image gradient.

Vessel boundaries can also be detected with the help of structural information in addition to the image gradient. Lorigo et al. presented the CURVES algorithm [34, 50], which extended the gradient magnitude based geodesic active contour method [28] using the arbitrary codimension framework [33]. The CURVES algorithm contained a heuristic factor. By adjusting this factor, the algorithm can enhance the detection and segmentation of tubular structures. Yan and Kassim also improved the geodesic active
contour method [28] by employing the capillary effects for the detection of thin vessel boundaries [29, 51].

On the other hand, the Hessian matrix based structural information is also useful in the detection of vessel boundaries. As mentioned in a review by Sato et al. [14], the eigenvalues of the Hessian matrix can provide valuable information about the shape and local structures of a boundary. Frangi et al. introduced the term “vesselness” [2] as a measurement of tubular structures by observing the ratio of eigenvalues of the Hessian matrix. Bullitt et al. [12] presented a work that found the vessel centerlines first and then located the vessel boundaries according to the eigenvalue ratio of the Hessian matrix. Descoteaux et al. [52] employed the vesselness measure [2] to detect the boundaries of tubular structures and incorporated it in [18] to perform segmentation. Westin et al. [32] utilized the Hessian matrix to detect the boundaries of planar or tubular structures and the Hessian matrix based boundary information was complemented with the codimension two segmentation method [34].

In the aforementioned approaches [2, 12, 14, 32, 52], the structural information of the Hessian matrix is quantified by the relation of eigenvalues along different principle directions of the Hessian matrix. Different from the gradient, which utilizes the first derivatives of an image, the Hessian matrix is based on the second derivatives of images to compute the curvatures of boundaries. The curvature in the normal direction of the boundaries of vessels, which are mainly of tubular shape, should be much larger than the curvatures in other principle directions. Compared with the image gradient, which is general and has responses independent of the shape and local structures of boundaries, the Hessian matrix can distinguish between types of boundaries (e.g. tubes, planes, blob surfaces or noise) so that the Hessian matrix based techniques can be tailored to the target tubular structures.

Some non-tubular vasculatures such as junctions or ending point can induce high curvature values along more than one principle directions. This can possibly lead to inaccurate detection of vessel boundaries using the Hessian matrix based methods. On the other hand, although the image gradient is more general in handling structures with different shapes, due to the presence of intensity inhomogeneity such as bias field, overlapping between vessels and other tissues or the speed related vessel intensity, the intensity difference between vessels and background regions are not consistent but are
varying. The boundaries of the low contrast vessels cannot provide large values of the gradient term $|\nabla G \ast I|$ for the methods based on image gradient to detect those vessel boundaries.

In this chapter, we propose a general edge detection approach based on weighted local variances, which quantify intensity similarity on both sides of an edge for edge detection. The weighted local variance based method is robust against changes in intensity contrast between vessels and image background regions, and is able to return strong and consistent edge responses to the boundaries of low contrast vessels. Different from the Hessian matrix based techniques, which analyze the shape and local structures of boundaries for the detection of tubular vascular structures, the proposed weighted local variance based scheme is a general technique that returns high detection responses on low contrast edges disregarding the shape and local structures of boundaries.

Using the edge detection results of the weighted local variance based method, which include the edge strength and the edge normal direction, blood vessels are extracted by the flux maximizing geometric flows [18]. In the experiments, the edge strength and the edge normal direction computed by the proposed method are studied using two synthetic volumes. The weighted local variance based vascular segmentation method is validated and compared using a time-of-flight (TOF) magnetic resonance angiography (MRA) and three phase contrast (PC) MRA image volumes. It is shown experimentally that the weighted local variance based method is capable of giving high and consistent edge strength in low contrast boundaries and the active contour based segmentation using weighted local variance is able to handle low contrast vessels.

2.2 Methodology

2.2.1 Edge detection and weighted local variance

In this section, we introduce the use of weighted local variance [36][37][38] for extracting edge information, including edge normal orientation and edge strength. The weighted local variance is a general edge detection technique, which considers the voxel intensity homogeneity within local regions. To extract edge information based on the weighted local variance, we first consider the directional derivative of a Gaussian function. The directional derivative of a Gaussian function $G(\vec{x})$ along a direction $\hat{n}$ at a position $\vec{x}$
in 2D is given by,

\[ G_n(\vec{x}; \sigma, \sigma_\perp) = -\frac{\vec{x} \cdot \hat{n}}{2\pi\sigma^2\sqrt{\sigma\sigma_\perp}} \exp\left( -\frac{(\vec{x} \cdot \hat{n})^2}{2\sigma^2} - \frac{1}{2} \frac{\|\vec{x} \times \hat{n}\|^2}{\sigma_\perp^2} \right), \]

and in 3D

\[ G_n(\vec{x}; \sigma, \sigma_\perp) = -\frac{\vec{x} \cdot \hat{n}}{(2\pi)^{3/2}\sigma^{5/2}\sigma_\perp} \exp\left( -\frac{(\vec{x} \cdot \hat{n})^2}{2\sigma^2} - \frac{(\vec{x} \cdot \hat{n}_1)^2 + (\vec{x} \cdot \hat{n}_2)^2}{2\sigma_\perp^2} \right), \tag{2.1} \]

where \( \hat{n}_1 \) and \( \hat{n}_2 \) are the unit vectors, which are perpendicular to each other and orthogonal to \( \hat{n} \), mathematically, \( \hat{n} = \hat{n}_1 \times \hat{n}_2 \). These filters are sensitive to an edge having the normal direction aligned with \( \hat{n} \). The value of \( \sigma \) determines the scale of an edge detectable by the filter while the value of \( \sigma_\perp \) specifies the size of the filters in directions orthogonal to the derivative direction. In the case that \( \sigma \neq \sigma_\perp \), \( G(\vec{x}) \) is an anisotropic Gaussian function, which is dependent on the orientation \( \hat{n} \); when \( \sigma = \sigma_\perp \), the above equations represent the directional derivatives of an isotropic Gaussian function, which is similar to the filter proposed in [16].

The goal of the weighted local variance (WLV) is to quantify voxel intensity homogeneity locally based on the directional derivatives of a Gaussian function. To achieve this, we first split \( G_n(\vec{x}) \) into two halves,

\[ \begin{align*}
G'_{1,n}(\vec{x}) &= \begin{cases} 
G_n(\vec{x}) & \text{if } G_n(\vec{x}) \leq 0 \\
0 & \text{otherwise}
\end{cases}, \\
G'_{2,n}(\vec{x}) &= \begin{cases} 
G_n(\vec{x}) & \text{if } G_n(\vec{x}) > 0 \\
0 & \text{otherwise}
\end{cases}.
\end{align*} \tag{2.2} \]

These two filters are then normalized to be sum-to-one, for \( i = \{1, 2\} \),

\[ G_{i,n}(\vec{x}) = \frac{G'_{i,n}(\vec{x})}{\int G'_{i,n}(\vec{y})d\vec{y}}. \tag{2.3} \]

Using these normalized filters, the value of weighted local variance is calculated. Broadly speaking, variance is a measure to estimate the sparseness of a set of variables. Similarly, based on the normalized filters, around the position \( \vec{x} \), the weighted local variance evaluates the intensity homogeneity within two local regions separated by an edge having normal direction aligned with \( \hat{n} \). These two local regions are associated with the
non-zero entries of $G_{1,\hat{n}}(\vec{x})$ and $G_{2,\hat{n}}(\vec{x})$. Hence, the weighted local variances (WLVs) for both split filters are defined as,

$$\text{WLV}_{i,\hat{n}} = \int G_{i,\hat{n}}(\vec{y})I(\vec{x} + \vec{y}) - \mu_{i,\hat{n}}(\vec{x}))^2 d\vec{y},$$  \hspace{1cm} (2.4)$$

where $I(\vec{x})$ represents the intensity at $\vec{x}$, and $\mu_{1,\hat{n}}$ and $\mu_{2,\hat{n}}$ are the weighted intensity averages of their corresponding filters, i.e., $\mu_{i,\hat{n}}(\vec{x}) = \int g_{i,\hat{n}}(\vec{y})I(\vec{x} + \vec{y})d\vec{y}, \ i = \{1, 2\}$.  

The weighted local variances, $\text{WLV}_{1,\hat{n}}$ and $\text{WLV}_{2,\hat{n}}$, are weighted sums of squared intensity differences between the intensities of the neighboring voxels $I(\vec{x} + \vec{y})$ and their corresponding weighted intensity averages, $\mu_{1,\hat{n}}(\vec{x})$ and $\mu_{2,\hat{n}}(\vec{x})$, respectively. As such, the variances aim to evaluate the intensity homogeneity in two local regions separated by an edge. To illustrate the idea, we use five examples consisting of horizontal edges having different levels of intensity contrast, a corner and two edges with different values of curvature. This is shown in Figure 2.1. The figure shows the values of $\sqrt{\text{WLV}_{1,\hat{n}}(\theta)}$ and $\sqrt{\text{WLV}_{2,\hat{n}}(\theta)}$ with various orientations, $\hat{n}(\theta) = [\cos \theta, \sin \theta]^T$. This demonstrates how WLV varies with the orientation of detection $\hat{n}(\theta)$. As shown in Figure 2.1, for all five examples, the variances vary as $\theta$ changes and attain small values when the corresponding filters, $G_{1,\hat{n}}(\theta)$ or $G_{2,\hat{n}}(\theta)$ along $\theta$, do not cross the edges. It is observed that the value of $\min(\sqrt{\text{WLV}_{1,\hat{n}}(\theta)}, \sqrt{\text{WLV}_{2,\hat{n}}(\theta)})$ is small when $\theta$ is approaching the edge normal orientation. A small WLV value implies that the voxel intensities are similar in the two local regions on two different sides of an edge.

Therefore, we define a confidence value for finding an edge having normal orientation $\hat{n}$, as

$$R_{\hat{n}}(\vec{x}) = \frac{\mu_{1,\hat{n}}(\vec{x}) - \mu_{2,\hat{n}}(\vec{x})}{\sqrt{\min(\text{WLV}_{1,\hat{n}}(\vec{x}), \text{WLV}_{2,\hat{n}}(\vec{x}))} + \epsilon},$$  \hspace{1cm} (2.5)$$

where the epsilon $\epsilon$ avoids singularity when either or both WLV$_{1,\hat{n}}(\vec{x})$ and WLV$_{2,\hat{n}}(\vec{x})$ are zero. The value of this constant is $10^{-3}$ in our implementation. On one hand, the denominator of Equation 2.5, based on the weighted local variances for evaluating intensity similarity between both sides of an edge, should be small when an edge is likely to be found. On the other hand, the numerator measuring the intensity change across an edge should be large if an edge is detected. Therefore, a high confidence value implies the presence of an edge having normal orientation $\hat{n}$. 

18
2.2.2 Properties

In the first and second rows of Figure 2.1, we show two horizontal edges. The former has intensity values 1 and 0 and the latter with lower contrast has intensity values 0.9 and 0.6, which give smaller intensity differences across the edge. Comparing the variances obtained from the high contrast edge (the top row of Figure 2.1) and the low contrast edge (the second row of Figure 2.1), the variances are generally smaller in the case of the lower contrast edge. Since the values of both terms, \([\mu_{1,\hat{n}}(\vec{x}) - \mu_{2,\hat{n}}(\vec{x})]\) and \(\min(\sqrt{WLV_{1,\hat{n}(\theta)}}, \sqrt{WLV_{2,\hat{n}(\theta)}})\), are reduced for low contrast edges, the return value of Equation 2.5 is not affected significantly by the change in intensity contrast. This concept can be mathematically illustrated by considering two arbitrary image patches, \(I\) and \(J\), which have different levels of intensity contrast and brightness but are related by \(I(\vec{x}) = cJ(\vec{x}) + b, c > 0\). The terms \(c\) and \(b\) are constants representing the differences in intensity contrast and brightness respectively. The numerator of the confidence value in Equation 2.5 for the image patch, \(I\) and \(J\), are related as,

\[
\mu_{1,\hat{n}}^I - \mu_{2,\hat{n}}^I = \int (cJ(\vec{x} + \vec{y}) + b)(G_{1,\hat{n}}(\vec{y}) - g_{2,\hat{n}}(\vec{y}))d\vec{y}
\]

since \(G_1\) and \(G_2\) are summed-to-one,

\[
= c(\mu_{1,\hat{n}}^I - \mu_{2,\hat{n}}^I). \quad (2.6)
\]

Moreover, for the denominator of the confidence value in Equation 2.5, we evaluate the WLVs (Equation 2.4) for the image patches, \(I\) and \(J\), for \(i = \{1, 2\}\),

\[
WLV_{i,\hat{n}}^I(\vec{x}) = \int \{G_{1,\theta}(\vec{y})(cJ(\vec{x} + \vec{y}) + b - \mu_{1,\hat{n}}^I(\vec{x} + \vec{y}))^2\}d\vec{y},
\]

\[
= \int \{G_{1,\theta}(\vec{y})(cJ(\vec{x} + \vec{y}) + b - c\mu_{1,\hat{n}}^I(\vec{x} + \vec{y}) - b)^2\}d\vec{y},
\]

\[
= c^2WLV_{i,\hat{n}}^J(\vec{x}). \quad (2.7)
\]

Therefore, according to Equation 2.5, the confidence value of the image patch \(I\) is,

\[
R_{\hat{n}}^I(\vec{x}) = \lim_{\epsilon \to 0} \frac{c[\mu_{1,\hat{n}}^I(\vec{x}) - \mu_{2,\hat{n}}^I(\vec{x})]}{c\sqrt{\min(WLV_{1,\hat{n}}^I(\vec{x}), WLV_{2,\hat{n}}^I(\vec{x}))} + \epsilon} \approx R_{\hat{n}}^I(\vec{x}). \quad (2.8)
\]
The brightness term $b$ is eliminated and the contrast term $c$ is approximately canceled for the calculation of confidence value defined in Equation 2.5, except that there is a small constant $\epsilon$.

With regard to the confidence value defined in Equation 2.5, the numerator (i.e. the intensity difference term) and the denominator (i.e. the WLV terms) measure respectively the intensity difference across an edge and intensity similarity between two local regions separated by the edge. Both the intensity difference and the value of WLV are altered according to the contrast term $c$ and this leads to the cancellation of the contrast term in Equation 2.8. In practice, in the presence of a low contrast edge (i.e. $c$ is small in Equations 2.6 – 2.8), the intensity similarity is high (i.e. the values of WLVs are small), which can compensate the reduced intensity change across low contrast edge. This enables the confidence value in Equation 2.5 to be retained at a high level for low contrast edges.

### 2.2.3 Computing edge normal orientation and edge strength

In the previous section, we introduce a confidence value in Equation 2.5 based on the weighted local variance for edge detection. In this section, we further elaborate the procedures to obtain edge strength and edge normal orientation using WLV based confidence values. The WLV based edge detection method is named **WLV-EDGE** hereafter to distinguish it from the calculation of WLVs in Equation 2.4.

Both the edge strength and the edge normal orientation of **WLV-EDGE** are quantified by calculating the confidence values in a set of discretized orientations. The set of discrete orientations is denoted as a set of unit vectors $\hat{n}_k, k = \{1, 2, 3, ..., K\}$. $\hat{n}_k$ is the k-th discrete orientation sample (see Figure 2.2 for the typical sets of 2D orientations and 3D orientations). Based on Equation 2.5, The confidence value obtained along the orientation $\hat{n}_k$ is denoted as $R_{\hat{n}_k}(\vec{x})$. It is noted that the orientation samples sweep across a semi-circle in 2D or the surface of a hemisphere in 3D instead of a complete circle or sphere due to the fact that the confidence value is conjugate, i.e. $R_{\hat{n}_k}(\vec{x}) = -R_{-\hat{n}_k}(\vec{x})$.

Although it is straightforward to estimate the edge normal orientation using the orientation associated with the largest confidence value, it is possible that multiple
orientation samples give the maximum confidence value, which can be problematic in determining the edge normal orientation in this situation. Therefore, the edge normal orientation is obtained according to the confidence values computed in different discrete orientations. This is achieved by considering the confidence values as a set of points,

{\hat{R}_{n_k}(\vec{x})\hat{n}_k}, k = \{1, 2, 3, \ldots, K\}.  \tag{2.9} 

Each voxel has its own point set to represent $K$ confidence values. The point set for each voxel is considered independently. The idea is that an orientation having a large output of $\hat{R}_{n_k}(\vec{x})$ is likely to be the edge normal orientation. Thus, the orientation having a large value of $\hat{R}_{n_k}(\vec{x})\hat{n}_k$ is represented as a point, $\hat{R}_{n_k}(\vec{x})\hat{n}_k$, located away from the origin in the Euclidean space. Using this representation, the edge normal direction is estimated by finding an orientation such that the points are mostly spanning away. Such edge normal orientation is found using the first principle direction of the point set. This is accomplished by performing eigen-decomposition on a matrix associated with the points,

$$\mathbf{M}(\vec{x}) = \mathbf{E}_k \left[ R_{n_k}^2(\vec{x})(\hat{n}_k \cdot \hat{n}_k^T) \right],$$  \tag{2.10} 

where $\mathbf{E}(.)$ is the expected value.

Three eigenvectors $\hat{e}_1(\vec{x})$, $\hat{e}_2(\vec{x})$ and $\hat{e}_3(\vec{x})$ corresponding to three eigenvalues $\lambda_1(\vec{x})$, $\lambda_2(\vec{x})$ and $\lambda_3(\vec{x})$ are obtained respectively, where $|\lambda_1(\vec{x})| \geq |\lambda_2(\vec{x})| \geq |\lambda_3(\vec{x})|$. Using the first principle direction, either one of the directions $\hat{e}_1(\vec{x})$ or $-\hat{e}_1(\vec{x})$ represents the edge normal orientation of the voxel $\vec{x}$. The final decision is based on the sign of the sum of confidence value samples projected along $\hat{e}_1(\vec{x})$, which is formulated as,

$$\hat{g}(\vec{x}) = \text{sign} \left\{ \mathbf{E}_k \left[ [R_{n_k}(\vec{x})\hat{n}_k] \cdot \hat{e}_1(\vec{x}) \right] \right\} \hat{e}_1(\vec{x}).$$  \tag{2.11} 

Meanwhile, the edge strength is computed as

$$s(\vec{x}) = \sqrt{\lambda_1^2(\vec{x}) + \lambda_2^2(\vec{x}) + \lambda_3^2(\vec{x})}. \tag{2.12}$$

The calculation of the edge strength in the above equation is the $L_2$ Norm of the matrix $\mathbf{M}(\vec{x})$ in Equation 2.10. Since the entries of this matrix $\mathbf{M}(\vec{x})$ are based on the confidence values, the edge strength calculated from the above equation inherits the
robustness of the confidence values against different edge contrasts (see Equation 2.8 and the discussion in Section 2.2.2). Therefore, the estimated edge strength can retain a high value despite low contrast edges.

It should be pointed out that the computations of $\hat{g}(\vec{x})$ and $s(\vec{x})$ are based on the values of $WLV$, which depend on the shape of the filters $g_{1,\hat{n}}$ and $g_{2,\hat{n}}$, as described in Equation 2.4. The shape of the filters, $g_{1,\hat{n}}$ and $g_{2,\hat{n}}$, are controlled by two parameters $\sigma$ and $\sigma_\perp$ (see Equations 2.1-2.3). We here briefly describe the influences of these two parameters and the criteria for setting their values.

Parameter $\sigma$ specifies the $WLV$ detection range in a direction perpendicular to an edge that has to be detected. It designates the minimum width of the vessels, which boundaries can be detected by $WLV$-EDGE. This is because those vessels with diameters smaller than or close to $\sigma$, the effective range of the filters, $g_{1,\hat{n}}$ and $g_{2,\hat{n}}$, used by $WLV$ in all orientations, is longer than the vessel width. Then, the detection range of either one of the filters encloses the entire width of the target vessel and, consequently, both the voxels of the background region and the target vessel are included inside the detection range. The values of $WLV$ are then boosted significantly even though the filters are properly aligned with the edge orientation. As such, for detecting the boundary of vessels with diameter smaller than or close to $\sigma$, $WLV$ cannot provide reliable edge information and thus $WLV$-EDGE is adversely affected. It is recommended that the value of $\sigma$ is set with the consideration of the width of the narrowest vessels. For example, the value of $\sigma$ can be set slightly smaller than 1-voxel-length in order to help detect the 1-voxel-width vessels.

For the parameter $\sigma_\perp$, it determines the $WLV$ detection range along the orientation parallel to the target edge. The value of $\sigma_\perp$ is suggested to be similar to the value of $\sigma$, given that the value of $\sigma$ is properly assigned. Otherwise, if the value of $\sigma_\perp$ is too large, even though $g_{1,\hat{n}}$ and $g_{2,\hat{n}}$ are aligned with the vessel edge orientation, they can inevitably include background voxels outside the target vessel located in the tangential directions of the edge. Furthermore, it is recommended that $\sigma_\perp$ should be at least equal to the longest length of a voxel even if $\sigma$ has a value smaller than the longest length of the voxel, so that the detection range of $WLV$ can include enough voxel samples to provide a reliable measurement for estimating $WLV$-EDGE. In our experiments presented in Section 2.3, the parameters $\sigma$ and $\sigma_\perp$ for the clinical cases were selected
according to the criteria described above.

2.2.4 Implementation

The estimations of edge strength and edge normal direction require probing the confidence value in a set of discrete orientations. It is a computation demanding process as it repeatedly calculates the confidence value for each voxel in different orientations. To speed up the proposed method, in our implementation, the calculation of the confidence value (Equation 2.5) is performed in the frequency domain. First, for the denominator of Equation 2.5, the weighted local variances as stated in Equation 2.4 are rewritten in the form of convolution, for $i = \{1, 2\}$,

$$WLV_{i,\hat{n}}(\vec{x}) = \int G_{i,\hat{n}}(\vec{y}) (I(\vec{x} + \vec{y}) - \mu_{i,\hat{n}}(\vec{x}))^2 d\vec{y},$$

$$= \int G_{(3-i),\hat{n}}(\vec{y}) I^2(\vec{x} - \vec{y}) d\vec{y} - \mu^2_{i,\hat{n}}(\vec{x}),$$

$$= I^2(\vec{x}) * G_{(3-i),\hat{n}}(\vec{x}) - \mu^2_{i,\hat{n}}(\vec{x}), \quad (2.13)$$

similarly, for the terms $\mu_{i,\hat{n}}$ appearing in the above equation and in the numerator of Equation 2.5,

$$\mu_{i,\hat{n}}(\vec{x}) = \int G_{(3-i),\hat{n}}(\vec{y}) I(\vec{x} - \vec{y}) d\vec{y},$$

$$= I(\vec{x}) * G_{(3-i),\hat{n}}(\vec{x}). \quad (2.14)$$

Then, the convolution is computed as the multiplication in the frequency domain using the Fast Fourier Transform algorithm. The running time of calculation of weighted local variances in a single orientation is reduced significantly from $O(m^2)$ to $O(m \log m)$ for an image with $m$ voxels.

On the other hand, it is always beneficial to have a larger number of orientation samples for WLV-EDGE. However, an excessive use of orientation samples can increase the running time of WLV-EDGE without any remarkable enhancement to the accuracy. We suggest that the number of orientation samples should be associated with the size of the filtering window since a large filtering window extends the effective range of the filtering process, which requires a finer angular resolution.
In 2D cases, for a semi-circle having a diameter equal to the dimension of the filtering window, we can have at least one equally spaced orientation sample per unit length for the semi-circle. In our implementation, there are 15 voxels in each dimension for the filtering window. The circumference of the corresponding semi-circle having diameter 15 is \( \frac{15}{2} \pi \). In order to have at least one equally spaced orientation sample per unit length, we should have \( \lceil \frac{15}{2} \pi \rceil \) samples, i.e. 24 samples. An example of the 24 orientation samples in 2D is shown in Figure 2.2a.

In 3D cases, the orientation samples are organized in a grid fashion along the longitudinal and the latitudinal directions. There are 24 angularly equally spaced latitudinal levels for the same reason as in 2D cases. At the latitudinal level of \( \frac{\pi}{2} \), there are 24 orientation samples along the longitudinal direction for the aforementioned reason. In each of the other latitudinal levels, the number of samples is reduced according to the circumference of semi-circles in the corresponding latitudinal levels along the longitudinal direction. As such, there are 281 orientation samples totally, which are roughly equally spaced, as shown in Figure 2.2b.

### 2.2.5 Segmentation using active contour model

The edge normal direction and the edge strength computed in Equations 2.11 and 2.12 are the edge information of WLV-EDGE. The robustness of WLV-EDGE against changes to the intensity contrast of the edges makes it suitable to detect blood vessel boundaries for active contour based vascular segmentation.

Due to the nature of vascular images such as the presence of intensity inhomogeneity like bias field, the presence of other tissues or fluctuation in the speed related intensity, it is possible that some vessels have lower intensity contrast than other structures. These low contrast vessels are difficult to detect in active contour models. The difficulty of detection is due to the fact that the boundaries of the low contrast vessels can not exert enough "force" to compete with other boundaries. This results in contour leakages if the contour is attracted by the edges of other unrelated tissues but not the weak edges of low contrast vessels. On the other hand, it can possibly lead to contours being halted before reaching the vessel boundaries as those boundaries cannot attract the contours to the desired position.
Although the formulation of WL-EDGE is general and not limited to the detection of vessel boundaries, the edge strength computed from WL-EDGE is robust to the changes of intensity contrast of edges. The computed edge strength also retains high even when the edge contrast is low. Therefore, the edge information computed from WL-EDGE is useful for active contour models to prevent the missing of weak edges.

Using the edge information extracted by WL-EDGE, we apply flux maximizing geometric flows FLUX \cite{18} to incorporate WL-EDGE for the extraction of blood vessels. There are two reasons that we choose FLUX to complement WL-EDGE (it is mentioned as WL-FLUX hereafter). First, some approaches such as geodesic active contour \cite{27,28} or geometric snake model \cite{30} do not utilize edge directional information for segmentation. Thus, they are not suitable for the evaluation of the proposed method, which returns both edge strength and edge normal direction. In addition, the proposed technique is independent of tubular structures and does not make use of shape and local structural information. Although there are other methods that employ both edge strength and edge normal direction such as CURVES \cite{34,50} or capillary active contour \cite{29,51}, these methods are specialized for detecting tubular structures, which is not the focus of the proposed method. On the other hand, FLUX utilizes both edge strength and edge normal orientation and is not limited to tubular structures. Furthermore, FLUX is state-of-the-art, and has been used to generate reference segmented vessels for BrainWeb \cite{53}. Therefore, FLUX is employed and incorporated with WL-EDGE to perform segmentation of blood vessels.

In FLUX, the motion of a closed active contour $C$ which surface is parametrized by $\bar{S}$ is given by,

$$C_t(\bar{S}) = \left\{ \nabla \cdot \vec{v}(C(\bar{S})) \right\} \cdot \vec{N}(\bar{S}),$$  \hspace{1cm} (2.15)

where $\vec{N}(\bar{S})$ is the normal direction of the contour $C(\bar{S})$ and $\vec{v}(S(\bar{S}))$ is a vector field associated with the gradient. In practice, as described in \cite{18}, the divergence of $\vec{v}$ is calculated using a multi-scale method and discretized as,

$$\nabla \cdot \vec{v}(\bar{x}) \approx \max_{r \in L} \left\{ \frac{1}{|L|} \sum_{\hat{n} \in L} \vec{v}(\bar{x} + r\hat{n}) \cdot \hat{n} \right\},$$  \hspace{1cm} (2.16)

where $L$ is a set of outward normals the bounding sphere having radius $r$, $L$ is a set
of discrete radii, which specifies the radii of targeted vessels. In our implementation, $\mathbb{L}$ consists of 562 orientation samples (in a sphere fashion), which is the double of $K = 281$ (in a hemisphere fashion) for WLV-EDGE in Equation 2.10 and Equation 2.11. In [18], Vasilevskiy and Siddiqi have proposed to obtain the vector field $\vec{v}$ as the gradient vector field of an image $I$ smoothed by an isotropic Gaussian filter $G$, i.e.

$$\vec{v}(\vec{x}) = \nabla(G \ast I(\vec{x})). \tag{2.17}$$

To formulate WLV-FLUX, the edge strength (Equation 2.12) and the edge normal orientation (Equation 2.11) extracted by WLV-EDGE are incorporated in FLUX. As such, the vector field $\vec{v}$ in Equation 2.17 is substituted using a vector field, $\vec{w}(\vec{x})$, based on the edge detection results obtained from WLV-EDGE. The vector field is given by,

$$\vec{w}(\vec{x}) = s(\vec{x})\hat{g}(\vec{x}), \tag{2.18}$$

where $s$ and $\hat{g}$ are the edge strength and the edge normal orientation respectively computed from WLV-EDGE in Equation 2.12 and Equation 2.11. As such, Equation 2.16 becomes

$$\nabla \cdot \vec{w}(\vec{x})$$

$$\approx \max_{r \in \mathbb{L}} \left\{ \frac{1}{|\mathbb{L}|} \sum_{\hat{n} \in \mathbb{L}} \vec{w}(\vec{x} + r\hat{n}) \cdot \hat{n} \right\},$$

$$= \max_{r \in \mathbb{L}} \left\{ \frac{1}{|\mathbb{L}|} \sum_{\hat{n} \in \mathbb{L}} s(\vec{x} + r\hat{n})\hat{g}(\vec{x} + r\hat{n}) \cdot \hat{n} \right\} \tag{2.19}$$

The zero level of a level set function [40] is utilized to represent the moving contour $C$. To perform segmentation, the motion of the level set $\Theta$ function is governed by,

$$\frac{\partial \Theta(\vec{x})}{\partial t} = \mathcal{L}(\vec{x})|\nabla \Theta(\vec{x})| + \kappa \nabla \cdot \left( \frac{\nabla \Theta(\vec{x})}{|\nabla \Theta(\vec{x})|} \right), \tag{2.20}$$

where $\mathcal{L}(\vec{x})$ is a function that determines the speed of the contour evolving in the normal direction and $\beta$ is a regularization term to specify resultant contour smoothness. The values of $\mathcal{L}(\vec{x})$ are $\nabla \cdot \vec{v}(\vec{x})$ (Equation 2.16) for FLUX; and $\nabla \cdot \vec{w}(\vec{x})$ (Equation 2.19) for WLV-FLUX.
The discretization scheme and related parameters of Equation 2.20 follow the sparse field level set method presented in [42]. The implementation is based on the Insight Segmentation and Registration ToolKit (ITK) [54]. For those entries of $\vec{v}$ and $\vec{w}$ in Equation 2.16 and Equation 2.19 having non-integer coordinates, the values are linearly interpolated.

2.3 Experiments

To validate the performance of WLV-EDGE, we have carried out two sets of experiments on both synthetic and clinical image volumes. The first set of experiments employs three synthetic image volumes with synthetic tubes and tori of various sizes to analyze the estimation accuracy of the edge strength and the edge normal direction under the effect of intensity variation (see Section 2.3.1). The estimation accuracy of WLV-EDGE is compared with two edge detection approaches. In the second set of experiments, using four magnetic resonance angiographic (MRA) image volumes, the performance of WLV-FLUX is compared with FLUX (see Section 2.3.2).

2.3.1 Synthetic image volumes

In this section, we present the results obtained from the first experiment, in which a synthetic and numerical image volume of size $24 \times 24 \times 100$ voxels was created. The image volume contains a synthetic and three-dimensional (3D) tube with the diameter of 7 voxels. Figure 2.3a shows one of the image slices (x and y axes are also shown). In the image volume, the intensity values inside the tube are consistent in the x-y plane. Although there is no intensity variation in the x- and y- directions, the x and y dimensions are necessary for creating the surface curvature of a 3D tube. Along the z-direction with $x = 12$, Figure 2.3b shows the cross section of the image volume and that the synthetic tube runs from the first slice (left) to the last slice (right). The intensity value is 0 in the background regions and varies along the tube (i.e. along the z-axis). The intensity profile along the tube surface ($x = 12$ and $y = 10$) is plotted in the top row of Figure 2.3c. This experiment aims to demonstrate the relationship between the estimated edge strength and the intensity contrast on the tube surface, and the robustness of the proposed method against the intensity change along a tube.
For our approach, the edge strength of WLV-EDGE is estimated using the Equation 2.12. The estimated edge strength is then compared with the estimated edge strengths computed from two other edge detection approaches, structure tensor (ST) [55] and smoothed image gradient (GRADIENT). For ST, the edge strength is defined as the square root of the trace of the structure tensor. The edge strength of GRADIENT is the magnitude of the filtering responses of applying the directional derivatives of a Gaussian function on an image volume. In the experiments, for ST, a numerical scheme of central finite difference was utilized to compute the image derivatives for the entries of the structure tensor. The image volume was pre-processed with a Gaussian smoothing filter prior to the calculation of the image derivatives. The value of $\sigma$ used in WLV-EDGE and the scale parameter of the Gaussian filter used in ST and GRADIENT were fixed to 2. The window size for the filtering processes of ST, GRADIENT and WLV-EDGE was $15 \times 15 \times 15$ voxels.

Along the z direction with $x = 12$ and $y = 10$, the estimated edge strengths for all the approaches are plotted along with the intensity profile, as shown in Figure 2.3c. The corresponding slices of the edge strengths estimated by the three methods along the z-direction with $x = 12$ are shown in Figure 2.3d. For comparison, the values of the plots and the slices of the edge strengths were normalized to have maximum value equal to 1. In the second and third rows in Figure 2.3c, it is observed that the edge strength profiles of ST and GRADIENT fluctuate according to the intensity change along the tube. Conversely, as plotted in the last row of Figure 2.3c, although there are two slight drops of edge strength computed from WLV-EDGE at the positions of 40 and 70, WLV-EDGE's edge strength profile is relatively flat and consistent along the tube, as compared with other two methods. These two slight drops of edge strength were caused by the fact that the estimated confidence values at these two positions were reduced slightly. It is because, in the denominator of Equation 2.5, the values of WLVs are increased at the positions where the intensity values are varying significantly.

The experimental results show that, as compared with the edge strength computed from ST and GRADIENT, the estimated edge strength of WLV-EDGE is relatively consistent given the intensity inhomogeneity along the tube. It is also observed that the edge strengths computed from ST and GRADIENT follow the trend of the intensity profile along the tube. On the contrary, WLV-EDGE obtains a fairly flat
edge strength profile. It is because the intensity differences between the tube and background are normalized by WLV in the calculation of confidence value using Equation 2.5, as illustrated in Equation 2.8 and discussed in Section 2.2.2.

In the second experiment, two synthetic and numerical image volumes were created and utilized. Both volumes are of $181 \times 217 \times 181$ voxels and are shown in Figure 2.4a and Figure 2.8. This experiment aims to further study the variation of estimated edge strength and the discrepancy of edge normal orientation estimated by ST, GRADIENT and WLV-EDGE under the effect of intensity inhomogeneity. Different from the first experiment, the intensity inhomogeneity of this experiment was specified according to different bias field models appearing in MR images [56]. Bias fields are inherent to MR imaging and they can cause smooth and slow intensity changes within different regions of an image. This intensity change induced by the bias field is multiplicative and causes variations in intensity contrast between vessels and background.

One of these two synthetic and numerical volumes consists of straight tubes of different radii and the other volume contains tori with various sizes. In the volume of straight tubes, as shown in Figure 2.4a, there are totally 25 circular tubes having radii equal to 2, 3, 4, 5 and 6 voxels. This image was formed by firstly assigning intensity value 1 for the voxels inside the tubes and 0 for the background regions, and then smoothing using a Gaussian filter with a scale parameter equal to 1. The smoothing aims at creating smooth intensity transitions from the interior of the tubes to the image background. For the volume of tori, as shown in Figure 2.8, there are totally 18 non-overlapping tori. The radius ($R$ in Figure 2.8d) of the torus centerline ranges from 12 voxels to 72 voxels, and the shortest distance ($r$ in Figure 2.8d) between the torus surface and the torus centerline ranges from 2 voxels to 6 voxels. Similar to the straight tube volume, this image was formed by firstly assigning intensity value 1 for the voxels inside the tori and 0 for the background regions, and then smoothing using a Gaussian filter with a scale parameter equal to 1. These tubes and tori, of different sizes, are located in different positions of the images in order to examine how the bias fields affect the performance of different methods.

In this experiment, the bias fields were applied according to the bias field model for MR imaging described in [56]. In this model, the resultant signal $s(x)$, the original signal $o(x)$, the noise induced by the imaging device $n(x)$ and the bias field $b(x)$ at
position \( \mathbf{x} \) are related as,

\[
o(\vec{x})b(\vec{x}) + \eta(\vec{x}).
\]

The noise generated by the imaging devices, \( \eta(s) \), was simulated by an additive Gaussian noise with \( \sigma = 0.05 \) in order to mimic the noise level observed in the clinical images. The standard deviation of the observed background region intensity in the clinical images was approximately 5% of the voxel intensity of the brightest vessel. The bias field term \( b(\vec{x}) \) was obtained from the BrainWeb [1], and had three different models, namely, FIELD\( \text{A} \), FIELD\( \text{B} \) and FIELD\( \text{C} \). All of the bias fields had 20% INU. Slices of these bias fields are shown in Figures 2.5a, 2.6a and 2.7a respectively. The slices of the synthetic volume after applying the bias fields are shown in Figures 2.5b, 2.5c, 2.6b, 2.6c, 2.7b and 2.7c.

In this experiment, WLV-EDGE is compared with ST and GRADIENT using the same set of parameters as in the previous experiment. The comparison of performance is based on two criteria, (a) the edge strength variation and (b) the edge normal orientation estimation accuracy. For the first criterion, the edge strength variation is quantified by evaluating the standard deviation of edge strengths estimated by different methods on the tube surfaces. A large value of standard deviation of a method implies that the method performance can be easily affected by the bias fields. In contrast, a method is robust to changes of intensity contrast if a small value of standard deviation is observed. Considering different approaches returning edge strengths in different magnitude scales, for a fair comparison, the edge strengths computed from each method were normalized to have unit mean values. Similar to the previous experiment, the edge strengths of ST, GRADIENT and WLV-EDGE were calculated using the square root of the trace of the structure tensor, smoothed image gradient magnitude and Equation 2.12 respectively.

For the second criterion, the measurement of estimation accuracy is based on angular discrepancy between the estimated edge normal orientation and the ground truth normal orientation of the tube or torus surfaces. The ground truth orientation of the tube surfaces is defined as the direction pointing outward from the tube centerline to the tube surface or the torus centerline to the torus surface. Figure 2.4b and Figure 2.8(d) show the ground truth normal orientations of the tubes and tori respectively.
The angular discrepancy is measured in radian as

$$\arccos(|(\hat{n} \cdot \hat{g})|),$$ (2.22)

where $\hat{n}$ and $\hat{g}$ are the unit vectors of the estimated edge normal direction and the ground truth normal orientation respectively. In ST, the surface normal direction is obtained based on the first principle direction of the structure tensor. The surface normal direction of GRADIENT is computed as,

$$\frac{\nabla(G \ast I(\vec{x}))}{|\nabla(G \ast I(\vec{x}))|}$$ (2.23)

For WLV-EDGE, the edge normal direction is computed according to Equation 2.11.

In Tables 2.1 and 2.2, we list the standard deviations of the estimated edge strength computed by ST, GRADIENT and WLV-EDGE after the bias fields, FIELDa, FIELDb and FIELDc, have been applied. Each sub-column in Table 2.1 represents the results of ST, GRADIENT and WLV-EDGE with the tubes having the same radius but in different positions. Also, each sub-column in Table 2.2 represents the results of ST, GRADIENT and WLV-EDGE with the tori having the same $r$ but with different $R$ and at different positions. From the tables, it is observed that WLV-EDGE consistently gives the smallest standard deviation. This shows that, as compared with ST and GRADIENT, WLV-EDGE is more robust for estimating edge strength in the presence of intensity variation.

For the edge strength analysis, it is noticed that there is a relationship between GRADIENT and WLV-EDGE when $\sigma = \sigma_{\perp}$ (see Appendix A). This relationship also exists in the above synthetic experiments, in which $\sigma = \sigma_{\perp} = 2$. As shown in Appendix A, except the multiplicative constant $\frac{1}{Z}$ in Equation A.2, the main difference between WLV-EDGE and GRADIENT is the denominator of Equation A.2, $\min(\sqrt{\text{WLV}_{1,\hat{n}(\theta)}}, \sqrt{\text{WLV}_{2,\hat{n}(\theta)}})$. Therefore, for the edge strength computed by WLV-EDGE, the difference in intensity contrast between tubes and background regions is normalized by this denominator. Thus, different from GRADIENT, the edge strength computed by WLV-EDGE is robust to intensity inhomogeneity induced by the multiplicative effect of bias fields. In contrast, ST and GRADIENT are based on differential operators and can give relatively large variations in edge strength under the
effect of bias fields. For the analysis of edge normal orientation estimation, as listed in the WLV-EDGE rows of Tables III and 2.4 for tubes having different radii, tori having different values of $r$ and $R$, and under the effects of different bias fields, WLV-EDGE produces slightly smaller discrepancies than ST and GRADIENT. As a conclusion, WLV-EDGE offers superior robustness of edge strength estimation against intensity inhomogeneity without sacrificing the accuracy of edge normal orientation estimation.

### 2.3.2 Clinical image volumes for segmentation experiments

We present and compare the segmentation results obtained from WLV-FLUX and FLUX using four clinical image volumes, a TOF-MRA image volume (Figure 2.9) having the size of $188 \times 168 \times 39$ voxels; and three PC-MRA speed image volumes (Figures 2.12, 2.15 and 2.18) having the sizes of $130 \times 286 \times 52$ voxels, $67 \times 257 \times 35$ voxels and $104 \times 252 \times 64$ voxels respectively. The third image volume has voxel size $0.4\text{mm} \times 0.4\text{mm} \times 0.8\text{mm}$, and for the rest of image volumes, they have voxel size $0.4\text{mm} \times 0.4\text{mm} \times 1.0\text{mm}$. All the data sets were acquired using a Philips 3T ACS Gyroscan MR scanner at the University Hospital of Zurich, Switzerland. All image volumes are axial brain scans. The standard TOF-MRA and PC-MRA imaging protocols were used without contrast agents. Details are as follows. For the first volume, $TE/TR = 3.4/23\text{ms}$ and flip angle = $20^\circ$. For the second volume, $TE/TR = 5.2/15.4\text{ms}$ and flip angle = $9^\circ$. For the third volume, $TE/TR = 5.4/18.4\text{ms}$ and flip angle = $9^\circ$. For the fourth volume, $TE/TR = 5.0/14\text{ms}$ and flip angle = $9^\circ$.

In these four experiments, the set of detection radii $R$ for both FLUX and WLV-FLUX (Equations 2.16 and 2.19) was set to 0.4mm, 0.8mm, 1.2mm, 1.6mm, 2mm and 2.4mm. Due to the presence of narrow vessels, which involved a few voxels across the boundaries, the parameters of FLUX and WLV-FLUX were chosen for the detection of vessels having the width of one voxel. Such parameters included the scale of the Gaussian smoothing operation for FLUX in Equation 2.17, and $\sigma$ for WLV-FLUX used in Equations 2.1. These two parameters were set to 0.3mm (slightly smaller than the x-y plane voxel spacing). For $\sigma_{\perp}$ of WLV-FLUX, the value of 1.0mm (the slice spacing) was chosen. It aims to guarantee that the filters used in Equation 2.2 have an effective range to cover adequate number of voxels in all directions $n$ for Equation 2.4 and Equation 2.5.
The smoothness regularization term $\kappa$ in Equation 2.20 was set to 0.05 for WLV-FLUX in order to maintain the contour smoothness. In FLUX, we observe that this regularization term can neutralize the effect of the evolution speed term in Equation 2.20 and hence can halt contours at the undesired positions. Since this term is not necessarily beneficial to the segmentation results of FLUX, we present the best results obtained from FLUX using $\kappa = 0$ and $\kappa = 0.05$.

For these clinical cases, the same set of initial seed points for WLV-FLUX and FLUX was produced by global thresholding of each image volume. The thresholds were selected carefully to ensure that the seed points were located inside the major vessels to provide proper initial conditions to start both algorithms, WLV-FLUX and FLUX. The algorithms were then stopped when the accumulated per-voxel update of the level set function was less than $10^{-4}$ after 10 iterations.

In the first experiment on TOF-MRA image volume, as shown in Figure 2.9 and Figure 2.10, based on visual assessment between MIP images and segmented images, WLV-FLUX selects most of the vessels from the image volume and its results have no leakage. Conversely, the segmentation results of FLUX have leakages at several positions, as shown in Figure 2.11a. The results of FLUX with regularization parameter $\kappa = 0.05$ are presented because, when $\kappa = 0.05$, it encourages a smooth resultant contour and showed less leakages compared with the results using $\kappa = 0$ for this image volume. However, leakages still occurred at three positions in a low contrast branch. Three arrows point at the leakage positions of FLUX in the axial view of Figure 2.11a (top-left). In Figure 2.11b and Figure 2.11c, four slices of these leakage regions are shown and the positions of leakages are indicated by arrows. At these positions, contours propagated through the weak vessel boundaries and this resulted in leakages. The leakages continued to expand and contaminated the results in other regions. This leads to difficulty in visualizing the results. Therefore, instead of presenting the final segmentation results of FLUX, the results shown in Figure 2.11 are an intermediate step that leakages just began. In the axial projection of Figure 2.9 (top-left), there is a vessel segment at the bottom right position missed by FLUX, as shown in Figure 2.11a. It should be pointed out that FLUX can discover this segment after a number of iterations of contour evolution. However, results including this segment were contaminated by the leaked contours and cannot be shown here.
The cause of the leakages is that the weak vessel boundaries cannot halt the FLUX contour, which can be attracted by the boundaries of other tissues attached on the vessels. In WLV-FLUX, the edge strength estimated from WLV-EDGE retains large values in weak boundaries so that the contour is able to stop on these weak boundaries and the vasculature is successfully captured without leakage.

In the second experiment, as shown in Figure 2.12, a PC-MRA image volume was used. The results of FLUX using $\kappa = 0$ for Equation 2.20 are presented. The use of $\kappa = 0$ disables the smoothness regularization term and encourages the contour to evolve into branches. Even though this regularization term was disabled, FLUX missed some branches, as shown in Figure 2.14a. Arrows indicates the missing branches in Figure 2.14b and Figure 2.14c. Figure 2.13 shows that the segmentation results of WLV-FLUX are satisfactory, and the low intensity branches are included in the segmentation results.

For the third experiment, another PC-MRA image volume was used, as shown in Figure 2.15. Similar to the previous experiment, the results of FLUX using $\kappa = 0$ for Equation 2.20 are presented. It is observed that FLUX can only locate the major vessels, as shown in Figure 2.17a. The missing branches are pointed at by arrows in Figure 2.17b and Figure 2.17c. In contrast, better segmentation results are obtained using WLV-FLUX where low contrast branches are captured, as shown in Figure 2.16.

In Figures 2.14b, 2.14c, 2.17b and 2.17c, the image slices show the white arrows pointing at the low contrast vessels missed by FLUX in the second and third experiments. As shown in these figures, the contours are halted inside the branches rather than the boundaries of the branches. The main reason is that the boundaries of the low contrast branches do not provide gradient magnitude strong enough to attract contours for propagation. For WLV-FLUX, the contours can evolve into the low contrast vessels because WLV-EDGE is capable of returning large and consistent edge strength on those low contrast boundaries.

Finally, a PC-MRA image volume, as shown in Figure 2.18, was utilized in the forth segmentation experiment. The results of FLUX using $\kappa = 0.05$ for Equation 2.20 are presented. When $\kappa = 0.05$, it reduces the chance of leakages by encouraging smooth resultant contours. In this case, both FLUX and WLV-FLUX are able to
capture the vessels, as shown in Figure 2.19 and Figure 2.20. Nonetheless, \textbf{WLV-FLUX} gives better segmentation results in the low contrast branches compared with \textbf{FLUX} because, for \textbf{FLUX}, leakages occurred in the low contrast branches indicated by the arrows in Figure 2.20 and Figure 2.21. The low contrast vessel boundaries could not halt the contours accurately, which were attracted by the noise near the vessel boundaries. Unlike the first experiment, which shows an intermediate evolution step for \textbf{FLUX} (see Figure 2.11), the results shown in Figure 2.20 are the final segmentation results obtained using \textbf{FLUX} as the leakages in this experiment were stopped before they contaminated the results in other regions. Compared with \textbf{FLUX}, \textbf{WLV-FLUX} is able to halt the contours at the low contrast vessel boundaries.

We have measured the computation times required for segmenting blood vessels using \textbf{WLV-FLUX} and \textbf{FLUX} in the above four clinical image volumes. For \textbf{WLV-FLUX}, the measured time includes the computation of \textbf{WLV-EDGE} and estimation of $\mathcal{L}$ in Equation 2.20. For \textbf{FLUX}, the measured time includes the computation of \textbf{GRADIENT} and estimation of $\mathcal{L}$ in Equation 2.20. The implementations of both \textbf{WLV-FLUX} and \textbf{FLUX} were based on programs written using Microsoft Visual C++ in Windows XP 32-bit environment and run on a Pentium 4-HT 3.2GHz PC with 1GB RAM. As the image volumes have different sizes, rather than the average running time, the individual running times are reported. For \textbf{WLV-FLUX}, the running times for the first, second, third and forth experiments were 3711, 6449, 1924 and 5634 seconds respectively. For \textbf{FLUX}, the running times were 3500, 5737, 1725 and 5441 seconds respectively. It is observed that the running time for \textbf{WLV-FLUX} is only slightly longer (around 9\%) than \textbf{FLUX}.

\section*{2.4 Discussion}

\subsection*{2.4.1 Edge detectors}

The main contribution of this chapter is the proposal of using weighted local variance (WLV) to quantify the intensity similarity between two sides of an edge. For boundary detection, this intensity similarity measure is complemented with the intensity difference across an edge to formulate a new WLV based edge detection method, namely \textbf{WLV-EDGE}. It is shown to be robust to the changes in the intensity contrast of edges.
Figure 2.1: The plots of the values of $\sqrt{WLV_{1,n(\theta)}}$ (second column), $\sqrt{WLV_{2,n(\theta)}}$ (third column), and $\min(\sqrt{WLV_{1,n(\theta)}}, \sqrt{WLV_{2,n(\theta)}})$ (forth column) against different values of $\theta$ obtained from horizontal edges having different levels of intensity contrast, a corner and two edges of different values of curvature.

Figure 2.2: (a) An example of 24 discrete 2D orientation samples for calculation of confidence values. (b) An example of 281 discrete 3D orientation samples for calculation of confidence values.
Figure 2.3: Single synthetic tube. (a) The 50th slice of the image volume, \( z = 50 \). (b) The 12th slice of the image volume, \( x = 12 \). (c) From top to bottom: The intensity profile of the tube along the \( z \)-direction with \( x = 12 \) and \( y = 10 \); The square root of the trace of the structure tensor computed by \( \text{ST} \); The \text{GRADIENT} magnitude; The edge strength computed by \( \text{WL V} \). (d) Along the \( z \)-direction with \( x = 12 \), it shows the slices of edge strengths detected by (from top to bottom) \( \text{ST}; \text{GRADIENT}; \text{WL V} \).

Figure 2.4: Multiple synthetic tubes. (a) The 50th slice of the image volume (181 × 217 × 181 voxels) consists of 25 synthetic tubes of radii equal to 2, 3, 4, 5 and 6. (b) An example of the ground truth normal orientations on the tube surfaces.
Figure 2.5: (a) The 50th slice of the bias field A obtained from the Brainweb [1]. (b) The 50th slice of the image volume after applying the bias field model A without noise. (c) The 50th slice of the image volume after applying the bias field model A and the additive Gaussian noise with the scale parameter equal to 0.05.

Figure 2.6: (a) The 50th slice of the bias field B obtained from the Brainweb [1]. (b) The 50th slice of the image volume after applying the bias field model B without noise. (c) The 50th slice of the image volume after applying the bias field model B and the additive Gaussian noise with the scale parameter equal to 0.05.

Figure 2.7: (a) The 50th slice of the bias field C obtained from the Brainweb [1]. (b) The 50th slice of the image volume after applying the bias field model C without noise. (c) The 50th slice of the image volume after applying the bias field model C and the additive Gaussian noise with the scale parameter equal to 0.05.
Figure 2.8: The third synthetic and numerical volume consists of 18 non-overlapping tori. The size of each torus is defined by two parameters $r$ and $R$ as shown in (d). (d) The description of the setting of a torus. $r$ specifies the shortest distance between the centerline of the torus (denoted as the dashed line inside the torus) and the surface of the torus. $R$ determines the radius of the torus centerline. $\hat{n}_1$ and $\hat{n}_2$ are two examples of the ground truth normal orientations of the torus surface. They are pointing outward from the torus centerline to the surface. Using the convention $\{r, R\}$, the sizes of all tori (in voxel) are $\{2, 12\}, \{2, 24\}, \{2, 36\}, \{2, 48\}, \{2, 60\}, \{2, 72\}, \{3, 18\}, \{3, 32\}, \{3, 46\}, \{3, 60\}, \{3, 76\}, \{4, 24\}, \{4, 48\}, \{4, 64\}, \{5, 48\}, \{5, 66\}, \{6, 54\}$ and $\{6, 72\}$. (a) A top-view of the torus iso-surfaces. (b) An elevation-view of the torus iso-surfaces. (c) A side-view of the torus iso-surfaces. In this subfigure, there are five layers. The top layer has two tori of $r = 6$, the next layer has two tori of $r = 5, \ldots$, and the bottom layer has six tori of $r = 2$. 
<table>
<thead>
<tr>
<th>Bias field models</th>
<th><strong>FIELDA</strong></th>
<th><strong>FIELDB</strong></th>
<th><strong>FIELDC</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Tube radius = 2 voxels</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.2317</td>
<td>0.2253</td>
<td>0.2400</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.2326</td>
<td>0.2206</td>
<td>0.2316</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.1701</td>
<td>0.1589</td>
<td>0.1581</td>
</tr>
<tr>
<td>Tube radius = 3 voxels</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.1751</td>
<td>0.1692</td>
<td>0.2001</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.1792</td>
<td>0.1752</td>
<td>0.1995</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.1344</td>
<td>0.1353</td>
<td>0.1350</td>
</tr>
<tr>
<td>Tube radius = 4 voxels</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.1224</td>
<td>0.1153</td>
<td>0.1428</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.1354</td>
<td>0.1043</td>
<td>0.1474</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.0874</td>
<td>0.0608</td>
<td>0.0997</td>
</tr>
<tr>
<td>Tube radius = 5 voxels</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.1599</td>
<td>0.1574</td>
<td>0.1685</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.1664</td>
<td>0.1501</td>
<td>0.1731</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.1010</td>
<td>0.0653</td>
<td>0.0831</td>
</tr>
<tr>
<td>Tube radius = 6 voxels</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.1720</td>
<td>0.1602</td>
<td>0.1984</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.1764</td>
<td>0.1501</td>
<td>0.2045</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.1010</td>
<td>0.0653</td>
<td>0.0857</td>
</tr>
</tbody>
</table>

Table 2.1: Standard deviations of edge strengths computed by different methods after applying different bias field models **FIELDA**, **FIELDB** and **FIELDC** on the volume shown in Figure 2.4. The edge strengths computed from each method, **ST**, **GRADIENT** and **WLV-EDGE**, are normalized to have unit mean values.
<table>
<thead>
<tr>
<th>Bias field models</th>
<th>FIELDA</th>
<th>FIELD B</th>
<th>FIELD C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tori, $r = 2$, $R = 12, 24, 36, 48, 60, 72$ voxels</td>
<td>ST</td>
<td>0.1611</td>
<td>0.0795</td>
</tr>
<tr>
<td></td>
<td>GRADIENT</td>
<td>0.1460</td>
<td>0.0855</td>
</tr>
<tr>
<td></td>
<td>WLV-EDGE</td>
<td>0.1345</td>
<td>0.0490</td>
</tr>
<tr>
<td>Tori, $r = 3$, $R = 18, 32, 46, 60, 76$ voxels</td>
<td>ST</td>
<td>0.0837</td>
<td>0.0844</td>
</tr>
<tr>
<td></td>
<td>GRADIENT</td>
<td>0.0929</td>
<td>0.0864</td>
</tr>
<tr>
<td></td>
<td>WLV-EDGE</td>
<td>0.0633</td>
<td>0.0630</td>
</tr>
<tr>
<td>Tori, $r = 4$, $R = 24, 48, 64$ voxels</td>
<td>ST</td>
<td>0.1005</td>
<td>0.0764</td>
</tr>
<tr>
<td></td>
<td>GRADIENT</td>
<td>0.0894</td>
<td>0.0808</td>
</tr>
<tr>
<td></td>
<td>WLV-EDGE</td>
<td>0.0648</td>
<td>0.0645</td>
</tr>
<tr>
<td>Tori, $r = 5$, $R = 48, 66$ voxels</td>
<td>ST</td>
<td>0.0941</td>
<td>0.1321</td>
</tr>
<tr>
<td></td>
<td>GRADIENT</td>
<td>0.1045</td>
<td>0.1057</td>
</tr>
<tr>
<td></td>
<td>WLV-EDGE</td>
<td>0.0998</td>
<td>0.1001</td>
</tr>
<tr>
<td>Tori, $r = 6$, $R = 54, 72$ voxels</td>
<td>ST</td>
<td>0.1304</td>
<td>0.1128</td>
</tr>
<tr>
<td></td>
<td>GRADIENT</td>
<td>0.1275</td>
<td>0.1003</td>
</tr>
<tr>
<td></td>
<td>WLV-EDGE</td>
<td>0.0800</td>
<td>0.0732</td>
</tr>
</tbody>
</table>

Table 2.2: Standard deviations of edge strengths computed by different methods after applying different bias field models **FIELDA**, **FIELD B** and **FIELD C** on the volume shown in Figure 2.8. The edge strengths computed from each method, **ST**, **GRADIENT** and **WLV-EDGE**, are normalized to have unit mean values.

Figure 2.9: The MIP images of a TOF-MRA image volume (First row: axial projection; second row: coronal projection and sagittal projection). This TOF-MRA image volume has dimension 188 × 168 × 39 voxels and voxel size 0.4mm×0.4mm×1.0mm.
<table>
<thead>
<tr>
<th>Bias field models</th>
<th>No bias field</th>
<th>FIELD A</th>
<th>FIELD B</th>
<th>FIELD C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tube radius = 2 voxels</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.0186 (0.0099)</td>
<td>0.0221 (0.0180)</td>
<td>0.0139 (0.0085)</td>
<td>0.0130 (0.0084)</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.0205 (0.0093)</td>
<td>0.0225 (0.0092)</td>
<td>0.0138 (0.0071)</td>
<td>0.0129 (0.0018)</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.0162 (0.0073)</td>
<td>0.0201 (0.0077)</td>
<td>0.0099 (0.0052)</td>
<td>0.0110 (0.0020)</td>
</tr>
<tr>
<td>Tube radius = 3 voxels</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.0155 (0.0103)</td>
<td>0.0222 (0.0132)</td>
<td>0.0107 (0.0053)</td>
<td>0.0112 (0.0063)</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.0169 (0.0075)</td>
<td>0.0216 (0.0070)</td>
<td>0.0124 (0.0058)</td>
<td>0.0114 (0.0060)</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.0130 (0.0058)</td>
<td>0.0168 (0.0058)</td>
<td>0.0091 (0.0042)</td>
<td>0.0104 (0.0073)</td>
</tr>
<tr>
<td>Tube radius = 4 voxels</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.0142 (0.0088)</td>
<td>0.0198 (0.0093)</td>
<td>0.0156 (0.0091)</td>
<td>0.0125 (0.0078)</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.0169 (0.0090)</td>
<td>0.0202 (0.0068)</td>
<td>0.0161 (0.0062)</td>
<td>0.0128 (0.0062)</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.0123 (0.0055)</td>
<td>0.0148 (0.0051)</td>
<td>0.0104 (0.0038)</td>
<td>0.0125 (0.0083)</td>
</tr>
<tr>
<td>Tube radius = 5 voxels</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.0246 (0.0114)</td>
<td>0.0285 (0.0123)</td>
<td>0.0296 (0.0145)</td>
<td>0.0252 (0.0131)</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.0268 (0.0111)</td>
<td>0.0282 (0.0107)</td>
<td>0.0310 (0.0148)</td>
<td>0.0260 (0.0112)</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.0230 (0.0099)</td>
<td>0.0240 (0.0124)</td>
<td>0.0296 (0.0155)</td>
<td>0.0242 (0.0117)</td>
</tr>
<tr>
<td>Tube radius = 6 voxels</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>0.0240 (0.0106)</td>
<td>0.0243 (0.0166)</td>
<td>0.0240 (0.0124)</td>
<td>0.0232 (0.0126)</td>
</tr>
<tr>
<td>GRADIENT</td>
<td>0.0248 (0.0103)</td>
<td>0.0266 (0.0103)</td>
<td>0.0246 (0.0120)</td>
<td>0.0222 (0.0109)</td>
</tr>
<tr>
<td>WLV-EDGE</td>
<td>0.0205 (0.0089)</td>
<td>0.0216 (0.0090)</td>
<td>0.0224 (0.0114)</td>
<td>0.0200 (0.0089)</td>
</tr>
</tbody>
</table>

Table 2.3: Angular discrepancies (in radian) between ground truth normal orientation and the edge normal orientation estimated by ST, GRADIENT and WLV-EDGE on the surfaces of the synthetic tubes, as shown in Figure 2.4.
Table 2.4: Angular discrepancies (in radian) between ground truth normal orientation and the edge normal orientation estimated by ST, GRADIENT and WLV-EDGE on the surfaces of the synthetic tubes, as shown in Figure 2.8.
Figure 2.10: Segmentation results of **WLV-FLUX**.

Figure 2.11: (a) Segmentation results of **FLUX**. (b) The 14th and 15th axial slices; (c) The 11th and 12th axial slices of the TOF-MRA image volume and the contours are corresponding to the segmentation results, as shown in (a). The white arrows in (a) are corresponding to the positions pointed at by the white arrows in (b) and (c) where leakages occur.
Figure 2.12: The MIP images of a PC-MRA speed image volume (First row: axial projection; second row: coronal projection and sagittal projection). This image volume has dimension $130 \times 286 \times 52$ voxels and voxel size $0.4\text{mm} \times 0.4\text{mm} \times 1.0\text{mm}$.

Figure 2.13: Segmentation results of WLV-FLUX.
Figure 2.14: Segmentation results of FLUX. (b) The 26th and 27th axial slices; (c) The 40th and 41st axial slices of the PC-MRA image volume and the contours are corresponding to the segmentation results, as shown in (a). The white arrows in (b) and (c) indicate the low contrast vessels, which are missed by FLUX.

and that it can give high and consistent edge strength for low contrast boundaries. As such, this method is ideal for handling low contrast vessel boundaries. Since the proposed method considers intensity similarity of local regions between both sides of an edge, it is different from other differential operators for edge detection, such as Sobel filters, Roberts filters, Prewitt filters, filters used by Canny edge detection scheme [16], \( \lambda_r \)-Space Representation [57], quadrature filters [9], difference of a Gaussian [58], Laplacian of Gaussian [47] as well as the Hessian matrix. Their detection principles rely entirely on the intensity changes across boundaries, and therefore, they can only return small edge strength responses for low contrast edges. Our method, on the other hand, not only relies on the intensity change across boundaries but also takes into account the intensity homogeneity in two local regions separated by an edge.

2.4.2 Edge detection and segmentation

In our experiments, the segmentation results of WLV-FLUX are compared against FLUX. The major distinction between FLUX and WLV-FLUX is that they use different edge information. FLUX utilizes the smoothed image gradient (Equation
Figure 2.15: The MIP images of a PC-MRA speed image volume (First row: axial projection; second row: coronal projection and sagittal projection). This image volume has dimension $67 \times 257 \times 35$ voxels and voxel size $0.4\text{mm} \times 0.4\text{mm} \times 0.8\text{mm}$.

Figure 2.16: Segmentation results of \textbf{WLV-FLUX}.

Figure 2.17: Segmentation results of \textbf{FLUX}. (b) The 25th and 26th axial slices; (c) The 29th and 30th axial slices of the PC-MRA image volume and the contours are corresponding to the segmentation results, as shown in (a). The white arrows in (b) and (c) indicate the low contrast vessels, which are missed by \textbf{FLUX}.
Figure 2.18: The MIP images of a PC-MRA speed image volume (First row: axial projection; second row: coronal projection and sagittal projection). This image volume has dimension $104 \times 252 \times 64$ voxels and voxel size $0.4\text{mm} \times 0.4\text{mm} \times 1.0\text{mm}$.

2.17) and WLV-FLUX employs WLV-EDGE (Equation 2.18). The mathematical comparison between the smoothed image gradient and the confidence value (as defined in Equation 2.5) of WLV-EDGE is provided in Appendix-A, where the confidence value is written as,

$$R_{\hat{n}}(\vec{x}) = \left( \frac{1}{Z} \right) \left( \frac{f_{\hat{n}}(\vec{x}) \ast I(\vec{x})}{\sqrt{\min(WLV_{1,\hat{n}}(\vec{x}), WLV_{2,\hat{n}}(\vec{x})) + \epsilon}} \right),$$  (2.24)

where $Z = - \int f_{1,\hat{n}}(\vec{u}) d\vec{u} = \int f_{2,\hat{n}}(\vec{u}) d\vec{u}$ is a constant. In the above equation, the confidence value $R_{\hat{n}}(\vec{x})$ of WLV-EDGE is composed of three parts: a constant $\frac{1}{Z}$; the numerator, which is based on the anisotropically ($\sigma \neq \sigma_\perp$) or isotropically ($\sigma = \sigma_\perp$) smoothed image gradient along direction $\hat{n}$; and the denominator, which involves WLVs. Analogous to the discussion in Section 2.2.2, the denominator, which is governed by the values of WLV, enables the confidence value of WLV-EDGE to (a) be more robust against the change in intensity contrast and (b) have strong and consistent edge strength responses on low contrast edges. Therefore, WLV-EDGE enables WLV-FLUX to capture low contrast branches in the experiments but such branches are not discovered by FLUX.

It is worth mentioning that smoothing image prior to the calculation of image gradient, as suggested in the gradient based approaches [18, 23–25, 27–30, 32, 34, 35,
Figure 2.19: Segmentation results of **WL-V-FLUX**.

Figure 2.20: Segmentation results of **FLUX**.

Figure 2.21: Segmentation results of **FLUX**. (a) The 17th and 18th coronal slices; (b) The 25th and 26th coronal slices of the PC-MRA image volume and the contours are corresponding to the segmentation results, as shown in Figure 2.20. The white arrows in (a) and (b) indicate the low contrast vessels, which are missed by **FLUX**.
\( \nabla (G \ast I) = (\nabla G) \ast I \), yields a connection to the filters used in Canny edge detector [16]. This is because they are grounded on applying directional derivatives of a Gaussian function along the x, y and z directions in an image. As such, the calculation of image gradient can be viewed as a procedure of edge detection prior to performing segmentation. To segment vessels in low contrast regions with the presence of intensity inhomogeneity, these segmentation methods need an edge detector, which is able to detect low contrast edges. Therefore, WLV-EDGE, which returns high and consistent edge strength for low contrast edges, is suitable for segmentation of vessels in low contrast regions.

2.4.3 Future extension – shape and structural information

At the current stage, the proposed method does not analyze the shape and local structures of boundaries. Currently, the focus of this chapter is to propose and validate a framework for boundary detection based on (a) intensity similarity among regions on both sides of an edge and (b) intensity difference across the edge, without considering the shape and local structures of boundaries. Utilizing shape information or structural information along with WLV-EDGE to provide more advanced features for vessel boundary detection is the future direction of this research. For instance, the eigenvalues along different principle directions of the matrix stated in Equation 2.10 can provide structural information of boundaries; the proposed method can be combined with the Hessian matrix (details are provided in Appendix B) for the analysis of the shape and local structures of boundaries; using the minimum curvature instead of the mean curvature in Equation 2.20, i.e. using the codimension two level set approach [33], can enhance the accuracy of WLV-FLUX in the extraction of tubular shaped vessels.

2.5 Conclusion

In summary, the robustness of WLV-EDGE against the change of intensity contrast of edges, and the capability of returning high and consistent edge strength on low contrast edges has been shown to be beneficial for the segmentation of vasculatures in low contrast regions. Such robustness of WLV-EDGE is validated in experiments
using three synthetic volumes. Moreover, in the presence of intensity inhomogeneity, such as bias field, the presence of other tissue or reduced speed related vessel intensity for narrow vessels in PC-MRA, \textbf{WLV-FLUX} makes use of the edge information of \textbf{WLV-EDGE} and is less likely than \textbf{FLUX} to have leakages or discontinuities in the four clinical segmentation experiments. It is experimentally shown that the WLV based edge detection approach can achieve high quality segmentation of vasculatures in magnetic resonance angiography.
CHAPTER 3

EFFICIENT IMPLEMENTATION FOR SPHERICAL FLUX COMPUTATION AND ITS APPLICATION TO VASCULAR SEGMENTATION

3.1 Introduction

In vector calculus, flux of a vector field over the closed boundary of a region is the quantification of the amount of the vectors, which flow into or out of that region along the boundary surface normal direction. Mathematically, in Euclidean space, the flux of a vector field $\vec{v}$ over the closed boundary of a region $C$ is defined as

$$\text{flux}_C = \int_{\partial C} \langle \vec{v}, \hat{n}_A \rangle \, dA,$$  

where $\partial C$ is the closed boundary of $C$, $\hat{n}_A$ is the outward normal of $\partial C$ and $dA$ is the infinitesimal area on $\partial C$. By specifying the vector field $\vec{v}$ as the gradient vector field of a gray scale image $I$, we have, by the divergence theorem,

$$\text{flux}_C = \int_{\partial C} \langle \vec{v}, \hat{n}_A \rangle \, dA \equiv \int_C \Delta I \, dV,$$

where $dV$ is the infinitesimal volume in $C$. The computation of flux associated with arbitrary regions is widely used for digital image analysis [18, 19, 39, 44, 46, 52, 62–71].

In particular, for some applications such as analysis of tubular structures, it is useful to compute flux in either a two-dimensional (2D) circular region or a three-dimensional (3D) spherical region. For instance, Siddiqi et al. have proposed to compute flux in discs or spheres of different radii to aid in obtaining the skeletons of target objects [63]. Pizer et al. have presented the use of flux for recognizing medial loci [64]. In [62], the disc-based or sphere-based calculations of flux have been utilized for centerline extraction of blood vessels in magnetic resonance angiography and colons in computed tomography. For segmentation of tubular structures, e.g. blood vessels in the brain,
Vasilevskiy and Siddiqi have proposed flux maximizing geometric flows [18] by approximating the divergence of image gradient as the flux estimated over the closed boundaries of discs or spheres of different radii. This segmentation algorithm is useful in extracting small and dim blood vessels because of its sensitivity to low contrast and narrow tubular structures. Its performance has been proved and validated as it has been employed for generating the ground truth for the BrainWeb project [53], which is widely used in medical image segmentation validation. In addition, Audette and Chinzei have extended the flux maximizing geometric flows for tissue identification [65].

Among the aforementioned techniques, it is essential to compute flux in discs or spheres. In this thesis, flux computed in a spherical region is called spherical flux. The continuous form of the spherical flux \( s(\vec{s}; r) \) is given as,

\[
s(\vec{x}; r) = \frac{1}{4\pi r^2} \int_{\partial B_r} \vec{v}(\vec{x} + r\hat{n}_A) \cdot \hat{n}_A dA,
\]

where \( B_r \) represents a spherical region of radius \( r \); \( dA \) is the infinitesimal area on boundary \( \partial B_r \); \( \hat{n}_A \) is the boundary surface normal vector at position \( r\hat{n}_A \); and \( \vec{v} \) is the gradient of an image \( I \). In practice, \( \vec{v} \) is obtained from a Gaussian smoothed image, which ensures the differentiability of the discrete image signal \( I \). In a discrete form, Equation 3.1 may be computed numerically as,

\[
s(\vec{x}; r) \approx \frac{1}{|K_r|} \sum_{k \in K_r} \vec{v}(\vec{x} + r\hat{k}) \cdot \hat{k},
\]

where \( K_r \) is a set of \( |K_r| \) outward normal samples swapping across the surface of \( \partial B_r \).

It is straightforward to estimate the values of spherical flux for an image using Equation 3.2 in the spatial domain. This is called the conventional spatial implementation hereafter. Although it is easy to implement the conventional spatial formulation, the computation time taken by the conventional spatial implementation is unsatisfactorily long when the radius \( r \) is large. The reason is that a large value of \( r \) boosts the boundary surface area of the spherical region. Thus, in Equation 3.2, a considerable number of orientation samples is needed to precisely approximate the surface integral, as stated in Equation 3.1.

To reduce the time required to compute spherical flux, we propose a more efficient implementation, which is formulated in the Fourier domain. It avoids the orientation
sampling for the discretization of the surface integral, which is given in Equation 3.1, and thus dramatically reduces the running time complexity of the spherical flux computation. The computation time required for the proposed implementation has been compared with that of the conventional spatial implementation (Equation 3.2) using both synthetic and clinical image volumes of various sizes. The computation accuracies of both implementations have also been validated. It is experimentally shown that the proposed efficient implementation is accurate and capable of radically reducing the running time for the spherical flux computation. The proposed implementation can definitely benefit flux-based applications such as tubular structure analysis. This is because techniques for such applications grounded on spherical flux, for example flux maximizing geometric flows [18], can have significant computation time reduction by employing the proposed implementation.

3.2 Conventional spatial implementation: orientation sampling strategy and time complexity

This section gives the orientation sampling strategy and the time complexity of the conventional spatial implementation for estimating the spherical flux \( s(\vec{s}, r) \) using the Equation 3.2. On the boundary surface of a spherical region \( \partial B_r \), with \(|\mathbb{K}_r|\) orientation samples, the sum of dot products of the image gradient \( \vec{v} \) and outward normals is computed for approximating the surface integral given in Equation 3.1. Bilinear interpolation is employed to retrieve the vector \( \vec{v} \) in Equation 3.2 if its position is not on the exact grid positions. Intuitively, the accuracy of the conventional spatial implementation is closely related to the number of orientation samples \(|\mathbb{K}_r|\) taken in Equation 3.2.

The size of the orientation sample set \( \mathbb{K}_r \) is a trade-off between the computation time and the accuracy. It is natural to have one orientation sample for each unit area in the unit of voxel-length on the surface of the spherical region \( \partial B_r \) for a good balance between the computation time and the accuracy. Therefore, the number of orientation samples taken by the conventional spatial implementation \( K \) is specified to ensure that there is at least one orientation sample for each unit area in the unit of voxel-length on the sphere surface. The orientation samples are organized in a grid fashion along the longitudinal and the latitudinal directions. For a sphere with radius \( r \), there are
angularly equally spaced latitudinal levels. These latitudinal levels are specified by elevation angles within a range \((-\pi/2, \pi/2]\). For the \(l\)-th latitudinal level with an elevation angle \(\theta_l\), there are \([2\pi r \cos \theta_l]\) orientation samples taken on the circumference of the circle associated with the \(l\)-th latitudinal level.

Using the above orientation sampling strategy, the number of orientation samples, \(|K_r|\), is directly proportional to the surface area of the sphere, i.e., \(|K_r| \propto r^2\). These orientation samples are roughly equally spaced for the spatial computation of spherical flux. Therefore, based on Equation 3.2, the time complexity for computing the spherical flux using the conventional spatial implementation in the entire image domain with \(m\) voxels is

\[
O(|K_r|m) = O(r^2m).
\]  

As the computation time increases quadratically with the value of \(r\), it hinders the implementation of multiscale spherical flux with different radii, especially when the radius \(r\) is large. This will be demonstrated in Section 3.4.

### 3.3 Methodology

In this section, we present our proposed implementation of the spherical flux in the Fourier domain (Section 3.3.1) [93]. The fast Fourier transform algorithm, which is used in our implementation, is also described in Section 3.3.2. Finally, for the detection of blood vessels of different widths, multiscale spherical flux is utilized and a method for reducing computation time is presented in Section 3.3.3.

#### 3.3.1 Our Efficient Implementation in the Frequency Domain

Although the estimation of spherical flux based on Equation 3.2 is straightforward in the spatial domain, as described in Section 3.1 and Section 3.2, the computation speed is unsatisfactorily slow when the size of the sphere \(r\) is large. To shorten the time for computing the spherical flux, we propose to reformulate the calculation of the spherical flux using the divergence theorem and spherical step function \(B(\vec{x}; r)\). As such, the spherical flux for the entire image is computed based on the convolution between the input image and a function. This function embeds the second derivative operator, the Gaussian smoothing function and the spherical step function \(B(\vec{x}; r)\).
Then, using the convolution theorem, the spherical flux can be estimated by multiplying the Fourier expressions of the image and that function, which are calculated using the fast Fourier transform algorithm. As is shown in the experiments (see Section 3.4), our implementation can lead to a significant computation gain, in particular when the sphere radius \( r \) is large.

**Reformulating the spherical flux computation using convolution operations**

We rewrite Equation 3.1 into a convolution operation between the input image \( I \) and a spatial function. By employing the divergence theorem and using a spherical step function,

\[
 s(\vec{x}; r) = \frac{1}{4\pi r^2} \int_{B_r} \text{div}(\vec{v}(\vec{x} + r\hat{n}_A))dA,
 = \int_{\Omega} B(\vec{y}_V; r) \left( \text{div}(\vec{v}(\vec{x} + \vec{y}_V)) \right) dV,
\]

(3.4)

where \( \Omega \) is the entire image domain, \( dV \) is the infinitesimal volume in the image domain, \( \vec{y}_V \) is a position inside the image domain, \( B(\vec{x}, r) \) is a spherical step function, which is given as

\[
 B(\vec{x}; r) = \begin{cases} 
 (4\pi r^2)^{-1}, & ||\vec{x}|| \leq r, \\
 0, & \text{otherwise}
 \end{cases}
\]

(3.5)

The main purpose of introducing the spherical step function in the above equation is to allow us to reformulate the calculation of spherical flux using the convolution operation. From Equation 3.4, we have

\[
 s(\vec{x}; r)
 = \int_{\Omega} B(\vec{y}_V; r) \left[ \text{div}(\vec{v}(\vec{x} + \vec{y}_V)) \right] dV,
 = \int_{\Omega} B(\vec{y}_V; r) \left( \Delta(G * I)(\vec{x} + \vec{y}_V) \right) dV,
 = (\Delta(B * G) * I)(\vec{x}; r)
\]

(3.6)

where \( * \) represents the convolution operator, \( \Delta \) represents the Laplacian operator and \( \vec{v} \) is replaced by the Gaussian smoothed image gradient \( \vec{\nabla}(G * I)(\vec{x}) \).
Evaluating the spherical flux in the Fourier domain

The above derivation leads to a convolution between the image $I$ and the function $\Delta(B \ast G)$. The convolution is implemented as a multiplication in the Fourier domain. Such multiplication merely involves the Fourier transformed image and the Fourier expression of the function of $\Delta(B \ast G)$. Denote $\mathcal{F}\{}$ and $\mathcal{F}^{-1}\{}$ be the Fourier transform and the inverse Fourier transform operators, and $\vec{\mu} = [\mu_1, \mu_2, \mu_3]^T$ is the position vector in the frequency domain, the value along each dimension $\vec{\mu}$ are in "cycle per voxel-length" and in a range of $[-0.5, 0.5)$. As such, $\mathcal{F}\{}I\mathcal{\}}(\vec{u})$ represents the Fourier coefficient of $I$ at the frequency $\vec{\mu}$ and $\mathcal{F}^{-1}\{}\mathcal{F}\{}I\mathcal{\}}(\vec{x}) \equiv I(\vec{x})$. To facilitate the discussion, we define $\tilde{\Upsilon}(\vec{\mu}; r) = \mathcal{F}\{}\Delta(B \ast G)\mathcal{\}}(\vec{u}; r)$. It is noted that $\tilde{\Upsilon}(\vec{\mu}; r)$ embodies three components, including the Fourier expression of the Laplacian operator $\Delta$ (which is equivalent to three second derivative operators), the Fourier expression of the Gaussian smoothing function $G$ and that of the spherical step function $B$. $\tilde{\Upsilon}(\vec{\mu}; r)$ is calculated by applying the Fourier transform on $\Delta G(\vec{x})$ and using three-dimensional Hankel transforms \[72\] to acquire the Fourier expression of $B(\vec{x}; r)$.

\[ \tilde{\Upsilon}(\vec{\mu}; r) = e^{-2(\pi ||\vec{\mu}||\sigma)^2 \frac{1}{r}} \left( \cos(2\pi r ||\vec{\mu}||) - \frac{\sin(2\pi r ||\vec{\mu}||)}{2\pi r ||\vec{\mu}||} \right). \]  

(3.7)

$\tilde{\Upsilon}(\vec{\mu}; r)$ is a spherically symmetric and real function, and its magnitude decays along the radial direction (see an example of $\tilde{\Upsilon}(\vec{\mu}; r)$ in Figure 3.1).

By Equations 3.6 and 3.7, the spherical flux can be calculated as,

\[ \mathcal{F}^{-1}\{} \left( \tilde{\Upsilon}(\vec{\mu}; r) \right) \mathcal{F}\{}I\mathcal{\}}(\vec{u}) \mathcal{\} \right), \]  

(3.8)

Given an image consist of $m$ voxels, the complexity is $O(m \log m)$ for both $\mathcal{F}\{}$ and $\mathcal{F}^{-1}\{}$, and $O(m)$ for the coefficient multiplication. Therefore, the complexity of the proposed implementation is,

\[ O(m \log m). \]  

(3.9)

The running time of the proposed implementation is independent of the value of $r$, i.e., the size of the sphere. Theoretically, our implementation has speed advantage over the conventional spatial implementation when $r$ is large and $m$ is small. As we will show in Section 3.4, the proposed implementation is much faster than the conventional spatial implementation, even when $s$ is small ($r = 1$) and $m$ is sufficiently large (details can
Figure 3.1: An example of the function $\tilde{\Upsilon}([\mu_1, \mu_2, 0]^T; 12)$. (a) The plot of the coefficients for the plane $[\mu_1, \mu_2, 0]^T$ in the Fourier domain. (b) The plot of the coefficients for the line $[\mu_1, 0, 0]^T$ in the Fourier domain. The shaded regions are not covered by the Subband 1. (c) The zooming of (b).
be found in Section 3.4) in the vascular image analysis applications, assuming that the orientation sampling strategy as described in Section 3.2 is used.

Applying $\tilde{T}(\bar{x}; r)$ to discrete image signals

In the above formulation, the power spectral density of $\tilde{T}(\bar{\mu}; r)$ is oscillating with a decaying magnitude in the entire Fourier domain and thus, $\tilde{T}(\bar{\mu}; r)$ is not a bandlimited signal. Nonetheless, when $\tilde{T}(\bar{\mu}; r)$ is multiplied with a bandlimited signal $F\{I\}(\bar{\mu})$, the components of $\tilde{T}(\bar{\mu}; r)$ at the frequency region beyond the bandwidth of $F\{I\}(\bar{\mu})$ are discarded. As such, the computation result of Equation 3.8 is inaccurate since it is not based on a perfectly reconstructed spherical flux function.

It is necessary to quantify the error incurred by applying the image signal to the spherical flux function which is not perfectly reconstructed. This error should be suppressed without introducing an excessive computational load. Denote $\hat{I}$ is the unknown original image signal before being captured by the imaging device. The discrete image signal $I(\bar{x})$ is acquired by performing discrete sampling on the original image signal $\hat{I}(\bar{x})$, i.e.

$$I(\bar{x}) = \hat{I}(\bar{x}) \left( \sum_{\bar{y} \in \cap} \delta(\bar{x} - \bar{y}) \right), \quad (3.10)$$

where $\cap(\bar{x}, \bar{y})$ is a set of integer coordinates in a cuboid, in which $\bar{x}$ is the minimum x-, y- and z-coordinates and $\bar{y}$ is the maximum x-, y- and z-coordinates. When the analytical form of spherical flux is applied to the discrete image signal, the same sampling procedure is implicitly taken on the spherical flux in the spatial domain. Denote the

\[\text{1The bandwidth of a spatial-discrete image signal can be determined by the image voxel spacings (i.e. the spatial distance between adjacent signal samples) according to the Nyquist sampling theorem [73].}\]
sampled version of $\tilde{\Upsilon}(\mu; r)$ is $\Upsilon(\mu; r)$, i.e.

$$\Upsilon(\mu; r) = \tilde{\Upsilon}(\mu; r) \ast \mathfrak{F} \left\{ \sum_{\bar{y} \in \mathbb{N}} \delta(\bar{x} - \bar{y}) \right\} (\mu)$$

$$= \tilde{\Upsilon}(\mu; r) \ast \left( \sum_{\bar{y} \in \mathbb{N}} \delta(\mu - \bar{v}) \right)$$

$$= \sum_{\bar{y} \in \mathbb{N}} \tilde{\Upsilon}(\mu - \bar{v}; r).$$ \hspace{1cm} (3.11)

An accurate estimate of the spherical flux based on the discrete image signal $I$ is,

$$s(\bar{x}; r) = \mathfrak{F}^{-1} \left\{ \mathfrak{F} \{ I \} (\mu) \left( \sum_{\bar{y} \in \mathbb{N}} \tilde{\Upsilon}(\mu - \bar{v}; r) \right) \right\} (\bar{x}).$$ \hspace{1cm} (3.12)

To simplify the discussion, we denote this accurate analytical form of spherical flux as $\Upsilon^\infty(\mu; r)$, which has the same bandwidth as the input image $I$, i.e.

$$\Upsilon^\infty(\mu; r) = \sum_{\bar{y} \in \mathbb{N}} \tilde{\Upsilon}(\mu - \bar{v}; r).$$ \hspace{1cm} (3.13)

When $\Upsilon^\infty(\mu; r)$ is applied to the Fourier transformed image signal $\mathfrak{F} \{ I \} (\mu)$, the value along each dimension of $\mu$ is in the range of $[0.5, 0.5)$, the same as the image signal bandwidth. The computation of $\Upsilon^\infty(\mu; r)$ is based on the infinite bandwidth of $\tilde{\Upsilon}(\mu; r)$.

It is noted that $\Upsilon^\infty(\mu; r)$ involves a sum-to-infinity series, which is not efficient for implementation. Indeed, the magnitude of $\tilde{\Upsilon}(\mu)$ (Equation 3.7) is decaying according to the term $e^{-2(\pi ||\mu||_2^2)\sigma^2}$. Therefore, ignoring the high frequency regions where the magnitudes of $\tilde{\Upsilon}(\mu)$ are negligibly small does not harm the computation accuracy of spherical flux. Noted that the decaying term $e^{-2(\pi ||\mu||_2^2)\sigma^2}$ of $\tilde{\Upsilon}(\mu)$ is a Gaussian function having a scale factor $\frac{1}{2\pi\sigma^2}$. We assume that the value of the Gaussian function and hence the value of $\tilde{\Upsilon}(\mu)$ are sufficiently small and can be treated as zero when the value of $\sigma$ is sufficiently large.
\[ ||\vec{\mu}|| \text{ is larger than four times of the scale factor } \frac{1}{2\pi\sigma}, \text{ i.e.,} \]
\[ \tilde{\Upsilon}(\vec{\mu}) \approx 0 \text{ for } ||\vec{\mu}|| \geq \frac{2}{\pi\sigma}, \quad (3.14) \]

where \( e^{-2(\pi||\vec{\mu}||\sigma)^2} \leq e^{-8} < 10^{-3} \). Based on this criterion, we ignore the computations involved in the regions, in which \( ||\vec{\mu}|| \geq \frac{2}{\pi\sigma} \) and thus the values of \( \tilde{\Upsilon}(\vec{\mu}) \) are negligible. It leads to a simplification of Equation 3.12 based on the scale factor of the Gaussian function, which is being applied on \( I(\vec{x}) \) for the calculation of the spherical flux. The region for computing \( \tilde{\Upsilon}(\vec{\mu}) \) is trimmed in each dimension separately. It is essential to correctly calculate the subbands of images having different lengths and voxel spacings along various dimensions.

We now consider the simplest case that \( \Upsilon_{\text{Subband}_1}(\vec{\mu}; r) \) directly substitutes \( \Upsilon_\infty(\vec{\mu}; r) \) in Equation 3.12,
\[ \Upsilon_\infty(\vec{\mu}; r) \approx \Upsilon_{\text{Subband}_1}(\vec{\mu}; r). \quad (3.15) \]

Here, \( \Upsilon_{\text{Subband}_1}(\vec{\mu}; r) \) is obtained by evaluating \( \tilde{\Upsilon}(\vec{\mu}; r) \) merely in the frequency \( \text{Subband}_1 \), which is defined as,
\[ \text{Subband}_1 = \left\{ \{ -0.5 \leq \mu_1 < 0.5 \} \cap \{ -0.5 \leq \mu_2 < 0.5 \} \cap \{ -0.5 \leq \mu_3 < 0.5 \} \right\} \quad (3.16) \]

\( \text{Subband}_1 \) exactly covers the bandwidth of the discrete image signal \( I(\vec{x}) \) (see Figure 3.2a). Equation 3.12 becomes,
\[ s_\infty(\vec{x}; r) \approx s_{\text{Subband}_1}(\vec{x}; r) = \mathcal{F}^{-1}\left\{ \left( \tilde{\Upsilon}(\vec{\mu}; r) \right) \left( \mathcal{F}\left\{ I(\vec{\mu}) \right\} \right) \right\}. \quad (3.17) \]

For the validity of the Equations 3.15 and 3.17, according to Equation 3.14, it is imposed that
\[ \sigma \geq \frac{2}{\pi \min_{\{u,v,w\} \in \mathbb{Z}^3 \setminus \text{Subband}_1} (||\vec{\mu}||)}, \]
and \( \min_{\vec{\mu} \in \mathbb{Z}^3 \setminus \text{Subband}_1} ||\vec{\mu}|| = 0.5 \). Therefore
\[ \sigma \geq \frac{4}{\pi} = 1.27. \quad (3.18) \]

\[^2\min_{\vec{\mu} \in \mathbb{Z}^3 \setminus \text{Subband}_1} ||\vec{\mu}|| = 0.5 \text{ at the coordinates } (0.5,0,0), (0,0.5,0) \text{ and } (0,0,0.5).\]
Noted that the value of $\sigma$ is in the unit of voxel-length.

For the calculation of spherical flux, $\sigma$ is the scale parameter for the Gaussian smoothing of the image $I(\vec{x})$ as a preprocessing step for the computation of flux. However, the value of $\sigma$ grounded on Subband$_1$ as hinted in Equation 3.18 may be large ($\sigma > 1$) for some applications such as detection and segmentation of small blood vessels (1 voxel width vessels). Applying Subband$_1$ with smaller value of $\sigma$ causes cropping of the frequency regions that contain large-valued $\Upsilon(\vec{\mu}; r)$ and thus hinders the calculation accuracy of the spherical flux. Figures 3.1b and 3.1c show an example of $\tilde{\Upsilon}([\mu_1, 0, 0]^T, 12)$ with $\sigma = 1$, where several ripples of $\tilde{\Upsilon}([\mu_1, 0, 0]^T, 12)$ are cropped as they are located outside frequency Subband$_1$.

To allow a more precise detection of small structures (i.e. a smaller value of $\sigma$, e.g. $\sigma = 1$) to be applied without discarding large-valued $\tilde{\Upsilon}(\vec{\mu}; r)$, a larger frequency subband is formulated to approximate Equation 3.13. This subband is defined as,

$$\text{Subband}_{1.5} = \left\{ \{(-1 < \mu_1 \leq 1) \cap (-0.5 \leq \mu_2 < 0.5) \cap (-0.5 \leq \mu_3 < 0.5) \} \right. $$
$$\left. \cup \{(-0.5 \leq \mu_1 < 0.5) \cap (-1 < \mu_2 \leq 1) \cap (-0.5 \leq \mu_3 < 0.5) \} \right. $$
$$\left. \cup \{(-0.5 \leq \mu_1 < 0.5) \cap (-0.5 \leq \mu_2 < 0.5) \cap (-1 < \mu_3 \leq 1) \} \right\}. \quad (3.19)$$

Figure 3.2b illustrates the coverage of Subband$_{1.5}$ in the Fourier domain. By employing Subband$_{1.5}$, $\Upsilon^\infty(\vec{\mu}; r)$ becomes

$$\Upsilon^\infty(\vec{\mu}; r) \approx \Upsilon^{\text{Subband}_{1.5}}(\vec{\mu}; r). \quad (3.20)$$

For Subband$_{1.5}$, based on the criterion stated in Equation 3.14, the range of acceptable values of $\sigma$ is,

$$\sigma \geq \frac{2}{\pi} \min_{\{u,v,w\} \in \mathbb{Z}^3 \setminus \text{Subband}_{1.5}} \left( k(u,v,w) \right), \quad (3.21)$$

where

$$\min_{\vec{\mu} \in \mathbb{Z}^3 \setminus \text{Subband}_{1.5}} ||\vec{\mu}|| = \frac{1}{\sqrt{2}} \quad \text{at the coordinates } (0, 0.5, 0.5), (0.5, 0, 0.5) \text{ and } (0.5, 0.5, 0). \quad (3.22)$$
Figure 3.2: The gray regions represent the coverages of different frequency subbands in the Fourier domain. (a) Subband$_1$; (b) Subband$_{1.5}$; (c) Subband$_2$.

Hence, the coverage of Subband$_{1.5}$ is sufficient for the calculation of spherical flux when $\sigma = 1$ and thus $\Upsilon^{\text{Subband}_{1.5}}(\vec{\mu};r)$ is employed in the rest of the chapters. In Figure 3.3, when approximating $\Upsilon^\infty(\vec{\mu};r)$, the differences between the formulations based on Subband$_1$ and Subband$_{1.5}$ are shown. Figure 3.3b shows that observable artifacts are generated in the function $\Upsilon^{\text{Subband}_1}(\vec{\mu};r)$ because of the insufficient coverage of Subband$_1$ for approximating $\Upsilon^\infty(\vec{\mu};r)$. In contrast, there is no similar artifact for $\Upsilon^{\text{Subband}_{1.5}}(\vec{\mu};r)$ (see Figures 3.3c and 3.3d).

For some applications that possibly require smaller value of $\sigma$, we suggest devising a valid frequency subband based on the criterion stated in Equation 3.14. As an example, Subband$_2$, which supports $\sigma \geq \frac{2}{\pi} = 0.64$ is illustrated in Figure 3.2c. The subscript ”2” of Subband$_2$ represents that this subband encloses doubled bandwidth along all dimensions. Comparing different subbands in Figures 3.2a, 3.2b and 3.2c, it is observed that the coverage of Subband$_{1.5}$ is between Subband$_1$ and Subband$_2$, and a larger subband coverage supports smaller value of $\sigma$.

Finally, grounded on $\Upsilon^{\text{Subband}_{1.5}}(\vec{\mu};r)$, $s^{\text{Subband}_{1.5}}(\vec{x};r)$ is computed as,

$$
s^{\text{Subband}_{1.5}}(\vec{x};r) = \hat{\mathfrak{F}}^{-1}\left\{ \left( \hat{\mathfrak{F}}\{I\}(\vec{\mu}) \right) \left( \Upsilon^{\text{Subband}_{1.5}}(\vec{\mu};r) \right) \right\}(\vec{x})
$$

$$
= \hat{\mathfrak{F}}^{-1}\left\{ \left( \hat{\mathfrak{F}}\{I\}(\vec{\mu}) \right) \left( \hat{\Upsilon}(\vec{\mu};r) + \hat{\Upsilon}(\vec{\mu} - \text{sgn}(\mu_1)\hat{z}_1; r)
\right.
+ \hat{\Upsilon}(\vec{\mu} - \text{sgn}(\mu_2)\hat{z}_2; r) + \left. \hat{\Upsilon}(\vec{\mu} - \text{sgn}(\mu_3)\hat{z}_3; r) \right) \right\}(\vec{x}) \quad (3.23)
$$

where $\text{sgn}(x) = \begin{cases} 
1, & \text{if } x \geq 0, \\
-1, & \text{otherwise.}
\end{cases}$
Reducing the computation time by exploiting Fourier coefficient redundancies

Based on the characteristics of \( I[x, y, z] \) and \( h_d[x, y, z] \), we describe two simplifications to further reduce the computation load of the spherical flux calculation in the Fourier domain.

First, \( \mathcal{F}\{I(\vec{x})\} \) and \( \tilde{\Upsilon}(\vec{\mu}; r) \) are real signals, almost half of the coefficients in \( \mathcal{F}\{I\} \) (\( \vec{\mu} \)) and \( \tilde{\Upsilon}(\vec{\mu}; r) \) are redundant. Computation time is shortened by ignoring the redundant coefficients. For the simplicity of the implementation, we treat the coefficients with negative values of \( \mu_3 \) to be redundant by exploiting the fact that \( \mathcal{F}\{I\} (\vec{\mu}) \equiv \mathcal{F}\{I\}^*(-\vec{\mu}) \) and \( \tilde{\Upsilon}(\vec{\mu}; r) \equiv \tilde{\Upsilon}^*(-\vec{\mu}; r) \), where the superscript * represents function conjugate. Therefore, Equation 3.23 is evaluated only when \( \mu_3 \) is non-negative. This halves the number of multiplication operations for evaluating Equation 3.23.

Second, \( \tilde{\Upsilon}(\vec{\mu}; r) \) is a spherically symmetric function. The spherical symmetry implies that

\[
\tilde{\Upsilon}(\vec{\mu}; r) \equiv \tilde{\Upsilon}(||\mu_1||, ||\mu_2||, ||\mu_3||^T; r).
\]

Therefore, evaluating \( \tilde{\Upsilon}(\vec{\mu}; r) \) when \( \mu_1, \mu_2 \) and \( \mu_3 \) are non-negative is adequate for acquiring \( \tilde{\Upsilon}(\vec{\mu}; r) \) with all values of \( \mu_1, \mu_2 \) and \( \mu_3 \). It facilitates the computation of
Equation 3.23, which is evaluated in the region of a non-negative frequency along the $z$-direction (i.e. $\mu_3 \geq 0$) and the values of $\tilde{\mathbf{T}}(\vec{\mu}; r)$ are retrieved only when $\mu_1$, $\mu_2$ and $\mu_3$ are all non-negative \footnote{In the proposed implementation, each evaluated value of $\tilde{\mathbf{T}}(\vec{\mu}; r) + \tilde{\mathbf{T}}(\vec{\mu} - \text{sgn}(\mu_1)\hat{z}_1; r) + \tilde{\mathbf{T}}(\vec{\mu} - \text{sgn}(\mu_2)\hat{z}_2; r) + \tilde{\mathbf{T}}(\vec{\mu} - \text{sgn}(\mu_3)\hat{z}_3; r)$ is utilized up to four times to multiply with the terms $\mathbf{F}\{I\}(\vec{\mu})$, $\mathbf{F}\{I\}(\vec{\mu} \circ (-\hat{z}_1))$, $\mathbf{F}\{I\}(\vec{\mu} \circ (-\hat{z}_2))$ and $\mathbf{F}\{I\}(\vec{\mu} \circ (-\hat{z}_3))$, where $\circ$ represents entrywise vector multiplication.}.

With the aid of these two simplifications, the number of times to evaluate $\tilde{\mathbf{T}}(\vec{\mu}; r)$ is reduced to one-eighth of those without the simplifications and number of multiplication operations taken is halved. Finally, although the above speed enhancement techniques are formulated based on Subband$_{1,5}$, they can be applied for various subbands as long as $I(\vec{x})$ and $\mathbf{F}^{-1}\{\mathbf{F}\{I\}(\vec{x}; r)\}$ are real signals and $\tilde{\mathbf{T}}(\vec{x}; r)$ is spherically symmetric. As such, they can be tailored for other subbands when different values of $\sigma$ are needed for the calculation of the spherical flux.

3.3.2 The Fast Fourier Transform Algorithm in Our Implementation

As previously elaborated, the proposed implementation makes use of the characteristics in the Fourier domain to speed up the calculation of the spherical flux. The fast Fourier transforms (FFT) is a necessity to efficiently compute $\mathbf{F}\{I\}(\vec{x})$ for Equations 3.13-3.23 and retrieve the calculation result of the spherical flux from the Fourier domain. FFT algorithms implicitly assume that images are discrete and periodic. The image periodicity causes artifacts appearing along the image boundaries when the spherical flux is computed. This artifacts can be greatly reduced by padding additional voxels outside the image to mirror-reflect the image content along the image boundaries prior to the spherical flux computation. The number of padding voxels depends on two values: the radius $r$ and the Gaussian smoothing scale factor $\sigma$. In practice, $r + 3\sigma$ is a good choice for setting the number of padding voxels.

It is well-known that the FFT algorithms are based on the divide-and-conquer strategy. In the standard FFT approach, an input signal is recursively partitioned into two equal halves. This divide-and-conquer strategy imposes a restriction on the dimensions of input images to be power-of-2. Such restriction is overcome by the recent development of the FFT algorithms, for instance, the Cooley-Tukey algorithm \cite{74}, the
Figure 3.4: Flow charts of the computation of the multiscale spherical flux. (a) The conventional spatial implementation. (b) The proposed implementation. \( L \) represents a set of scales. \( \sigma \) represents the scale parameter of the Gaussian function being applied on the image \( I(\vec{x}) \).

codelt method [75], the prime factor algorithm [76] or the Rader’s method [77]. These algorithms permit FFT to be applied on an image of any size and the time complexity is still \( O(m \log m) \).

In this work, the FFT routine in our implementation utilizes a publicly available and cross platform library FFTW [75, 78], which implements the aforementioned FFT algorithms and supports three-dimensional input data without the limitation of data size.

### 3.3.3 Multiscale Spherical Flux for Vessel Segmentation

One of the applications of tubular structure analysis is vascular segmentation in angiography. Due to the variation of vessel width, computing the values of the spherical
flux in spheres having different scales (or different radii) is needed to precisely detect vessels of different widths as in [18, 65]. In the conventional spatial implementation, the multiscale spherical flux involves repetitive calculation of spherical flux in different scales, i.e. evaluating Equation 3.1 using different values of $r$.

For instance, as described in [18, 65], a set of scales (radii) is specified by users according to the sizes of target structures. Different values of spherical flux are computed in the set of spheres of the user-defined scales. All values of the spherical flux in different scales are compared to retrieve the maximally magnitude-valued spherical flux, which is called *multiscale spherical flux*, and is given as,

$$s(\vec{x}; l(\vec{x})), \quad (3.25)$$

where $l(\vec{x})$ is the scale among the set of scales (radii of spheres) $\mathbb{L}$ that returns the maximally magnitude-valued spherical flux at the position $\vec{x}$, i.e.

$$l(\vec{x}) = \arg \max_{r \in \mathbb{L}} |s(\vec{x}; r)|. \quad (3.26)$$

While the spherical flux is computed within a set of scales $\mathbb{L}$, the $\tilde{\Upsilon}(\vec{\mu}; r)$ is repetitively evaluated for different values of $r$. To further reduce minimize the computation load of multiscale spherical flux, the components of $\tilde{\Upsilon}(\vec{\mu}; r)$, which are independent of the scale parameter $r$, are pre-calculated and stored in the pre-processing step. These components include the Gaussian term $e^{-2(\pi ||\vec{\mu}||\sigma)^2}$ and the radial frequency $||\vec{\mu}||$. They are firstly evaluated and buffered. Their values are retrieved directly upon the evaluation of Equation 3.23.

### 3.4 Experiments

The proposed implementation was studied using two synthetic and numerical volumes of $180 \times 180 \times 180$ voxels and five phase contrast magnetic resonance angiographic (PC-MRA) image volumes of various sizes. Details of the image volumes are given in the Section 3.4.3 and Section 3.4.4. In this section, the spherical flux computation times and accuracies for both the proposed implementation and the conventional spatial implementation are compared. In all the experiments, the value of $\sigma$ being used in Equation 3.7 for the proposed implementation and the Gaussian smoothing process...
for the conventional spatial implementation was 1 voxel, the set of scales \( L \) (radii of spheres) in Equation 3.26 for both the proposed implementation and the conventional spatial implementation was \( 1, 2, \ldots, 10 \) voxels.

### 3.4.1 Flux maximizing geometric flows

The proposed method is implemented along with a multiscale spherical flux based vessel extraction algorithm, Flux maximizing geometric flows [18] (FMGF). The segmentation performances of FMGF based on both the proposed implementation and the conventional spatial implementation were examined.

In [18], the multiscale spherical flux (Equation 3.25) is utilized as an evolution speed term to drive an active contour \( C \) for extracting the vasculatures,

\[
C_t(x) = \left( s(x; l(x)) + \kappa \nabla \cdot \left( \frac{\nabla C(x)}{|\nabla C(x)|} \right) \right) N(x),
\]

where \( N(x) \) is the outward normal of the contour \( C \) and \( \kappa \) is a weight attached to the mean curvature term. The value of \( \kappa \) was set to 0.03 for all the segmentation experiments. As regions inside vessels have higher intensity than the background regions, the values of \( s(x; l(x)) \) are negative. Negative values can expand the evolving contours inside those vascular regions. In contrast, the evolving contours are shrunk at the positions near and outside those vessels where \( s(x; l(x)) \) is positive. As a result, the evolving contours are eventually halted over the vessel boundaries.

Based on [18], a summary for implementing multiscale spherical flux, which is subsequently utilized by FMGF (Equation 3.27), is shown in Figure 3.4a. Another summary for the proposed implementation based on Equation 3.23 is provided in Figure 3.4b. The differences between the proposed implementation and the conventional spatial implementation are highlighted using the gray boxes. The bold boxes represent the crucial steps that have different time complexities, which are \( O(mr^2) \) in the conventional spatial implementation, and \( O(m \log m) \) in the proposed implementation.

### 3.4.2 Experiment set-up

In this chapter, both the proposed implementation and the conventional spatial implementation were written in a single-thread fashion using C++ and were compiled
using Microsoft Visual C++.Net 2003 in the Windows XP 32-bit environment. The representation of contours \( C \) of Equation 3.27 was based on the sparse field levelset framework [42], which was implemented in the Insight-Tool Kit library [54]. The evolution of the levelset function was halted when the per-voxel change of the levelset function accumulated over 10 iterations was less than \( 10^{-5} \). The experiments were conducted on a PC with a Pentium 4-HT 3.2GHz CPU and 1GB RAM.

### 3.4.3 Synthetic and numerical image volumes

Two synthetic and numerical image volumes were prepared for the validation of the proposed implementation. These image volumes, as shown in Figure 3.5, contain tubes and tori having various radii and intensity values. The intensity ranges between 0.6 and 1 for these structures, and is 0 for the background regions. These volumes were generated to mimic the appearance of vasculatures in the PC-MRA image volumes. To study the effect induced by noise, which commonly exists in the clinical images, the synthetic volumes were corrupted with an additive zero mean Gaussian noise having a set of standard deviation values 0.01, 0.02, 0.03, 0.05 and 0.1 (see Figure 3.6).

The performance of the proposed implementation was studied and compared with the conventional spatial implementation in three aspects. First, the accuracy of the segmentation and the similarity of the segmentation results obtained by using FMGF based on the both implementations. Second, the mean absolute difference between the values of the multiscale spherical flux computed by the both implementations. Third, the times for computing the multiscale spherical flux using the both implementations. To eliminate the boundary effect when evaluating the segmentation accuracy, all accuracies were measured in the center 150 × 150 × 150 voxel regions of the image volumes.

In Table 3.1, the segmentation accuracies are presented in terms of ”True positive”, ”True negative”, ”False positive”, ”False negative”, ”Positive predictive value” and ”Negative predictive value”. These values were obtained by comparing the ground truth with the segmentation results obtained by using FMGF based on the proposed implementation and the conventional spatial implementation. Noted that the ground truth was known in the experiments because the image volumes were synthetic. The proposed implementation was also validated by calculating the Dice similarity coefficients (DSC) [79] for the segmentation results obtained by using FMGF based on the
proposed and conventional spatial implementations. The DSC measure quantifies the similarity between two sets, $S_A$ and $S_B$ as

$$\text{DSC}(S_A, S_B) = \frac{2|S_A \cap S_B|}{|S_A| + |S_B|}. \quad (3.28)$$

The DSC value ranges from 0 to 1, where the DSC value of 1 indicates a perfect agreement between the sets $S_A$ and $S_B$. As stated in [79], the DSC measure is capable of precisely reporting the similarities between the sets $S_A$ and $S_B$ if $|\bar{S}_A| >> |S_A|$ and $|\bar{S}_B| >> |S_B|$. In vascular segmentation, the non-vessel regions occupy a very large proportion in the entire vascular image volume. Thus, it is illustrative to study the DSC values computed from the segmentation results obtained by using the proposed implementation of FMGF and the conventional spatial implementation of FMGF. These DSC values are listed in Table 3.1 in order to show the similarities between the segmentation results based on the two implementations.

It is observed that, across different noise levels, 0.01, 0.02, 0.03, 0.05 and 0.1, the positive predictive value and the negative predictive value (Table 3.1) are around 88% and 99.9% respectively for tubes, and around 79% and 99.9% respectively for tori (see the segmentation examples in Figure 3.7). For both tubes and tori, the differences between the positive predictive values obtained from the proposed implementation and the conventional spatial implementation are very small. Similar observations for the negative predictive values are obtained in Table 3.1. These observations reveal that, using FMGF, both the proposed implementation and conventional spatial implementation give very similar segmentation accuracy. In addition, the DSC values listed in Table 3.1 for different cases are all above the value of 0.97. Since a DSC value larger than 0.7 generally represents a good agreement [79], the DSC values listed in Table 3.1 indicate that the segmentation results based on the two implementations for each case are very similar to each other.

To further validate the similarity between the multiscale spherical flux obtained using the proposed implementation and the conventional spatial implementation, the mean absolute difference between two implementations was calculated (the last column of Table 3.2). As the magnitudes of the multiscale spherical flux vary from image to image, for estimating the mean absolute difference, those values were normalized into
a range between $-1$ and $1$. The mean absolute difference is calculated as,

\[
E_i \left( \mathbb{E}_{\vec{x} \in \Omega_i} \left( \frac{s^{\text{FAST}}(\vec{x}; r) - s^{\text{CONV}}(\vec{x}; r)}{\max_{\vec{y} \in \Omega_i} |s^{\text{FAST}}(\vec{y}; r)|} \right) \right),
\]

where $\Omega_i$ is the center $150 \times 150 \times 150$ voxel region of the $i$-th image volume, $s^{\text{FAST}}(\vec{x}; r)$ and $s^{\text{CONV}}(\vec{x}; r)$ are the multiscale spherical voxel computed by the conventional spatial implementation and the proposed implementation respectively, and $\mathbb{E}(\cdot)$ is the expected value. As listed in the last column of Table 3.2, the mean absolute difference is 0.0060, which is negligible. Therefore, the proposed implementation is capable of providing the same computation results as obtained in the conventional spatial implementation.

In addition to the good agreement between the computation results and segmentation results obtained by using the proposed implementation and the conventional spatial implementation, the former possesses a significant speed advantage as well. On average, as listed in Table 3.2, the proposed implementation computes the multiscale spherical flux with 10 scales for one image volume in around 11 seconds. The average was estimated by averaging the running times of computing multiscale spherical flux in 10 different noise corrupted image volumes (5 for tubes and 5 for tori). The average computation time of the conventional spatial implementation for the same set of image volumes is 19970 seconds. To illustrate the characteristics of different implementations in computing multiscale spherical flux, the running times of the conventional spatial implementation for different steps (or scales) are listed in Table 3.3 and plotted in Figure 3.8. The corresponding running times of the proposed implementation are also listed in Table 3.3. Figure 3.4 shows the details of the pre-processing steps in Table 3.3 for the both implementations. In Figure 3.8, the running time of the conventional spatial implementation soars as the scale increases (exact times can be found in Table 3.3). It is because a large scale requires more orientation samples on the spherical boundary surface (see Figure 3.9). In contrast, as shown in Table 3.3, the running time of the proposed implementation is relatively constant for all scales, except that, when $s = 1$, computation time is slightly shorter than other scales. It is because there is no comparison between the estimated spherical flux and the result buffer, as shown in Figure 3.4. It is experimentally shown that the proposed implementation offers significant computation time reduction as compared to the conventional spatial implementation when the value of $s$ grows (see the computation time percentages, which are enclosed
Table 3.1: The segmentation accuracies of the synthetic and numerical image volumes, as shown in Figure 3.5, using the proposed implementation based and the conventional spatial implementation based flux maximizing geometric flows. The values obtained by the conventional spatial implementation are enclosed by the brackets. The values for "True positive", "True negative", "False positive" and "False negative" are in voxels.

<table>
<thead>
<tr>
<th></th>
<th>0.01</th>
<th>0.02</th>
<th>0.03</th>
<th>0.05</th>
<th>0.1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>σ =</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>True positive</td>
<td>504742</td>
<td>504412</td>
<td>504685</td>
<td>504657</td>
<td>504346</td>
</tr>
<tr>
<td></td>
<td>(504738)</td>
<td>(504377)</td>
<td>(504341)</td>
<td>(504341)</td>
<td>(504304)</td>
</tr>
<tr>
<td>True negative</td>
<td>2800095</td>
<td>2801176</td>
<td>2800646</td>
<td>2801808</td>
<td>204712</td>
</tr>
<tr>
<td></td>
<td>(2799940)</td>
<td>(2801206)</td>
<td>(2800985)</td>
<td>(2800111)</td>
<td>(2799902)</td>
</tr>
<tr>
<td>False positive</td>
<td>70155</td>
<td>69074</td>
<td>69604</td>
<td>68442</td>
<td>65538</td>
</tr>
<tr>
<td></td>
<td>(70310)</td>
<td>(69044)</td>
<td>(69265)</td>
<td>(70139)</td>
<td>(70348)</td>
</tr>
<tr>
<td>False negative</td>
<td>8</td>
<td>338</td>
<td>65</td>
<td>93</td>
<td>404</td>
</tr>
<tr>
<td></td>
<td>(12)</td>
<td>(373)</td>
<td>(409)</td>
<td>(409)</td>
<td>(446)</td>
</tr>
<tr>
<td>Tubes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>False negative</td>
<td>8</td>
<td>338</td>
<td>65</td>
<td>93</td>
<td>404</td>
</tr>
<tr>
<td></td>
<td>(12)</td>
<td>(373)</td>
<td>(409)</td>
<td>(409)</td>
<td>(446)</td>
</tr>
<tr>
<td>Positive predictive</td>
<td>87.80%</td>
<td>87.96%</td>
<td>87.88%</td>
<td>88.06%</td>
<td>88.50%</td>
</tr>
<tr>
<td>values</td>
<td>(87.77%)</td>
<td>(87.96%)</td>
<td>(87.92%)</td>
<td>(87.79%)</td>
<td>(87.76%)</td>
</tr>
<tr>
<td>Negative predictive</td>
<td>100.00%</td>
<td>99.99%</td>
<td>100.00%</td>
<td>100.00%</td>
<td>99.99%</td>
</tr>
<tr>
<td>values</td>
<td>(100.00%)</td>
<td>(99.99%)</td>
<td>(99.99%)</td>
<td>(99.99%)</td>
<td>(99.98%)</td>
</tr>
<tr>
<td>DSC values</td>
<td>0.9855</td>
<td>0.9846</td>
<td>0.9847</td>
<td>0.9831</td>
<td>0.9830</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tori</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>True positive</td>
<td>323681</td>
<td>323680</td>
<td>323675</td>
<td>323677</td>
<td>323698</td>
</tr>
<tr>
<td></td>
<td>(323284)</td>
<td>(323582)</td>
<td>(323591)</td>
<td>(323594)</td>
<td>(323615)</td>
</tr>
<tr>
<td>True negative</td>
<td>2963672</td>
<td>2963751</td>
<td>2963608</td>
<td>2963920</td>
<td>2964878</td>
</tr>
<tr>
<td></td>
<td>(2966166)</td>
<td>(2966134)</td>
<td>(2966301)</td>
<td>(2966424)</td>
<td>(2966431)</td>
</tr>
<tr>
<td>False positive</td>
<td>87520</td>
<td>87441</td>
<td>87584</td>
<td>87272</td>
<td>86314</td>
</tr>
<tr>
<td></td>
<td>(85026)</td>
<td>(85058)</td>
<td>(84891)</td>
<td>(84768)</td>
<td>(84761)</td>
</tr>
<tr>
<td>False negative</td>
<td>127</td>
<td>128</td>
<td>133</td>
<td>131</td>
<td>110</td>
</tr>
<tr>
<td></td>
<td>(224)</td>
<td>(226)</td>
<td>(217)</td>
<td>(214)</td>
<td>(193)</td>
</tr>
<tr>
<td>Positive predictive</td>
<td>78.72%</td>
<td>78.72%</td>
<td>78.70%</td>
<td>78.76%</td>
<td>78.95%</td>
</tr>
<tr>
<td>values</td>
<td>(79.19%)</td>
<td>(79.19%)</td>
<td>(79.22%)</td>
<td>(79.24%)</td>
<td>(79.24%)</td>
</tr>
<tr>
<td>Negative predictive</td>
<td>100.00%</td>
<td>100.00%</td>
<td>100.00%</td>
<td>99.99%</td>
<td>100.00%</td>
</tr>
<tr>
<td>values</td>
<td>(99.99%)</td>
<td>(99.99%)</td>
<td>(99.99%)</td>
<td>(99.99%)</td>
<td>(99.99%)</td>
</tr>
<tr>
<td>DSC values</td>
<td>0.9730</td>
<td>0.9729</td>
<td>0.9735</td>
<td>0.9724</td>
<td>0.9720</td>
</tr>
</tbody>
</table>

In general, it is observed that the proposed implementation utilizes between 0.02% and 0.79% of computation time taken by the conventional spatial implementation for evaluating the spherical flux of an image volume. Regarding the total running times for multiscale spherical flux with 10 scales, the proposed implementation needs 0.05% (the forth column in Table 3.2) of the time taken by the conventional spatial implementation.
Figure 3.5: The synthetic and numerical image volumes. (a) The isosurfaces of the synthetic and numerical tubes. (b) The slice of the tube image volume at $z = 20$. The tubes have radii 1, 2, 4, 6 and 8 voxels, and intensity values 0.6, 0.7, 0.8, 0.9 and 1. (c) The isosurfaces of the synthetic and numerical tori. (d) Top: A vertical cross section of a torus; bottom: the slice of the torus image volume at $y = 90$. The intensity values of the tori are 0.8 for the gray tori and 1 for the white tori. Expressing the sizes of tori in voxels using the representation (small radius, large radius), from top to bottom, the tori have sizes $(1, 24)$, $(1, 48)$, $(1, 56)$, $(2, 24)$, $(2, 32)$, $(2, 40)$, $(2, 48)$, $(2, 56)$, $(4, 24)$, $(4, 48)$, $(4, 64)$, $(6, 36)$, $(6, 60)$, $(8, 48)$ and $(8, 60)$. 
Figure 3.6: Top: The image slices of the noise corrupted synthetic and numerical tubes at $z = 20$; bottom: the image slices of the noise corrupted synthetic and numerical tori at $y = 90$. From left to right, the image volumes were corrupted by an additive zero mean Gaussian noise with standard deviations 0.01, 0.02, 0.03, 0.05 and 0.1, respectively.

Figure 3.7: Top: The isosurfaces of the initial levelset function (the first column) and the segmentation results of noise corrupted synthetic tubes (from the second column to the sixth column). Bottom: The isosurfaces of the initial levelset function (the first column) and the segmentation results of noise corrupted synthetic tori (the second column to the sixth column). From the second column to the sixth column, the segmentation results of the image volumes corrupted by an additive zero mean Gaussian noise with standard deviations 0.01, 0.02, 0.03, 0.05 and 0.1. The results are obtained by applying the flux maximizing geometric flows along with the proposed implementation.
Table 3.2: (Second and third columns): The total running times (in seconds) for multiscale spherical flux computation with 10 scales based on the proposed implementation and the conventional spatial implementation; (the last column): the mean absolute difference between the multiscale spherical flux values computed by the both implementations. The times listed are obtained by averaging the running times of computing multiscale spherical flux in 10 different noise corrupted image volumes (5 for tubes and 5 for tori), which were generated for the segmentation experiments on the synthetic and numerical image volumes, as shown in Figure 3.5. The values in the second, third and last columns are rounded to 4 significant digits.

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Proposed Implementation (PI)</th>
<th>Conventional Spatial Implementation (CSI)</th>
<th>$\frac{PI}{CSI} \times 100%$</th>
<th>Mean absolute difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>180 × 180 × 180</td>
<td>10.82 seconds</td>
<td>19970 seconds</td>
<td>0.05%</td>
<td>0.0060</td>
</tr>
</tbody>
</table>

Figure 3.8: The plot of the average running times for multiscale spherical flux computation using the conventional spatial implementation against different steps (or scales). The times listed are obtained by averaging the running times of computing multiscale spherical flux in 10 different noise corrupted image volumes (5 for tubes and 5 for tori), which were generated for the segmentation experiments on the synthetic and numerical image volumes, as shown in Figure 3.5.

Figure 3.9: The number of orientation samples taken for the conventional computation of spherical flux with various radii.
Table 3.3: The running times (in seconds) of different steps for multiscale spherical flux computation based on the proposed implementation (top) and the conventional spatial implementation (bottom). The values listed in the table are obtained by averaging the running times of computing multiscale spherical flux in 10 different noise corrupted image volumes (5 for tubes and 5 for tori), which were generated for the segmentation experiments on the synthetic and numerical image volumes, as shown in Figure 3.5. The values listed in the table are obtained by averaging the running times of computing multiscale spherical flux for 10 different steps of different multiscale approximations. The values in the brackets are the percentages, which are given as \( \frac{\text{CSI}}{\text{PI}} \times 100\% \). The values are rounded to 4 significant digits, except for the values enclosed by brackets, which are given as \( \text{CSI} \times 100\% \).

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Pre-processing</th>
<th>Proposed implementation (PI)</th>
<th>Conventional spatial implementation (CSI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.02%</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0.04%</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>0.05%</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
<td>0.07%</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>0.05%</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>6</td>
<td>0.04%</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>7</td>
<td>0.03%</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>8</td>
<td>0.02%</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>9</td>
<td>0.02%</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>10</td>
<td>0.01%</td>
</tr>
</tbody>
</table>

The values in the brackets are the percentages, which are given as \( \frac{\text{CSI}}{\text{PI}} \times 100\% \). The values are rounded to 4 significant digits, except for the values enclosed by brackets, which are given as \( \text{CSI} \times 100\% \).
There were five clinical phase contrast magnetic resonance angiographic (PC-MRA) image volumes being employed in the experiments. These image volumes were upsampled along the z-direction using bilinear interpolation to acquire isotropic voxels prior to subsequent processing. The maximum intensity projections (MIPs) of these image volumes are shown in Figures 3.10a-e. Their resolutions and the voxel sizes are tabulated in Table 3.4. All the clinical cases are the axial brain scans and were acquired using a Philips 3T ACS Gyroscan MR scanner at the University Hospital of Zurich, Switzerland. The standard PC-MRA imaging protocols were utilized without contrast agents.

As mentioned in the previous section regarding the synthetic image experiments, the proposed implementation was evaluated by measuring the mean absolute difference between the values of the multiscale spherical flux obtained by using the both implementations and comparing the times for computing the multiscale spherical flux with 10 scales using the both implementations in the experiments on the above five clinical PC-MRA image volumes. The computation of the DSC values of the clinical segmentation results is the same as the computation described in the synthetic experiments (Equation 3.28). The mean absolute difference utilized in this section was measured analogous to the synthetic image experiments (Equation 3.29), except that the mean absolute difference was estimated individually for each clinical image volume. Thus, the mean absolute difference is given as,

$$E_{\vec{x} \in \Omega} \left( \frac{s_{\text{FAST}}(\vec{x}; r) - \max_{\vec{y} \in \Omega} s_{\text{FAST}}(\vec{y}; r)}{\max_{\vec{y} \in \Omega} s_{\text{FAST}}(\vec{y}; r)} \right) \left( \frac{s_{\text{CONV}}(\vec{x}; r) - \max_{\vec{y} \in \Omega} s_{\text{CONV}}(\vec{y}; r)}{\max_{\vec{y} \in \Omega} s_{\text{CONV}}(\vec{y}; r)} \right)$$

(3.30)

where $\Omega$ is the whole image domain excluding the positions, which have distances away from the image boundary within 15 voxels to eliminate the boundary effect when

<table>
<thead>
<tr>
<th>Case</th>
<th>Resolutions</th>
<th>Voxel sizes</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>174 × 296 × 150</td>
<td>0.39mm × 0.39mm × 0.39mm</td>
</tr>
<tr>
<td>b</td>
<td>296 × 296 × 130</td>
<td>0.39mm × 0.39mm × 0.39mm</td>
</tr>
<tr>
<td>c</td>
<td>276 × 276 × 162</td>
<td>0.37mm × 0.37mm × 0.37mm</td>
</tr>
<tr>
<td>d</td>
<td>234 × 270 × 166</td>
<td>0.39mm × 0.39mm × 0.39mm</td>
</tr>
<tr>
<td>e</td>
<td>280 × 280 × 162</td>
<td>0.37mm × 0.37mm × 0.37mm</td>
</tr>
</tbody>
</table>

Table 3.4: The resolutions and voxel sizes of the image volumes shown in Figure 3.10.
evaluating the differences.

In the fifth column of Table 3.5, the mean absolute differences between the multiscale spherical flux obtained in the both implementations for each clinical case are listed. Among all the clinical cases, the mean absolute differences are only ranged between 0.0020 and 0.0049. Similar to the previous synthetic image experiments, such differences are very small. As such, the proposed implementation can give comparable results of the multiscale spherical flux computation, as compared with the results obtained in the conventional spatial implementation.

More importantly, the computation time of multiscale spherical flux is sharply reduced by using the proposed implementation, as listed in the second and the third columns of Table 3.5. For the proposed implementation, the computation times of multiscale spherical flux for the five clinical cases are around 28 seconds, 41 seconds, 44 seconds, 46 seconds and 24 seconds. Comparing to the computation times for the same set of image volumes using the conventional spatial implementation, which are 24240 seconds, 36460 seconds, 40610 seconds, 34970 seconds and 43130 seconds respectively, the proposed implementation utilizes 0.10% of the computation time taken by the conventional spatial implementation on average (the forth column of Table 3.5). In Table 3.6, the computation times required by the both implementations for different steps (or scales) are listed. The computation time of the conventional spatial implementation shoots up as $s$ increases while the proposed implementation has fairly consistent computation times among all scales. Thus, the amount of computation time reduction gained by the proposed implementation becomes more significant when the value of $s$ is large (see the computation time percentages, which are enclosed by brackets in Table 3.6).

On the other hand, the vasculatures in the clinical image volumes extracted by FMGF based on multiscale spherical flux computed by the proposed implementation and the conventional spatial implementation are shown in Figures 3.11a–e and 3.12a–e respectively. The initial levelset functions for these cases were generated by a global thresholding scheme. In this global thresholding scheme, the regions having the highest 0.5% negative values of the multiscale spherical flux of each volume were selected. The selected regions were evolved under the pure curvature flows with $\kappa = 0.2$ and then utilized as the initial seed points for FMGF. This thresholding scheme ensures
all the initial seed points were located inside the vessels. The segmentation results shown in Figures 3.11a-e and 3.12a-e illustrate that the proposed implementation of FMGF is capable of delivering similar segmentation results obtained by using the conventional spatial implementation. This observation is further validated by the DSC values presented in last column of Table 3.5. In the table, the DSC values indicating the segmented region similarities between the proposed implementation of FMGF and the conventional spatial implementation of FMGF for all five cases are at least 0.9327, which are all above the DSC value of 0.7 for good agreement between results, as suggested in [79].

Finally, the segmentation results of FMGF were compared with the results obtained by using a widely used vessel extraction method based on vesselness measures, which was proposed by Frangi et al. [2]. The scale set used by the vesselness measure was the same as the scale set used by the multiscale spherical flux, which was 1, 2, ..., 10 voxels. The other parameters used by the vesselness measure were specified strictly according to the descriptions in [2]. In the literature, similar to other works [2, 80], the vessel extraction results were obtained by firstly thresholding the multiscale vesselness responses. In this chapter, 0.005 was used as the threshold value for all five clinical cases. This threshold value was acquired manually so that no excessive widening of main vessels was observed. To eliminate noisy regions in the thresholding results, the thresholded regions which were not connected with the positions having the highest 0.5% vesselness responses were removed in the final vessel extraction results. The vesselness extraction results are shown in Figure 3.13. It is observed that there are many high curvature small vessels missing in the vesselness extraction results (Figures 3.13a-e). It is because the high curvature small vessels deviate from the assumption of vessels made by the vesselness measure that vessels are mainly elongated tubular shapes with circular cross sections. In contrast, the FMGF does not rely on this assumption for detecting vasculatures and thus it is capable of discovering the high curvature small vessels (see Figures 3.11a-e and 3.12a-e). Comparing the extracted vessels of FMGF (Figures 3.11a-e and 3.12a-e) with the axial views of the corresponding image volumes (the upper-left images of Figures 3.10a-e), FMGF is able to deliver promising vascular segmentation results. With the aid of the proposed implementation, which remarkably reduces the computation time of multiscale spherical flux to 0.10% of the time taken by the conventional spatial implementation, the proposed implementation is a good
Table 3.5: (Second and third columns): The running times (in seconds) for multiscale spherical flux computation based on the proposed implementation and the conventional spatial implementation; (the fifth column): the mean absolute differences between the spherical flux values computed by the conventional spatial implementation and the proposed implementation; (the last column) The estimated DSC values based on the segmentation results obtained by using the proposed implementation and the conventional spatial implementation. The values in the second, third, fifth and last columns are rounded to 4 significant digits.

3.5 Discussion

We have presented a new and efficient implementation for computing the spherical flux in an image volume. The conventional spatial implementation has two major limitations that adversely affect its usefulness in tubular structure analysis, for example detection and segmentation of the blood vessels in the brain. First, the conventional spatial implementation is not computationally efficient in the spatial domain. Second, the computation time increases quadratically with the sphere size. Our implementation overcomes these limitations by reformulating the spherical flux computation so that most of the computations are performed in the Fourier domain. We have also presented a general scheme for selecting frequency subband while maintaining high computation quality as compared with the conventional spatial implementation.

Using two synthetic image volumes of $180 \times 180 \times 180$ voxels (straight tubes and tori), we have shown experimentally that our proposed implementation can achieve significant computational gain. For computing 10 scales in the multiscale spherical flux, the proposed implementation needs 0.05% of the running time taken by the con-
Figure 3.10: The maximum intensity projections of the five clinical PC-MRA image volumes used in the experiments. For each sub-figure, the top-left image is the axial view, the bottom image is the coronal view, and the top-right image is the sagittal view.
Figure 3.11: The axial views of the segmentation results of the five clinical PC-MRA image volumes based on FMGF using the proposed implementation. The corresponding maximum intensity projections are shown in Figures 3.10a-e.
Figure 3.12: The axial views of the segmentation results of the five clinical PC-MRA image volumes based on FMGF using the conventional implementation. The corresponding maximum intensity projections are shown in Figures 3.10a-e.
Figure 3.13: The axial views of the vessel extraction results of the five clinical PC-MRA image volumes based on the vesselness measure, proposed by Frangi et al. [2]. The corresponding maximum intensity projections are shown in Figures 3.10a-e.
Table 3.6: The running times (in seconds) of different steps for multiscale spherical flux computation based on the proposed implementation (top) and the conventional spatial implementation (bottom). The values in the brackets are the percentages, which are given as \( \frac{\text{PI}}{\text{CSI}} \times 100\% \). The values are rounded to 4 significant digits, except the values enclosed by brackets.
ventional spatial implementation, assuming that one orientation sample is taken for each unit area in the unit of voxel-length on the spherical region boundary. Given that the multiscale spherical flux is normalized between $-1$ and $1$, our proposed implementation can give comparable computation accuracy with mean absolute difference $0.0060$ between the both implementations. Our implementation has been tested on five clinical phase contrast magnetic resonance angiographic (PC-MRA) image volumes of $174 \times 296 \times 150$, $296 \times 296 \times 130$, $276 \times 276 \times 162$, $234 \times 270 \times 166$, $280 \times 280 \times 162$ voxels. It is found that, on average, our proposed implementation needs $0.10\%$ of the running time taken by the conventional spatial implementation, and can also give comparable accuracy with mean absolute difference $0.0036$.

With the recent rapid technological advances in minimally invasive surgery and endovascular treatments, there is a growing need to perform analysis of tubular structures in magnetic resonance angiography and computed tomographic angiography, such as detection and segmentation of the blood vessels in the brain. We believe that our implementation will benefit tubular structure analysis and contribute in improving surgery and treatment efficacy.
CHAPTER 4

THREE DIMENSIONAL CURVILINEAR STRUCTURE DETECTION AND VASCULAR SEGMENTATION USING OPTIMALLY ORIENTED FLUX

4.1 Introduction

Analysis of curvilinear structures in volumetric images has a wide range of applications, for instance centerline extraction [62, 81, 83], detection and segmentation [10, 13, 18, 84], vascular image enhancement [14, 19, 82] or visualization [85]. In particular, low-level detectors which are sensitive to curvilinear structures are the foundations of the aforementioned applications. One classic low-level detector is the multiscale based image intensity second-order statistics. Lindeberg [15] conducted in-depth research regarding the use of the Gaussian smoothing function with various scale factors for extracting multiscale second-order statistics. Koller et al. [10] exploited the image intensity second-order statistics to form Hessian matrices for the analysis of curvilinear structures in three dimensional image volumes. Frangi et al. [2] introduced the vesselness measure based on eigenvalues extracted from the Hessian matrix in a multiscale fashion. Krissian et al. [13] studied the relation between the Hessian matrix and the image gradient computed in multiple scales for the detection of tubular structures. Manniesing et al. [82] made use of multiscale Hessian matrix based features to devise a nonlinear scale space representation of curvilinear structures for vessel image enhancement.

Grounded on the multiscale based Hessian matrix, Sato et al. [14] presented a thorough study on the properties of the eigenvalues extracted from the Hessian matrix in different scales, and their performance in curvilinear structure segmentation and visualization. The study showed that the eigenvalues extracted from the Hessian matrix can be regarded as the results of convolving the image with the second derivative of a Gaussian function. This function offers differential effects which computes the difference between the intensity inside an object and in the vicinity of the object.
However, if the intensity around the objects is not homogeneous due to the presence of closely located adjacent structures, the differential effect given by the second derivatives of Gaussian is adversely affected.

Another recently developed low-level detector for curvilinear structure analysis is the image gradient flux. It is a scalar measure which quantifies the amount of image gradient flowing in or out of a local spherical region. A large magnitude of the image gradient flux is an indication of the presence of a curvilinear structure disregarding the structure direction. Bouix et al. proposed to compute the image gradient flux for extracting centerlines of curvilinear structures [62]. Siddqi et al. [18] showed promising vascular segmentation results by evolving an image gradient flux driven active surface model. However, the major disadvantage of the image gradient flux is its disregard of directional information.

On the other hand, segmentation of three dimensional curvilinear objects, particularly vascular structures has a wide range of applications. Over the past decades, incorporating curvilinear structure-specific image features in active contour models for vessel segmentation has been intensively studied. For instance, Lorigo et al. [34] developed the CURVES algorithm based on the geodesic active contour model [28], which aims at driving active contours to the boundaries where image intensity is rapidly changing. The CURVES algorithm employs the minimal curvature regularization term to prevent the evolving contours from vanishing inside narrow vascular structures. Yan and Kassim refined the geodesic active contour model by introducing the capillary force [29] to encourage contours to propagate into small vessels. The contour dynamics of these segmentation methods are governed by the image intensity gradient. It is possibly problematic if the structure intensity fluctuates along and inside structures. The intensity fluctuation can halt the evolving contours inside structures, and such intensity fluctuation commonly exists in angiographic images. Furthermore, low contrast structure boundaries cannot exert enough image force to compete against other forces generated from the intensity fluctuations along the structures. The evolving contours can finally stop inside structures instead of at the boundaries of the structures.

To extract reliable image features for segmentation of three dimensional curvilinear structures, the intensity profiles along the structure cross-sectional plane are commonly considered to be symmetric with respect to the structure center. Classic differential
operators, such as the second derivatives of Gaussian [10] and the Hessian matrix [14][2], which are based on convolving an image with symmetric filter functions [14], were proposed for the detection of curvilinear structures. The differential operators quantify the difference between the intensity inside a local region defined by a scale parameter and those in the vicinity of that local region. Exploiting the Hessian matrix, Toledo et al. [86] developed an active contour model based on the eigenvalues and eigenvectors extracted from the Hessian matrix. In [46], Descoteaux et al. fused the Hessian matrix and the flux measure [18] to formulate an active contour model to segment vascular objects. The flux measure was introduced by Vasilevskly and Siddiqi in [18]. It drives the active contours to segment vessels by using a discretized Laplacian operator, which inspects the intensity changes that occur at the boundary of a local sphere with a predefined radius. Analogous to the original Laplacian operator, the discretized version is isotropic and sensitive to symmetric structures. To handle vessels with various widths, these symmetric operators are always incorporated in multiscale frameworks. However, they commonly return faint responses at structure boundaries. This is because the local intensity variations across the structure boundaries are not symmetric with respect to those boundaries. At the boundaries, the active contours driven by the responses of these operators can evolve randomly according to the image noise attached along the object boundaries. It can lead to subsequent contour leakages.

In this chapter, we propose a novel detector of curvilinear structures, called optimally oriented flux (OOF). Specifically, the oriented flux encodes directional information by projecting the image gradient along some axes, prior to measuring the amount of the projected gradient that flows in or out of a local spherical region. Meanwhile, OOF discovers the structure direction by finding an optimal projection axis which minimizes the oriented flux. OOF is evaluated for each voxel in the entire image. The evaluation of OOF is based on the projected image gradient at the boundary of a spherical region centered at a local voxel. When the local spherical region boundary touches the object boundary of a curvilinear structure, the image gradient at the curvilinear object boundary produces an OOF detection response. Depending on whether the voxels inside the local spherical region have stronger intensity, the sign of the OOF detection response varies. It can be utilized to distinguish between regions inside and outside curvilinear structures.
The major advantage of the proposed method is that the OOF based detection is localized at the boundary of the local spherical region. Distinct from the Hessian matrix, OOF does not consider the region in the vicinity of the structure where a nearby object is possibly present. As such, the OOF detection result is robust against the disturbance introduced by closely located objects. With this advantage, utilizing OOF for curvilinear structure analysis is highly beneficial when closely located structures are present. Moreover, the computation of OOF does not introduce additional computation load compared to the Hessian matrix. Validated by a set of experiments, OOF is capable of providing more accurate and stable detection responses than the Hessian matrix, with the presence of closely located adjacent structures.

To segment curvilinear objects such as vessels without leakages, an OOF-based approach that inspects the symmetry of image gradients for active contour evolution is proposed. The proposed segmentation model considers both the image gradient symmetry with respect to the structure center, and the image gradient antisymmetry with respect to the structure boundary. Analyzing both the gradient symmetry and asymmetry helps devise image features to encourage contour propagation even through there exists intensity fluctuation along structures, and simultaneously avoids contour leakages. Through the experiments using a noise corrupted synthetic image volume and real vascular image volumes, the proposed method is compared with two well founded published approaches, the flux method [18] and the CURVES algorithm [34]. The ability of the proposed method to correctly segment curvilinear structures, particularly vasculatures without leakages is validated. It consistently delivers promising segmentation results in all cases. It is therefore well suited to perform segmentation of curvilinear structures.

4.2 Methodology

4.2.1 Optimally oriented flux (OOF)

The notion of oriented flux along a particular direction refers to the amount of image gradient projected along that direction at the surface of an enclosed local region. The image gradient can flow either in or out of the enclosed local region. Without loss of generality, our elaboration focuses on the situation where the structures have stronger
intensity than background regions. As such, optimally oriented flux (OOF)\cite{88} aims at finding an optimal projection direction that minimizes the inward oriented flux for the detection of curvilinear structure.

The outward oriented flux along a direction $\hat{\rho}$ is calculated by projecting the image gradient $\vec{v}(\vec{x})$ along the direction of $\hat{\rho}$ prior to the computation of flux in a local spherical region $B_r$ with radius $r$. Based on the definition of flux\cite{61}, the computation of the outward oriented flux along the direction of $\hat{\rho}$ is,

$$ f(\vec{x}; r, \hat{\rho}) = \frac{1}{4\pi r^2} \int_{\partial B_r} \left( (\vec{v}(\vec{x}+r\hat{n}_A) \cdot \hat{\rho} ) \hat{\rho} \right) \cdot \hat{n}_A dA, \quad (4.1) $$

where $dA$ is the infinitesimal area on $\partial B_r$, $r\hat{n}_A$ is a position at $\partial B_r$. As $\partial B_r$ is a sphere surface, the sphere outward normal at $r\hat{n}_A$ is $\hat{n}_A$. The above equation is rewritten as,

$$ f(\vec{x}; r, \hat{\rho}) = \frac{1}{4\pi r^2} \int_{\partial B_r} \left\{ \sum_{i=1}^{3} \sum_{j=1}^{3} \left[ (\vec{v}(\vec{x}+r\hat{n}_A) \cdot \hat{z}_i)(\hat{n}_A \cdot \hat{z}_j) \right] \right\} dA = \hat{\rho}^T Q(\vec{x}; r) \hat{\rho}, \quad (4.2) $$

where $\hat{z}_1$, $\hat{z}_2$ and $\hat{z}_3$ are the basis vectors along the x-, y- and z-directions. $Q(\vec{x}; r)$ is a tensor that the entry at the $i$th row and $j$th column ($i, j \in \{1, 2, 3\}$) is,

$$ q_{i,j}(\vec{x}; r) = \frac{1}{4\pi r^2} \int_{\partial B_r} (\vec{v}(\vec{x}+r\hat{n}_A) \cdot \hat{z}_i)(\hat{n}_A \cdot \hat{z}_j)dA. \quad (4.3) $$

### 4.2.2 Analytical computation of OOF

The main idea of OOF is to identify the direction $\hat{\rho}$ that maximizes the inward oriented flux (i.e. the outward oriented flux attains the minimum),

$$ \arg \min_{\hat{\rho}} f(\vec{x}; r, \hat{\rho}). \quad (4.4) $$

It is not easy to discretize any one of the surface integrals of Equation 4.1 and 4.3 to estimate oriented flux and find the optimal axis which minimizes the inward oriented flux. Nevertheless, computation of OOF can be achieved analytically by acquiring the values of the entries of $Q(\vec{x}; r)$, that only involves convolving an image with a set of filters. This formulation avoids discretization along the local spherical region surface and reduces computation complexity as compared with discretizing the integral of either
Equation 4.1 or Equation 4.3. By using the fast Fourier transform, the complexity of evaluating $Q(\vec{x}; r)$ is $O(m \log m)$, where $\forall \vec{x} \in \Omega$ and $\Omega$ is the image domain containing $m$ voxels. The proposed method introduces no additional computation load compared to some traditional approaches, such as Hessian matrix based methods [2, 13, 14].

We begin the elaboration of the analytical computation from Equation 4.3,

$$q_{i,j}(\vec{x}; r) = \frac{1}{4\pi r^2} \int_{\partial B_r} \left( \left( \vec{v}(\vec{x} + r\hat{n}_A) \cdot \hat{z}_j \right) \hat{z}_i \right) \cdot \hat{n}_A dA$$ \hspace{1cm} (4.5)

Assuming that image gradient $\vec{v}$ is continuous, by the divergence theorem,

$$q_{i,j}(\vec{x}; r) = \frac{1}{4\pi r^2} \int_{B_r} \frac{\partial}{\partial \hat{z}_j} \vec{v}(\vec{x} + \vec{y}_V) \cdot \hat{z}_i dV,$$ \hspace{1cm} (4.6)

where $\vec{y}_V$ is the position vector inside the sphere $B_r$ and $dV$ is the infinitesimal volume in $B_r$. The continuous image gradient $\vec{v}(\vec{x})$ is acquired by convolving the discrete image with the first derivatives of Gaussian with a small scale factor (i.e., 1 voxel length),

$$\vec{v}(\vec{x}) = \left( \nabla G(\vec{x}) \right) * I(\vec{x}).$$ \hspace{1cm} (4.7)

where $*$ is the convolution operator, $G(\vec{x})$ is the Gaussian function and with scale factor 1 in all our implementations. Furthermore, the volume integral of Equation 4.6 is operated in the entire image domain $\Omega$ by employing a spherical step function,

$$B(\vec{x}; r) = \left\{ \begin{array}{ll} (4\pi r^2)^{-1}, & ||\vec{x}|| \leq r, \\
0, & \text{otherwise}. \end{array} \right.$$ \hspace{1cm} (4.8)

$$q_{i,j}(\vec{x}; r) = \int_{\Omega} B(\vec{y}; r)((G_{\hat{z}_i, \hat{z}_j} * I)(\vec{x} + \vec{y}_V)) dV = ((B * G_{\hat{z}_i, \hat{z}_j})(\vec{x}; r)) * I(\vec{x}),$$ \hspace{1cm} (4.9)

where $G_{\hat{z}_i, \hat{z}_j}$ is the second derivative of $G$ along the axes $\hat{z}_i$ and $\hat{z}_j$. The next step is to obtain the analytical form of $B * G_{\hat{z}_i, \hat{z}_j}(\vec{x}; r)$. The analytical form can be found in the frequency domain. With the aid of the fast Fourier transform, the convolution in Equation 4.9 is computed efficiently by using Fourier coefficient multiplication. By employing Fourier transforms on $G_{\hat{a}_i, \hat{a}_j}$ and Hankel transforms [72] on $B_r(\vec{x})$,

$$\hat{\mathcal{F}} \left\{ B * G_{\hat{z}_i, \hat{z}_j} \right\} (\vec{u}; r) = \frac{\hat{z}_i^T (\vec{\mu} \vec{\mu}^T) \hat{z}_j e^{-2(\pi ||\vec{\mu}|| \sigma)^2}}{r} \frac{1}{||\vec{\mu}||^2} \left( \cos(2\pi r ||\vec{\mu}||) - \frac{\sin(2\pi r ||\vec{\mu}||)}{2\pi r ||\vec{\mu}||} \right).$$ \hspace{1cm} (4.10)
Based on the above formulation, the tensor $Q(\vec{x}; r)$ is computed as

$$Q(\vec{x}; r) = \mathcal{F}^{-1} \left\{ \begin{bmatrix} z_1^T(\vec{u}\vec{u}^T)\hat{z}_1 & z_1^T(\vec{u}\vec{u}^T)\hat{z}_2 & z_1^T(\vec{u}\vec{u}^T)\hat{z}_3 \\ z_2^T(\vec{u}\vec{u}^T)\hat{z}_1 & z_2^T(\vec{u}\vec{u}^T)\hat{z}_2 & z_2^T(\vec{u}\vec{u}^T)\hat{z}_3 \\ z_3^T(\vec{u}\vec{u}^T)\hat{z}_1 & z_3^T(\vec{u}\vec{u}^T)\hat{z}_2 & z_3^T(\vec{u}\vec{u}^T)\hat{z}_3 \end{bmatrix} \right\} \times \mathcal{F} \{ I \} \left( \frac{e^{-2i\pi||\vec{u}||r}}{r||\vec{u}||^2} \left( \cos(2\pi r||\vec{u}||) - \frac{\sin(2\pi r||\vec{u}||)}{2\pi r||\vec{u}||} \right) \right) (\vec{x}). \quad (4.11)$$

With the analytical form of $Q(\vec{x}; r)$, the optimal projection axis which maximizes inward oriented flux can be computed analytically. Denote the optimal direction is $\hat{\sigma}(\vec{x}; r)$, maximizing inward oriented flux is equivalent to minimizing $f(\vec{x}; r, \hat{\sigma}(\vec{x}; r))$, subject to the constraint $||\hat{\sigma}(\vec{x}; r)|| = [\hat{\sigma}(\vec{x}; r)]^T \hat{\sigma}(\vec{x}; r) = 1$. This leads to the Lagrange equation,

$$[\hat{\sigma}(\vec{x}; r)]^T Q(\vec{x}; r) \hat{\sigma}(\vec{x}; r) + \lambda(\vec{x}; r)(1 - [\hat{\sigma}(\vec{x}; r)]^T \hat{\sigma}(\vec{x}; r)), \quad (4.12)$$

The optimal direction $\hat{\sigma}(\vec{x}; r)$ is acquired based on the first derivative of the above Lagrange equation,

$$\frac{\partial}{\partial \hat{\sigma}(\vec{x}; r)} [\hat{\sigma}(\vec{x}; r)]^T Q(\vec{x}; r) \hat{\sigma}(\vec{x}; r) + \lambda(\vec{x}; r)(1 - [\hat{\sigma}(\vec{x}; r)]^T \hat{\sigma}(\vec{x}; r)) = 0, \quad (4.13)$$

It is in turn being solved as a generalized eigenvalue problem,

$$Q(\vec{x}; r) \hat{\sigma}(\vec{x}; r) = \lambda(\vec{x}; r) \hat{\sigma}(\vec{x}; r). \quad (4.14)$$

For volumetric images, there are at most three distinct pairs of eigenvalues and eigenvectors. Noted that the eigenvalues can be positive (outward oriented flux), zero (no oriented flux) or negative (inward oriented flux).

These eigenvalues are denoted as $\lambda'_i(\vec{x}; r)$, for $\lambda'_1(\vec{x}; r) \leq \lambda'_2(\vec{x}; r) \leq \lambda'_3(\vec{x}; r)$, and the corresponding eigenvectors are $\vec{\omega}'_i(\vec{x}; r)$. Inside a curvilinear structure having stronger intensity than the background, the first two eigenvalues would be much smaller than the third one, $\lambda'_1(\vec{x}; r) \leq \lambda'_2(\vec{x}; r) << \lambda'_3(\vec{x}; r)$ and $\lambda'_3(\vec{x}; r) \approx 0$. The first two eigenvectors span the normal plan of the structure and the third eigenvector is the structure direction.

### 4.2.3 Eigenvalues and eigenvectors

The major difference between the eigenvalues-eigenvector pairs extracted from OOF and those from the Hessian matrix is that the computation of OOF and thus, its
eigenvalues and eigenvectors are grounded on the analysis of image gradient on the local sphere surface ($\partial B_r$ in Equation 4.3). In contrast, as pointed out by Sato et al. [14], the computation of the Hessian matrix is closely related to the results of applying the second derivative of Gaussian function on the image. This function computes the weighted intensity average difference between the regions inside the structure and in the vicinity of the structure. As such, the coverage of this function extends beyond the boundary of target structures and possibly includes structures nearby. As a result, the weighted intensity average difference computed by the second derivative of Gaussian function can be affected by the adjacent objects. It can be harmful to the detection accuracy of the Hessian matrix when closely located adjacent structures are present.

On the contrary, the evaluation of OOF is performed on the boundary of a local spherical region $\partial B_r$. Detection response of OOF is induced from the intensity discontinuities at the object boundary when the local sphere surface touches the object boundary of the structure. The detection of OOF is localized at the boundary of the local spherical region. The localized detection avoids the inclusion of objects nearby. Therefore, the OOF based detection is robust against the disturbance introduced by closely located adjacent structures.

The eigenvalues extracted from OOF are the values of oriented flux along the corresponding eigenvectors,

$$
\lambda'_i(\bar{x}; r) = [\bar{\omega}'(\bar{x}; r)]^T Q(r, \bar{x}) \bar{\omega}'(\bar{x}; r) = f(\bar{x}; r, \bar{\omega}'_i(\bar{x}; r)).
$$

(4.15)

The image gradient at the object boundary of a strong intensity curvilinear structure points to the centerline of the structure. Inside the structure, when the local spherical region boundary $\partial B_r$ (see Equation 4.1) touches the object boundary, at the contacting position of these two boundaries, the image gradient $\bar{v}(\bar{x})$ is aligned in the opposite direction of the outward normal $\hat{n}_A$, hence $\lambda'_1(\bar{x}; r) \leq \lambda'_2(\bar{x}; r) < 0$. On the other hand, the image gradient is perpendicular to the structure direction, the projected image gradient along $\bar{\omega}'_3(\cdot)$ has zero or very small magnitude, thus $\lambda'_3(\cdot) \approx 0$. In contrast, if OOF is computed for a voxel which is just outside the curvilinear structure, at the position where $\partial B_r$ touches the curvilinear structure boundary, the image gradient $\bar{v}(\bar{x})$ is in the same direction as the outward normal $\hat{n}_A$. It results in a large positive eigenvalue, that is $\lambda'_3(\cdot) >> 0$.

Combining multiple eigenvalues to tailor a measure for identifying structures in a
specific shape is now possible. For instance $\Lambda'_12(\vec{x}; r) = \lambda'_1(\vec{x}; r) + \lambda'_2(\vec{x}; r)$ can provide responses at the curvilinear object centerline with circular cross section. According to Equations 4.1 and 4.15,

$$\Lambda'_12(\vec{x}; r) = \frac{1}{4\pi r^2} \int_{\partial B_r} \left( [\mathbf{W}'_{12}(\vec{x}; r)]^T \vec{v}(\vec{x} + \vec{h}) \right) \cdot \left( [\mathbf{W}'_{12}(\vec{x}; r)]^T \hat{n}_A \right) dA,$$

where $\mathbf{W}'_{12}(\vec{x}; r) = \begin{bmatrix} \vec{\omega}'_1(\vec{x}; r) \\ \vec{\omega}'_2(\vec{x}; r) \end{bmatrix}$. The term involving the projection of $\hat{n}_A$ in the second half of the surface integral of the above equation is independent to the image gradient. This term varies along the boundary of the spherical region $\partial B_r$. It is a weighting function that makes the projected image gradients at various positions on the sphere surface contribute differently to the resultant values of $\Lambda'_12(\vec{x}; r)$. The values of $||[\mathbf{W}'_{12}(\vec{x}; r)]^T \hat{n}_A||$ on the local spherical region surface are shown in Figures 4.1a-c. A large value of $||[\mathbf{W}'_{12}(\vec{x}; r)]^T \hat{n}_A||$ represents the region where $\Lambda'_12(\vec{x}; r)$ is sensitive, as the projected image gradient at that region receives a higher weight for the computation of $\Lambda'_12(\vec{x}; r)$. The large valued regions of $||[\mathbf{W}'_{12}(\vec{x}; r)]^T \hat{n}_A||$ are distributed in a ring shape around the axis $\vec{\omega}_3(\vec{x}; r)$. In a curvilinear structure having circular cross section, the image gradient at the object boundary points to the centerline of the structure. Therefore, at the centerline of the structure, $\Lambda'_12(\vec{x}; r)$ delivers the strongest response if $r$ and the radius of the structure are matched.

4.2.4 Regarding multiscale detection

Multiscale detection is an essential technique for handling structures with various sizes. The multiscale detection of OOF involves repetitive computations of OOF using a set of radii ($r$ in Equation 4.1). The radius set should cover both the narrowest and the widest curvilinear structures in an image volume. Since the evaluation of OOF is localized at
the spherical region boundary, the spherical region has to touch the target structure boundary to obtain detection responses of OOF. As such, linear radius samples should be taken for OOF with the consideration of the voxel length in order to properly detect vessels in a given range of radii. It also ensures that a structure with non-circular cross section can induce detection responses of OOF obtained in at least one radius sample. We suggest that radius samples are taken in every 0.5 voxel length according to the Nyquist sampling rate.

For different values of $r$, the area covered by the surface integral of Equation 4.1 varies. The denominator of Equation 4.1, $4\pi r^2$ (the surface area of the spherical region) normalizes the oriented flux response over radii and hence, the computation of Equation 4.1 is scale-invariant. Such normalization is essential to aggregating OOF responses in a multiple scale setting. This OOF normalization scheme is distinct to the average-outward-flux (AOF) measure [87], which divides the outward flux by the surface area of the spherical region to attain the AOF-limiting-behavior. The AOF measure works only on the gradient of a distance function of a shape with its boundary clearly delineated. OOF, in contrast, is applied to a gradient of a gray-scale image, where no explicit shape
boundary is embedded and noise is possibly present.

In Figures 4.2a-f, we show two examples of evaluating OOF on image volumes consisting of one synthetic tube (Figures 4.2a and c) and two closely located synthetic tubes (Figures 4.2b and d) using multiple radii. The trace of the matrix $Q(\vec{x}; r)$ (Equations 4.12), which is equal to the sum of the eigenvalues of $Q(\vec{x}; r)$, is utilized to quantify the detection response strength of OOF. The trace of the matrix $Q(\vec{x}; r)$ is computed using multiple radii in both of the synthetic image volumes. In Figures 4.2e and f, it is observed that the trace of $Q(\vec{x}; r)$ is negative for all radii inside the tubes. It attains its maximal negative values at the tube centers and with the radius $r$ matching the tube radius, i.e. $r = 4$. The magnitudes of the trace of $Q(\vec{x}; 4)$ decline at positions away from the tube centers. In these positions, it attains its maximal magnitudes with smaller values of $r$ when approaching the tube boundaries. Therefore, making use of the trace of $Q(\vec{x}; r)$ as well as the eigenvalues of $Q(\vec{x}; r)$, (the trace of $Q(\vec{x}; r)$ is equal to the sum of its eigenvalues), with maximal negative values or maximal magnitudes over radii is capable of delivering a strong detection responses inside curvilinear structures.

When OOF is computed using multiple radii, the spherical regions of OOF with large radii possibly overshoot the narrow structure boundaries. The computation of OOF with overshot radii can include the objects nearby and adversely affects the detection responses of OOF (see Figure 4.2e, $r = 5$ and 6 versus Figure 4.2f, $r = 5$ and 6). In which, utilizing the eigenvalues or the trace of the matrix $Q(\vec{x}; r)$ with the maximal negative values or maximal magnitudes over radii as mentioned above can eliminate the responses obtained by using overshot radii. Furthermore, it excludes the OOF responses associated with undersized radii at the center of curvilinear structures (see Figures 4.2e and f, $r = 1, 2$ and 3). In the case that the radius of the spherical region $r$ matches the target structures, OOF avoids the inclusion of objects nearby. It therefore reports the same response at the centerlines of the tubes with $r = 4$ despite the presence of closely located structures (see Figure 4.2e, $r = 4$ versus Figure 4.2f, $r = 4$).

4.2.5 Vessel segmentation based on oriented flux symmetry

For the detection of curvilinear structures, we first describe analyzing the image gradients at a local spherical region boundary which touches the object boundary. These
image gradients are projected along a direction on the object’s cross-sectional plane. As shown in Figure 4.3a, along the cross-sectional plane of a curvilinear structure, the image gradients point to the structure center and form a symmetric pattern. When the local spherical region centers at the middle of the object, the projected gradients are symmetric (see Figure 4.3b). The symmetry of the projected gradients implies that both the magnitudes and the orientations of the projected gradients are symmetric with respect to the spherical region center. When the local spherical region centers at other positions, the projected gradients are aligned in various patterns (see Figures 4.3c-e). At the object boundary, the projected gradients at the local spherical region boundary point along the same direction (Figure 4.3d). As such, the projected gradient magnitudes are symmetric but the projected gradient orientations are antisymmetric with respect to the spherical region center. This pattern of image gradients is referred to as the antisymmetric pattern (Figure 4.3d). In the positions slightly inside or outside the structure, the projected image gradients are similar to the patterns as shown in Figures 4.3c and e respectively, in which both the projected gradient magnitudes and orientations are antisymmetric. This pattern of projected gradients is considered as neither symmetric nor antisymmetric. In summary, there are three situations discussed regarding various positions located,

- at the structure centers, the projected gradients are symmetric (the projected gradient magnitudes and orientations are symmetric);
- at object boundaries, the projected gradients are antisymmetric (the projected gradient magnitudes are symmetric but their orientations are antisymmetric); and
- slightly inside or outside the object, the projected gradients are neither symmetric nor antisymmetric (the projected gradient magnitudes and orientations are antisymmetric).

### 4.2.6 Oriented flux symmetry

In this section, two measures are devised to measure the symmetric gradient patterns and the antisymmetric gradient patterns. These two measures jointly quantify the gradient symmetry. They are therefore conveying reliable detection responses to identify the aforementioned three situations. This detection scheme is referred to as oriented
At the center (Figure 4.3b) $f(\vec{x}; r, \hat{\rho}) \approx 0$, $|a(\vec{x}; r, \hat{\rho})| \approx 0$, $|A_{12}(\vec{x}; r)| \approx 0$, $\Gamma_{12}(\vec{x}; r) \approx 0$. At the boundary (Figure 4.3d) $\ast \approx 0$, $\ast \approx 0$, $\ast \approx 0$, $\ast \approx 0$. Slightly inside or outside the object (Figures 4.3c and e) $> 0$, $> 0$, $> 0$, $> 0$, $\approx 0$.

| Location of $\vec{x}$, relative to a curvilinear object | $f(\vec{x}; r, \hat{\rho})$ | $|a(\vec{x}; r, \hat{\rho})|$ | $|A_{12}(\vec{x}; r)|$ | $\Gamma_{12}(\vec{x}; r)$ | $\Xi(\vec{x})$ |
|-----------------------------------------------------|----------------------------|----------------|----------------|----------------|----------------|
| At the center (Figure 4.3b)                          | $>> 0$                     | $\approx 0$   | $>> 0$         | $\approx 0$   | $>> 0$         |
| At the boundary (Figure 4.3d)                        | $\ast \approx 0$          | $\ast > 0$    | $\ast \approx 0$ | $\ast > 0$    | $\ast = 0$     |
| Slightly inside or outside the object (Figures 4.3c and e) | $> 0$                     | $> 0$         | $> 0$          | $> 0$          | $\approx 0$    |

Table 4.1: The analysis of the response magnitudes of various measures obtained at different positions $\vec{x}$. In the second, the third and the fifth columns, $\hat{\rho}$ is given as the direction on the structure cross-sectional plane, pointing from object centers to $\vec{x}$. In the second to the fifth columns except the entries with $\ast$, $r$ is given as the distance from $\vec{x}$ to the nearest object boundary; in the entries with $\ast$, $r$ is assumed to be a value smaller than the structure radius.

*flux symmetry* [89]. In oriented flux symmetry detection scheme, the first measure to help quantify the gradient symmetry is introduced on the basis of OOF [88] (see Section 4.2.1). Whereas the idea of OOF is to focus on finding the optimal projection orientation to maximize the resultant value of the oriented flux measure, we now aim at quantifying the image gradient symmetry. In this aspect, the oriented flux is regarded as a measure sensitive to the symmetric image gradient pattern. The oriented flux measure detects curvilinear structures grounded on its high sensitivity to the symmetric gradient pattern, as shown in Figure 4.3b. Given that Equation 4.1 is evaluated when $\hat{\rho}$ is a direction on the structure cross-sectional plane and $r$ is equal to the structure radius, $s(\vec{x}; r)$ attains its maximal value at the structure centers. The gradient symmetry decreases with respect to the positions away from the centers and thus, the strength of the oriented flux detection response declines accordingly. To identify the antisymmetric gradient pattern, the second measure is devised as,

$$ a(\vec{x}; r, \hat{\rho}) = \frac{1}{4\pi r^2} \int_{\partial B_r} (\vec{v}(\vec{x} + r\hat{n}_A) \cdot \hat{\rho}) dA. $$

(4.16)

This measure helps quantify the antisymmetry of the image gradients that contributes to the resultant value of the above oriented flux measure (Equation 4.1). The image gradients are obtained by convolving the image with the first derivatives of Gaussian function. It is referred to as *oriented flux antisymmetry* (OFA). It is sensitive to antisymmetric gradient patterns occurred at object boundaries. The OFA measure and the oriented flux measure alternatively return strong detection responses at the structure centers and at the structure boundaries (see the second and the third columns of Table 4.1).
4.2.7 Quantifying gradient symmetry along structure cross-sectional planes

Developing a measure to indicate the middle of vascular structures is now possible by aggregating the OFA measure and the oriented flux measure. It is achieved by first performing the eigen decomposition on the tensor $Q(\vec{x}; r)$ to obtain the optimal projection axis which maximizes the magnitude of the oriented flux measure [88]. There are three pairs of resultant eigenvalues and eigenvectors, denoted as $\lambda_j(\vec{x}; r)$ and $\hat{\omega}_j(\vec{x}; r)$ respectively, where $|\lambda_1(\vec{x}; r)| \geq |\lambda_2(\vec{x}; r)| \geq |\lambda_3(\vec{x}; r)|$. To detect curvilinear structures, the amount of the image gradients pointing to the structure center along its cross-sectional plane spanned by $\hat{\omega}_1(\vec{x}; r)$ and $\hat{\omega}_2(\vec{x}; r)$ [88] is evaluated,

$$\Lambda_{12}(\vec{x}; r) = \frac{1}{4\pi r^2} \int_{\partial B_r} \left( [\hat{\omega}_1(\vec{x}; r) \hat{\omega}_2(\vec{x}; r)]^T \vec{v}(\vec{x} + r\hat{n}_A) \right) \cdot \left( [(\hat{\omega}_1(\vec{x}; r) \hat{\omega}_2(\vec{x}; r)]^T \hat{n}_A \right) dA,$$

$$= f(\vec{x}; r, \hat{\omega}_1(\vec{x}, r)) + f(\vec{x}; r, \hat{\omega}_2(\vec{x}, r))$$

$$= \lambda_1(\vec{x}; r) + \lambda_2(\vec{x}; r). \quad (4.17)$$

Utilizing Equation 4.16, an OFA based measure associated with $\hat{\omega}_1(\vec{x}; r)$ and $\hat{\omega}_2(\vec{x}; r)$ is used to inspect the symmetry of gradients along structure cross-sectional planes. Denote $\gamma_1(\vec{x}; r)$ and $\gamma_2(\vec{x}; r)$ are the oriented flux antisymmetry along the directions $\hat{\omega}_2(\vec{x}; r)$ and $\hat{\omega}_2(\vec{x}; r)$, i.e. $\gamma_1(\vec{x}; r) = a(\vec{x}; r, \hat{\omega}_1)$ and $\gamma_2(\vec{x}; r) = a(\vec{x}; r, \hat{\omega}_2)$,

$$\Gamma_{12}(\vec{x}; r) = \frac{1}{4\pi r^2} \left| \int_{\partial B_r} \left( [\hat{\omega}_1(\vec{x}; r) \hat{\omega}_2(\vec{x}; r)]^T \vec{v}(\vec{x} + r\hat{n}_A) \right) dA \right|$$

$$= \sqrt{\gamma_1^2(\vec{x}; r) + \gamma_2^2(\vec{x}; r)}. \quad (4.18)$$

The above equation evaluates the magnitude of the sum of the projected image gradients at $\partial B_r$ on the detected structure cross-sectional plane. A moderate or large resultant value signals the situation that $\vec{x}$ is not located at the structure center (see the fifth column in Table 4.1). As presented in the second row, the forth and the fifth columns in Table 4.1, $|\Lambda_{12}(\vec{x}; r)| >> \Gamma_{12}(\vec{x}; r)$ in the middle of a curvilinear structure. Also, both $\Gamma_{12}(\vec{x}; r)$ and $\Lambda_{12}(\vec{x}; r)$ are robust against the intensity fluctuation along structure because they are evaluated along its cross-sectional planes. Besides, $\Lambda_{12}(\vec{x}; r)$ cannot give a very large magnitude outside the middle of structures (see the forth column in Table 4.1), including the positions either inside structure and closed to the structure boundaries, at the boundaries, or slightly outside the structure. It is
because the gradients are not symmetric at these positions. Based on $\Lambda_{12}(\vec{x}; r)$ and $\Gamma_{12}(\vec{x}; r)$, a measure that only reports positive responses in the middle of structure is, $\max (0, -\Lambda_{12}(\vec{x}; r) - \Gamma_{12}(\vec{x}; r))$. As the target object radius is unknown, the detection response at $\vec{x}$ is the maximum response among those responses computed in a set of radii. It therefore retrieves the most significant responses induced by the image gradients located at the object boundaries. As such,

$$\Xi(\vec{x}) = \max_{r \in L} \left( \max (0, -\Lambda_{12}(\vec{x}; r) - \Gamma_{12}(\vec{x}; r)) \right),$$

(4.19)

where $L$ is the radius set and is specified to include all possible radii of the target structures.

Regarding the proposed active contour based segmentation algorithm, the measure $\Xi(\vec{x})$ guides the evolving contours to expand along and inside curvilinear structures, even though there exists intensity fluctuation along them. To illustrate this idea, $\Xi(\vec{x})$ is evaluated using a noise corrupted synthetic tube with a radius of 4 voxels (Figures 4.4a-d). In this example, the radius set for $\Xi(\vec{x})$ is specified as $L = \{1, 1.5, 2, \ldots, 6\}$ voxels. A sharp intensity drop is observed along the tube. This synthetic tube exaggerates the situation where a sudden intensity change is present along a structure. Many existing active contour approaches [28][34][29] can misidentify sudden intensity drops as parts of object boundaries. On the contrary, the measure $\Xi(\vec{x})$ can consistently deliver positive detected values in the middle of the synthetic tube despite the intensity drop (Figure 4.4e). In each sub-figure of Figure 4.4j, it is observed when the detection radius of $\Xi(\vec{x})$ differs from the structure radius (all cases, except the one with "$L = \{4\}$"), the detection responses are smaller than that with a matched radius (in the case of "$L = \{4\}$"). It is because the symmetric gradient pattern vanishes as the spherical region radius differs from the structure radius. Thus, acquiring the maximum response obtained among all radii as in Equation 4.19 offers a reliable measure to quantify the symmetric gradient patterns.

As a major component of $\Xi(\vec{x})$, the magnitude of $\Lambda_{12}(\vec{x}; r)$ is insignificant at object boundaries (see the third row, the forth column in Table 4.1). This observation is validated using the above synthetic tube (see the grey solid line in Figure 4.4i, and it is a plot along the dotted line (i) in Figure 4.4c). Along the tube boundary, the response magnitude of $\Lambda_{12}(\vec{x}; r)$ is small and slightly fluctuating along the tube boundary. The
Figure 4.3: Illustrations of image gradients which form various patterns. The black arrows and grey solid lines represent image gradients and structure boundaries respectively. (a) Image gradients along a curvilinear structure cross-sectional plane. (b-e) Four examples showing image gradients located at the local spherical region boundaries (black dotted circles), projected along \( \hat{\rho} \).

Figure 4.4: (a) An xy-plane which shows the cross-section of the synthetic tube with a 4 voxel radius. (b) An xz-plane of the synthetic tube. (c) The numbers represent the intensity of various parts of the image in (b). (d) An xz-plane of the synthetic tube corrupted by additive Gaussian noise with standard deviation 0.1. (e-g) The xz-planes which show different measures. The black line in (g) showing the boundary where \( \max_{\hat{\rho}} |s(\vec{x}; p(\vec{x}), \hat{\rho})| \) is maximal along the vertical directions from the tube center to the image background. (h-i) The profiles of different measures obtained along the lines shown as dotted lines in (c). (j-k) The values of \( \Xi(\vec{x}) \) and \( \max_{r \in L} |a(\vec{x}; r, \hat{\rho})| \), which are obtained using one radius for each sub-figure.
response magnitude exhibits no significant change at the position where the tube intensity drops from 1.0 to 0.6. This implies that the response fluctuates randomly instead of following the tube intensity. Returning faint and randomly fluctuating response magnitudes along object boundaries is common to the approaches that extract image features by using symmetric measures, such as the oriented flux measure, the Hessian matrix [2][86][46] and the discretized Laplacian operator used by the flux method [18]. Since the local intensity variations across the object boundaries are not symmetric with respect to the boundaries, these symmetric measures deliver noisy responses at object boundaries. Evolving an active contour according to the symmetric measure based responses can lead to subsequent contour leakages. In the proposed method, $\Xi(\vec{x})$ is obtained by subtracting $\Lambda_{12}(\vec{x}; r)$ from $\Gamma(\vec{x}; r)$. It keeps the resultant values of $\Xi(\vec{x})$ zero at the object boundaries (see in the third row, the forth to the sixth columns in Table 4.1). It avoids the interference in the detection results incurred by the fluctuating responses of $\Lambda_{12}(\vec{x}; r)$ along boundaries.

4.2.8 The oriented flux symmetry based active contour model

To locate the structure boundaries in the proposed active contour model, the OFA measure which can capture the antisymmetric gradient patterns occurred at object boundaries is utilized. Suppose that $C$ is a closed contour, $\vec{S}$ is a position vector on the surface of $C$ and $\vec{N}(\vec{S})$ is the contour inward normal at $\vec{S}$, one of the criteria of finding the desired segmentation solution is,
\[ \arg\max_{\vec{c}} \int_{C} a(\vec{S}; r, \vec{N}(\vec{S}))dS, \]
where $dS$ is the infinitesimal area on $C$. Regarding the value of $r$, for positions inside curvilinear structures or slightly outside the structures, a proper value is the distance from those positions to the closest object boundary. It ensures that the responses of $s(\vec{x}; r)$ computed at the various positions, such as those shown in Figures 4.3c and e are significant and produced by the image gradient at the object boundaries. It is illustrated in Figure 4.4k, when $r$ is small (1 or 2 voxels), the OFA responses are concentrated in the vicinity of the tube boundary. As $r$ grows, more OFA responses can be observed in the regions further away from the tube boundary, despite the generally weaker responses than those obtained using smaller values of $r$. Therefore,
large values of \( r \) can guide the evolving contours which are located further away from object boundaries. Meanwhile, a small valued \( r \) is beneficial to precisely indicate the boundaries. Hence, \( r \) is estimated at each location by observing the OFA measure along the direction giving the strongest detection response, which is maximal among a set of radii. The estimated value at position \( \vec{x} \) is denoted as \( p(\vec{x}) \)

\[
p(\vec{x}) = \arg \max_{r \in \mathbb{P}} \left( \max_{\hat{\rho}} |s(\vec{x}; r, \hat{\rho})| \right) = \arg \max_{r \in \mathbb{P}} \left| \frac{1}{4\pi r^2} \int_{\partial B_r} \vec{v}(\vec{x} + r\hat{n}_A) dA \right|.
\]

(4.20)

To recognize structures adjacent to the strong edges of undesired objects, \( \mathbb{P} \) can contain only the smallest radius in \( \mathbb{L} \) discussed in the previous section. This avoids the detection being adversely affected by the strong edges of adjacent objects. For detection of curvilinear structures with complicated geometry (e.g. high curvature vessel or bifurcation) or irregular cross-sections, \( \mathbb{P} \) can be defined as the same as \( \mathbb{L} \). It ensures that various positions inside or slightly outside the structures can reach the nearest object boundary by those radii in \( \mathbb{P} \). In Figure 4.4g, the value of \( \max_{\hat{\rho}} |a(\vec{x}; p(\vec{x}), \hat{\rho})| \) is presented. Its profiles along the dotted lines in Figure 4.4c are given in Figures 4.4h and i. Along the vertical direction in Figure 4.4g, from the tube center to the image background regions (in the upper half and in the lower half of Figure 4.4g), the locations where \( \max_{\hat{\rho}} |a(\vec{x}; p(\vec{x}), \hat{\rho})| \) attains its maximum are shown as two black lines. These black lines are located along the tube boundaries, which become distinctive in the image of \( \max_{\hat{\rho}} |a(\vec{x}; p(\vec{x}), \hat{\rho})| \). It illustrates that evolving contours according to the OFA measure with the detection radius \( p(\vec{x}) \) can facilitate the detection of object boundaries.

The OFA measure is not limited to the detection of curvilinear structures as the oriented flux measure does. The OFA measure can also highlight the boundaries of various kinds of structures, which deviates from the curvilinear ones. However, this flexibility implies that the OFA measure is sensitive to all intensity changes, including the intensity fluctuation along curvilinear structures. As shown by the black solid line in Figure 4.4h, a large value of \( \max_{\hat{\rho}} |a(\vec{x}; p(\vec{x}), \hat{\rho})| \) is observed when the tube intensity drops from 1.0 to 0.6 inside the synthetic tube. Nonetheless, \( \Xi(\vec{x}) \) retains a high detection response as compared to \( \max_{\hat{\rho}} |a(\vec{x}; p(\vec{x}), \hat{\rho})| \) (see the black solid line and the black dotted line in Figure 4.4h). On the contrary, \( \max_{\hat{\rho}} |a(\vec{x}; p(\vec{x}), \hat{\rho})| \) is large at the tube boundary as compared to \( \Xi(\vec{x}) \) (see the black solid line and black dotted line.
in Figure 4.4i). These two measures alternatively deliver higher responses than their counterparts at the structure centers and at the object boundaries (also see the last two columns of Table 4.1). Hence, the desired resultant contour maximizes the following energy functional,

$$
E(C) = \int_{\text{Inside}(C)} \Xi(V) dV + \int_{C} a(S; p(S), N(S)) dS, \quad (4.21)
$$

where $V$ and $dV$ are the position vector inside $C$ and the infinitesimal volume respectively. The desired resultant contour is,

$$
\arg \max_{C} E(C).
$$

By using the gradient descent approach, the contour $C$ is evolved iteratively to acquire the desired solution. The evolving contour is represented as the zero level of a level set function $\Theta(x)$ \[40\]. The dynamic of the level set function can be described as \[41\]

$$
\Theta_t(x) = L(x)|\nabla \Phi(x)|, \quad \text{where} \quad L(x) = \Xi(x) - \text{div} \left( \frac{1}{4\pi p(x)} \int_{\partial B_p(x)} \bar{v}(x + p \hat{n}) dA \right). \quad (4.22)
$$

Considering the large positive responses of $\Xi(x)$ in the middle of curvilinear structures, the regions with large values of $\Xi(x)$ can be used as the seed positions to initialize the contour evolution. The function $L(x)$ is positive inside curvilinear structures to keep the contour expanding. It is negative at the positions slightly outside the structure boundaries. This eventually stops the evolving contour over the structure boundaries.

### 4.2.9 Fourier expressions of the oriented flux measure and the oriented flux antisymmetry measure

Studying the Fourier expressions helps devise the efficient computation algorithm for the proposed measures. It also reveals the orthogonality of the oriented flux measure and the OFA measure if they are regarded as two types of image filters. The Fourier

\footnote{The implementation is based on \[42\] and a publicly available library, ”The Insight Segmentation and Registration Toolkit“ (http://www.itk.org). The level set evolution is stopped when the increment of the segmented voxels over 20 iterations is less than 0.01% of them.}
expression of the OFA measure $s(\vec{x}; r, \hat{\rho})$ can be found by first rewriting Equation 4.16 as,

$$a(\vec{x}; r, \hat{\rho}) = \int_{\Omega} D(\vec{y}; r)(\hat{\rho} \cdot (\vec{\nabla} g) * I)(\vec{x} + \vec{y})dV$$

$$= \left( (\hat{\rho} \cdot (\vec{\nabla} g) * D)(\vec{x}; r) \right) * I(\vec{x}),$$

(4.23)

where $G$ is the Gaussian filter employed for smoothing the input image as discussed in Section 4.2.6, and $D(\vec{x}; r)$ is a spherical impulse function,

$$D(\vec{x}; r) = \begin{cases} \frac{(4\pi r^2)\text{ }^{-1}}{2\pi} & \text{if } ||\vec{x}|| = r, \\ 0 & \text{otherwise}. \end{cases}$$

By employing the Hankel transform [73],

$$\Phi(\vec{u}; r, \hat{\rho}) = \sqrt{-1} \frac{\hat{\rho} \cdot \vec{u}}{r ||\vec{u}||} e^{-2(\pi ||\vec{u}||^2 \sigma)} \sin(2\pi r ||\vec{u}||).$$

(4.24)

Besides, as stated in Chapter 4.2.2, the computation of $f(\vec{x}; r, \hat{\rho})$ is based on the filter formulated in the Fourier domain,

$$\Psi(\vec{u}; r, \hat{\rho}) = \left( \frac{\hat{\rho} \cdot \vec{u}}{r ||\vec{u}||^2} e^{-2(\pi ||\vec{u}||^2 \sigma)} \right) \left( \cos(2\pi r ||\vec{u}||) - \frac{\sin(2\pi r ||\vec{u}||)}{2\pi r ||\vec{u}||} \right).$$

(4.25)

As such, the computation of the oriented flux measure and the OFA measure are considered as two filtering operations. To facilitate the discussion, we denote $\Phi(\vec{u}; r, \hat{\rho}) = \mathcal{F} \left\{ (\hat{\rho} \cdot (\vec{\nabla} g) * D) \right\}(\vec{u}; r)$. These two functions are related by,

$$\Psi(-\vec{u}; r, \hat{\rho})\Phi(-\vec{u}; r, \hat{\rho}) = -\Psi(\vec{u}; r, \hat{\rho})\Phi(\vec{u}; r, \hat{\rho}),$$

$$\lim_{||\vec{u}|| \to 0} \Psi(\vec{u}; r, \hat{\rho}) = \lim_{||\vec{u}|| \to 0} \Phi(\vec{u}; r, \hat{\rho}) = 0,$$

and thus,

$$\int_{\Omega} \mathcal{F}^{-1} \left\{ \Psi \right\}(\vec{x}; r, \hat{\rho})\mathcal{F}^{-1} \left\{ \Phi \right\}^*(\vec{x}; r, \hat{\rho})d\vec{x} = \int_{\text{Image bandwidth}} \Psi(\vec{u}; r, \hat{\rho})\Phi(\vec{u}; r, \hat{\rho})d\vec{u} = 0,$$

where the superscript $*$ is function conjugate. Hence, given the same radius $r$ and orientation $\hat{\rho}$, the oriented flux measure and the OFA measure can be regarded as two orthogonal image filters. They convey two distinct types of information - the gradient
symmetry and the gradient antisymmetry. Fusing this information, the measure \( \Xi(\vec{x}) \) (Equation 4.19) judges which type of the information is more significant at a given position. It delivers responses only if that position exhibits a greater degree of gradient symmetry than that of antisymmetry.

On the other hand, the level set evolution speed \( \mathcal{L}(\vec{x}) \) is independent of the evolving contour. It is therefore evaluated prior to the level set evolution process, in which,

\[
[a(\vec{x}; r, \hat{z}_1) \ a(\vec{x}; r, \hat{z}_2) \ a(\vec{x}; r, \hat{z}_3)]^T = \frac{1}{4\pi r^2} \int_{\partial B_r} \vec{v}(\vec{x} + r\hat{n})dA. \tag{4.26}
\]

With the aid of the aforementioned Fourier expressions, \( \mathcal{L}(\vec{x}) \) is evaluated efficiently, with complexity \( O(|\mathcal{L}|m \log m) \). It is noted that, whereas the complicated formulation of \( \mathcal{L}(\vec{x}) \), its complexity is comparable to that of the FFT-based multiscale Hessian techniques (see [88] for details). Finally, the divergence in Equation 4.22 is evaluated using the central difference scheme.

### 4.3 Experimental Results

In this section, the proposed method is compared and validated in two aspects. First, as a low level curvilinear structure detector, OOF is compared against the Hessian matrix by using both synthetic data and real clinical cases. On the other hand, the oriented flux symmetry active contour method is compared with two well founded published vascular segmentation techniques, the CURVES algorithm [34] (CURVES) and the flux method [18] (FLUX).

The differential terms of the Hessian matrix are obtained by convolving image with the first derivatives of Gaussian functions. The eigenvalues and eigenvectors extracted from the Hessian matrix and \( \mathbf{Q}(\vec{x}; r) \) for OOF (Equation 4.14) are represented as \( \lambda_i^H(\vec{x}; r) \), \( \vec{\omega}_i^H(\vec{x}; r) \) and \( \lambda_i^Q(\vec{x}; r) \), \( \vec{\omega}_i^Q(\vec{x}; r) \), respectively. The order of the eigenvalues and the notation of sums of the first two eigenvalues (\( \Lambda_{12}^H(\vec{x}; r) \) and \( \Lambda_{12}^Q(\vec{x}; r) \)) are analogous to those described in Section 4.2.2.

On the contrary, prior to performing segmentation using the segmentation methods, the image volumes are smoothed by a Gaussian filter with a scale factor of 1 smallest voxel length, for noise reduction for CURVES and FLUX, and for ensuring differentiability of the discrete image signal for the proposed method. Based on visual
assessments of the clinical data, the widths of the target structures are all less than 3mm. The radius set used for \textbf{FLUX} and the proposed method \( \mathbb{L} \) covers the radii from 1 voxel-length (the physical length depends on the voxel sizes of different images) to 3mm (or 8 voxel-length in the synthetic case). The second radius set for the proposed method \( \mathbb{R}' \) is the same as \( \mathbb{L} \) in all tests except the forth real vascular image case. For \textbf{CURVES}, for each case, we present the structure which reports no leakage and that segmented region has the largest number of segmented voxels among those obtained using different heuristic parameter values used by \textbf{CURVES}.

### 4.3.1 Synthetic data

The proposed method, OOF, is examined in this section using synthetic images containing tori with various sizes. There are 10 synthetic volumetric images in the size of \( 100 \times 100 \times 100 \) voxels being generated for the synthetic experiments. The main purpose is to verify the performance of OOF and compare OOF with the Hessian matrix when closely located structures are present.

The configurations of the tori in the synthetic images are shown in Figure 4.5. The number of tori in different synthetic images varies and depends on the values of \( d \) and \( R \). The tori are placed in a layer fashion along the \( z \)-direction. The strategy to generate the first layer of tori is to place a torus with \( D = 10 \) at the altitude \( z = 8 \). The center of that torus is randomly selected among the positions \((x = 45, y = 45, z = 8)\), \((x = 35, y = 45, z = 8)\), \((x = 45, y = 35, z = 8)\) and \((x = 35, y = 35, z = 8)\). We keep deploying adjacent tori centered at the same position of the first torus but having larger values of \( D \) in an interval of \( 2R + d \) until \( D \leq 42 \). Each successive layer of tori is generated in a \( 2R + d \) interval of altitude \( z \) for \( z \leq 90 \). The center of each layer of tori is randomly selected among the positions of \((x = 35, y = 35)\), \((x = 45, y = 35)\), \((x = 35, y = 45)\) and \((x = 45, y = 45)\). The background intensity of these images is 0 and the intensity inside the tori is assigned to 1. The torus images are smoothed by a Gaussian kernel with scale factor 1 to mimic the smooth intensity transition from structures to background. Each synthetic image is corrupted by two levels of additive Gaussian noise, with standard deviations of \( \sigma_{\text{noise}} = \{0.75, 1\} \). Finally, 20 testing cases are generated for this experiment.

The experiment results are based on the measures obtained in the estimated object
scales of the both methods. For the testing objects with circular cross section such as the tori used in this experiment, computing the sums of the first two eigenvalues \( \Lambda'_{12}(\vec{x}; r) \) and \( \Lambda'_{12}(\vec{x}; r) \) at structure centerlines is useful to determine the structure scales. The reason is that \( \Lambda'_{12}(\vec{x}; r) \) of the Hessian matrix quantifies the second order intensity change occurred along the radial direction of a circle on the normal plane of the structure. Meanwhile, for OOF, \( \Lambda'_{12}(\vec{x}; r) \) evaluates the amount of gradient pointing to the centerlines of tubes with circular cross section. Based on the above observation, the object scale is obtained as

\[
\ell'_{\text{H}}(\vec{x}) = \arg \max_{r \in L_{\text{H}}} \left( -\frac{r^2}{3} \Lambda'_{12}(\vec{x}; \sqrt{3}) \right)
\]

for the Hessian matrix (see [10, 90] for details regarding the structure scale detection and [15] for Hessian matrix based feature normalization over scales) and

\[
\ell'_{\text{Q}}(\vec{x}) = \arg \max_{r \in L_{\text{Q}}} \left( -\Lambda'_{12}(\vec{x}; r) \right)
\]

for OOF. The set of discrete detection scales of OOF and detection scales of the Hessian matrix are represented as \( L_{\text{Q}} \) and \( L_{\text{H}} \) respectively. These scales cover the structure radii ranged from 1 to 6 voxel length. The radii of OOF are taken for each 0.5 voxel length and there are in total 11 different radii in \( L_{\text{Q}} \). Meanwhile, the same number of scales are logarithmically sampled for the Hessian matrix scale set \( L_{\text{H}} \) so as to minimize the detection error of the Hessian matrix [14].

There are two measures being studied for the comparison of OOF and the Hessian matrix, “Angular discrepancy” and “Response fluctuation”. For objects with circular cross section and having stronger intensity than the background, the third eigenvector represents the structure direction. At the estimated structure scales, we measure the angular discrepancy of the Hessian matrix and OOF by

\[
\arccos \left( |\vec{g}(\vec{t}) \cdot \omega'_{3\text{H}}(\vec{x}; \ell'_{\text{H}}(\vec{t}))| \right), \quad \arccos \left( |\vec{g}(\vec{t}) \cdot \omega'_{3\text{Q}}(\vec{x}; \ell'_{\text{Q}}(\vec{t}))| \right),
\]

respectively, where \( \vec{g}(\vec{t}) \) is the ground truth direction, which is defined as the tangent direction of the torus inner-tube centerline at the position \( \vec{t} \), \( \vec{t} \in T \), where \( T \) is a set of samples taken in every unit voxel length at the inter-tube centerlines of the tori. Bilinear interpolation is applied if \( \vec{t} \) does not fall on an integer coordinate. The value of the angular discrepancy is in a range of \( [0, \pi/2] \) and a small value of the angular discrepancy represents an accurate estimation of structure direction.

The second measure, “Response fluctuation” for the tori having circular cross section is defined as the ratio between the variance and the mean absolute value of \( \Lambda_{12}(\cdot) \).
The “Response fluctuation” of the Hessian matrix and OOF are defined as

\[
\frac{\text{Var}_{\vec{t}\in T}(\Lambda'_{12}(\vec{x}; l'_{H}(\vec{t})))}{\text{Mean}_{\vec{t}\in T}(\left|\Lambda'_{12}(\vec{x}; l'_{H}(\vec{t}))\right|)}; \quad \frac{\text{Var}_{\vec{t}\in T}(\Lambda'_{12}(\vec{x}; l'_{Q}(\vec{t})))}{\text{Mean}_{\vec{t}\in T}(\left|\Lambda'_{12}(\vec{x}; l'_{Q}(\vec{t}))\right|)},
\]

respectively. A small value of fluctuation implies a stable response, which is robust against the adverse effects introduced by the interference of closely located structures as well as image noise.

The results based on the above measurements for different combinations of noise levels and torus separations are presented and listed in Table 4.2. In Table 4.2, it is observed that both the Hessian matrix and OOF perform better when the inner-tube radii of tori rise. It is because structures having low curvature surfaces such as large inner-tube radius tori are easier to be detected than the tori having small inner-tube radii. To evaluate the performance drops of OOF and the Hessian matrix in handling images having closely located structures, the changes of the mean angular discrepancy and response fluctuation in various cases are investigated in Table 4.3. In the entries of Table 4.3, a small value represents high robustness against the reduction of torus separation (Table 4.3a); the increment of noise level (Table 4.3b); and both of them (Table 4.3c).

As previously mentioned, the detection of OOF is localized at the boundary of local spherical regions. The OOF detection responses are merely induced from the intensity discontinuities taken place at the structure boundary, when the local sphere surface of OOF touches the structure boundary. In contrast to OOF, the Hessian matrix based detection relies on the computation of the weighted intensity average difference between the regions inside the structure and in the vicinity of the structure, where a nearby object is possibly present. As the correct detection scale of the Hessian matrix increases for recognizing large scale structures, the detection coverage of the correct scale of the Hessian matrix expands. It increases the chances to include adjacent structures. Hence, the increments of mean angular discrepancies and response fluctuations of the Hessian matrix are larger than those of OOF, especially when \( R \) increases, in the cases that the torus separation is reduced from 5 voxel length to 2 voxel length (the second and the forth columns versus the first and the third columns of Table 4.3a).

Moreover, in the situation where noise is increased (Table 4.3b), it is observed that
OOF (the second and the forth columns) has less increment of the mean angular discrepancies than the Hessian matrix (the first and the third columns), particularly when $R$ increases. Although the Gaussian smoothing taken by the Hessian matrix partially eliminates noise from the image volume, the smoothing process also reduces the edge sharpness of the structure boundaries. In particular, the scale factor of the Gaussian smoothing process of the Hessian matrix has to rise to deal with large scale structures. Consequently, the Hessian matrix performs detection based on the smoothed object boundaries which are easier to be corrupted by image noise. For OOF, the detection does not require Gaussian smoothing using a large scale factor ($\sigma = 1$ for OOF). It retains the edge sharpness of the structure boundaries. Therefore, the OOF detection has higher robustness against image noise than the Hessian matrix. As expected, when the torus separation is reduced to 2 voxel length and the noise level is raised to $\sigma_{\text{noise}} = 1$, OOF has higher robustness than the Hessian matrix, against the presence of both closely located adjacent structures and high level noise than the Hessian matrix (Table 4.3c).

To summarize the results of the synthetic data experiments (Tables 4.2 and 4.3), OOF is validated in several aspects, the structure direction estimation accuracy, the stability of responses, the robustness against the disturbance introduced by closely located structures and the increment of noise levels. In some applications, an accurate structure direction estimation is vital. For instance, vascular image enhancement, the estimated structure direction is to avoid smoothing along the directions across object boundaries. Furthermore, for tracking curvilinear structure centerlines (a centerline tracking example is in [81]), estimated structure direction is to guide the centerline tracking process. Also, small response fluctuation facilitates the process to extract curvilinear structures or locate object centerlines by discovering the local maxima or ridges of the response.

On the other hand, the structure direction estimation accuracy and the stability of structure responses of OOF are robust against the reduction of structure separation and the increment of noise levels. As such, employing OOF to provide information of curvilinear structures is highly beneficial for curvilinear structure analysis.
Figure 4.5: The description of the tori. These tori have been used in the synthetic data experiments. The center of the tori in each layer is randomly selected from the positions of \((x = 35, y = 35), (x = 45, y = 35), (x = 35, y = 45)\) and \((x = 45, y = 45)\). The values of \(d\) and \(R\) are fixed to generate a torus image. In the experiments, there are 10 torus images generated by using 10 pairs of \(\{d, R\}\), \{2, 1\}, \{2, 2\}, \{2, 3\}, \{2, 4\}, \{2, 5\}, \{5, 1\}, \{5, 2\}, \{5, 3\}, \{5, 4\}\) and \{5, 5\}.

4.3.2 Application example - blood vessel extraction

In this section, we demonstrate an example on utilizing OOF to supply information of curvilinear structures for extracting vessels in a vascular image. The vascular image utilized in this example is a phase contrast magnetic resonance angiographic (PCMRA) image volume (Figure 4.6a) and the image intensity represents the blood flow speed inside the vasculature. The challenges to extraction algorithms are the presence of closely located vessels due to the complicated geometry of vascular structures, and the small and low intensity vessels in images with relatively high background noise level.

To perform comparison between OOF and the Hessian matrix, we replace the Hessian matrix based information used by a vessel extraction measure with the similar information extracted from OOF. It is reminded that the main goal of this experiment is to validate OOF as a general curvilinear structure detector. Therefore, measures having heuristic parameters which involve different values for certain kinds of structures are not preferred in this example, such as the vesselness measure [2] or majority of techniques in [14] for integrating multiple eigenvalues which involve heuristic parameters. On the other hand, the sum of the first two eigenvalues employed in the synthetic experiments is designed to provide responses at centerlines of curvilinear structures. It is not suitable for vessel extraction, which requires a measure to give vessel detection responses in the entire image region. We make use of the geometric mean of the first two eigenvalues, which was suggested for the detection of vessels in [10, 14],

\[
\mathcal{G}'(\bar{x}; r) = \begin{cases} 
\sqrt{\lambda_1'(\bar{x}; r)\lambda_2'(\bar{x}; r)}, & \lambda_1'(\bar{x}; r) \leq \lambda_2'(\bar{x}; r) < 0, \\
0, & \text{otherwise},
\end{cases}
\] (4.29)
This measure is computed in a set of discrete scales to obtain the maximum responses,

\[ G^\mathbf{H}(\vec{x}) = \max_{r \in L^\mathbf{H}} \left( \frac{r^2}{3} G^\mathbf{H}(\vec{x}; r) \right) \]

\[ G^\mathbf{Q}(\vec{x}) = \max_{r \in L^\mathbf{Q}} \left( G^\mathbf{Q}(\vec{x}; r) \right). \]  

(4.30)

There are 15 radii and scales being employed for \( L^\mathbf{Q} \) and \( L^\mathbf{H} \) respectively to cover the structure radii ranged from 1 to 8 voxel length. The vessel measure response is retrieved as the resultant values of Equation 4.29 obtained in the estimated object scales. The binary extraction results are obtained by thresholding the vessel measure responses. The thresholding value is found empirically so that neither over-segmentation nor under-segmentation of major vessels is observed and the same amount of voxels for both the methods are selected. Finally, 4% of voxels having the highest vessel measure responses among all voxels are thresholded as the extraction results.

The vessel extraction results are shown in Figures 4.6b and c. The interesting positions in the results are highlighted by the numbered arrows in Figures 4.6b and c. In the regions pointed at by the fifth and sixth arrows in Figures 4.6b and c, the Hessian based method misidentifies closely located vessels as merged structures. On the contrary, the OOF based method is capable of discovering the small separation between the closely located vessels. This result is consistent with the findings in the synthetic experiments, where OOF is more robust than the Hessian matrix when handling closely located structures (Table 4.3a).

In Figure 4.6c, it is found that several vessels with weak intensity (arrows 1, 2, 3, 4 and 7) are missed by the Hessian based method where the OOF based method has no problem to extract them (Figure 4.6b). The reason is that the noise level relative to the weak intensity structures is higher than those relative to strong intensity structures. Coherent to the synthetic experiments, in which OOF shows higher robustness against image noise as compared to Hessian matrix (see Table 4.3b). The vessel extraction results in this real case experiment reflects that robustness against image noise is important on extracting vessels with weak intensity.

### 4.3.3 Active contour segmentation on synthetic data

Using a noise corrupted synthetic spiral, as shown in Figures 4.7a-c, we examine the ability of different approaches to segment an elongated structure, where the intensity
Figure 4.6: (a) A phase contrast magnetic resonance angiographic image volume with the size of $213 \times 143 \times 88$ voxels. (b) The vessel extraction results obtained by using the optimally oriented flux based method. (c) The vessel extraction results obtained by using the Hessian matrix based method.

is changing along the structure and image noise is present. The inner radius and the intensity of the spiral are gradually reduced from 4 voxels and a value of 1 at the bottom of the structure, to 1 voxel and a value of 0.5 at the top of the structure. This synthetic spiral is corrupted using additive Gaussian noise with standard deviation equal to 0.1.

The active contours of all methods are initialized inside the bottom part of the spiral (Figure 4.7d). In Figure 4.8, the segmentation results of various approaches are shown. In which, the contour of CURVES cannot propagate along the spiral to reach the top of the structure. As the image intensity declines along the structure, the image gradient generated by the image noise inside the spiral exerts higher image force than
that exerted by the weak boundary of the structure having low intensity value. Thus, the evolving contour of CURVES is halted inside the structure. Besides, the contour of FLUX (Figure 4.8) penetrates the object boundary and it results in contour leakages. It is because the symmetric discretized Laplacian operator used by FLUX returns faint responses along object boundaries. The contour is randomly evolved at the low contrast spiral boundaries and leaks through these boundaries. On the contrary, for the proposed method, the measure $\Xi(\vec{x})$ allows the contour to propagate along structure and the OFA measure stops the evolving contour at object boundaries. The proposed method is therefore capable of segmenting the entire spiral without leakages (see Figure 4.8, the proposed method).

4.3.4 Active contour segmentation on real vascular images

There are four angiographic images employed in this experiment, including two intracranial phrase contrast magnetic resonance angiographic (PC-MRA) image volumes (Figures 4.9a and b), one intracranial time-of-flight MRA (TOF-MRA) image volume (Figure 4.9c) and one cardiac computed tomographic angiographic (CTA) image volume (Figure 4.9d). The voxel intensity of these images was scaled to be in the range
Figure 4.9: The image volumes used in the real vascular image experiment. (a, b) The perspective maximum intensity projections, along the axial, the sagittal and the coronal directions of two intracranial PC-MRA volumes; (c) the axial perspective maximum intensity projection (left) and the 53th image slice (right) of an intracranial TOF-MRA volume; (d) The 182th (left) and 214th (right) slices of a cardiac CTA volume. The red circles indicate the aorta and the blue dots are the manually placed initial seed points.
Figure 4.10: (a, b, e, f) The segmentation results of the clinical cases shown in Figs. 4.9a, b, c and d respectively, by using CURVES. (c, d, g, h) The segmentation results of the clinical cases shown in Figs. 4.9a, b, c and d respectively, by using FLUX.

Figure 4.11: The segmentation results obtained by using the proposed method from the four angiographic images shown in Figs. 4.9a-d.
of 0 and 1. The experimental settings of different approaches are the same as those settings in the synthetic data experiments except the procedures of contour initialization. For the PC-MRA and TOF-MRA image volumes, the initial level set function is obtained by thresholding the 0.1% image voxels, which produce the highest values of $\Xi(\vec{x})$ among all voxels in the image. The initial contours are only placed in the middle of vessels with large detected values of $\Xi(\vec{x})$. For the CTA image volume, the object of interest - coronary arteries are connected with the aorta, which is not a part of the target region. They share the same intensity range. We manually select two spheres with a radius of 3mm at two positions where the aorta is connected with the left coronary artery, and the right coronary artery. The level set update (i.e. the contour evolution) is disabled within these two spheres for all methods. Two initial seeds are placed in the left coronary artery and the right coronary artery (see the blue dots in Figure 4.9d). In this CTA image, the radius set $\mathbb{P}$ of the proposed method contains only the smallest radius in $\mathbb{L}$ to avoid the disturbance introduced by the edges adjacent to the arteries (see Figure 4.9d).

It is noted that the vessel intensity of the flow-sensitive PC-MRA images fluctuates significantly because of the variation of blood flow speeds inside the vessels with different sizes. This intensity fluctuation produces image gradient along vascular structures and stops the evolving contours of CURVES inside the vessels (see Figures 4.10a and b). For FLUX, the faint responses detected by the symmetric discretized Laplacian operator cannot precisely position the boundaries of the vessels. The evolving contours leak through the object boundaries and are subsequently guided by image noise as shown in Figures 4.10c and d. In TOF-MRA image, the non-vascular tissues can report intensity values similar to those of vascular regions (see Figure 4.10c, right). It greatly reduces the intensity contrast of the vessel boundaries where a non-vascular structure with similar intensity is nearby. As a result, the weak vessel boundaries cannot exert enough image force to draw the evolving contours of CURVES along the vessels and causes under-segmentation (Figure 4.10e). In Figure 4.10g, the contour of FLUX expands beyond the weak vessel boundaries and follows the non-vascular structures. In the CTA image, the evolving contours of both CURVES and FLUX (Figures 4.10f and h) leak through the arteries and follow the edges of the heart chamber surface. The contour evolution of FLUX and CURVES in this case was manually stopped for contour visualization.
In contrast, the measure \( M(\vec{x}) \) of the proposed method encourages contours to expand along vessels despite the intensity variation of vessels. On the other hand, the OFA based measure, as stated in Equation 4.21, is capable of halting the evolving contours at the vessel boundaries. It can segment the vessels without leakage (Figures 4.11a-c). Based on the visual comparison between the segmented vessels of the proposed method, and the original image volumes shown in Figures 4.9a-c, the proposed method is able to deliver faithful segmentation results. It can also withstand the disturbance introduced by the irrelevant edges adjacent to the target structures. Thus, the proposed method successfully segment the coronary arteries as presented in Figure 4.11d.

### 4.4 Perspectives

In this chapter, we have presented the use of optimally oriented flux (OOF) for detecting curvilinear structures. With the aid of the analytical Fourier expression of OOF, no discretization and orientation sampling are needed. It therefore leads to a highly efficient computation of OOF. Computation-wise, it has the same complexity as in the computation of the most commonly used approach, Hessian matrix. Furthermore, computation of OOF is based on the image gradient at the boundary of local spheres. It focuses on the detection of intensity discontinuities occurring at the object boundaries of curvilinear structures.

The OOF based detection avoids including adjacent objects. Thus, it exhibits the robustness against the interference introduced by closely located adjacent structures. This advantage is validated and demonstrated by a set of experiments on synthetic and real image volumes. In addition, in the experiments, it is observed that OOF has higher structure direction estimation accuracy and stable detection responses under the disturbance of high level image noise. With the aforementioned high detection accuracy and robustness, OOF, as opposed to the Hessian matrix, is better able to supply information of curvilinear structures, for curvilinear structure analysis.

A potential extension of the proposed method is the use of OOF based features for the detection or segmentation of vessel walls. This can be achieved by exploiting the OOF detection responses obtained in multiple radii at each voxel. In the image modalities (for instance black blood MR images or ultrasonic images) where vessel walls
are distinguishable from vessel lumen and non-vascular regions, the OOF detection responses would be strong when the detection sphere touches either the inner-wall or the outer-wall boundaries. This requires the responses obtained from using two different detection radii to be jointly considered to properly detect vessel walls. With the aid of the prior information regarding the desired vessel walls, such as the cross-section shapes of the vessels and the thickness of the target vessel walls, the OOF based detection can deliver promising vessel wall detection results.

Based on OOF, the proposed active contour model is devised based on various measures which aim at locally quantifying the image gradient symmetry. In the proposed application vascular segmentation, since tissue intensity can vary spatially due to the presence of multiplicative bias field, the proposed model avoids encapsulating the regional intensity variance information [48][91]. Albeit the three dimensional formulation of the proposed method, it is general to cope with curvilinear structures in two, three or higher dimensions, if any. Also, we are acquiring more data sets and segmenting ground truth in order to perform quantitative comparison to other approaches.

Besides, analogous to the studies in [34][14], introducing geometric constraints to the proposed active contour model may be beneficial. The major concern is that these geometric constraints require intensive parameter searching. Furthermore, vast numbers of application specific constraints or supplementary information have been proposed recently, for instance, detecting only structures with circular cross sections, regularization based on the curvature of structure centerlines or structure radii, disallowing bifurcation, exploiting training data or interactive segmentation (see [24] for a comprehensive survey). The proposed measures can provide useful features to detect curvilinear structures along with these constraints or supplementary information for particular applications.

Regarding the proposed active contour model, the oriented flux symmetry based formulation expands the evolving contours in the middle of curvilinear structures where the image gradients are symmetric. The contours are eventually driven to the object boundaries, in order to maximize the gradient asymmetry measure along the contour inward normal direction. Benefited from the oriented flux asymmetry measure and the oriented flux measure, the proposed model is capable of segmenting entire structures without contour leakages, in both the experiments using the synthetic image and the
real images of different modalities. It is experimentally demonstrated that the oriented flux symmetry based active contour model achieves promising segmentation results.
<table>
<thead>
<tr>
<th>d = 5, $\sigma_{\text{noise}} = 0.75$</th>
<th>Angular discrepancy</th>
<th>Response fluctuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>Hessian matrix</td>
<td>OOF</td>
</tr>
<tr>
<td>1</td>
<td>0.406 (0.250)</td>
<td>0.309 (0.176)</td>
</tr>
<tr>
<td>2</td>
<td>0.232 (0.197)</td>
<td>0.180 (0.093)</td>
</tr>
<tr>
<td>3</td>
<td>0.109 (0.111)</td>
<td>0.110 (0.065)</td>
</tr>
<tr>
<td>4</td>
<td>0.063 (0.068)</td>
<td>0.062 (0.054)</td>
</tr>
<tr>
<td>5</td>
<td>0.054 (0.075)</td>
<td>0.059 (0.027)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>d = 2, $\sigma_{\text{noise}} = 0.75$</th>
<th>Angular discrepancy</th>
<th>Response fluctuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>Hessian matrix</td>
<td>OOF</td>
</tr>
<tr>
<td>1</td>
<td>0.408 (0.260)</td>
<td>0.304 (0.178)</td>
</tr>
<tr>
<td>2</td>
<td>0.305 (0.215)</td>
<td>0.227 (0.129)</td>
</tr>
<tr>
<td>3</td>
<td>0.162 (0.155)</td>
<td>0.135 (0.072)</td>
</tr>
<tr>
<td>4</td>
<td>0.098 (0.127)</td>
<td>0.087 (0.055)</td>
</tr>
<tr>
<td>5</td>
<td>0.079 (0.125)</td>
<td>0.065 (0.033)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>d = 5, $\sigma_{\text{noise}} = 1$</th>
<th>Angular discrepancy</th>
<th>Response fluctuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>Hessian matrix</td>
<td>OOF</td>
</tr>
<tr>
<td>1</td>
<td>0.518 (0.288)</td>
<td>0.409 (0.239)</td>
</tr>
<tr>
<td>2</td>
<td>0.331 (0.252)</td>
<td>0.246 (0.148)</td>
</tr>
<tr>
<td>3</td>
<td>0.204 (0.218)</td>
<td>0.169 (0.109)</td>
</tr>
<tr>
<td>4</td>
<td>0.112 (0.158)</td>
<td>0.110 (0.080)</td>
</tr>
<tr>
<td>5</td>
<td>0.107 (0.159)</td>
<td>0.082 (0.044)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>d = 2, $\sigma_{\text{noise}} = 1$</th>
<th>Angular discrepancy</th>
<th>Response fluctuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>Hessian matrix</td>
<td>OOF</td>
</tr>
<tr>
<td>1</td>
<td>0.532 (0.305)</td>
<td>0.414 (0.243)</td>
</tr>
<tr>
<td>2</td>
<td>0.435 (0.278)</td>
<td>0.319 (0.192)</td>
</tr>
<tr>
<td>3</td>
<td>0.279 (0.243)</td>
<td>0.200 (0.132)</td>
</tr>
<tr>
<td>4</td>
<td>0.181 (0.220)</td>
<td>0.125 (0.095)</td>
</tr>
<tr>
<td>5</td>
<td>0.157 (0.217)</td>
<td>0.097 (0.088)</td>
</tr>
</tbody>
</table>

Table 4.2: The performance of optimally oriented flux and the Hessian matrix obtained in the synthetic data experiments. The entries in the columns of ”Angular discrepancy” include two values, the mean and the standard deviation (the bracketed values) of the resultant values of Equation 4.27. The values in the columns of ”Response fluctuation” are the results based on Equation 4.28.
Changes of mean angular discrepancy  & Changes of response fluctuation  \\
<table>
<thead>
<tr>
<th>R</th>
<th>Hessian matrix</th>
<th>OOF</th>
<th>Hessian matrix</th>
<th>OOF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>+0.002</td>
<td>-0.005</td>
<td>+0.013</td>
<td>+0.006</td>
</tr>
<tr>
<td>2</td>
<td>+0.073</td>
<td>+0.048</td>
<td>+0.052</td>
<td>+0.035</td>
</tr>
<tr>
<td>3</td>
<td>+0.053</td>
<td>+0.025</td>
<td>+0.041</td>
<td>+0.023</td>
</tr>
<tr>
<td>4</td>
<td>+0.035</td>
<td>+0.024</td>
<td>+0.033</td>
<td>+0.031</td>
</tr>
<tr>
<td>5</td>
<td>+0.025</td>
<td>+0.005</td>
<td>+0.034</td>
<td>+0.012</td>
</tr>
</tbody>
</table>

(b)  
Changes of mean angular discrepancy  & Changes of response fluctuation  \\
<table>
<thead>
<tr>
<th>R</th>
<th>Hessian matrix</th>
<th>OOF</th>
<th>Hessian matrix</th>
<th>OOF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>+0.112</td>
<td>+0.100</td>
<td>+0.050</td>
<td>+0.045</td>
</tr>
<tr>
<td>2</td>
<td>+0.099</td>
<td>+0.067</td>
<td>+0.044</td>
<td>+0.040</td>
</tr>
<tr>
<td>3</td>
<td>+0.095</td>
<td>+0.059</td>
<td>+0.036</td>
<td>+0.010</td>
</tr>
<tr>
<td>4</td>
<td>+0.049</td>
<td>+0.047</td>
<td>+0.030</td>
<td>+0.026</td>
</tr>
<tr>
<td>5</td>
<td>+0.053</td>
<td>+0.023</td>
<td>+0.021</td>
<td>+0.004</td>
</tr>
</tbody>
</table>

(c)  
Changes of mean angular discrepancy  & Changes of response fluctuation  \\
<table>
<thead>
<tr>
<th>R</th>
<th>Hessian matrix</th>
<th>OOF</th>
<th>Hessian matrix</th>
<th>OOF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>+0.126</td>
<td>+0.104</td>
<td>+0.068</td>
<td>+0.052</td>
</tr>
<tr>
<td>2</td>
<td>+0.203</td>
<td>+0.139</td>
<td>+0.106</td>
<td>+0.079</td>
</tr>
<tr>
<td>3</td>
<td>+0.170</td>
<td>+0.090</td>
<td>+0.085</td>
<td>+0.039</td>
</tr>
<tr>
<td>4</td>
<td>+0.118</td>
<td>+0.062</td>
<td>+0.068</td>
<td>+0.054</td>
</tr>
<tr>
<td>5</td>
<td>+0.103</td>
<td>+0.037</td>
<td>+0.054</td>
<td>+0.029</td>
</tr>
</tbody>
</table>

Table 4.3: The changes of mean angular discrepancy and response fluctuation from the case of ”$d = 5, \sigma_{\text{noise}} = 0.75”$ to other three cases presented in Table 4.2.
CHAPTER 5

CONCLUSION AND FUTURE DEVELOPMENT

With advances in image acquisition techniques, such as high power magnetic resonance angiography, vascular images reach higher resolutions. Thus, smaller and thinner blood vessels can be revealed using the new acquisition techniques. The geometry of the smaller and thinner vessels are more complicated than that of the relatively large vessels visible in previously available imaging techniques. The complicated vessel geometry includes bifurcation, high curvature vessels, closely located and disconnected (kissing) vessels and crossing vessels (in 2D cases). These vessels deviate from the tubular shape assumption of vessels taken by different algorithms [2, 10, 13–15, 29, 32–34]. Although some studies [3, 14, 29] have investigated how to handle bifurcation, there are needs for techniques to deal with high curvature vessels, kissing vessels and crossing vessels.

Besides, the partial volume effect occurred in thin vessels also fluctuates the intensity contrast between thin blood vessels and image background regions. The partial volume effect occurs when a single image voxel represents multiple types of tissue. This effect commonly exists on the boundaries of vessels where the voxel intensity is composed of vessels and backgrounds. When the partial volume effect happens on the boundaries of thin vessels, the intensity contrast between the entire cross section of the thin vessels and its background is reduced (see an example in Figure 5.1). As
such, the intensity contrast between thin vessels and their image background fluctuates depending on how the partial volume effect interferes with the intensity of the vessel boundary voxels.

In this thesis, we propose three edge detection and segmentation techniques for extracting vessels in magnetic resonance angiographic images. The first proposed method introduces the weighted local variance-based edge detector, which is robust against large intensity contrast changes and is capable of returning accurate detection responses on low contrast edges. The use of the weighted local variance-based edge detector for vascular segmentation leads to more accurate segmentation results for small and low contrast vessels. On the other hand, our second approach radically improves the computation efficiency of a well founded vessel detection algorithm. As compared to the conventional computation method of that algorithm, the proposed one is over a thousand times faster without segmentation performance deterioration. The third method is a curvilinear structure descriptor which is robust against the disturbance induced by closely located objects. This descriptor also inspires the formulation of a set of orthogonal image filters for accurate and efficient vascular structure detection. Preliminary experimental results show that the aforementioned methods are very suitable for vascular segmentation in magnetic resonance angiographic images.

Whereas our experiments are conducted mainly using magnetic resonance angiographic images, the proposed methods can be applied on various image modalities. As an example, the fourth clinical data in the experiment in Section 4.3.4 (see Figure 4.9d) is a computed tomographic angiographic image. This experiment reveals that the proposed method delivers a better segmentation result in cardiac CTA images as compared to the other state-of-the-art methods (compare Figure 4.11d against Figures 4.10f and h). In this thesis, all the proposed methods employ no image modality specific information - the weighted local variance based edge detection exploits the information of intensity changes and intensity homogeneity; the efficient flux implementation makes use of the Fourier technique to perform fast segmentation of tubular structures; the optimally oriented flux computes flux on projected image gradient to perform curvilinear structure detection. These methods are also not confined to handling three-dimensional images. Whenever the interested vessels are curvilinear structures in the images and do not overlap with the undesired curvilinear objects, the proposed methods work well
regardless of the image modalities.

If the desired structures are connected with the undesired ones or the target objects deviate from the curvilinear shape, introducing local image features, capturing image modality specific information or employing prior information regarding the desired structures would help correctly segment those structures. These also promote the detection accuracies and robustness of the algorithms. For instance, Lesage et al. have demonstrated the computation of circular flux on a local plane perpendicular to the vessel direction for vessel tracking in CT images [92]; supplying additional blood flow coherence information [94] is helpful for the detection of low contrast vascular structures in PC-MRA images; when segmenting only the coronary arteries in cardiac CTA images, the undesired but connected aorta can be excluded by constraining the algorithms from detecting structures of excessive radii.

On the other hand, the use of non-invasive magnetic resonance angiography is beneficial to perform early diagnosis of life-threatening vascular diseases, such as aneurysms. Aneurysms are abnormal vascular structures caused by the abnormal dilation of vessel walls. Early aneurysm detection is crucial and extremely challenging, as aneurysms gives a few or no symptoms before becoming significantly large and lethal. They are also barely visible in non-invasive angiographic images and proceed in a large shape variability [23]. The most problematic situation of aneurysm detection in non-invasive angiographic images is in the middle part of the aneurysm, where the blood flows randomly. As such, the aneurysm intensity can be as low as the image background. Both intensity based and geometric based segmentation approaches cannot fully discover the aneurysms. Recent approaches have relied on capturing information in addition to the image intensity. Such information includes blood flow orientation [94], prior knowledge associated with a specific kind of abnormality [95, 101], or training data [96–98].

One of our important future goals for developing new segmentation techniques is to handle complicated vessel geometry and also provide more accurate segmentation results of vascular structures, including both normal vessels and abnormal vascular structures. It is observed that the existing approaches to segmenting abnormal vasculatures is designated to a specific kind of abnormality, such as brain aneurysm segmentation [97,98] or abdominal aneurysm segmentation [100]. Considering our proposed approaches which have taken care of vessel intensity variation, computation
time required and the complicated geometry of normal vascular structures, developing a general technique to segment abnormal vascular structures is now possible. This is a challenging but rewarding task. I believe that devising an accurate technique for the detection of abnormal vascular structures in angiographic images can contribute in improving the efficacy and also reducing the cost of diagnosis, surgery and treatment of vascular diseases.
APPENDIX A

RELATIONSHIP BETWEEN CONFIDENCE VALUE AND THE SMOOTHED IMAGE GRADIENT

The term of intensity average difference in Equation 2.5 can be written as,

\[ \mu_1 - \mu_2 = \int (G_{1,\hat{n}}(\bar{y}) - G_{2,\hat{n}}(\bar{y})) I(\bar{x} + \bar{y}) d\bar{y}, \]

\[ = \int \left( \frac{G'_{1,\hat{n}}(\bar{y})}{\int G'_{1,\hat{n}}(\bar{u}) d\bar{u}} - \frac{G'_{2,\hat{n}}(\bar{y})}{\int G'_{2,\hat{n}}(\bar{u}) d\bar{u}} \right) I(\bar{x} + \bar{y}) d\bar{y}, \]

since the directional derivative of a Gaussian function is antisymmetric, we define \( Z \), i.e. \( Z = -\int G'_{1,\hat{n}}(\bar{u}) d\bar{u} = \int G'_{2,\hat{n}}(\bar{u}) d\bar{u} \) (see Equations 2.1-2.3),

\[ = -\frac{1}{Z} \int (G'_{1,\hat{n}}(\bar{y}) + G'_{2,\hat{n}}(\bar{y})) I(\bar{x} + \bar{y}) d\bar{y}, \]

\[ = -\frac{1}{Z} \int G_{\hat{n}}(\bar{y}) I(\bar{x} + \bar{y}) d\bar{y}, \]

\[ = \frac{1}{Z} G_{\hat{n}}(\bar{x}) * I(\bar{x}), \tag{A.1} \]

where \( * \) represents the convolution operator. Therefore, the intensity average difference of the numerator in Equation 2.5 is equal to the results of convolving the directional derivative of a Gaussian function \( G_{\hat{n}}(\bar{x}) \) with a multiplicative constant \( \frac{1}{Z} \).

\[ R_{\hat{n}}(\bar{x}) = \frac{\mu_{1,\hat{n}}(\bar{x}) - \mu_{2,\hat{n}}(\bar{x})}{\sqrt{\min(WLV_{1,\hat{n}}(\bar{x}), WLV_{2,\hat{n}}(\bar{x}))} + \epsilon}, \]

\[ = \left( \frac{1}{Z} \right) \frac{G_{\hat{n}}(\bar{x}) * I(\bar{x})}{\sqrt{\min(WLV_{1,\hat{n}}(\bar{x}), WLV_{2,\hat{n}}(\bar{x}))} + \epsilon}, \]

128
thus, the confidence value, as defined in Equation 2.5, can be written in terms of the smoothed image gradient \( \nabla (G(\vec{x}) \ast I(\vec{x})) \) if \( \sigma = \sigma_\perp \),

\[
\begin{align*}
&= \left( \frac{1}{Z} \right) \left( \frac{\partial}{\partial \hat{n}} (G(\vec{x}) \ast I(\vec{x}))}{\sqrt{\min(WLV_{1,\hat{n}}(\vec{x}), WLV_{2,\hat{n}}(\vec{x})) + \epsilon}} \right), \\
&= \left( \frac{1}{Z} \right) \left( \frac{\hat{n} \cdot \nabla (G(\vec{x}) \ast I(\vec{x}))}{\sqrt{\min(WLV_{1,\hat{n}}(\vec{x}), WLV_{2,\hat{n}}(\vec{x})) + \epsilon}} \right).
\end{align*}
\] (A.2)
APPENDIX B

INCORPORATING WL V-EDGE IN THE HESSIAN MATRIX

It is straightforward to utilize the Hessian matrix to complement with WL V-EDGE. It can be achieved by rewriting the Hessian matrix $H$ as follows,

$$
H = \begin{pmatrix}
(G \ast I)_{xx} & (G \ast I)_{xy} & (G \ast I)_{xz} \\
(G \ast I)_{yx} & (G \ast I)_{yy} & (G \ast I)_{yz} \\
(G \ast I)_{zx} & (G \ast I)_{zy} & (G \ast I)_{zz}
\end{pmatrix}
$$

and then replacing the first derivative operation of the Hessian matrix with the edge detection results of WL V-EDGE in Equation 2.18, i.e.

$$
H_{WLV-EDGE} = \begin{pmatrix}
(\vec{w} \cdot (1, 0, 0)^T)_x & (\vec{w} \cdot (1, 0, 0)^T)_y & (\vec{w} \cdot (1, 0, 0)^T)_z \\
(\vec{w} \cdot (0, 1, 0)^T)_x & (\vec{w} \cdot (0, 1, 0)^T)_y & (\vec{w} \cdot (0, 1, 0)^T)_z \\
(\vec{w} \cdot (0, 0, 1)^T)_x & (\vec{w} \cdot (0, 0, 1)^T)_y & (\vec{w} \cdot (0, 0, 1)^T)_z
\end{pmatrix}, \quad (B.1)
$$

As such, the robustness of WL V-EDGE against changes of intensity contrast can be inherited in the Hessian matrix.
REFERENCES


analysis of white matter lesions in mr images: Method and validation,” *IEEE

[80] M. Sofka and C.V. Stewart, “Retinal vessel centerline extraction using multiscale
matched filters, confidence and edge measures,” *IEEE Transactions on Medical

[81] S.R. Aylward and E. Bullitt, “Initialization, noise, singularities, and scale in
height ridge traversal for tubular object centerline extraction,” *IEEE Transactions

scale space representation of vessel structures,” *Medical Image Analysis*, vol. 10,

anisotropy for 2D vessels segmentation,” in *IEEE Conference on Computer Vision

Segmentation,” in *International Conference on Information Processing in Medical


[86] R. Toledo, X. Orriols, P. Radeva, X. Binefa, J. Vitria, C.C. Morales, and J.J. Vil-
lanueva, “Eigensnakes for vessel segmentation in angiography,” in *International


[88] M.W.K. Law and A.C.S. Chung, “Three dimensional curvilinear structure detec-
tion using optimally oriented flux,” in *European Conference on Computer Vision*,


