SEGMENTATION, REGISTRATION, AND SELECTIVE WATERMARKING OF RETINAL IMAGES

A Dissertation

by

DI WU

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2005

Major Subject: Computer Science

SEGMENTATION, REGISTRATION, AND SELECTIVE WATERMARKING OF RETINAL IMAGES

A Dissertation

by

DI WU

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Approved as to style and content by:

Jyh-Charn (Steve) Liu
(Chair of Committee)

Donald Friesen
(Member)

Ricardo Gutierrez-Osuna
(Member)

Andrew Chan
(Member)

Valerie Taylor
(Head of Department)

May 2005

Major Subject: Computer Science

ABSTRACT

Segmentation, Registration, and Selective Watermarking of Retinal Images. (May 2005)

Di Wu, B.S., Nankai University (China);
M.S., Nankai University (China)
Chair of Advisory Committee: Dr. Jyh-Charn (Steve) Liu

In this dissertation, I investigated some fundamental issues related to medical image segmentation, registration, and watermarking. I used color retinal fundus images to perform my study because of the rich representation of different objects (blood vessels, microaneurysms, hemorrhages, exudates, etc.) that are pathologically important and have close resemblance in shapes and colors. To attack this complex subject, I developed a divide-and-conquer strategy to address related issues step-by-step and to optimize the parameters of different algorithm steps.

Most, if not all, objects in our discussion are related. The algorithms for detection, registration, and protection of different objects need to consider how to differentiate the foreground from the background and be able to correctly characterize the features of the image objects and their geometric properties.

To address these problems, I characterized the shapes of blood vessels in retinal images and proposed the algorithms to extract the features of blood vessels. A tracing algorithm was developed for the detection of blood vessels along the vascular network. Due to the noise interference and various image qualities, the robust segmentation techniques were used for the accurate characterization of the objects' shapes and verification. Based on the segmentation results, a registration algorithm was developed, which uses the bifurcation and cross-over points of blood vessels to establish the correspondence between the images and derive the transformation that aligns the images. A Region-of-Interest (ROI) based watermarking scheme was proposed for image authenticity. It uses linear segments extracted from the image as reference locations for embedding and detecting watermark. Global and locally-randomized synchronization schemes were proposed for bit-sequence synchronization of a watermark. The scheme is robust against common image processing and geometric distortions (rotation and scaling), and it can detect alternations such as moving or removing of the image content.

To My Wife and Parents

ACKNOWLEDGMENTS

I would like to thank Dr. Jyh-Charn Liu for his guidance, support and patience throughout my graduate study. He has been the constant source of inspiration and help. His suggestions and deep insight on academic matters contributed greatly to the completion of this dissertation as well as to my academic achievements.

I would like also to express my appreciation to the members of my committee: Dr. Donald Friesen, Dr. Ricardo Gutierrez-Osuna, and Dr. Andrew Chan. They gave me very helpful suggestions on improving the quality of this dissertation. I am grateful to them for their guidance and support. They spent time discussing my research with me although they were very busy.

I also appreciate my dear wife, Ye Huang, for her dedication and patience. Without her support, I would not have finished my Ph.D.

TABLE OF CONTENTS

CHAPTER Page

I INTRODUCTION 1

A. Retinal Image Segmentation 2
B. Image Registration 3
C. Image Watermarking 4

II ADAPTIVE DETECTION OF BLOOD VESSELS 6

A. Introduction 6
B. Enhancement and Feature Extraction 9
C. Tracing 18
  1. Forward Detection 22
  2. Bifurcation Point Detection 26
  3. Backward Verification 28
D. Experiments and Performance Evaluation 31

III IMAGE REGISTRATION 40

A. Introduction 40
B. Registration 45
  1. Extraction of Registration Points 45
  2. Matching of Registration Points 49
  3. Estimation of Transformation 50
C. Performance Evaluation 53
D. Application to Detection of Laser Scars 57

IV ROI-BASED WATERMARKING 63

A. Introduction 63
B. ROI Detection and Synchronization 67
  1. Blood Vessel Segmentation 67
  2. Delaunay Tessellation Ordering 72
  3. Randomized Flow Ordering 76
  4. Performance Comparison of Flow Ordering Algorithms 79
C. Watermark Embedding and Detection 80
D. Experimental Results 85

CHAPTER Page

V CONCLUSION 92

REFERENCES 94

VITA 108

LIST OF TABLES

TABLE Page

I Prediction error (in degree) of directions with curvature-threshold = 0.1. 26

II Performance of our method for 10 normal images. 34

III Performance of our method for 10 abnormal images. 34

IV Registration scores for test images: "R" denotes the registration is rejected, and "A" denotes the registration is accepted. 56

V Segmentation scores under different distortions. 71

VI Average displacement $(\Delta x, \Delta y)$ of centroid over all distortions. . . . 73

VII Authentication results of the testing images against distortions and manipulations. "Yes" means that an image is decided to be authentic, and "No" means that an image is decided to be inauthentic. 91

LIST OF FIGURES

FIGURE Page

1 (a) A normal retinal image, and (b) an abnormal retinal image. . . . 10

2 The performance of contrast enhancement for (a) the normal image and (b) the abnormal image in Figure 1. 12

3 Experimental results for a normal image: (a) the inverted green channel (I_1), (b) the output (I_2) after illumination equalization, (c) the contrast-enhanced map (I_3), and (d) the GS-filtering map (I_4). 13

4 Comparison of different feature maps for small vessels and their detection outcomes: (a) the green channel of a cropped image, (b) the 2-D Gaussian filter response map, (c) the GS-filtering map, and (d-f) their respective tracing outcome. 17

5 Flowchart of the tracing algorithm. 20

6 Weighted-sum estimation of θ_{n+1} by (a) extrapolation and by (b) local greedy search, and examples when (c) $c_{p1} > c_{p2}$ and (d) $c_{p1} < c_{p2}$ in the estimation on θ_{n+1}. 23

7 Bifurcation point detection: (a) a cropped image, (b) the *OFR* values along different orientations around a bifurcation point (blue dot in (a)), and (c) the *OFR* values around a non-bifurcation point (green dot in (a)). 29

8 False detection cases (a) and (c), and their correction (b) and (d) in the backward verification. The red dots represent detected bifurcation points. 30

9 First row: detection results for the images in Figure 1. Second and third rows: the hand-1 and hand-2 maps of the same images. . . 33

10 The average detection performance over 20 images: our method vs. the filter response analysis method and the multi-threshold probing method. 35

FIGURE Page

11 (a) A reconstructed vessel from segment features, and (b) the width function and curvature of the vessel. 38

12 Two images photographed (a) before and (b) after laser surgery. Some laser scars are located at the center of the second image. . . . 46

13 (a) An coarse blood vessel map and (b) its thinned centerlines as well as the extracted feature points that are highlighted in blue. . . 48

14 Registration result of the two images shown by: (a) the superposition, (b) the checkerboard. 54

15 Registration result of the two images using their fine vascular maps: (a) the superposition, (b) the checkerboard. 58

16 Application of Registration result to detection of laser scars: (a) a cropped retinal image containing laser scars; (b) the difference map between pre- and post-laser-surgery images, (c) the detected laser scars in yellow color. 59

17 Another example of detection of laser scars by registration: (a) a cropped retinal image containing laser scars; (b) the difference map between pre- and post-laser-surgery images, (c) the detected laser scars in yellow color. 62

18 (a)-(b) Two sample retinal images (077 and 163); (c)-(d) the centerlines of the detected vascular networks of the two sample images. . 68

19 The segmentation inconsistency due to the distortion: the highlighted area by rectangles and circles show the difference in the segmentation maps of an original image and the respective JPEG-compressed image with quality factor 80. 70

20 DT-ordering results for the two training images: the triangles are generated by the Delaunay triangulation. Each vertex is the middle point of a selected flow, except vertex 0 that is the centroid of all pixels on the centerlines. The number around each vertex is the index of the flow after sorting. For good visual effects, 30 longest flows are selected in this example. 75

FIGURE Page

21 Geometric structure of the vascular network for (a-b) the original training images, (c-d) the Gaussian-filtered images, and (e-f) the rotated images. The red dot near the center of each image denotes the centroid of the centerline pixels. The covariance matrix is illustrated by the red ellipse, whose orientation and size are determined by the eigenvectors and eigenvalues of the matrix. 77

22 Performance comparison over the four local ordering algorithms and two global ordering algorithms for the two training images. . . . 81

23 Performance comparison of the watermark recovery schemes based on block-mean and Wiener filter. 84

24 Watermark embedding: (a) the red squares present the locations where the watermark bits are embedded. (b) The watermarked image with $\alpha = 9$. (c) The watermarked image with $\alpha = 6$. 86

25 (a) Histogram of the similarity value of the in-authentication class and (b) histogram of the similarity value of the authentication class, where the two curves are the approximated normal distributions. 88

26 Two manipulation examples for a testing image: (a) two blocks in the image are exchanged; (b) two blocks in the image are replaced by the uniform regions with mean intensity of the entire image. . . . 89

CHAPTER I

INTRODUCTION

Digital Image Processing is an important field in the engineering discipline. It is used mainly for two purposes: a) improving the visual appearance of images to a human viewer, and b) preparing images for measurement of the features and structures present.

In this work, I investigate several challenging problems in the area of medical image analysis. The retinal images are chosen in our study for two main reasons. First, the application of computational algorithms to retinal images is a rapidly developing field, which has led to great advancements in retinal structure analysis. Automated methods can help detection and control of retinal diseases such as diabetic retinopathy. Screening of diabetic retinopathy by automated algorithms could reduce the occurrence of blindness by 50% [1][2] and lessen the expenses associated with examinations. The automated algorithms are needed to be able to screen patients for diabetic retinopathy and other conditions in a manner that is repeatable, reliable, and cost effective.

The color retinal images affected by diseases contain many patterns of different shapes and sizes. For example, microaneurysms are small red round spots. Exudates are yellow, waxy spots with distinct margins. Hemorrhages are dark patches with a brush-stroke or flame-like shape. In addition, retinal images exhibit diverse qualities due to the different photographing conditions, disease deterioration, and noise and artifact interference. For these reasons, retinal images are the ideal image objects on which one can test basic algorithms for medical image processing.

The journal model is *IEEE Transactions on Image Processing.*

A. Retinal Image Segmentation

A retinal image contains many distinct types of objects. In my work, I select blood vessels in the images as the target of segmentation. Retinal blood vessel morphology can be an important indicator for diseases such as diabetes, hypertension, and arteriosclerosis. Measurement and analysis of the features of the blood vessels, such as diameter, branching angle, length, or tortuosity, can aid in diagnosis, treatment, and further clinical study. Thus, a reliable method of vessel detection and quantification would be valuable. Detection of large blood vessels is relatively straightforward in noise-free images, but the detection process becomes much more complicated when lesions, central reflex, and other pathological changes affect the retinal images. Detection of small vessels (diameters that are 3 pixels or less) is difficult because of their low contrast levels. However, morphological changes of small vessels often signify the onset of diseases, and analysis of small vessels must be carefully considered.

I proposed an adaptive detection scheme for large and small blood vessels in color retinal images. Our scheme consists of three functions: adaptive contrast enhancement, feature extraction of blood vessels, and tracing. Enhancement of blood vessels is extended from an adaptive histogram equalization technique. Feature extraction of small blood vessels is performed by using the standard deviation of Gabor filter responses along different orientations. Tracing of the vascular network consists of three major functions: forward detection, backward verification, and bifurcation detection. Combining extrapolation and local greedy search reduces the prediction errors of vessel directions by 15-20%. Only two sample images and their hand-traced maps are needed for parameter training and calibration. The experiment results show that the performance is better than that of two most recently proposed methods which have been tested on the same image data. The majority of small vessels could be captured by our method.

B. Image Registration

Integrated analysis of multiple images for a subject provides critical information on the progression of pathological conditions. The first step in the integrated analysis is to align key image objects, i.e., *registration*, so that a *fusion* step can be taken for display and analysis of the data set.

Image registration can be categorized by considering the type of sensor used, the position of the sensors, the time difference between the images, and differences between the image subjects. Using this classification, registration can be mainly divided into three categories [3]. 1) Multimodal registration, where images of the same subject have been taken by different types of sensors, for example, computed tomography (CT), positron emission tomography (PET), and magnetic resonance (MR) images of the brain. 2) Stereo registration, where the same subject is imaged with two similar sensors at the same moment, but from different positions. 3) Temporal registration, where the same subject is photographed from the same viewpoint, but at different times, for example, the analysis of myocardial function under conditions of rest and stress.

In this dissertation, I focused on the registration of retinal images. Temporal registration can help to track disease evolution while stereo registration can combine or enhance pathological information to facilitate the diagnosis. Despite the importance of this topic and prior research effort, retinal image registration has remained a difficult problem. We address some of the opening problems by developing a new image registration method. The coarse blood vessel map is extracted from each retinal image by enhancement and thresholding. The feature points are then identified from the centerline pixels. The feature points in the two images are corresponded by their local structures. The images are mapped onto a common frame by the affine transformation derived from the corresponded feature points. The desirable registration performance is obtained on the retinal images with good and poor qualities.

C. Image Watermarking

The third issue I addressed is related to authenticity protection of medical images using digital watermarking. Digital watermarking has been recently proposed as a solution for prohibiting copyright infringement of multimedia data. By embedding indelible and imperceptible labels and/or signatures into various digital media, extraction of the embedded information without ambiguity can be used to identify the copyright owner or legitimate recipients to prevent these data from being illegally distributed or misused.

Most commercial watermarking methods are developed to protect the copyright of broadcast or multimedia data. They embed watermarks in the entire image without taking the image content into account. For medical images, regions that contain disease symptoms are of major concern to the physicians and patients. For example, in retinal images the image viewer usually cares more about the vasculature and lesions than the background. The portions that attract more attention from an image viewer are called the regions of interest (ROI). It is desirable to embed robust watermarks in ROI to give them better protection.

I proposed a ROI based watermarking scheme that uses linear segments extracted from the image as reference locations for embedding and detecting watermark. Using the vascular network of the retinal images as an in-depth case study, the (nearly) linear patterns of the vascular network dispersed across the retinal image are extracted for synchronization, embedding, and detection of invisible watermarks. Global and locally-randomized synchronization schemes are proposed for bit-sequence synchronization of a watermark. On the synchronized locations, one can decide the shapes and strengths of the watermark signals for embedding, and multiple watermarks can be embedded easily. The scheme is robust against common image processing and geometric distortions (rotation and scaling), and it can detect alternations, such as moving or removing of the image content. Experimental results on a set of ten public images show that with a high probability, images can be authenticated under different attacks.

Details on the three major research themes and their experimental results are presented in Chapter II, III and IV, respectively.

CHAPTER II

ADAPTIVE DETECTION OF BLOOD VESSELS

A. Introduction

Automatic detection of blood vessels in retinal images can help physicians for the purposes of diagnosing ocular diseases, patient screening, and clinical study, etc. Blood vessel appearance provides information on pathological changes caused by some diseases including diabetes, hypertension, and arteriosclerosis. The most effective treatment for many eye-related diseases is the early detection through regular screenings. Thus, the development of a reliable, repeatable process to accurately detect and trace the structural changes of the blood vessels in retinal images is needed.

The previous work on extracting blood vessels in retinal images can be divided into two major approaches: pixel classification and vessel tracing. The pixel classification approach [4]-[12][13][14] determine whether or not a pixel belongs to a blood vessel, based on analysis outcomes of image features at proximity of the pixel under consideration. This approach requires further processing to generate vessel geometrical properties.

In the threshold probing scheme [7], pixels are grouped using a set of thresholds on their 2-D matched filter responses, followed by detection of vessels using local and global features. In [13], the hypothesized vessels are segmented by multiple thresholds. They are then verified by some criteria designed based on the curve-linear structure of blood vessels. The postprocessing is performed to locate vessel boundaries by non-maxima suppression of Sobel detector response.

A primitive-based method (PBM) is proposed in [15], where for every pixel over twenty features are computed based on the detected vessel centerlines and straight lines of blood vessels. A *KNN* classifier is constructed using selected features by the sequential forward selection method with the training data to detect vessel pixels.

In [9], retinal images are segmented by morphological filters using linear structuring elements. The segmentation results are validated by the cross-curvature property, and the vessels are detected as the patterns whose curvatures are linearly coherent. The parameters in the morphological filters are optimized for detection of large vessels. The segmentation performance is sensitive to the Gaussian noise and uniform noise.

A statistical method for vessel segmentation is propsed in [16]. The a-priori distribution of pixel labels is modelled by a Markov random field. The posterior probability of pixel labels is estimated by simulated annealing and iterated conditional modes.

Vessel tracing schemes [17][18][19][20] work by starting from some initial vessel points and tracing the vasculature by detecting vessel boundaries and directions. The vessel direction and boundaries can be determined either by edge detectors such as Sobel edge detecors, gradient operators, or by the matched filters that approximate the intensity profile of vessel cross-sections.

In [21], the initial points for tracing are chosen from the local minima of Gaussian kernel responses on a set of lines across the image subjection to pre-specified criteria. The vessel boundaries and directions are estimated using the directional matched low-pass differentiator templates.

In [22], the tracing algorithm matches the vessel profiles and determines the vessel locations using Gaussian filter. The locations of next vessel segment is estimated by Kalman filter. The twin Gaussian model developed in [23] characterizes the blood vessels with central light reflex. In [24], an amplitude modified second-order Gaussian filter is developed to find the highest response for determining the vessel width. These filters are effective for direct matching of large vessels, but their effectiveness diminishes with small vessels.

In [25], an unsupervised fuzzy algorithm is proposed for vessel tracing to resolve the ambiguity in determining the exact position of a vessel segment. The fuzzy C-means clustering is used to classify vessel and non-vessel regions, without any assumptions for the shape of the vessels and without utilizing edge information that is usually corrupted by noise.

Detection of large blood vessels is relatively straightforward in noise-free images, but the detection process becomes much more complicated when lesions, central reflex and other pathological changes affect the retinal images [7]. Detection of small vessels (diameters that are 3 pixels or less) is difficult because of their low contrast levels [22][23][24][26]. However, knowing that subtle changes in small blood vessels may have significant implications on prognostic of serious diseases, like diabetes or hypertension, analysis of small blood vessels must be carefully considered.

In this chapter, we propose a novel approach for detection of large and small blood vessels in color retinal fundus images. Our scheme consists of three major features: adaptive contrast enhancement, small vessel feature extraction, and tracing of vessel shapes. In addition to the classical illumination and adaptive equalization schemes that are known to increase the contrast of blood vessels, we developed a Gabor-based filtering technique that is highly effective for enhancement of small blood vessels while removing their surrounding noise. The tracing algorithm uses different types of feature information based on the most likely vessel shapes for decision making. Marked performance gain is achieved for both large and small blood vessel detection, with little parameter adjustment needed for the detection process.

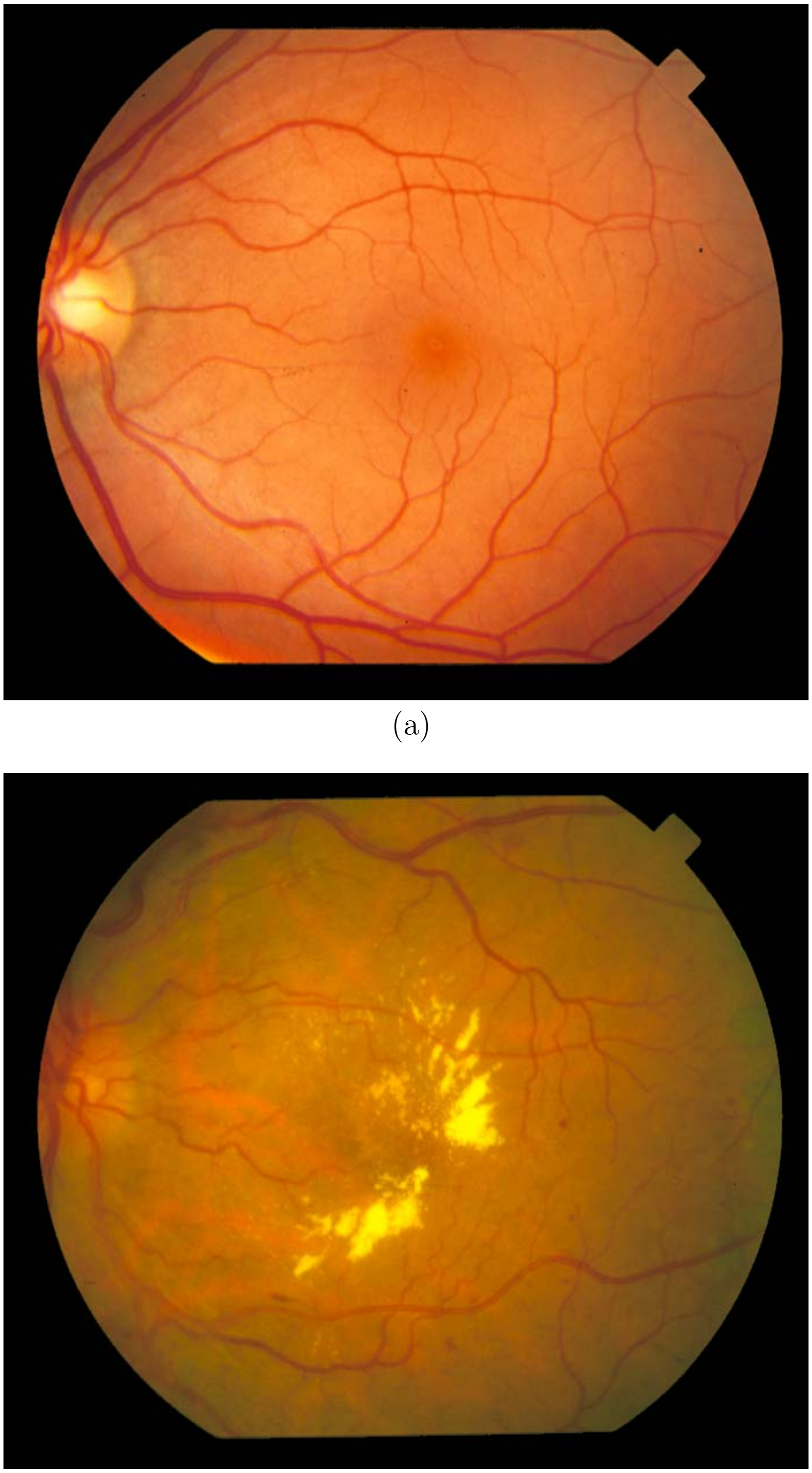
The rest of the chapeter is organized as follows. Section B discusses the contrast enhancement and feature extraction schemes for retinal images. Section C discusses the tracing algorithm. Experiments and analysis of their results are given in section D.

B. Enhancement and Feature Extraction

The color fundus images under consideration have RGB channels. The dynamic range of the pixel in each channel is 8-bit, and the image size is 700×605 . A broad range of retinal images that represent normal and abnormal symptoms are tested in our study based on the image collection in [27]. Two sample images given in Figure 1 are used in the rest of this chapter to illustrate effects of different algorithm steps on normal and abnormal images, respectively. A normal retinal image consists of mostly blood vessels and the background, but an abnormal retinal image has multiple artifacts of distinct shapes and colors caused by different diseases.

As a preprocessing step, the green channel of a color retinal image is extracted and inverted in its intensity, which is denoted as I_1. The illumination equalization scheme proposed in [28] is applied to I_1 to neutralize illumination fluctuation. That is, the intensity of each pixel in I_1 is adjusted by $I_2(p) = I_1(p) + m - A(p)$, where $m = 128$ for an 8-bit image, $A(p)$ is the local average intensity within a 21×21 window centered at pixel p. The output is then normalized into the range [0,255]. Denote the output of this step as I_2. Using I_2 as its input, the objective of feature extraction and contrast enhancement processes is to enhance the contrast of blood vessels as much as possible. Contrast enhancement is particularly important to detection of small blood vessels, because the intensity of small vessels declines significantly with reduction of the vessel width.

The first enhancement step is aimed at general enhancement of all vessels, based on the adaptive histogram equalization (AHE) technique [29]. While AHE is found



(a)

(b)

Figure 1: (a) A normal retinal image, and (b) an abnormal retinal image.

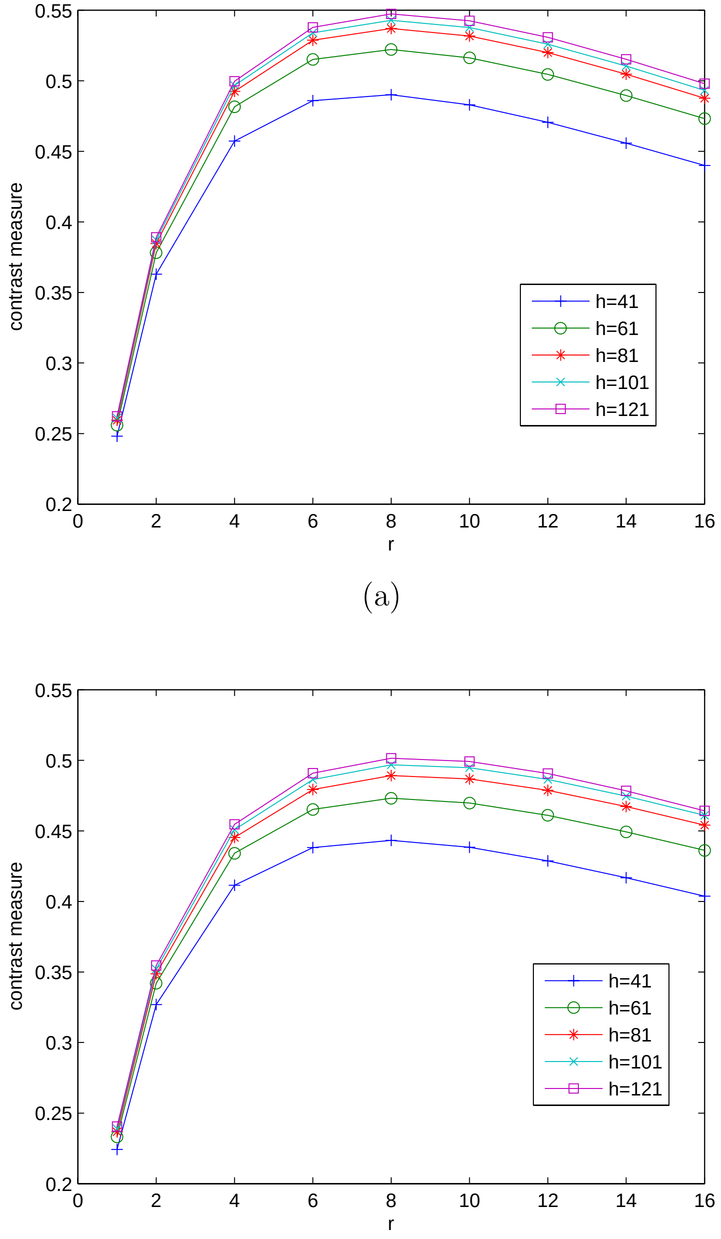
to be much more effective than the classical histogram equalization approach, the empirical study shows that the contrast gain can be further increased by introduction of an enhancement exponent to AHE:

$$I_3(p) = (\sum_{p' \in R(p)} s(I_2(p) - I_2(p'))/h^2)^r \cdot M \tag{2.1}$$

where $M = 255$, and $R(p)$ denotes the neighborhood of pixel p, specified by a square window with length h. (2.1) reduces to AHE when $r = 1$. $s(d) = 1$ if $d > 0$, and $s(d) = 0$ otherwise. This step produces an output I_3 from its input I_2.

Increase of r also increases the contrast between vessel pixels and the background. However, when r becomes too large, it will enhance the noise in the background, and cause the shape distortion of the blood vessels in the image. To balance the tradeoff, we found that the contrast measure $(\mu_v - \mu_b)/(\mu_v + \mu_b)$ proposed in [30] is an excellent performance indicator for optimization of r and h values, where μ_v and μ_b represent the average intensity value of vessel pixels and that of background pixels on I_2. Using the hand-traced maps of blood vessel in [27] as the reference, we carried out extensive experiments to determine the optimal enhancement parameter values.

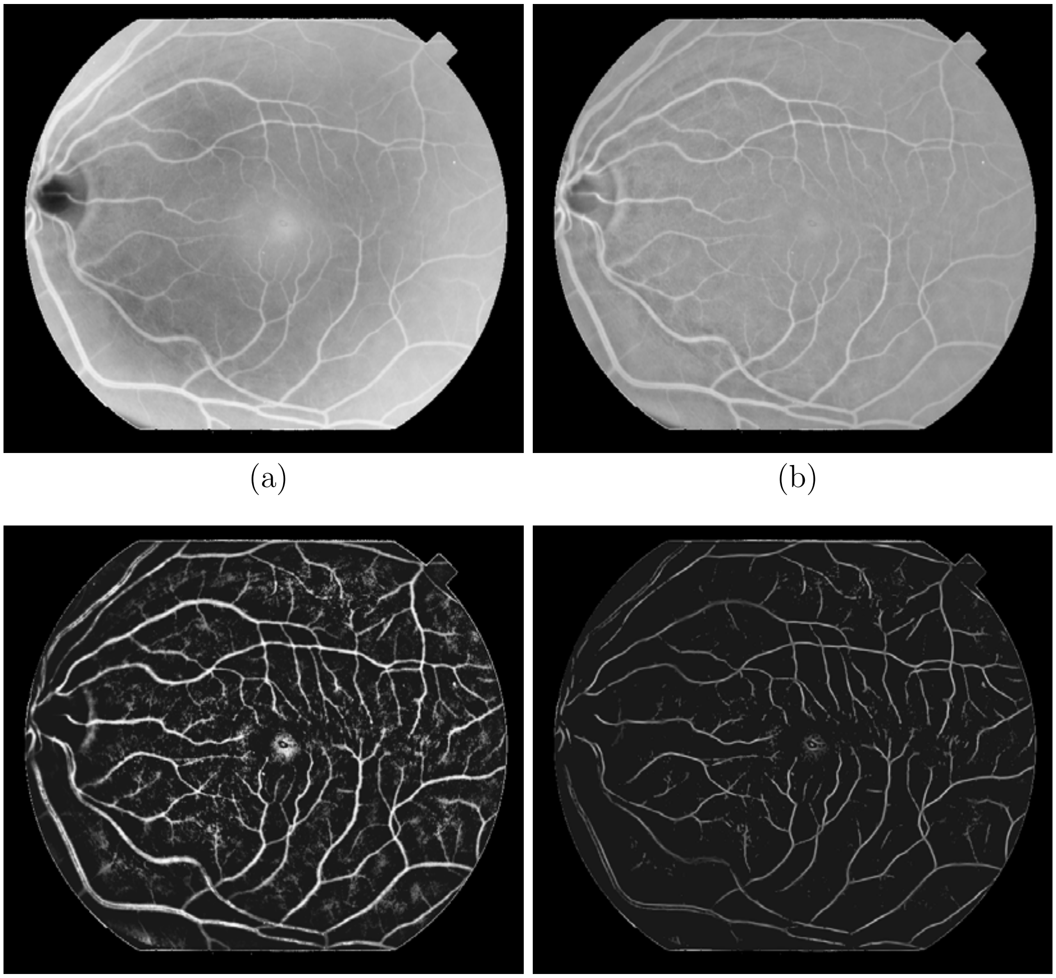
The value of r is tested from 1 to 16 at increment of 1, and the value of h from 41 to 121 at increment of 20. The performance curves for images in Figure 1 are shown in Figure 2, where each curve represents the performance for a given value of h over the range of r. Based on the tradeoff between the enhancement performance and processing cost, we chose $h = 81$ and $r = 8$ for the rest of discussion in this chapter. Figure 3(a-c) illustrates the effects of the first two steps. Illumination equalization and the modified AHE are intensity based algorithms. Together they increase significantly the contrast of blood vessels but also bring out the background noise. Subsequent testing results suggest that I_3 is adequate for large blood vessel detection, but it is



(a)

(b)

Figure 2: The performance of contrast enhancement for (a) the normal image and (b) the abnormal image in Figure 1.



(a) (b)

(c) (d)

Figure 3: Experimental results for a normal image: (a) the inverted green channel (I_1), (b) the output (I_2) after illumination equalization, (c) the contrast-enhanced map (I_3), and (d) the GS-filtering map (I_4).

not sufficient for detection of small blood vessels. With decrease of the blood vessel sizes, their contrast against the background diminishes. As a result, the effectiveness of Gaussian or double-Gaussian like filters is significantly compromised because of low contrast and small sizes of blood vessels.

Although large and small blood vessels are physically connected, the significant differences in their sizes and contrast levels suggest that they should be treated as two different type objects, and be handled by using different enhancement and detection mechanisms for better results. For further enhancement of small vessels after the modified *AHE* step, we take a shape-based enhancement approach. Knowing that numerous background noises would present at proximity of small vessels, the enhancement filter for small vessels should have directional selectivity. That is, when the filter is positioned at the center of a small vessel, it should have maximum and minimum responses, respectively, along the vessel direction and its perpendicular direction. To eliminate spotty noise, only the pixels along the blood vessel direction should be enhanced, and others suppressed. By putting these factors together, we propose using the standard deviation (std) of the Gabor filter [31]-[35] responses along different orientations, denoted as GS, for enhancement of small blood vessels,

$$I_4(x,y) = std_\phi(q_\phi(x,y)) \tag{2.2}$$

where $q_\phi = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I_3(x-u, y-v) g_\phi(u,v)dudv$ is the Gabor filter response along a particular orientation ϕ, and the real part of the Gabor filter is used for convolution with the modulation axis parallel to the envelope axis, as used in most cases [32], expressed by

$$g_\phi(u,v) = \exp(-\pi(\frac{u'^2}{\sigma_u^2} + \frac{v'^2}{\sigma_v^2}))\cos(2\pi f u') \tag{2.3}$$

where $(u', v') = (u\cos\phi + v\sin\phi, -u\sin\phi + v\cos\phi)$. The filtering output in 2.2 is called the *GS response*.

The Gabor filter has weak responses along all orientations on the smooth (background) surface, so the GS response is also weak. On the other hand, when it is positioned on a linear-pattern object, such as the small blood vessels, the Gabor filter produces relatively large differences in its responses when the orientation parameter changes, so that the GS response is much stronger than that of the background.

We adopt the procedure in [35] for parameter setting of the Gabor filter to produce a single peak response on the straight line with width d, and to differentiate two closely spaced lines. That is, the frequency parameter f is set to β/d with $\beta \in [0.5, 1]$. σ_u and σ_v, which determine the spread of Gabor filter, are specified by $\sigma_u = n\lambda d/\alpha\beta\pi$ and $\sigma_v = \kappa\sigma_u$, where n is the number of orientations, $\lambda = \sqrt{2ln2/\pi}$, $\alpha \in [1, 1.5]$, and $\kappa = 0.85$ [35]. As Gabor filter is enveloped by a Gaussian function, the range of convolution in the actual implementation is set to be $r < 1.5\sigma_v$ because the filter coefficients beyond this range become negligible, where $r = \sqrt{(x-u)^2 + (y-v)^2}$. In the rest of our discussion, we chose $n = 12$, $\alpha = 1.5$ and $\beta = 0.5$.

Blood vessels have varying diameters along different branches. When the value of d is large, e.g., 10, most small vessels are either depressed or smeared by neighboring noise. A simple histogram counting of the hand-labeled blood vessel maps shows that diameters of blood vessels range from 1 to 12 pixels, where the highest frequency count falls on the diameter of 3-pixel wide. As a result, we chose $d = 3$ for enhancing and preserving of small vessels, and it is also used as the threshold to differentiate large and small vessels.

The GS filtering is performed as follows. For each pixel in I_3 with intensity larger than the medium value m_d of the pixel intensity range, i.e. $m_d = 128$ for 8-bit gray level images, the GS response is calculated by taking the standard deviation of the Gabor filter response along n quantized orientations ϕ_i, $\phi_i = i\pi/n$, $i = 1, \ldots, n$. For pixels in I_3 whose intensities are less than m_d, they are mostly on the background, and thus their GS responses are simply set to zero. The GS filtering output (denote by I_4) is illustrated in Figure 3(d), where the output has been normalized and quantized to the range of [0,255]. The skeletons of small vessels are clearly extracted with high contrasts against the background, suggesting that I_4 is the ideal feature map for detection of small vessels in the tracing phase.

Given that the value of d was chosen to favor enhancement of small vessels, large vessels have inconsistent responses to GS filtering. Some large blood vessels produce high GS responses on their central lines or boundaries, yet others were depressed, obviously because of mismatched vessel diameters. Regardless of this minor problem, large vessels in I_3 are strong enough for their reliable detection. As a result, we use I_3 as the feature map for detection of large vessels in the tracing phase.

Figure 4 illustrates (a) the green channel of a cropped image, (b) its Gaussian filter response map based on the 2-D Gaussian filter specified in [2], and (c) its GS-filter response map. Clearly, GS filtering brought out more small vessels that otherwise could not be captured by the Gaussian filter. In addition to its plausible visual effects, the numerical contrast measure of the GS response map is 0.80, which is much higher than those of the green channel (0.05) and the Gaussian output (0.20). To have a better understanding on the effectiveness of the three different feature maps, i.e., Figure 4(a-c), we applied our tracing method (to be discussed next) to them using identical parameters and gave the results in Figure 4(d-f). Clearly, the tracing using the GS-filtering map captures more small vessels than the other two.

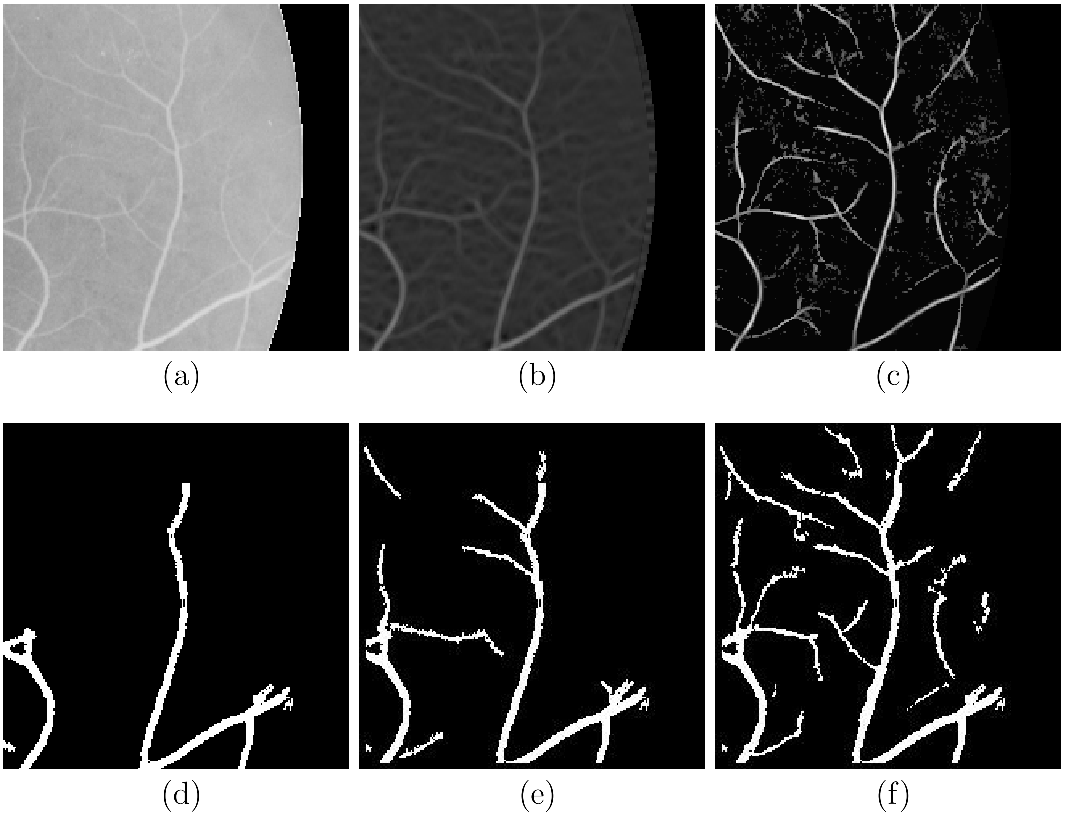


Figure 4: Comparison of different feature maps for small vessels and their detection outcomes: (a) the green channel of a cropped image, (b) the 2-D Gaussian filter response map, (c) the GS-filtering map, and (d-f) their respective tracing outcome.

C. Tracing

The objective of blood vessel tracing is to create the final vascular map based on the geometric properties of blood vessels. Using the feature maps generated by the enhancement and feature extraction sub-systems, tracing starts from some automatically selected initial points, at which initial estimations on widths and directions of vessels will also be made. The initial tracing points are chosen on both I_3 and I_4 by the following procedure:

Step 1) In I_3, pixels on a 9×9 mesh grid similar to the one in [12] are analyzed. A pixel on the grid is considered a candidate for further analysis if it locates at the middle of two consecutive edge points that have similar responses but at opposite orientations, and the distance between the two edge points is an initial estimate of the vessel diameter. Local maxima of gradients calculated by Sobel operator are considered as edge points of blood vessels.

Step 2) In I_4, local maxima points (in a square window) whose intensity values are also greater than m_d are selected as candidates, where the length of the square window is set to be equal to the width of the widest vessel in the sample images. The initial estimate of vessel width is equal to the value of the parameter d used in GS-filtering.

Step 3) The *oriented feature responses* (*OFR*s), which will be introduced shortly, of candidates chosen from the first two steps are computed to verify the candidates. The candidates that have two local maxima on *OFR* along two opposite orientations are identified as initial blood vessel points, whose two orientations are the initial estimate of the vessel directions.

Most candidate initial points selected in the first and second steps are initial points on large and small vessels, respectively. The third step confirms the selections in the first two steps by rejecting candidates that are likely noise or on lesions.

In our work, Sobel operator is chosen to locate the vessel edges. The edges in an image are defined as the pixels where gray level intensities abruptly change, and they can be identified by the gradients or differences of neighboring intensities. In a digitized image, the magnitude M and direction θ of the edge pixel at (x, y) are computed as,

$$M = [I_x(x,y)^2 + I_y(x,y)^2]^{1/2}, \tag{2.4}$$

$$\theta = \arctan \frac{I_y(x,y)}{I_x(x,y)}, \tag{2.5}$$

where $I_x(x, y)$ and $I_y(x, y)$ are computed by Sobel operator as

$$\begin{aligned} I_x(x,y) =& [I(x+1,y+1) + 2I(x+1,y) + I(x+1,y-1)] \\ & - [I(x-1,y+1) + 2I(x-1,y) + I(x-1,y-1)], \\ I_y(x,y) =& [I(x+1,y+1) + 2I(x,y+1) + I(x-1,y+1)] \\ & - [I(x+1,y-1) + 2I(x,y-1) + I(x-1,y-1)], \end{aligned} \tag{2.6}$$

All initial points are stored in a list after they are sorted in the descending order of vessel widths. In each iteration, the initial point with the largest vessel width in the list is removed and is used to start tracing of a new branch. On each branch, the vascular network is constructed segment by segment, where each segment is assumed to have a fixed width and direction, and its length is equal to the width, using three major routines, *forward detection*, *backward verification*, and *bifurcation identification*. The interactions between the three routines are illustrated in Figure 5.

In the forward detection routine, we use the latest detected segments to predict the most likely location of the next segment, perform local greedy search, and then move to the new location based on the two pieces of information. The bifurcation-

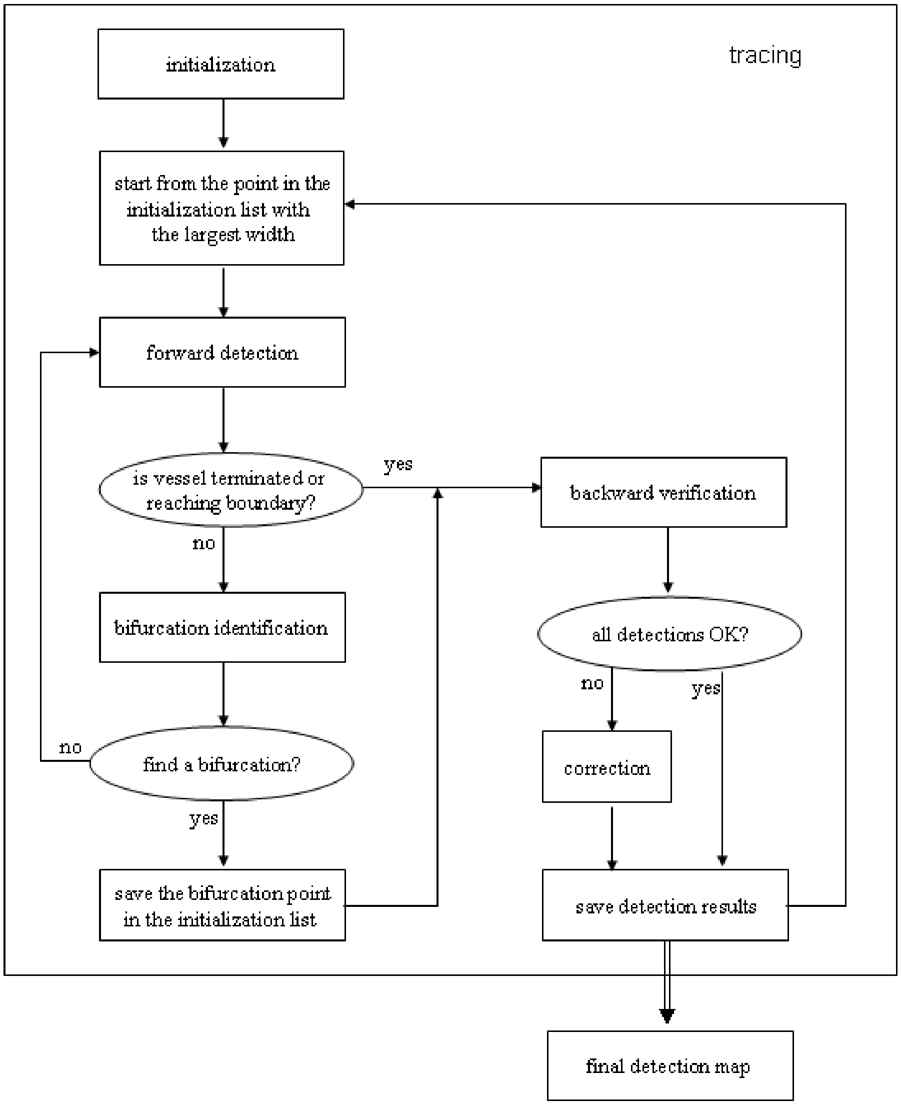

tracing
initialization
start from the point in the initialization list with the largest width
forward detection
is vessel terminated or reaching boundary?
yes
no
backward verification
bifurcation identification
all detections OK?
no
yes
find a bifurcation?
no
yes
correction
save the bifurcation point in the initialization list
save detection results
final detection map

Figure 5: Flowchart of the tracing algorithm.

identification routine is invoked along with the forward detection routine to identify branches. After an identified bifurcation point is saved into the initialization list, the current iteration is terminated, and tracing of the next branch can be started from the initial point that has the largest diameter. The iteration can also be terminated when the current branch is detected to be terminated. The backward-verification routine is invoked at the end of the iteration to verify detection results.

Feature maps I_3 and I_4 are used adaptively for detection of large vessels and small vessels. The contrast between (large and small) blood vessels and their neighboring background is strong in I_3, but there is significant noise at proximity of the small vessels. On the other hand, small vessels have sharp skeletons in I_4 but their diameters can be more accurately measured on I_3. As a result, we use I_3 to detect large vessels, and use I_3 and I_4 jointly to detect small vessels, where I_4 is used to determine the presence of small vessels and their directions, and I_3 is used to measure their diameters. Detection of bifurcation points is based on multiple local maxima of the tracing filter response *OFR*, where each local maximum of the *OFR* represents a branch surrounding a bifurcation point. *OFR* is essentially the Gaussian filter response of the vessel on its feature map, i.e., I_3 for large vessels, and I_4 for small vessels, along a certain orientation. That is,

$$OFR_\theta(x,y) = \begin{cases} \iint_{H(x,y)} I_3(x-u, y-v)\xi_\theta(u,v)dudv & \text{for large vessels} \\ \iint_{H(x,y)} I_4(x-u, y-v)\xi_\theta(u,v)dudv & \text{for small vessels} \end{cases}, \tag{2.7}$$

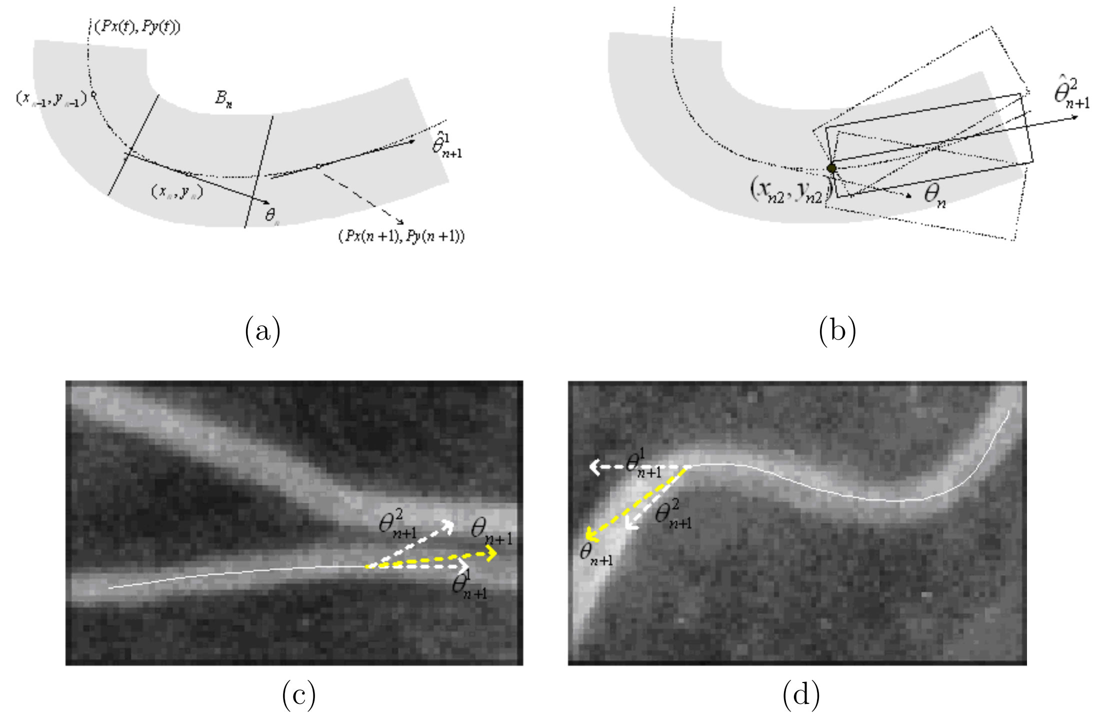
where $\xi_\theta(u,v) = \exp(-\pi v'^2/\sigma_v^2)$ is a two-dimensional Gaussian filter oriented toward θ, and the convolution range is specified by $H(x,y) = \{(u,v) : 0 \leq u' \leq 2w, |v'| \leq 1.5\sigma_v\}$. Referring to Figure 6(b), the filter is not centered at (x,y). Instead, the pixel under consideration is on one edge of the filter. The filter is designed to explore vessel branches around the pixel. The dimension of the filter, as a function the local vessel width w, was determined empirically.

1. Forward Detection

The forward detection routine consists of *forward extrapolation* and *local greedy search.* Let $B_1, \ldots, B_n$ denote the n latest detected consecutive segments, the goal of forward extrapolation is to estimate the location of the next new segment B_{n+1} from $B_1, \ldots, B_n$, where B_i is characterized by its middle point on the centerline, (x_i, y_i), width w_i, and direction θ_i, see Figure 6(a). The *central curve function* (*CCF*) of a vessel represents the geometric central points of the vessel that can be expressed by $(x, y) = (x(t), y(t))$, where t takes values in an interval. The *CCF* will be approximated by its cubic spline construct, called *CCF'* [36].

The cubic splines are the spline functions of degree 3, and they are widely used in practice. A spline function of degree k having knots $t_0, t_1, \ldots, t_n$ is a function S such that: 1) On each interval $[t_{i-1}, t_i)$, S is a polynomial of degree $\leq k$. 2) S has a continuous $(k-1)$st derivative on $[t_0, t_n]$. Hence, S is a continuous piecewise polynomial of degree at most k having continuous derivatives of all orders up to $k-1$. For the construction of a cubic spline function based on a set of given points, please refer to [36].

Let P_x and P_y denote the two cubic splines that approximate $x(t)$ and $y(t)$ respectively via interpolation on knots (i, x_i) and (i, y_i). As illustrated in Figure 6(a), the central point (x_{n+1}, y_{n+1}) of B_{n+1} can be approximated by $(P_x(n+1), P_y(n+1))$. $\hat{\theta}^1_{n+1}$, which denotes the estimate of θ_{n+1}, can be calculated by the tangential orientation of *CCF'* at $(P_x(n+1), P_y(n+1))$. Similarly, $\hat{w}_{n+1}$, which denotes the estimate of w_{n+1}, can be calculated by the value of the cubic spline function at $(n+1, w_{n+1})$, which is constructed from interpolation on $\{(i, w_i)\}$. The first estimated location of B_{n+1} by forward extrapolation is characterized by $(P_x(n+1), P_y(n+1))$



(a) (b)

(c) (d)

Figure 6: Weighted-sum estimation of θ_{n+1} by (a) extrapolation and by (b) local greedy search, and examples when (c) $c_{p1} > c_{p2}$ and (d) $c_{p1} < c_{p2}$ in the estimation on θ_{n+1}.

and $\hat{\theta}^1_{n+1}$.

$\hat{\theta}^1_{n+1}$ is affected by the extrapolation error at the high curvature area. To offset the prediction errors in $\hat{\theta}^1_{n+1}$, we propose a local greedy search scheme for the strongest vessel feature response along the current trajectory. In the design of the local greedy search scheme, we observed that adjacent segments have small changes in direction and feature response. Most blood vessels have local maxima of intensity in both feature maps, and thus the presence of local maxima in next segment suggests that it would be a good indicator on the location of the next segment. As a result, the second estimation component for the direction of B_{n+1}, denoted by $\hat{\theta}^2_{n+1}$, is chosen by

$$\hat{\theta}^2_{n+1} = \arg_\theta \max OFR_\theta(x_{n2}, y_{n2}) \quad \text{for} \quad \theta \in (\theta_n - \Delta\theta, \theta_n + \Delta\theta), \tag{2.8}$$

subject to the constraint that

$$OFR_{\hat{\theta}^2_{n+1}}(x_{n2}, y_{n2}) > C_{stop} \cdot OFR_{\theta_n}(x_n, y_n), 1 > C_{stop} > 0 \tag{2.9}$$

where (x_{n2}, y_{n2}) is the end point on the centerline of B_n , $C_{stop} = 0.5$ is set empirically. The constraint condition (2.9) is designed to prevent run-away errors. That is, if one cannot find the next segment that has similar feature strength as that of the current one, most likely the current vessel branch ends and the tracing routine terminates at the current point. $\Delta\theta$ is the angular range for prediction of the new segment, and it should be less than the smallest angle between two vessels. $\Delta\theta = 20^\circ$ is adequate to serve our purposes. Estimation errors in $\hat{\theta}^2_{n+1}$ are most significant when the blood vessels are adjacent to lesion and other vessels.

Combining two estimations through the extrapolation and local greedy search, the final estimate of θ_{n+1} is given by

$$\hat{\theta}_{n+1} := C_{p1} \cdot \hat{\theta}^1_{n+1} + C_{p2} \cdot \hat{\theta}^2_{n+1} \tag{2.10}$$

where $0 \le C_{p1}, C_{p2} \le 1$, and $C_{p1} + C_{p2} = 1$. We note that the two weight coefficients are not constants, and they are adaptive to local curvature changes. Two cases need to be considered. In the first case, when the vessel under consideration is adjacent to a brighter vessel, as the example shown in Figure 6(c), the local greedy search would choose the brighter vessel as a (falsely) better choice. However, the current vessel usually has only small change in its direction, $\hat{\theta}^1_{n+1}$ would be a better choice than $\hat{\theta}^2_{n+1}$. On the other hand, when the vessel's direction undergoes sharp change at the new segment, as the example shown in Figure 6(d), $\hat{\theta}^2_{n+1}$ would be a better choice than $\hat{\theta}^1_{n+1}$. In other words, local curvature should be used for adaptive setting of parameters C_{p1} and C_{p2} in (2.10), where the local curvature can be estimated by using *CCF'*.

The standard definition of the curvature, denote by κ, is given by

$$\kappa = \frac{d\theta}{dl} = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{3/2}} \tag{2.11}$$

It can be calculated either from direction difference between two consecutive segments, or using *CCF* with $x(t)$ and $y(t)$ approximated by $P_x(t)$ and $P_y(t)$, respectively.

By studying the distribution of curvature in sample images, we found that 0.1 is a reasonable value as the curvature threshold to switch the two coefficients based on the following simple technique. Let $\alpha \in [0, 0.5]$, when the local curvature is greater than 0.1, we have $C_{p1} = \alpha$ and $C_{p2} = 1 - \alpha$. Otherwise, we have $C_{p1} = 1 - \alpha$ and $C_{p2} = \alpha$. Different α values are tested for the normal image in Figure 1 and its hand-labeled vessel map, and the results are tabulated in Table I. It was found that the median prediction error was minimized when $\alpha = 0.4$. This represents a performance gain of 15-20% over the prediction errors of using only $\hat{\theta}^1_{n+1}$ or $\hat{\theta}^2_{n+1}$. This parameter setting is used for all the experiments in this chapter.

Given $(P_x(n+1), P_y(n+1))$, $\hat{\theta}_{n+1}$ and $\hat{w}_{n+1}$, next we use them to make the final

Table I: Prediction error (in degree) of directions with curvature-threshold = 0.1.

$\hat{\theta}^1_{n+1}$	$\hat{\theta}^2_{n+1}$	$\hat{\theta}_{n+1} := C_{p1} \cdot \hat{\theta}^1_{n+1} + C_{p2} \cdot \hat{\theta}^2_{n+1}$				
		$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$	$\alpha = 0.5$
7.56	8.24	7.40	7.40	7.15	6.73	7.20

decision on the location and width of B_{n+1}. On the virtual centerline specified by $(P_x(n+1), P_y(n+1))$ and $\hat{\theta}_{n+1}$, we look for two edge points within the range of $\hat{w}_{n+1}$ along the two opposite perpendicular directions of the virtual centerline, where an edge point is the point that has the largest gradient value within the search range. Note that the virtual centerline is only a reference line to search for edge points in the new segment B_{n+1}. The true centerline is the collection of the middle points of opposite edge points. The value of θ_{n+1} can be attained by calculating the tangential orientation of the *CCF'* of the vessel at the end of the iteration. The width of B_{n+1} is acquired by averaging the distance of each edge point to the detected central line and then multiplying by 2. In general, the virtual centerline slightly deviates from the true centerline. Similarly, small but non-negligible errors exist between θ_{n+1} (w_{n+1}) and $\hat{\theta}_{n+1}$ $(\hat{w}_{n+1}$).

2. Bifurcation Point Detection

We search for bifurcation points on the central curve of a vessel (composed of the centerlines of the segments) following the forward detection steps. As mentioned earlier, detected bifurcation points are saved into the initialization list, so that they will be used as starting points for tracing on new branches, where the average width of the vessel is saved as the initial estimate of new branches' widths.

A bifurcation point is the junction of three vessel branches, and the *OFR* along each of the three vessel branches is relatively strong. A pixel on the centerline is considered to be located at proximity of a bifurcation point when three local maxima of its *OFR* values, denoted as OFR_{max}, are detected along three different orientations. The pixel located at (x, y) has a local maximum of *OFR* along the orientation θ' if

$$OFR_{\theta'}(x, y) = \max_{\theta} OFR_{\theta}(x, y) \quad \text{for} \quad \theta \in [\theta' - \Delta\theta, \theta' + \Delta\theta] \tag{2.12}$$

and

$$OFR_{\theta'}(x, y) \geq \mu_{OFR}(x, y) + C_b \cdot \sigma_{OFR}(x, y) \tag{2.13}$$

where $C_b > 0$ is a constant, $\mu_{OFR}(x, y)$ and $\sigma_{OFR}(x, y)$ are respectively the average and standard deviation of the *OFR*s along all discretized orientations. The constraint (2.13) is designed to eliminate false detections caused by noise. Experimental results shows that $C_b = 0.7$ suffices when the angular resolution is 10°.

We note that multiple pixels around a bifurcation point will meet the detection criteria (2.12) and (2.13). However, only the pixel that has the highest total sum of its three OFR_{max} values will be chosen as the representative location of the bifurcation point. Two of the three local maxima OFR_{max} point to directions of the two new branches, and the last one points to the backward direction of the vessel toward B_n. If the current vessel is a small one, the bifurcation point represents the location that connects three small vessels, and thus only the *OFR*s defined for small vessels need to be evaluated.

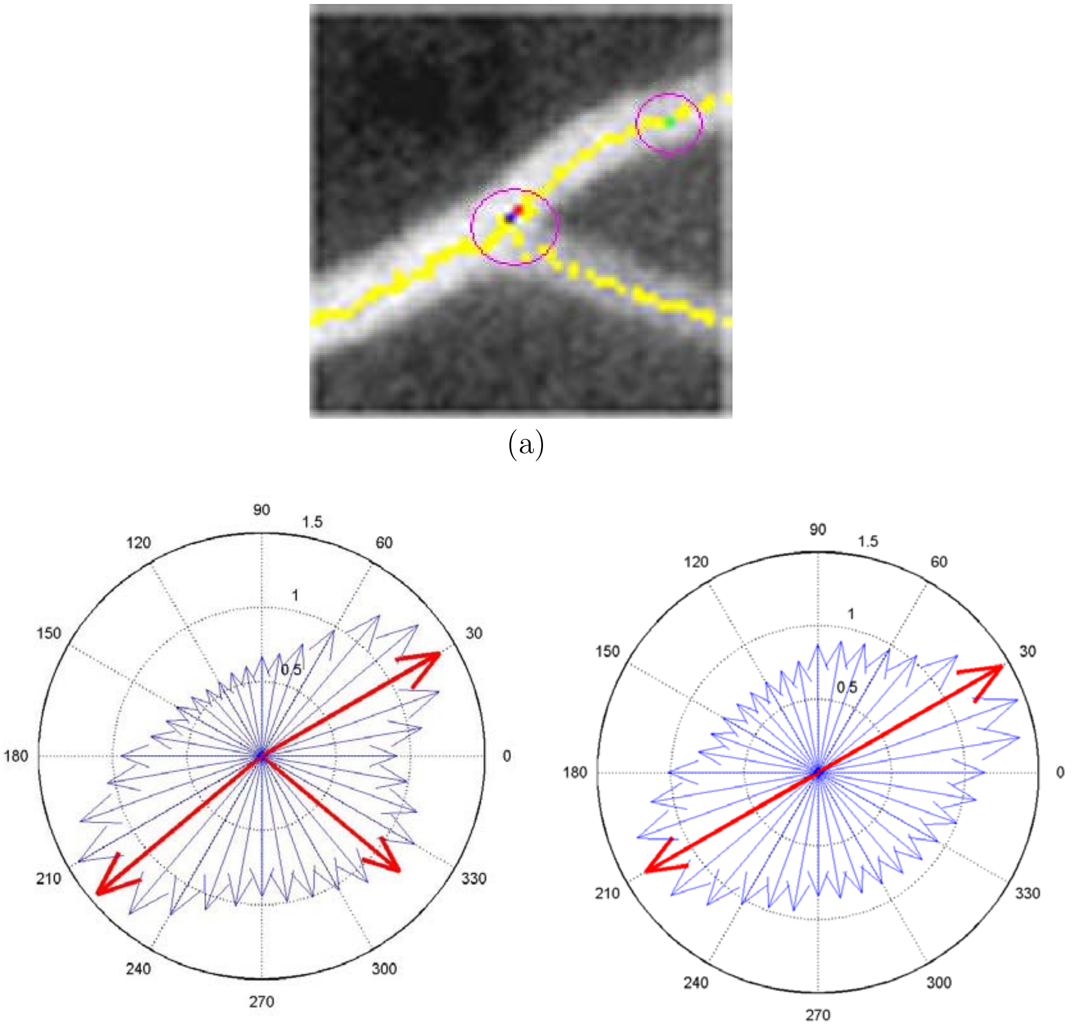
If the current vessel is a large one, the new branches can be large or small. If the bifurcation point represents the attaching point of a small vessel to a large one, then it can be identified by the presence of one OFR_{max} for small vessels along one orientation, and the other two OFR_{max} for large vessels along the forward and backward directions. If the bifurcation point represents the splitting point of a large vessel into two small ones, then two of the OFR_{max} values for small vessels should point to orientations other than the backward direction of the current vessel.

Figure 7 shows an example of bifurcation point detection. In Figure 7(a), the yellow line represents the central line of the blood vessel detected by forward detection. The OFR values around the blue point in Figure 7(a) are plotted in Figure 7(b), where the three OFR_{max} values are marked by red arrows. By definition, this example represents a bifurcation point. Although the red dot in Figure 7(a) also meets the detection criteria, it is not chosen as the bifurcation point because the blue dot has a larger total sum of OFR_{max} values. OFR values around a randomly chosen non-bifurcation point (the green dot) are plotted in Figure 7(c), where only two OFR_{max} values exist along the forward and backward directions.

We tested the effect of image resolution on identification of bifurcation points. Using the image in Figure 7(a) as an example, the bifurcation point in the image could still be identified by the method just mentioned when the image size is reduced until to 40% of the original size. That is because the sizes of vessels and the Gaussian kernel (used to calculate OFR) are proportional to the image size and they are also scaled down. When the image size is below 40% of the original size, the sizes of the vessels in the image are only 2 pixels or less, which reaches the limit of the Gaussian kernel, causing the failure of identification of bifurcation points.

3. Backward Verification

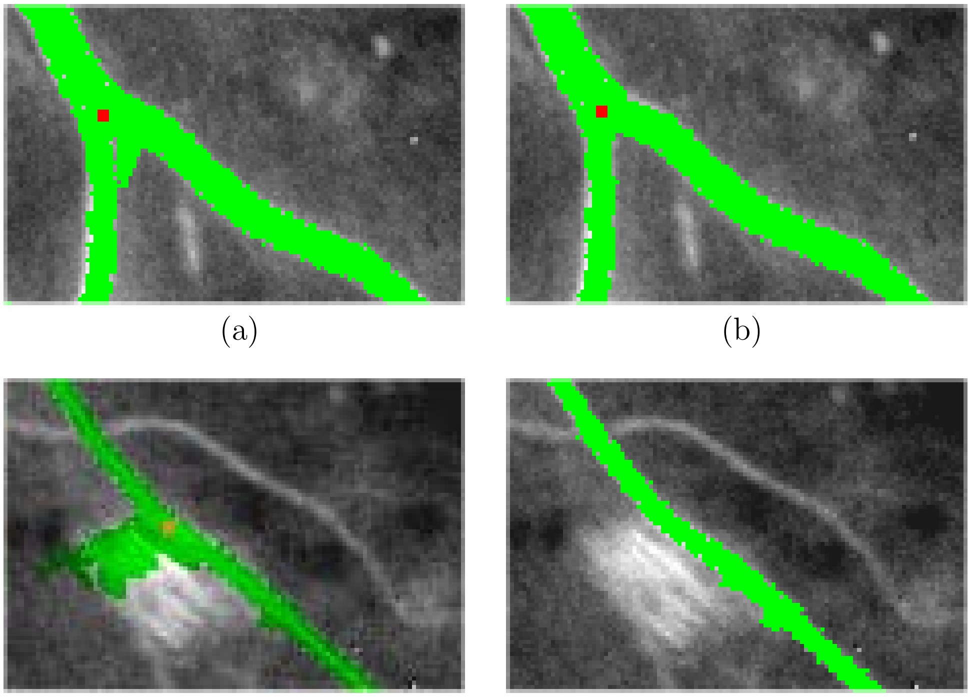
The backward verification process is designed to eliminate false detections of the two routines mentioned above. It uses vessel widths and OFR values of consecutive segments for outlier detection [37] based on Grubbs' algorithm [38]. This algorithm assumes that the finite observations of a study are taken from a Gaussian population. To detect the outlier, one first calculates $Z = |x - \bar{x}|/\sigma'$ where x is the sample being



(a)

(b) (c)

Figure 7: Bifurcation point detection: (a) a cropped image, (b) the *OFR* values along different orientations around a bifurcation point (blue dot in (a)), and (c) the *OFR* values around a non-bifurcation point (green dot in (a)).



(a) (b)

(c) (d)

Figure 8: False detection cases (a) and (c), and their correction (b) and (d) in the backward verification. The red dots represent detected bifurcation points.

examined, and $\bar{x}$ and σ' are respectively the mean and standard deviation of the observations. [38] provides a table of critical values for different observation sizes, where an outlier is detected when the test outcome of x exceeds the critical value for the given data size. Experimental results show that Grubb's algorithm is very effective in capturing false detection due to lesions or hemorrhages that are located close to vessels or bifurcation points.

Examples of the backward verification are given in Figure 8. Figure 8(a) shows an example in which the width of the segment next to a bifurcation point is overestimated. The location and width of the overestimated segment are corrected by using its adjacent segments, and the middle points are estimated through cubic-spline interpolation on the adjacent segments. The corrected detection result is shown in Figure

8(b). In the example shown in Figure 8(c), a hemorrhage patch adjacent to a blood vessel was falsely detected as a vessel branch. Our method limits its exploration range based on the width constraints, and the correction routine removes the hemorrhage from the vascular map, i.e., see Figure 8(d), because several outliers were detected in the sequence of segment widths of the false branch, and the branch's average width was found to be larger than that of its parent vessel. Obviously, once the vascular network is correctly identified, it is relatively easy to detect the hemorrhage in a separate process.

D. Experiments and Performance Evaluation

The performance of our scheme is evaluated against its objectives: detection of large and small blood vessels. Separate performance measurement on detection of large and small vessels is consistent with the hand-traced vascular maps provided in [7], where coarse-grained and fine-grained labeling, called hand-1 and hand-2 maps, respectively, are provided to illustrate different detection resolutions. The hand-1 maps are used as the reference maps in calculating the detection rates of our scheme. The hand-1 maps appear to use stricter detection rules for the vascular boundaries, and thus they have thinner blood vessels.

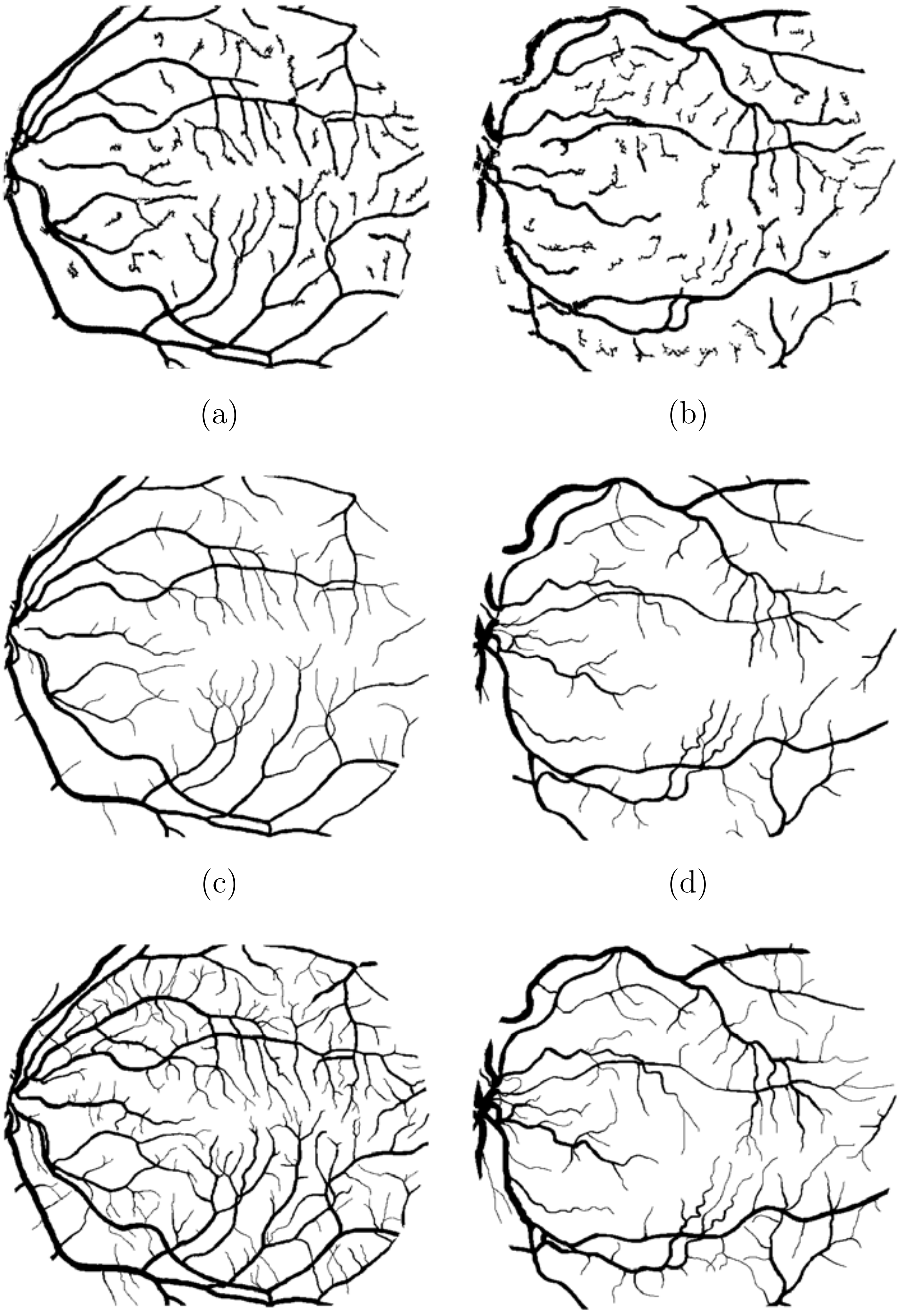
It is relatively straightforward to evaluate the performance of large blood vessel detection algorithms, but it is much more complex to evaluate the performance of small vessel detection algorithms because even hand traced maps can have substantial amount of inconsistent labeling. The most widely used performance criteria for blood vessel detection are *true positive rate* (TPR) and *false positive rate* (FPR), both of which are pixel level indicators. When a pixel is marked as a detected vascular pixel, the detection is a true positive if the pixel is also on the hand-traced vascular network.

Otherwise, the detection is a false positive. TPR is defined as the number of true positive divided by the total number of hand-traced (vascular) pixels. FPR is defined as the number of false positive divided by the total number of non-vessel pixels on the hand-trace map.

We have implemented our solutions in C#, and all the experiments were tested in MS Windows environment. The majority of the computational cost was due to the contrast enhancement, where the interpolation technique proposed in [39] for calculation of local histogram in the adaptive contrast enhancement algorithm was adopted to reduce the computational cost. The twenty retinal images used in [7] were chosen for comparative performance evaluation. The image set contains 10 normal (pathology-free) images and 10 abnormal (pathological) images under various illumination conditions.

Key parameters used in the evaluation experiments were determined based on statistical analysis of a normal image and another abnormal image, together with their hand-traced maps. The parameters derived from the training phase were assigned to our algorithms for testing on the remaining images. Figure 9 shows the computer-traced and hand-traced maps for the sample images given in Figure 1.

The numerical results on TPR and FPR for the twenty normal and abnormal images being tested are given in Table II and Table III, respectively. For normal images, TPR values range from 80% to 91%, and the corresponding FPR values range from 2.8% to 5.5%. For abnormal images, the TPRs range from 73.8% to 86.5%, and the FPRs range from 2.1% to 5.3%, respectively. The overall average performance of our method over the twenty tested images is 84.3% for TPR and 3.9% for FPR, which are marked as a single point in Figure 10. For comparison, the average performance curves reported in [7] (filter response analysis), and in [13] (multi-threshold probing) are also included in Figure 10.



(a) (b) (c) (d) (e) (f)

Figure 9: First row: detection results for the images in Figure 1. Second and third rows: the hand-1 and hand-2 maps of the same images.

Table II: Performance of our method for 10 normal images.

TPR (%)	91.0	88.0	89.9	82.7	90.6	85.1	82.7	86.2	86.2	80.0
FTR (%)	5.0	4.3	5.5	4.1	4.2	2.9	2.6	3.8	4.2	2.8

Table III: Performance of our method for 10 abnormal images.

TPR (%)	84.5	82.6	84.9	79.1	73.8	84.4	83.9	86.5	78.1	85.2
FPR (%)	4.5	3.5	3.6	4.0	3.8	4.1	4.5	2.1	3.4	5.3

To get a rough idea on the performance upper bound of computer-based detection systems, we calculated the "TPR" and "FPR" of the hand-2 maps. It was found that the "performance" of the second set of hand-traced maps to be 89.5% TPR and 4.4% FPR, which are also marked as a single point in Figure 10. The TPR of our method is lower than that of the hand-2 maps, but its FPR value is much lower. For the two schemes in [7] and [13], the multi-threshold probing method will need to increase its FPR to 5% to achieve the same level of TPR as ours, and the filter response analysis method needs to increase its FPR value significantly to achieve the same goal. In addition to its performance benefits, our scheme requires very little calibration. In contrast, the two approaches that are being compared in this study need ten and eight sets of parameter values, respectively, see the plotted points in Figure 10. For these two approaches, when TPR is over 80%, a small increase of TPR will have to be at the cost of large increase of FPR, suggesting the difficulty of further improving vessel detection performance without significantly increasing false positive detections.

The performance of our method is mainly attributed to the effectiveness of the contrast enhancement and feature extraction routines, especially for small vessels. A close study of 20 images shows that small vessels account for 42% of overall vessel

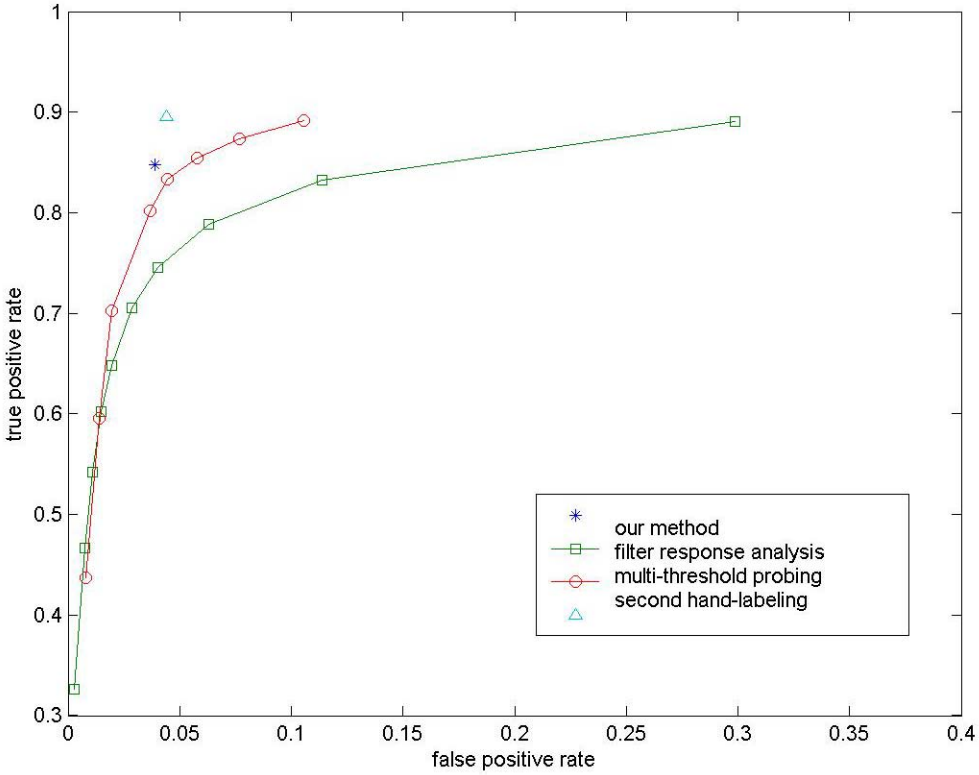


Figure 10: The average detection performance over 20 images: our method vs. the filter response analysis method and the multi-threshold probing method.

pixels, while 75% of small vessels were captured by our method. Except for lesions and noise, a close comparison between our detection results and the hand-traced vessel maps shows that the differences between the two maps are mostly located at vessel boundaries. Our scheme tends to yield thicker small vessels than those in the hand-traced maps because of its sensitivity to weak signals.

In addition to the two methods mentioned above that we compared in details, it is worth mentioning the difference between our scheme and the PBM method proposed in [15], which used nineteen of the twenty images that we used for performance evaluation. We do not make direct numerical comparison between our scheme and PBM based on two simple reasons. First, the performance of PBM is the testing outcome of their scheme on each image using the other eighteen images for training, but our scheme only needs to use two training images. The performance sensitivity of the size of the training set cannot be compared without their substantial experiments. Second, our experiences show that for a moderate test set of twenty images, the performance outcome on any abnormal image can have significant impact on the overall detection performance of twenty cases. Without knowing the nature of the abnormal image that has been left out from their test set, we cannot make objective statement about the differences between the two methods.

In terms of design complexity, our scheme is much simpler. It requires only the hand-traced maps of two (normal and abnormal) images for parameter training, and the performance outcome is insensitive to the choice of the training images. On the other hand, for PBM the sensitivity of feature selection is unclear. From over twenty feature types, one would need to observe the incremental performance outcomes of addition or removal of specific features, in order to decide the final picks. Normalization and weighing of different features that have significantly different dynamic ranges can become a challenging issue.

The imperfection of the hand-2 tracing maps is mainly due to different interpretation of the blood vessel boundaries. By the fact that only 89% of vessels are identified by both operators, one needs to exercise caution in interpreting the TPR and FPR values as the performance measure.

Figure 11 shows the vessel reconstructed from segment features including middle points, widths and directions. In the figure, the central curve S, shown in magenta color, is generated by Hermite interpolation [36] using the middle points (marked as "x") and directions of the segments. That is, assuming that the sequence of middle points and directions are $\{(x_i, y_i)\}$ and $\{\theta_i\}$, respectively, $0 \le i \le n$, S is constructed so that $S(x_i) = y_i$ and $S'(x_i) = \tan\theta_i$, for $0 \le i \le n$, where $\tan\theta_i$ gives the value of first derivative of S at (x_i, y_i). S is calculated by

$$S(x) = \sum_{i=0}^{n} y_i A_i(x) + \sum_{i=0}^{n} \tan(\theta_i) B_i(x) \tag{2.14}$$

With the aid of the functions

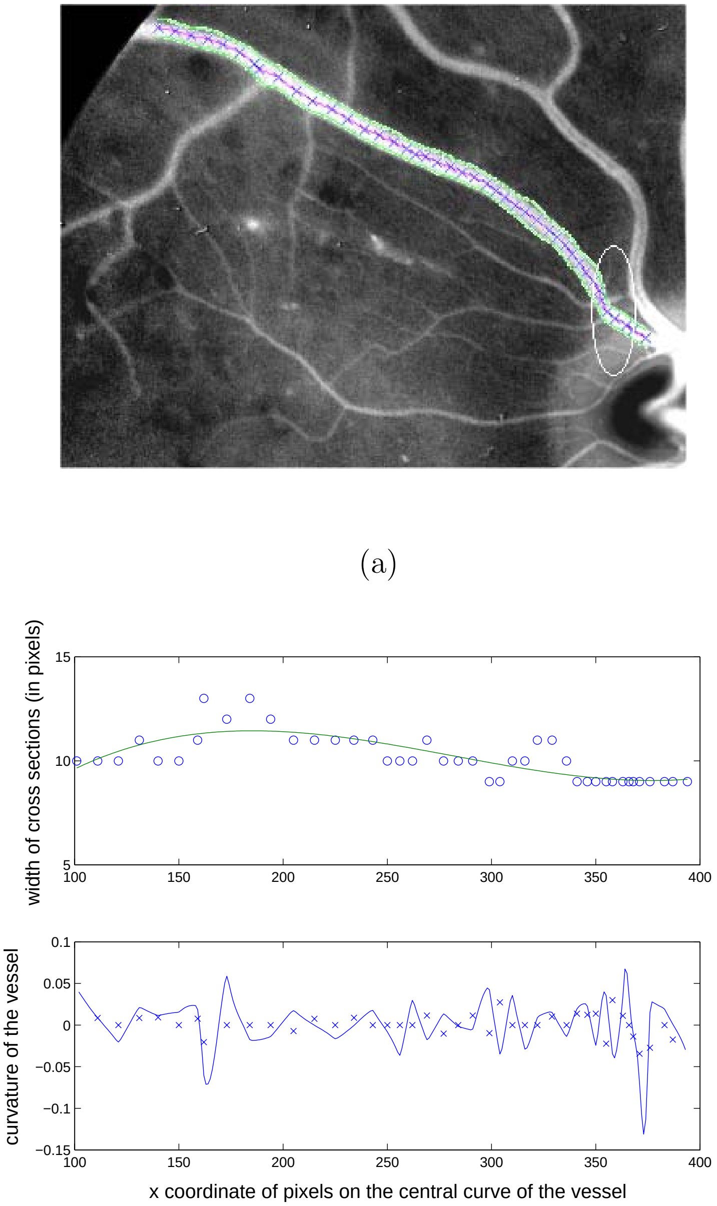
$$l_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j} \quad (0 \le i \le n) \tag{2.15}$$

A_i and B_i can be expressed as follows

$$A_i(x) = [1 - 2(x - x_i) l_i'(x_i)] l_i^2(x), \quad B_i(x) = (x - x_i) l_i^2(x) \tag{2.16}$$

Two vessel edges are plotted in green color, which are determined by the width function and the central curve. It can be seen that the reconstructed vessel closely matches the original one.

The width function of the vessel is plotted at the top of Figure 11(b). It is constructed by fitting a polynomial of degree 3 to the segment widths (marked by "o" in the plot) with least square error [36]. Round errors occur when the measured



(a)

(b)

Figure 11: (a) A reconstructed vessel from segment features, and (b) the width function and curvature of the vessel.

segment widths are quantized to pixels (integers). However, the approximated width function still shows that the diameter of this vessel changes significantly from left to right, indicating the abnormality of the vessel diameter distribution caused by a blockage of blood flow at the location circled in Figure 11(a). The curvature of the vessel is evaluated and illustrated at the bottom of Figure 11(b). The line in the plot stands for the result calculated numerically by (2.11) on the Hermite-interpolated central curve, and the dots marked in "x" are the curvature values of each segment approximated by the direction difference between two consecutive segments divided by the distance of the middle points of the two segments. The two methods of curvature evaluation both attain the maximum of absolute values at the same area of the vessel, circled in Figure 11(a). It indicates that although the latter method is relatively simple, it does capture the high curvature area by using features of discrete segments.

CHAPTER III

IMAGE REGISTRATION

A. Introduction

The objective of image registration is to align images acquired for an object so that the same object imaged at different times can be analyzed. Many algorithms have been proposed in the literature for image registration. Generally speaking, image registration algorithms can be classified as either *feature-based* or *intensity-based* [40]. The intensity-based matching approach compares and matches intensity differences of an image pair without explicit extraction of image features. The intensity-based approach tends to be less compute intensive, but is more sensitive to illumination changes and significant changes of the imaging positions. For retinal image analysis, this approach may be susceptible to occlusion, background changes caused by pathologies. It is particularly difficult for images that are affected by diseases such as cataract, which may significantly interfere with the photo qualities.

In [41], the similarity between images is measured by their intensities. The best rigid transformation is obtained using simulated annealing. This technique searches for the global optimum based on pyramid sampling for speed and accuracy. For performance evaluation, the registration results are compared with the ones obtained through the exhaustive search.

In [42], the Fourier coefficients of an image is first performed to the log-polar coordinate system. It converts the rotation and scaling differences of two images to their translation differences. The rotation and scaling parameters are then computed from the correlation of the log-polar pairs. The translation parameters were estimated by correlating the rotated and scaled images

Feature-based registration methods usually search for the transformation that optimizes the correspondence between the extracted features. For retinal image registration, the bifurcation points of blood vessels are the widely used features in this approach. The main advantage of the feature-based approach is its robustness against illumination changes.

In [9], a method based on Bayesian-Hough transform is proposed for temporal and multimodal registration of fluorescein images and green images (green channel of a color image). Each transformation that matches one pair of feature points in an image to another pair of feature points in the other image is scored by the *a posterior* probability. The best 20 transformations are used to derive the final transformation. The computational cost of this approach is relatively high, taking 5 to 7 minutes to register two images. The performance of this method was only evaluated using visual evaluation. In [43] a second-order polynomial transformation model was developed by characterizing retina as a rigid quadratic surface and retinal imaging as a weak perspective camera. The transformation parameters were estimated by a hierarchical algorithm. Registration error was evaluated by the median distance of the corresponding centerlines of blood vessels and the percentile of overlapping between images. This method achieved favorable performance results, less than one pixel error on average, even the overlapping area between two images is small.

In [44], a method based on a global point mapping was evaluated on three transformation models. The affine transformation based scheme achieved better performance results than the similarity and second-order polynomial based schemes, in terms of the superposition percentage. The Constrainted parameters in translation and rotation makes the proposed method unsuitable for registration of images with large displacements. Due to the small number of feature points being extracted, this method may fail to register images that have poor qualities.

In [45], the set of transformation parameter vectors that match every two control point pairs under rigid global transformation were first obtained. The distribution of Euclidean distance of each vector to its nearest neighbor was characterized, and then parameters of the distribution were estimated to identify the cluster of vectors that have correct matching. Bilinear and second order polynomial transformations were then obtained by linear regression of the control points that generated the "correct" cluster. Only two retinal images were tested for their performance evaluation.

In [46], the squared error of the edge direction histogram of a feature point pair was used to evaluate the correspondence between the two points. The final transformation composed of translation and scaling was validated by a sequential similarity detector.

In [47], automatically extracted vascular networks were used to measure the degree of similarity between images. Three optimization techniques were investigated to search for a global optimal transformation. Results for multimodal registration on fluorescent angiogram (FA), red-free retinal image (RF), and Indocyanine Green Chorioangiography (ICG) showed the effectiveness of combining genetic algorithms with the affine and bilinear transformation models. The allowed ranges of the transformation parameters indicate the slight change in viewpoint between the matched images. The registration of 512×512 images takes 4.5 minutes on an Indigo SGI.

In [48], the extracted blood vessel centerlines and bifurcation points were used for transformation initialization and image alignment. A Dual-Bootstrap iterative closest point method was proposed to refine the initial low-order transformations to the high-order transformation. It expanded the initial small matching region to the entire overlap region between images. The experiments on a large image set show the good performance.

In [49], the sequential similarity detection technique was used to evaluate the alignment of two images based on sub-sampled boundary points on blood vessel. The Hough transform technique was used to find the optimal translation and rotation parameters. The performance was examined by visual inspection of the corresponding positions of a manually selected cross-marker in the registered images.

In [50], the correlation between features extracted by a robust creaseness operator measures the quality of the alignment in two images, and the iterative simplex algorithm was used to optimize the alignment process.

In [51], the end points of the extracted blood vessel segments are used as feature points. Mismatched feature-point pairs were eliminated according to constrains on the parameter values of translation and scaling, and the correlation scores between the pixel intensities around each feature point. The initial transformation obtained from the least-square fitting on the remaining feature points is then iteratively refined. A common coordinate system for two images was used with the location of the identified optic nerve as the common origin. Similar to the intensity-based approach, this method is also sensitive to the color or intensity changes between the image pair. It can also be affected by the detection errors of locating optic nerves.

An active contouring based registration technique was proposed in [52], with only simulated images tested. The contour of blood vessel and bifurcation points in one image is placed on the other, and then the contour is adjusted until a matching can be made. Active contouring usually requires human guidance. Because only local features are used for adjustment of each snaxel, the method is susceptible to large translation between the two images.

In [53], *edge points* of two images extracted using edge detector are recorded in two matrices. Two images are registered by aligning the matrices on a template area to find the best transformation comprising of only translation. This method is unsuitable for registration of image pairs with rotation and scaling changes.

A recent survey on registration of medical images can be found in [54]. In general, the feature-based approach is more reliable than its intensity-based counterpart.

Despite the extensive work done in this area, feature-based registration of retinal images needs to pay special attentions to following issues:

1. Image registration fails when not enough features can be detected due to inadequate illumination, lesions, or the small overlapping of two images.
2. Feature-extraction under significant changes of image conditions, e.g., bifurcation points and vessels, is difficult. Inconsistent imaging qualities are commonplace because of pathological changes and extended time periods between images acquisitions.
3. Uneven/clustered placements of feature points may lead to *localized registration decisions* around the recognized feature points.
4. The high similarity between registration features of an image may lead to significant mismatches.
5. Precise registration is critical to capture subtle changes of small vessels that are three pixels or less in width. Such changes are important indicators to onset of various diseases.

Image registration is useful for long term and short term tracking of retinal changes, e.g., diabetic retinopathy, glaucoma, etc. Such changes can occur within a short time, e.g., before and after laser surgery, or within a long time, e.g., progression of lesion development or effectiveness of therapeutic treatments. It can also be used to form a mosaic image of the whole retina from a series of snapshots taken from different angles. Registration performed in real-time can assist control and adjustment of laser surgery and other related procedures [12][55][56]. Retinal images to be matched can be acquired through different modalities [57], i.e., color fundus image, FA, RF, ICG. The two images shown in Figure 12 represent an example of retinal image registration. The first and second images were taken before and after a laser surgery. Image registration will help detection and confirmation of the laser surgery (or incidental laser exposures).

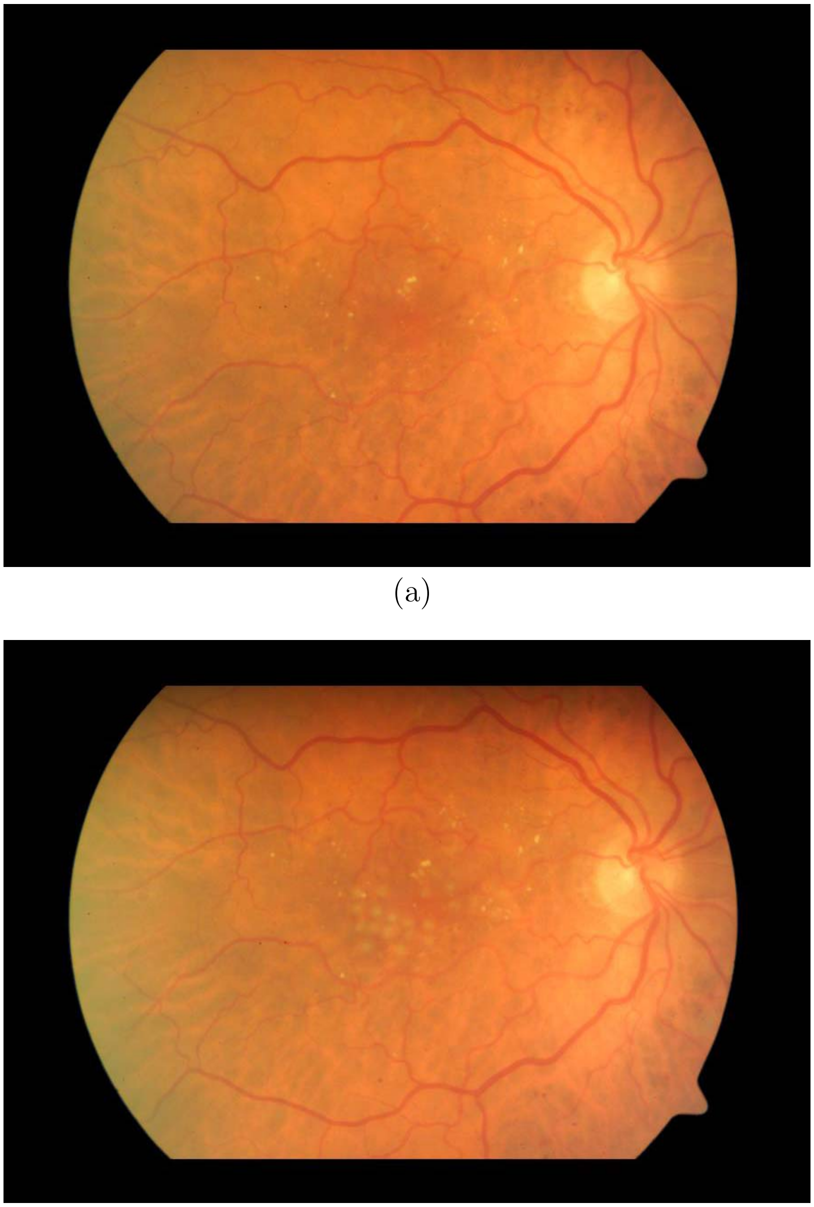
We propose here a feature-based registration scheme for retinal images using bifurcation and cross-over points extracted from the blood vessel maps. The local structures of feature points are used to establish the correspondence between feature points in two images. An affine transformation is then derived to map the images to a common frame. To achieve accuracy and speed, both the coarse and fine maps of blood vessels are extracted.

B. Registration

Four major steps are involved to register two images together. The first two steps are automated extraction and matching of image features from the two images. The third step is deriving a transformation between corresponded features so that in the fourth step, the two images can be transformed and merged into one common map. In our work, we use the bifurcation and cross-over points extracted from the blood vessel maps as registration features. These feature points are corresponded by their structural similarity. Then, an affine transform is estimated from the correspondences and is used to map two images into a common frame.

1. Extraction of Registration Points

In the previous methods for retinal image registration, vascular network is the most widely used landmark because of its full coverage of the whole retina, time-invariant



(a)

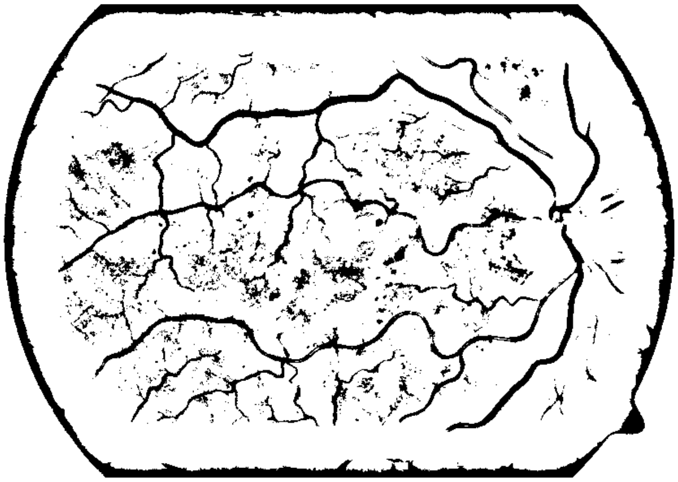
(b)

Figure 12: Two images photographed (a) before and (b) after laser surgery. Some laser scars are located at the center of the second image.

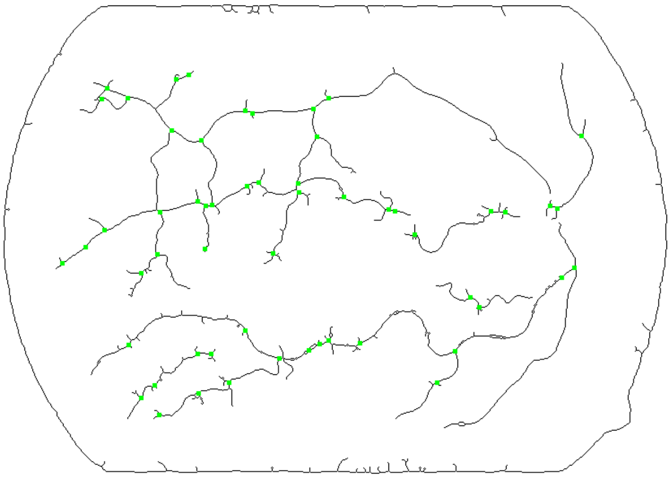
structures (except for certain severally diseased conditions), and easy localization of anchor points. Due to the diverse qualities of the images and interference of noise and lesions, full extraction of the vascular network is too time-consuming and unreliable. Speed is important for applications such as real-time surgery control, and high volume screening of retinal injuries. As a result, we choose the bifurcation points and cross-over points of the vascular network as registration features of retinal images.

Using bifurcation points for registration allows fast extraction of the vascular network at certain loss of precision in detection of vessel shapes. We propose a fast vessel detection method, which extracts a coarse map of blood vessels, and then look for the registration points from the map. This approach was found to be quite adequate for registration applications. The fast blood vessel extraction scheme is similar to the scheme described in chapter II. First, the green channel of the color fundus image is extracted, and then its intensity values are inverted so that blood vessels have higher intensities than the background. The illumination variance across the image is first reduced by the equalization scheme proposed in [28], and then our contrast enhancement scheme (2.1) is applied to enhance the contrast of blood vessels. The enhanced map is quantized and normalized to the range of [0,255], and then segmented by thresholding (threshold = 100, an empirical setting) to obtain a coarse blood vessel map. The coarse blood vessel map of the image in Figure 12(a) is shown in Figure 13(a).

Figure 13(a) shows that our scheme can extract the majority of large vessels and some background noise. The thinning algorithm proposed in [58] is applied to the coarse blood vessel map to obtain centerlines of blood vessels, and short lines are removed. Pixels that are adjacent to three or more other (centerline)pixels are candidate registration points, and they are inspected by using *OFR*s to determine if they are indeed bifurcation points. A candidate registration point is accepted as



(a)



(b)

Figure 13: (a) An coarse blood vessel map and (b) its thinned centerlines as well as the extracted feature points that are highlighted in blue.

a bifurcation point if it has three or four local maxima of *OFR*. For the example depicted in Figure 13(b), black lines are centerlines of the blood vessels in the coarse vascular network, and blue points represent the identified bifurcation points.

2. Matching of Registration Points

The objective of matching of registration points is to establish the best correspondence between registration points, so that the affine transform can be obtained from the best matched registration points. Local structure descriptors of registration points are used to establish a correspondence relationship between the two images being registered. *OFR* is a well suited descriptor for orientations, widths, and branch angles [9] of blood vessels adjacent to a registration point u. Local maxima of *OFR* give orientations of adjacent vessel branches. The angle between orientations of local maxima is indeed the angle between two branches. And the magnitudes of *OFR* represent the widths of branches.

Let Δt denote the atomic unit for measurement of the orientation, and $n = 2\pi/\Delta t$ is the number of *OFR* observations. The *OFR* vector for u observed at the n orientations is expressed as

$$\vec{u} = (OFR_{\Delta t}(u), \ldots, OFR_{n\Delta t}(u)), \tag{3.1}$$

where $OFR_{i\Delta t}(u)$ is the *OFR* at point u along direction $i\Delta t$. Knowing that two registering images can be taken under highly different illuminating conditions, the *OFR* vector is normalized so that $\|\vec{u}\| = 1$, where $\|\cdot\|$ is the Euclidean norm. The similarity between two feature points u and v is measured by their Euclidean distance

$$d(\vec{u}, \vec{v}) = \sqrt{\sum_{i=1}^{n}[OFR_{i\Delta t}(u) - OFR_{i\Delta t}(v)]^2} \tag{3.2}$$

For a registration point u in the first image, its corresponding point in the second image is determined by a back-and-forth search scheme. That is, for u we first find a registration point u' in the second image, where the difference between *OFR* vectors of the two points is smallest. Reversely, for the just chosen u', we search for u'' in the first image, so that the difference between their *OFR* vectors is the smallest. If $u = u''$, a correspondence between u and u' is established. Otherwise, no match.

The angular difference between the images caused by movements of eyes or/and camera should be considered in searching of the correspondence points. That implies the possible shift between *OFR* vectors of corresponded registration points. To accommodate this condition, (3.2) is revised to the following expression, where the angular difference between the two registering images is upper-bounded by $J\Delta t$, and J was empirically set to be 2 when $\Delta t = \pi/18$.

$$d(\vec{u},\vec{v}) = \min_{|j|\leq J} \sqrt{\sum_{i=1}^{n}[OFR_{i\Delta t}(u) - OFR_{mod(i+j,n)\cdot\Delta t}(v)]^2} \tag{3.3}$$

3. Estimation of Transformation

Four transformation models have been widely used for mapping of corresponded registration points in the literature. The first is the *affine transformation*, which maps straight lines to straight lines, and preserves parallelism of lines. The affine transform can be implemented by the combination of a linear transformation and translation. An affine transformation is defined by the following six parameters:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a_1 & a_2 \\ b_1 & b_2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_0 \\ b_0 \end{pmatrix} \tag{3.4}$$

where (x, y) and (x', y') are the coordinates of the corresponding points in the two images, a_i and b_i are the parameters that specifies the transformation.

The second transformation is the *bilinear transformation*, which is the simplest polynomial transformation to map straight lines into curves. It has eight parameters and is expressed by

$$\begin{aligned} x' = &\quad a_0 + a_1 x + a_2 y + a_3 xy \\ y' = &\quad b_0 + b_1 x + b_2 y + b_3 xy \end{aligned} \tag{3.5}$$

The third model, the *projective transformation* maps straight lines to straight lines without preserving parallelism of lines. This transformation is expressed by

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} u/w \\ v/w \end{pmatrix} \tag{3.6}$$

where w represents the extra homogeneous coordinate. The projective transformation contains nine parameters, while only the ratio of parameters is important.

The fourth model, the *quadratic transformation*, proposed in [43], is derived from the combination of rigid transformation, weak-perspective camera model, and a quadratic surface model of retina. It is defined by twelve parameters as follows.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} \end{pmatrix} \begin{pmatrix} x^2 & xy & y^2 & x & y & 1 \end{pmatrix}^T \tag{3.7}$$

Among these transformation models, the quadratic transformation has the best smoothness but the highest computing costs, with 12 parameters that need to be estimated. The affine transformation has the least degree of freedom. It outperforms the bilinear and projective transformations in terms of registration accuracy [47]. A further important consideration about the transformation choices is that many retinal images have poor illumination conditions, or significant presence of lesions over the whole image. For these images, only a small number of blood vessels can be detected reliably, and thus only limited number of registration points can be used. Putting these factors together, we adopt the affine transformation in our study.

The six parameters of the affine transformation can be estimated by solving the following system of linear equations, based on k pairs of corresponded registration points.

$$\begin{pmatrix} x'_1 & y'_1 \\ x'_2 & y'_2 \\ \vdots & \vdots \\ x'_k & y'_k \end{pmatrix} = \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_k & y_k & 1 \end{pmatrix} \begin{pmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_0 & b_0 \end{pmatrix} \tag{3.8}$$

where (x_i, y_i) and (x'_i, y'_i) are the coordinates of two corresponded registration points. Since the affine transformation has six parameters, three pairs of corresponded registration points are enough for parameter estimation. However, considering that registration may be accurate only around the employed registration points, it is better to use all available correspondences of registration points to improve the performance. When $k > 3$, as in most practical cases, we solve the system using the singular-value decomposition (SVD) method [59].

The SVD method is based on the fact that any matrix A with M rows and N columns can be factored as

$$A = U \cdot [diag(w_j)] \cdot V^T, \tag{3.9}$$

where $U(M \times M)$ and $V(N \times N)$ are unitary matrices (i.e., $UU^T = I, VV^T = I$), and $[diag(w_j)]$ is an $M \times N$ diagonal matrix. The pseudo-inverse of A is defined as

$$A^+ = V \cdot [diag(1/w_j)] \cdot U^T \tag{3.10}$$

Thus, the solution of the equation $Ax = b$ is given by $x = A^+b$.

Using the estimated affine transformation, one image can be mapped to the coordinate system where the other image locates, and the two images are then fused.

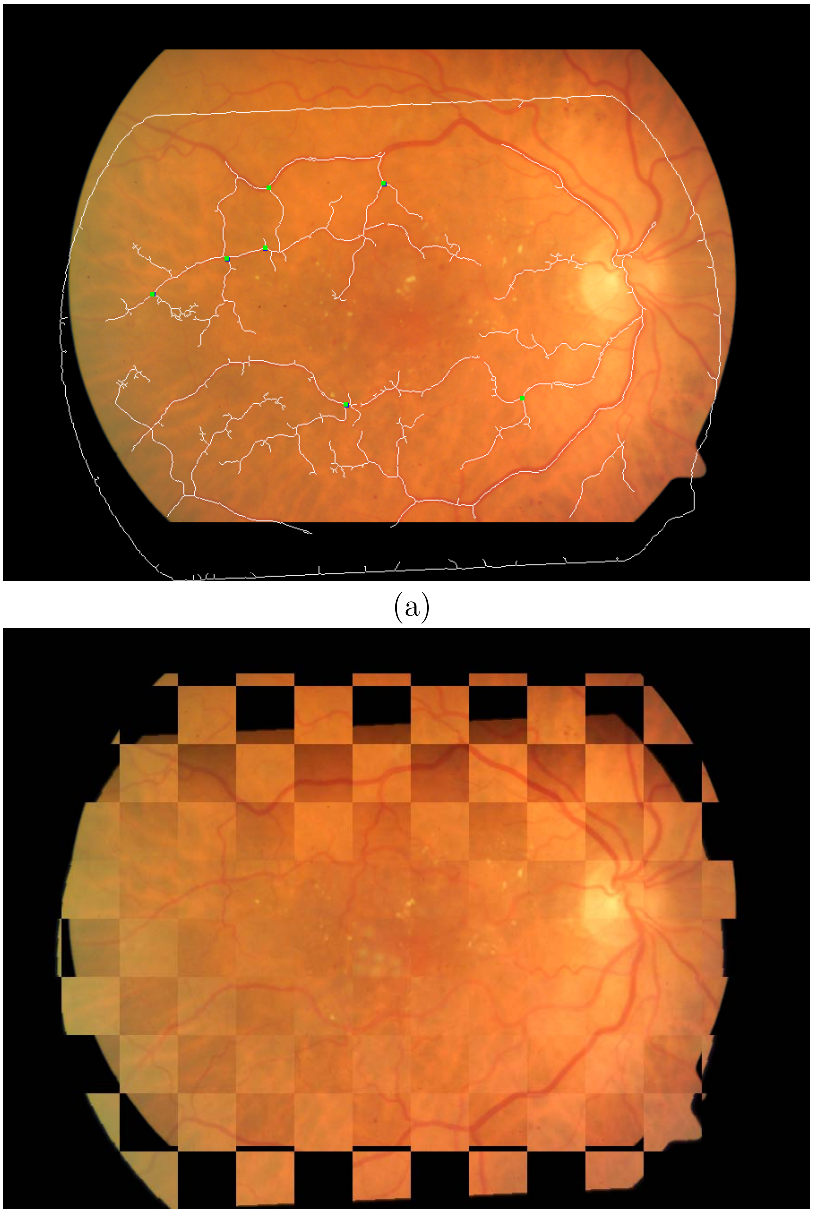
C. Performance Evaluation

Our retinal image registration scheme was implemented in C#, and its performance was evaluated on a (Pentium 4 CPU 2GHz, 512MB RAM) PC that runs MS Windows XP. Fourteen pairs of retinal images ranging from good to poor image qualities were used to test our scheme. The average registration time for two 900×600 images is about 15 seconds, much faster than most existing registration methods [43][47][60].

The accuracy of the registration scheme is further assessed visually and by quantitative measures. Visual assessment of registration results is based on a checkerboard mosaic image of the two images, and by superposition of the transformed extracted vessels of one image onto the other image. A good registration is one with a good superposition of blood vessel centerlines. Figure 14(a) shows an example of superposition, where the green dots in the image represent the registration points extracted and used for registration.

The checkerboard is composed by the interleaving grids extracted from the reference and transformed images. A good registration is one with smooth connection of blood vessels from grid to grid. Figure 14(b) shows an example of the checkerboard composition.

Typical numerical measures of registration algorithms include standard deviation, entropy, cross-entropy, spatial frequency [44]. In our study, we measure the registration performance by the *registration error*, similar to the one defined in [43] as follows. Let A be the image mapped into the coordinate system of B. The registration error between A and B is the median of the distances from all centerline pixels on A to their nearest centerline pixels on B.


(a)

(b)

Figure 14: Registration result of the two images shown by: (a) the superposition, (b) the checkerboard.

These, and other similar measures work fine for good quality images, but they are not applicable to images that have significantly different image qualities, or when occlusion occurs. For a blood vessel located in the overlapping area of A and B, it may appear in one image but does not show up in the other image. Therefore, we modify the performance measure as follows.

Assuming the image with less centerline pixels to be A, we define the neighborhood as a square window with size 50 by 50. For a centerline pixel p in the neighborhood of a feature point of A, it is considered a *matched pixel* if in the neighborhood of p one can find a centerline pixel of B. The correspondence of p is the centerline pixel of B in the neighborhood of p with the least distance to p. The performance measure, *median distance*, is defined to be the median of all distances of every matched pixel to its correspondence.

If the distance of a matched pixel to its correspondence is less than a specified threshold, the pixel is considered to be a *well-matched pixel*. The registration *score* is defined as the ratio of total number of well-matched pixels to the total number of matched pixels in A. In this work, we choose 5 pixels as the threshold for well-matched pixels, and 0.5 as threshold of registration score empirically.

The registration results for the thirteen tested image pairs are shown in Table IV. Ten image pairs were successfully registered, with median difference between the matched centerlines less than 2 pixels. Three image pairs were failed to be matched, which can be captured by the large median differences and lower registration scores. For comparison, the registration performance of the test images using their fine vessel maps (detected by the algorithm proposed in Chapter II) is also included. The performance data (except the last row) are worse than that of using coarse vessel maps since more vessels are detected and appear in the fine vessel maps, and most of them do not have correspondence.

Table IV: Registration scores for test images: “R” denotes the registration is rejected, and “A” denotes the registration is accepted.

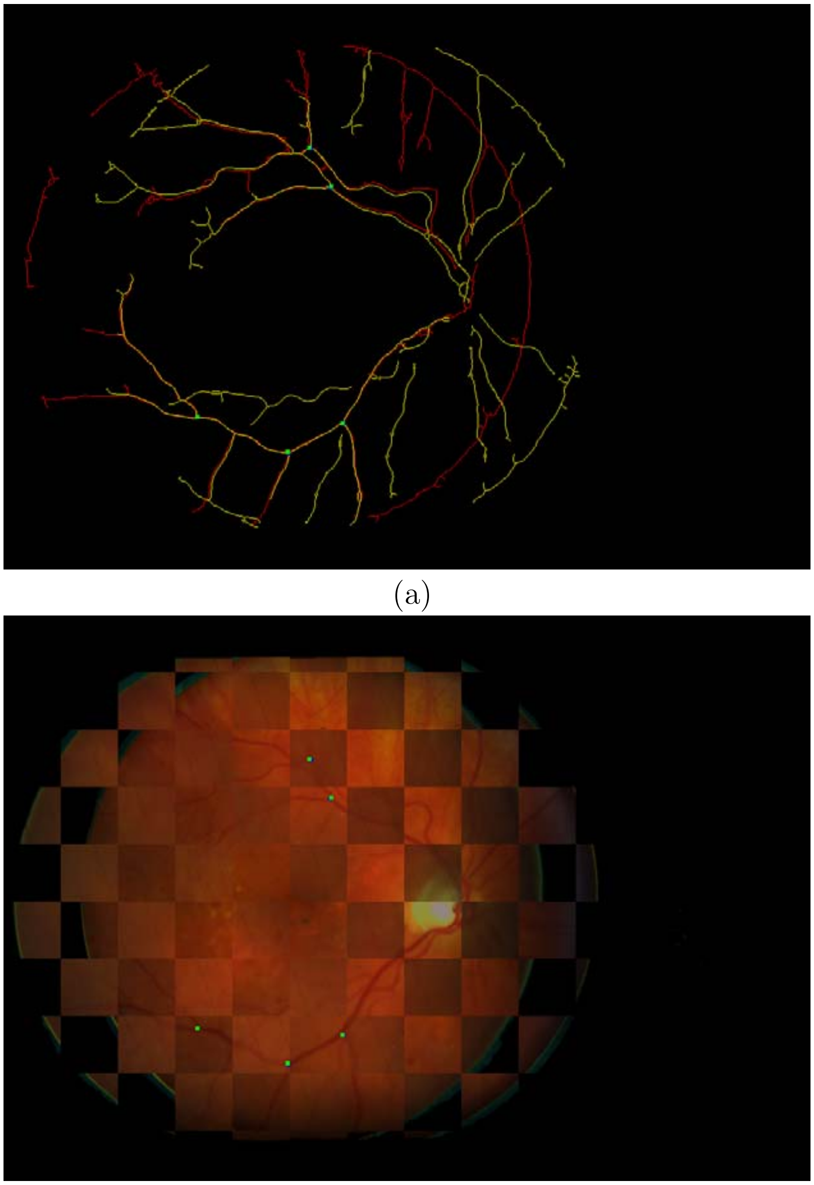
image	using coarse vessel map			using fine vessel map	
	median distance	score	decision	median distance	score
1	4.0	0.17	R	16.8	0.11
2	3.7	0.21	R	8.7	0.10
3	1.8	0.76	A	2.5	0.69
4	1.0	0.90	A	2.4	0.81
5	1.0	0.96	A	2.3	0.87
6	1.1	0.89	A	1.0	0.86
7	1.0	0.93	A	1.4	0.85
8	1.2	0.80	A	2.3	0.69
9	1.0	0.72	A	2.1	0.60
10	1.2	0.82	A	1.9	0.70
11	1.0	0.96	A	2.2	0.87
12	1.1	0.94	A	1.9	0.83
13	3.4	0.36	R	2.6	0.78

A main factor of failed registration is inconsistent detection of the bifurcation points, due to the nature of coarse detections of the vascular networks. False registration can be reduced by using a more robust and precise blood-vessel detection scheme like the one presented in Chapter II. Figure 15 shows an example of the superposition of the vessel centerlines in the reference and transformed images using the vascular maps detected by the algorithm presented in Chapter II. The superimposed map is relatively clean with little noise due to more precise detection of blood vessels. Obviously, computing costs increase with precision of the vasculature map algorithms. By switching between coarse and fine grained blood vessel maps, one can make flexible tradeoff between speed and accuracy of the registration process.

Another reason that may lead to false registration is due to significant similarities among local structures around registration points. Such significant similarities would lead to incorrect correspondence between feature points. This problem can be remedied by using additional reference structures, such as the centerline of the blood vessels or the location of optic disks, to constrain the correspondence, or using robust statistical techniques for feature extraction and matching [61]-[66].

D. Application to Detection of Laser Scars

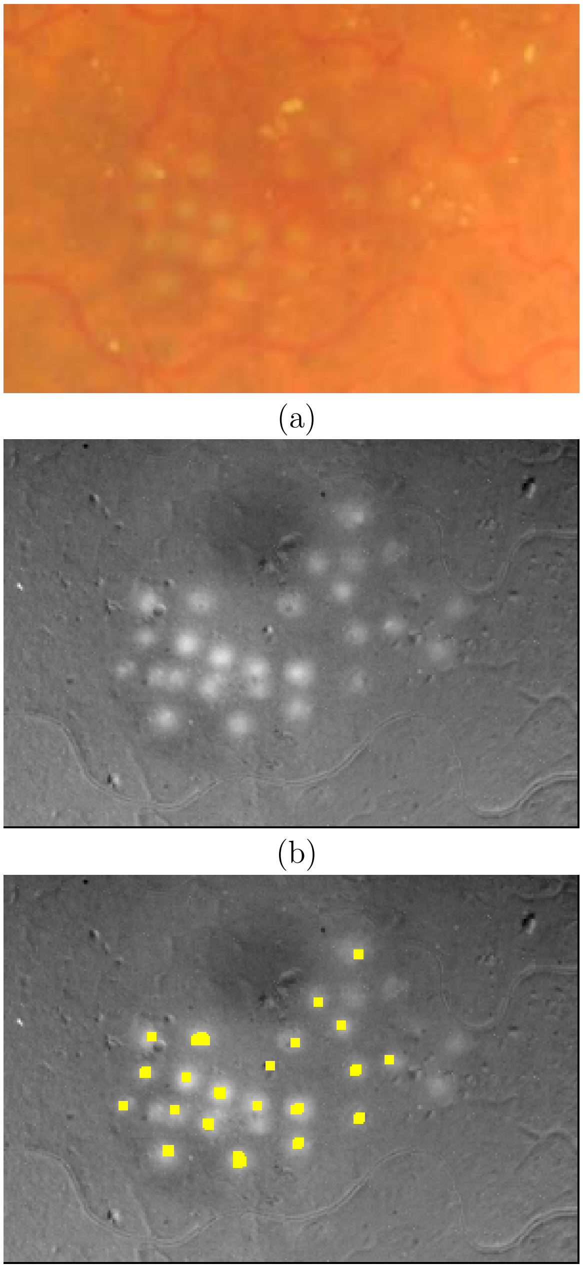
In this section, we show one application of retinal image registration. The registration algorithm is applied to the image pairs taken before and after laser surgery for detection of laser scars. The detection of laser scars in retinal images is importatnt as it can help verify surgery results, and inspect physicians for incidental laser exposure. However, the various intensity and contrast of laser scars in the images make the general detection techniques such as global thresholding and edge detection unreliable. The similarity between the lesions and laser scars in terms of color and shape



(a)

(b)

Figure 15: Registration result of the two images using their fine vascular maps: (a) the superposition, (b) the checkerboard.



(a)

(b)

(c)

Figure 16: Application of Registration result to detection of laser scars: (a) a cropped retinal image containing laser scars; (b) the difference map between pre- and post-laser-surgery images, (c) the detected laser scars in yellow color.

causes the problem of laser-scar detection more complicated [3][67][68][69], as shown in Figure 16(a).

The interference of noise and lesions on the detection of laser scars can be removed or mitigated by using registration technique. In our experiments, we applied the proposed registration scheme to the images photographed before and after laser surgery. On the common frame, each image is first processed by illumination equalization, and the intensity difference between two images is calculated. The difference map is then normalized and quantized. Figure 16(b) shows a cropped difference map between the images in Figure 12 after they are registered. In the figure, the laser scars show the higher contrast against background than in the post-surgery image. The other objects, such as blood vessels and exudates, are almost gone in the difference map. That is because they appear in the both pre- and post-surgery images with similar intensity and they are canceled out in the difference map. Thus, the difference map between the two aligned images taken before and after surgery is an ideal one on which one can further process for laser-scar detection.

Using the difference map, the detection of laser scars can be performed automatically by a computer-based algorithm. In the difference map, the laser scars have a roughly round shape and blur contour. The intensity of a laser scar is highest at the center, and it decreases gradually with increase in distance from the center, making the two-dimensional Gaussian model an ideal analytical model for a laser scar. Based on the observation, we develop a model fitting approach for laser-scar detection.

The Gaussian model is expressed by

$$z(p) = A \cdot G(p, \mu, \Sigma) + B, \tag{3.11}$$

where

$$G(p, \mu, \Sigma) = \exp[-\frac{1}{2}(p-\mu)^T \Sigma^{-1}(p-\mu)], \tag{3.12}$$

μ and Σ are the scar center and covariance matrix, respectively, A is the amplitude of the scar, B is the intensity of the background, and $z(p)$ is the estimated intensity at the pixel by the model.

To capture the laser scars, we first locate the pixels in the difference map that are local maxima of intensity. Then the regions around each local maximum are matched with the Gaussian template. Each patch is defined by the local square window centered at each local maximum. The parameters of the matched Gaussian template are estimated by the maximum likelihood method (MLE). Thus, we have

$$\hat{\mu} = \frac{1}{T} \cdot \sum_{i=1}^{n} [I(p_i) \cdot p_i], \tag{3.13}$$

where T is the total sum of the intensities of the pixels inside the window, and

$$\hat{\Sigma} = \frac{1}{T} \sum_{i=1}^{n} [I(p_i) \cdot (p_i - \hat{\mu})(p_i - \hat{\mu})^T]. \tag{3.14}$$

Let the cost function be the square errors between real data and the estimated Gaussian spot in each patch, A and B are chosen to minimize the cost function, and we have

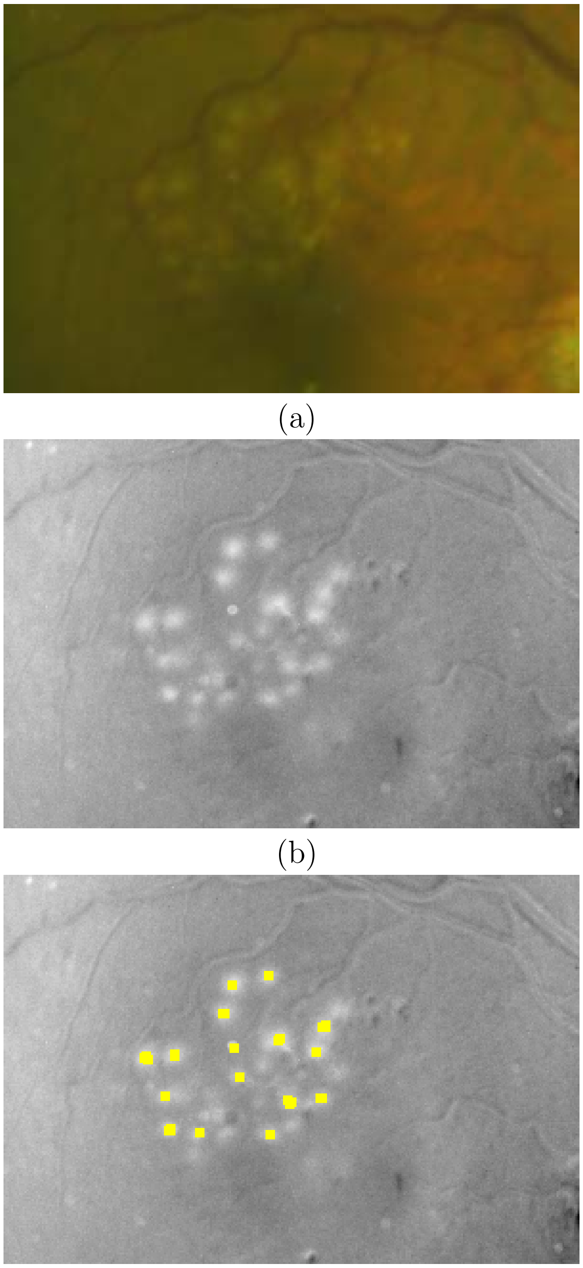
$$\hat{A} = \frac{\sum_{i=1}^{n} I(p_i) \cdot G(p_i; \hat{\mu}, \hat{\Sigma})}{\sum_{i=1}^{n} G(p_i; \hat{\mu}, \hat{\Sigma})} \tag{3.15}$$

$$\hat{B} = \frac{1}{n} \sum_{i=1}^{n} [I(p_i) - \hat{A} \cdot G(p_i; \hat{\mu}, \hat{\Sigma})] \tag{3.16}$$

The matching error is evaluated in terms of the measure

$$T_1 = \frac{\sum_{i=1}^{n} [I(p_i) - z(p_i)]^2}{\sum_{i=1}^{n} [I(p_i) - \frac{T}{n})]^2} \tag{3.17}$$

The size of each laser scar will be derived from the parameters. Figure 16(c) shows the detection results. The majority of laser scars in the image were captured by the model fitting algorithm. Another example of laser-scar detection is shown in Figure 17, where most laser scars (with different sizes) were detected.



(a)

(b)

(c)

Figure 17: Another example of detection of laser scars by registration: (a) a cropped retinal image containing laser scars; (b) the difference map between pre- and post-laser-surgery images, (c) the detected laser scars in yellow color.

CHAPTER IV

ROI-BASED WATERMARKING

A. Introduction

Digital watermarking is becoming increasingly important to protect authenticity and ownership of digital images [70][71][72][73][74]. Digital watermarks can be roughly classified as fragile or robust, based on their abilities to recover the watermark after the watermarked image is altered. As its name implies, a fragile watermark is made sensitive to image alternations, so that it becomes undetectable even after minute alteration is made to the marked image. The schemes proposed in [75][76] embed the (fragile) watermark in the least significant bit plane for perceptual transparency. The method proposed in [77] uses a two-component Gaussian mixture model to describe the statistical characteristics of the image and to generate the watermark. Robust watermarks are designed to tolerate moderate level of distortions, so that a partially damaged watermark can still be rendered for authentication or other purposes.

In most conventional watermarking techniques the watermark is embedded into an entire image for full coverage/protection [78][79][72], but it imposes some limitations on applications that are involved with multi-stage information processing, such as target identification. Even though the low intensity-level watermark added to the entire host image may appear to be unnoticeable to untrained eyes, it may interfere the analysis accuracy of computer algorithms or human readers (radiologists, ophthalmologists, etc.) For applications involved with analysis of image contents, one must exercise cautions in how and where to apply the watermark not to compromise the critical information of the host image. In an informal test, a professional human reader picked up small watermark blocks at the intensity level of 5, while regular readers could not notice them.

Selective watermarking [80] is more effective than full-image watermarking in protecting the authenticity of images that may go through legitimate, multiple steps of processing. Most image analysis applications have specific object types that are considered to be regions of interest (ROI) of the analysis, i.e., analysis-ROI. The remaining areas are less critical to the analysis function, and thus can be used for watermarking, i.e., the watermark-ROI. This way, one or more watermarks can be embedded to the watermark-ROI, and the integrity of the analysis-ROI can be protected by digital signature or similar hash functions. The user can make flexible tradeoff between the sizes of analysis-ROI and watermark-ROI to balance analyzability and authenticity protection.

Generally speaking, three major issues need to be addressed in the design of a watermarking system. The first issue is regarding the generation of watermark. A watermark is typically a small pseudo-random sequence that may follow the Gaussian [81], uniform, or bipolar distribution [82][83]. The second issue is on how to embed the watermark into the host image. Watermark embedding can be done in the spatial domain [84][85] or a transform domain, such as the discrete cosine transform (DCT) [81], discrete Fourier transform (DFT) [79], or discrete wavelet transform (DWT) [72]. The third issue is on how to recover the watermark from the watermarked image. Watermark verification is usually done by testing the similarity between the original and recovered watermarks using correlation [85] or hypothesis test [84], with or without the presence of the original host image.

For selective watermarking, one could use shapes, sizes, or other image features to divide the host image into the analysis- and watermark-ROI. A recent work on selective watermarking is reported in [86], where feature points were extracted from the host image by using the Mexican-Hat wavelet to partition the image into a Voronoi graph. Each region is independently watermarked by spread-spectrum modulation. Another scheme reported in [85] divided an image into triangles by applying Delaunay tessellation to feature points extracted by Harris detector. A triangle-shaped watermark is transformed to the exact shape of each triangle and then embedded into it. The watermark is later verified by a statistical estimator.

The region-based watermarking method proposed in [84] makes the watermark immune to geometric transformations by extracting multiple regions using the *iterated conditional modes* method. The watermark is embedded to the bounding rectangle of each region after rotation. The presence of watermark is verified by using the maximal response of the hypothesis test over all regions, plus a threshold-based decision rule. False acceptance/rejection ratios of this method are both low when watermarked images are attacked by popular image processing operations and geometrical distortions.

Multi-purpose watermarking is an emerging area of research. In [87], two watermarks are embedded to the Y channel for authentication, and to the I and Q channels for compression, of a color image. In [88] an MD5 signature computed on a disease region is encrypted with the patient information to form a digital envelope, whose bit stream is embedded into the least significant bits of selected pixels. Image integrity is verified by comparing the signature stored in the digital envelope against the one computed from the image.

The method proposed in [86] exploits both spectral and spatial information to embed robust watermark on user-defined ROI. The method in [89] protects the integrity of an ROI by embedding the watermark around the ROI. The watermark bit stream is generated by progressive coding of a signature image, which is then compressed by a hierarchical segmentation scheme.

Common attacks against a watermarking system include pixel level alternation (e.g., linear or nonlinear filtering, JPEG compression, noise addition) [90][91][92], breaking the synchronization between the image and the watermark (e.g., rotation, shear, cropping) [93][79], producing fake original or watermarked images (e.g., inversion attack [94]), or removing the watermark (e.g., collusion attack [95]). Many more exotic attacks can be added to the list, yet for practical purposes most watermarking techniques proposed in the literature focused on robustness against compression, noise addition, low-pass filtering [90][91][92], and geometric distortions [79].

In this chapter, we propose a selective watermarking system for image authentication. We use the vascular network in the retinal image as the watermark-ROI to order and embed watermarks. Automatically detected blood vessels are used for synchronization of embedding and recovery of the watermark. Two different ordering techniques are developed to compare their robustness against different attack schemes. Watermark is detected by a blind detector [96] without using the original image. Selection and marking of the watermark-ROI, i.e., blood vessels, can be done in the spatial domain, or other domain like discrete wavelet transform (DWT,) but for simplicity we will only consider the spatial domain here.

Different from the schemes proposed in [84][85], where the watermark is marked in each region, we use selected linear segments across the image as the watermark-ROI, so that analytical-ROI, which can be lesion such as exudates, hemorrhages, etc, can be marked independently. It allows flexible protection of specific areas in multi-stage processing without compromising the watermark. Testing results show that our scheme is robust against common image processing and geometric attacks.

The rest of this chapter is organized as follows. Section B describes the ROI detection and synchronization. Section C describes the watermark embedding and detection. In Section D, the experiment results on testing images are presented.

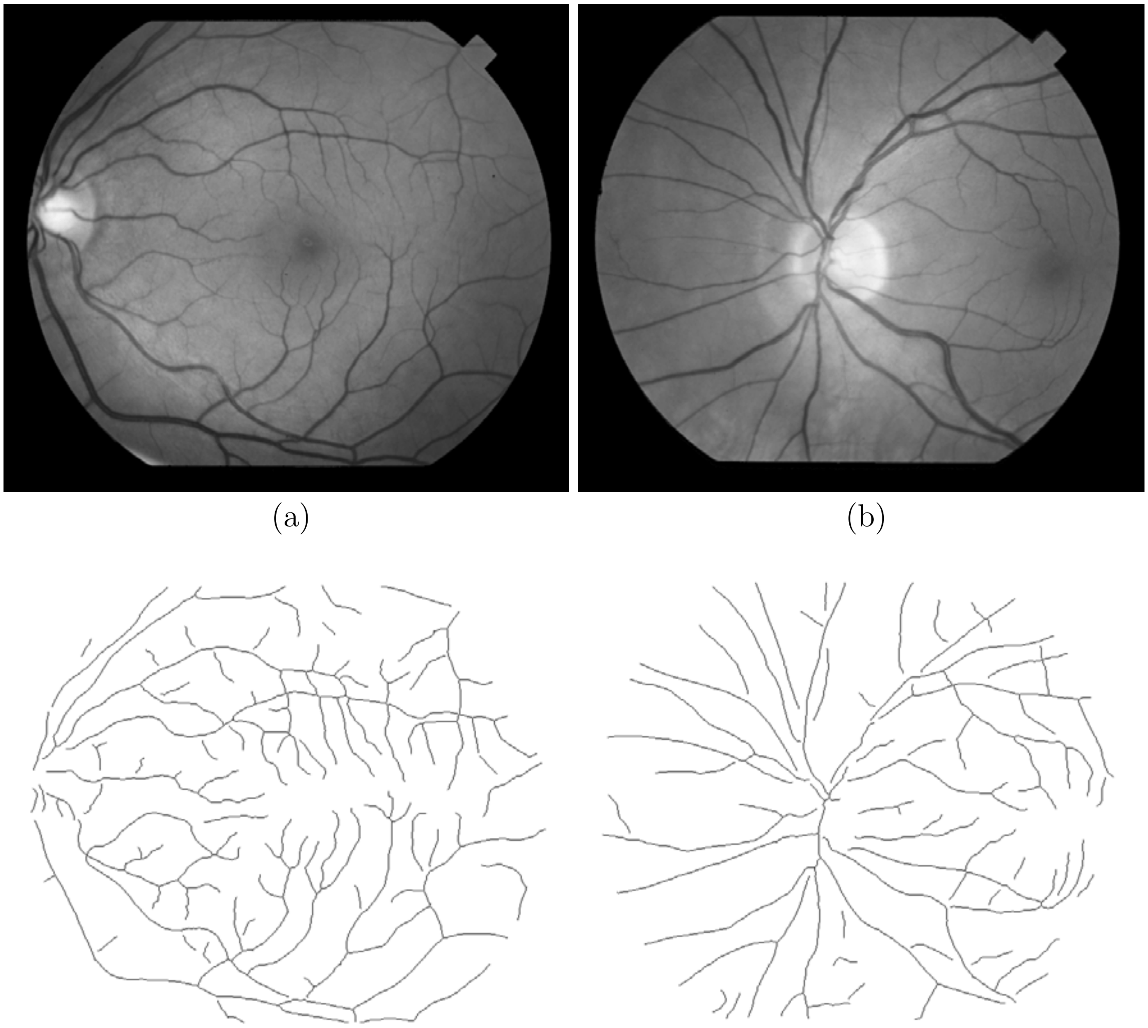
B. ROI Detection and Synchronization

Blood vessels are usually broadly dispersed across an image, but occupy only a fraction of the image area. We will discuss how to automatically locate, order, and embed/recover the watermark on the vascular network of retinal images, so that the vascular network and their surrounding areas can be used as either analysis-ROI or watermark-ROI. In our subsequent discuss, we use the two training images in Figure 18 to examine algorithmic steps. Then, eight additional testing images, all of them obtained from the STARE retinal image database [7], will be used to test the performance of our scheme.

The first step of watermark embedding is extraction of blood vessels. Then, detected blood vessels are divided between bifurcation points, and ordered in a rotation-invariant fashion. Following the ordering sequence, watermarks are embedded into the chosen flow segments. For watermark detection, blood vessels are first segmented and ordered in a similar manner, and then a statistical test is used to detect presence of the watermark.

1. Blood Vessel Segmentation

Blood vessel segmentation, and ordering of the segmented vessels are required for both watermark embedding and recovery. First, blood vessels are extracted by the algorithm proposed in Chapter II. Then, the center lines of extracted blood vessels are located by a thinning algorithm [58]. For ordering of blood vessels, the bifurcation points of the central lines are removed, so that vascular central lines are divided into disconnected flows. The length of each flow is defined as the sum of Euclidean distance of consecutive pixels on the centerline of the flow, and the width of a flow is defined as the average diameter of the cross-sections of the actual blood vessel at the



(a) (b)

(c) (d)

Figure 18: (a)-(b) Two sample retinal images (077 and 163); (c)-(d) the centerlines of the detected vascular networks of the two sample images.

flow.

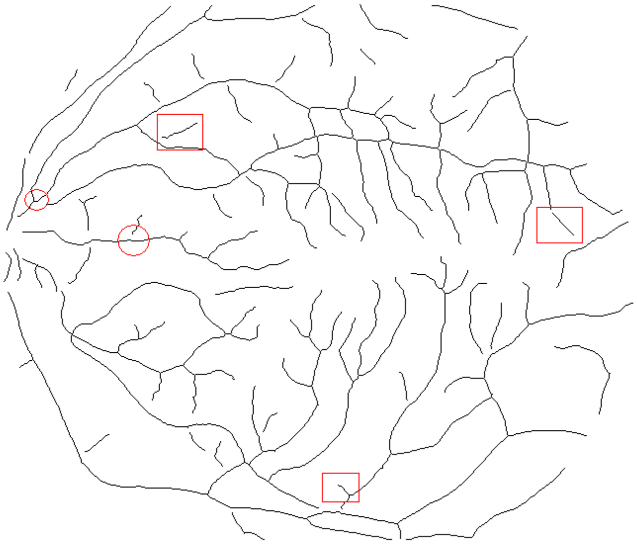
The performance of blood vessel segmentation is affected by geometric distortions and image processing routines. For instance, the two graphs in Figure 19 are the blood vessel maps detected from an image before and after it is compressed by JPEG routine, where the highlighted areas are the inconsistent detection outcomes due to compression. Segmentation inconsistencies are inevitable, especially for small blood vessels, when the original image is altered. As a result, we will only use relatively large blood vessels for watermarking, to minimize the effects of segmentation variations. For the tested images with size 700×605, only the blood vessels whose diameters are equal to or larger than 3 pixels are selected for watermarking.

To examine the effects of different distortions on the segmentation outcomes, we adopt the score function in [85] for robustness evaluation,

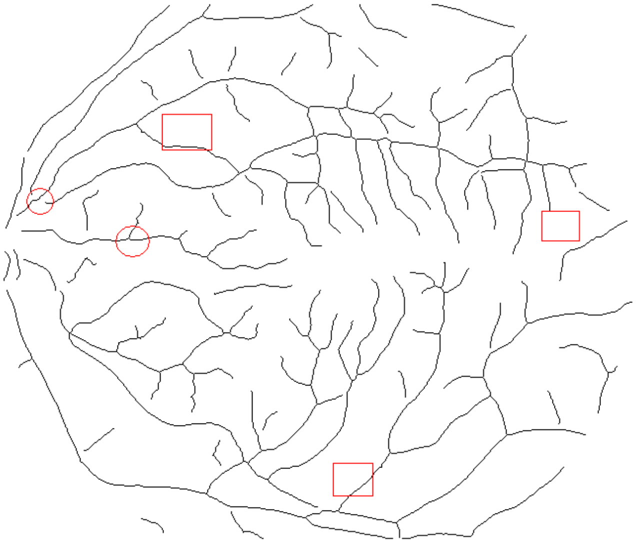
$$g = \frac{N_{pre} - (N_{add} + N_{ls})}{N_{ini}} \tag{4.1}$$

where N_{ini} denotes the number of pixels on the central lines of the segmentation map of the original image, and N_{add}, N_{ls}, N_{pre}represent the number of pixels that have been created, destroyed and preserved after distortion. The score function $s = 1$ when $N_{cre} = N_{des} = 0$, but it decrease to be smaller than one after the image is altered. Nineteen different processing routines, i.e., additive noise (salt-pepper and Gaussian), 3×3 low-pass filters, JPEG compression and geometric distortions (scaling and rotation) are applied to the two training images in Figure 18, and the results are given in Table V.

As shown in Table IV, the segmentation scores under these distortions range from 0.72 to 0.96, and from 0.77 to 0.96, respectively, for the two training images, where scaling has the most significant effects among all distortion effects, because



(a)



(b)

Figure 19: The segmentation inconsistency due to the distortion: the highlighted area by rectangles and circles show the difference in the segmentation maps of an original image and the respective JPEG-compressed image with quality factor 80.

Table V: Segmentation scores under different distortions.

Distortion	Sample image 1	Sample image 2
salt-pepper noise (21.8dB)	0.72	0.79
salt-pepper noise (28.0dB)	0.88	0.92
Gaussian noise (33.6dB)	0.82	0.89
Gaussian noise (40.5dB)	0.90	0.94
mean filtering	0.93	0.94
median filtering	0.94	0.96
Gaussian filtering	0.96	0.96
JPEG90	0.95	0.95
JPEG80	0.92	0.94
JPEG70	0.90	0.93
JPEG60	0.92	0.94
5% downsized	0.91	0.91
10% downsized	0.86	0.89
15% downsized	0.81	0.80
20% downsized	0.77	0.77
10^o rotation	0.90	0.91
20^o rotation	0.92	0.96
30^o rotation	0.90	0.95
40^o rotation	0.93	0.92

change of sizes of blood vessels (after scaling) directly affects the parameter setting of the detection filter. Otherwise, the blood vessel detection is highly robust against low-pass filtering, JPEG compression and rotation. It is observed that even when the score drops to 0.72, shapes and relative locations of blood vessels change little.

In addition to consistency in segmentation of large vessels, the location of the centroid of all centerline pixels was found to be relatively invariant to the distortions. We evaluated the sensitivity of displacement of the centroid with respect to the number of centerline pixels used in calculating the centroid. The displacement of the centroid, i.e., the distance between the centroid of the original image and that of a distorted image, is calculated for both training and testing images, and for each of the distortions under consideration. The pixels are selected by choosing one pixel from every t consecutive pixels on the centerlines, where t varies from 1 to 6. Table VI shows that for all the images, the average displacements over the distortions are small, less than 3.3 pixels. This property is used to design a rotation-invariant, scaling-invariant ordering scheme of flows.

2. Delaunay Tessellation Ordering

The objective of the flow ordering is to synchronize the embedding and recovering of the watermark. Knowing that detection of bifurcation points, especially those connected to relatively small blood vessels, are susceptible to noise distortion, the spanning tree based technique is not considered. Furthermore, selection of image features for rotation-invariant, scaling-invariant watermarking needs to reliably identify the same set of watermarking locations after attacks. Obvious object features, such as length or width of blood vessels cannot be used for this purpose, because of the close resemblance of blood vessels.

Instead, we propose a global and a local flow-ordering algorithm, using flow

Table VI: Average displacement $(\Delta x, \Delta y)$ of centroid over all distortions.

Image\\t	1	2	3	4	5	6
077	1.10,-1.33	1.19,-1.35	1.24,-1.36	1.13,-0.90	1.16,-1.03	0.89,-1.16
163	-0.07,-3.26	0.04,-3.06	0.15,-3.33	0.15,-3.33	0.31,-3.18	0.35,-2.96
001	0.13,1.68	0.01,1.42	-0.10,1.95	0.16,1.29	-0.21,2.32	-0.61,1.67
081	0.78,-2.04	0.71,-2.08	0.92,-2.14	0.38,-2.31	1.10,-1.99	1.15,-2.66
082	0.69,-1.68	0.69,-1.62	0.53,-2.12	0.67,-1.58	0.74,-1.36	0.90,-1.89
162	2.00,-1.89	1.91,-1.82	2.03,-2.10	1.97,-2.19	1.61,-1.88	1.72,-1.92
235	0.55,-0.29	0.64,-0.38	0.20,-0.01	0.98,-0.34	0.58,-0.09	0.76,0.26
236	1.34,-0.81	1.42,-0.83	1.24,-0.92	1.39,-1.27	1.62,-1.11	1.51,-0.72
239	0.33,-1.57	0.44,-1.82	0.04,-1.42	0.58,-1.94	0.23,-1.83	0.33,-1.78
255	-0.99,2.48	-0.95,2.43	-1.07,2.39	-0.98,2.24	-0.87,2.28	-0.86,2.31

lengths, widths and relative locations of flows as their ordering criteria. Through extensive experiments, we observed that relative locations of flows remain highly constant under different distortions, where the flow location is defined as the middle point of a flow. As a result, in the first (global) order scheme we adopt the well known *Delaunay tessellation* (DT) [97] to order the flows during embedding and recovery of the watermark. In the local ordering scheme, a set of random points are generated by a key. Following the generating order of these points, flows in areas adjacent to these points are then selected based on certain flow features.

In addition to its low computing cost [98], the tessellation configuration does not change if the vertex displacement is within its stability area [99]. Furthermore, the Delaunay tessellation is unique if cocircularity does not occur to P [97], where P is the set of nodes being connected. Cocircularity refers to the condition that the existence of a circle that intersects with at least four points of P, and remaining points of P are outside of the circle. This condition can be easily eliminated, because a middle point can be replaced by its adjacent pixel on the flow to disrupt the cocircularity with little effect on the Delaunay tessellation of flows. DT ordering can be easily implemented as follows.

Let P denote the set of flow locations of n longest flows in the image, plus the node o, which is the centroid of the centerline pixels of all flows. After the Delaunay tessellation, all points in p are interconnected by a number of non-overlapping triangles. Next, we order flows based on the following rules, using an array D and a stack E to record the vertices and edges of triangles, respectively.

1) 1) Draw a line L to connect o with the middle point of the longest flow. Starting from o, find its adjacent triangle T_1 that intersects with L;

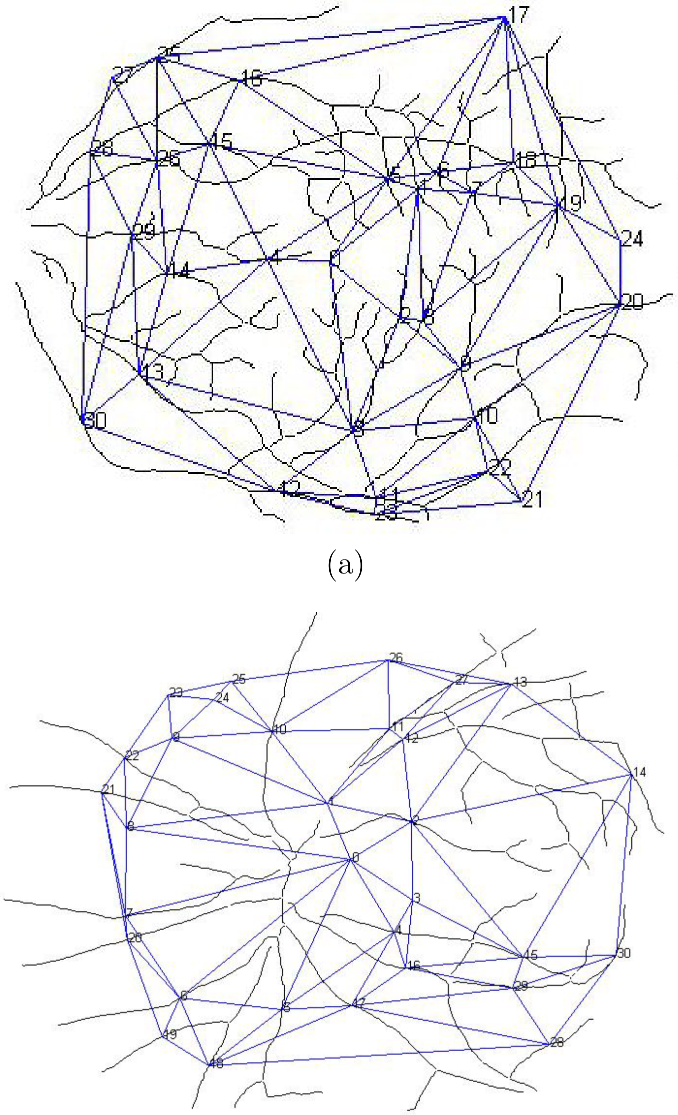
2) Add the other two vertices (except o) of T_1 into D, and the three edges of T_1 into E in the counterclockwise order with respect to o. The chosen vertices and edges are marked so that they cannot be added to D and E again later.

3) Remove the edge e on the top of E. If e is shared by another triangle that has not been completely marked, add the un-marked vertex of the triangle into D, and add the un-marked edges into E in the counterclockwise order.

4) If E is not empty goes to Step 3).

5) Vertices in D represent a rotation-invariant ordering of the selected flows.

An example of DT-ordering is illustrated in Figure 20. The triangles represent the Delaunay tessellation, and the number around each vertex is the index of the flow after ordering. Here, each vertex is the middle point of a selected flow, except for vertex 0, which is the centroid of all pixels on the centerlines.



(a)

(b)

Figure 20: DT-ordering results for the two training images: the triangles are generated by the Delaunay triangulation. Each vertex is the middle point of a selected flow, except vertex 0 that is the centroid of all pixels on the centerlines. The number around each vertex is the index of the flow after sorting. For good visual effects, 30 longest flows are selected in this example.

3. Randomized Flow Ordering

Relative locations of the vascular flows remain highly constant under different distortions. First, the location of the centroid, i.e., the red dot located at center of Figure 21, of all center-line pixels is highly invariant to distortions. In our experiments, the average displacement of the centroid location under different distortions is less than 3 pixels. Second, the covariance matrix of all centerline pixels is also relatively constant to the distortions. As shown in Figure 21, the red ellipse on each image illustrates the covariance matrix, whose orientation and size are determined by the eigenvectors and eigenvalues of the matrix, respectively. For most non-rotation distortions, the ellipse of the distorted image has nearly identical orientation and size as that of the original image, which is shown in Figure 21(b). For rotations, the ellipse of the distorted image is also rotated with the same degree as the image, but its size remains unchanged, as shown in Figure 21(c). In fact, the ellipse represents the spatial distribution of the pixels on a vascular network, and the orientation of the ellipse always points to the same location of the vascular network.

The covariance matrix is a widely used descriptor of a geometric structure. That is, the location covariance matrix of the vascular network can be calculated by

$$C = \frac{1}{M}\sum_{i=1}^{M}[(p_i - o)(p_i - o)^T], \tag{4.2}$$

where C is a 2×2 matrix, M is the number of all centerline pixels, and p_i and o are respectively coordinates of pixels on the centerlines and the centroid, expressed as column vectors.

The two eigenvalues of C, λ_1, λ_2 (assuming $\lambda_1 > \lambda_2$), and their corresponding eigenvectors, ϕ_1, ϕ_2, can be used to construct a rotation-invariant coordinate system. ϕ_1 and ϕ_2, commonly called the principal directions [100], are perpendicular to each

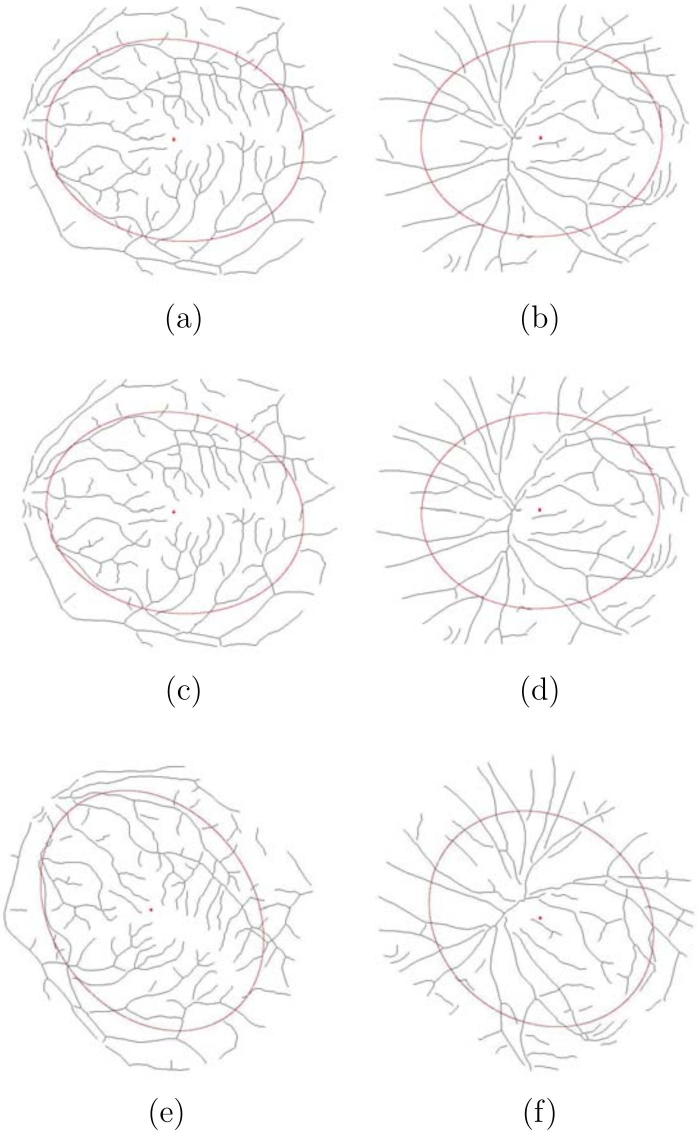

(a) (b) (c) (d) (e) (f)

Figure 21: Geometric structure of the vascular network for (a-b) the original training images, (c-d) the Gaussian-filtered images, and (e-f) the rotated images. The red dot near the center of each image denotes the centroid of the centerline pixels. The covariance matrix is illustrated by the red ellipse, whose orientation and size are determined by the eigenvectors and eigenvalues of the matrix.

other. Let θ be the angel between ϕ_1 and x-axis. We build a new coordinate system by taking o as the origin, the direction pointed by ϕ_1 as x-axis, and one perpendicular direction as y-axis. The directions of x-axis and y-axis follow the right-hand rule so that the z-axis points to the reader.

Each random point is first translated and scaled by

$$\begin{aligned} x'_i &= (2x_i - 1) \cdot \lambda_1 / l \\ y'_i &= (2y_i - 1) \cdot \lambda_2 / l \end{aligned} \tag{4.3}$$

where l is a scale factor so that the dynamic range of (x'_i, y'_i) is within the image size. Then, a transformation (rotation plus translation) with respect to the new coordinate system is performed by

$$\begin{aligned} x''_i &= x_0 + x'_i \cos(\theta) - y'_i \sin(\theta) \\ y''_i &= y_0 + x'_i \sin(\theta) + y'_i \cos(\theta) \end{aligned} \tag{4.4}$$

The transformation generates a sequence of locations, denoted by $\{q_i = (x''_i, y''_i), i = 1, 2, \ldots, n\}$. When an image is geometrically transformed, so are the centroid point and the covariance matrix. Thus, the locations determined based on o and C remain invariant to the structure of vascular network in the image.

Let A_i denote the circular area centered at q_i with radius r. Among the flows that pass through A_i, the flow can be selected for ordering based on one of the following rules:

1) *rand-nearest*: Select the flow that is nearest to q_i, where the distance of a flow to q_i is the minimum of the Euclidean distances of all centerline pixels of the flow to q_i.

2) *rand-longest*: Select the longest flow that passes through A_i.

3) *rand-widest*: Select the widest flow that passes through A_i.

4) *rand-most*: Select the flow that has the most pixels in A_i.

It is possible that no flows pass through A_i. In this case, q_i is recorded. After all locations are processed, the neighborhood for each recorded location is extended to the entire image, and the same rule is applied to the recorded locations for assigning flows for them. We note that by setting the initial radius smaller than the image size limits the scope of incorrect ordering due to image distortion. This is an important factor that contributes to the better performance of this scheme than that of DT-ordering.

4. Performance Comparison of Flow Ordering Algorithms

We used the two training images as benchmark to test the performance of a) DT-ordering method combined with flow length (global-length) and flow width (global-length), and b) randomized ordering schemes in conjunction with the four flow ordering rules, under distortions mentioned earlier. Synchronization ratio (s-ratio), which is defined as the number of correct orders divided by the total number of selected flows, is the primary performance measure. Here, a particular order is considered correct if a flow is selected in both the original and distorted images with the same order in the set of selected flows.

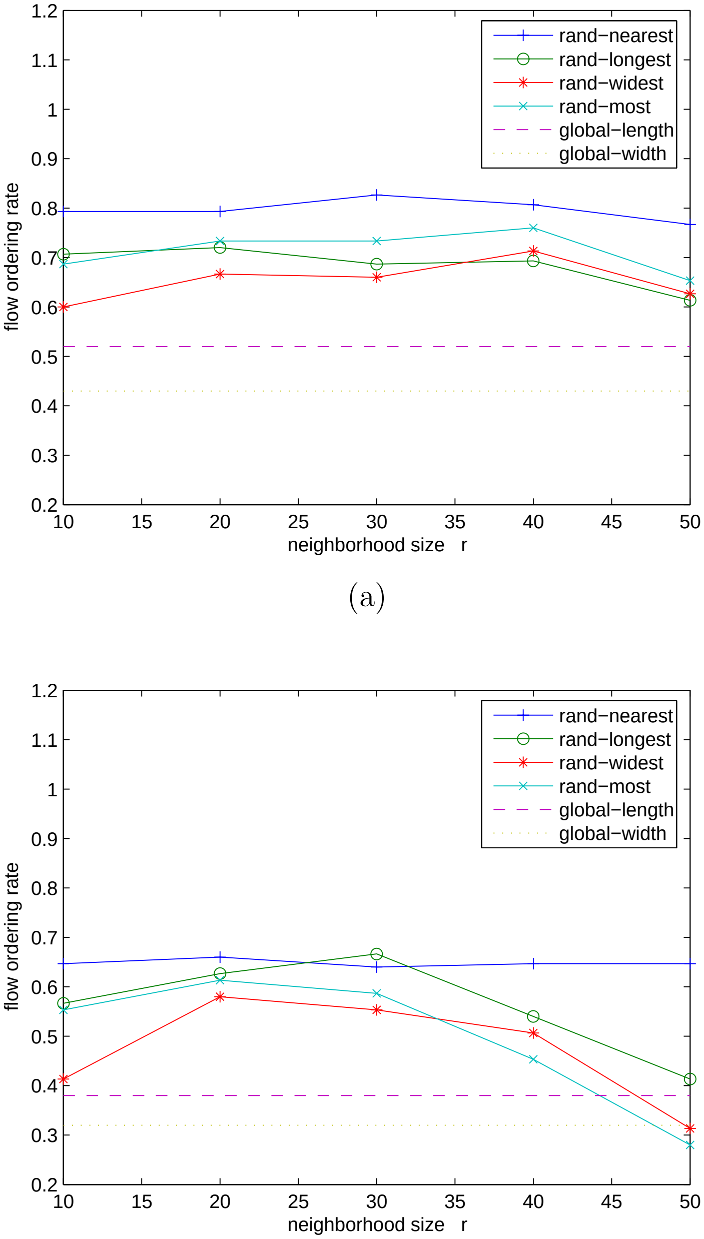
The s-ratios for $n = 50$ are plotted in Figure 22, where the x axis represents the value of r, and the y axis represents the average s-ratios over all distortions, where the s-ratios of the global ordering schemes is independent of the r value. It is clear from Figure 22 that s-ratios of all four randomized ordering algorithms are higher than that of the two DT-ordering algorithms, because in DT-ordering misordered flows cause mis-ordering of remaining flows. Among the four randomized ordering rules, rand-nearest performs best, and rand-widest worst. The rand-nearest rule attains the higher s-ratios (greater than 0.8 for the first image, and around 0.65 for the second one) when r ranges from 20 to 40. It is thus chosen for design of the watermarking scheme.

C. Watermark Embedding and Detection

In this section, we present the watermark embedding and detection routines using a watermark that consists of a sequence of uniformly distributed random 1/0 bits, whose seed is protected by a key k_2. The total number of watermark bits, the number of selected flows, and the number of bits on each flow are user inputs. Locations of the m blocks on a flow can be randomized using a secret key k_3, but for simplicity we assume that watermarking locations are evenly distributed on a flow.

Each watermark bit is encoded by modifying the intensities of pixels in a neighborhood of the determined location. To reduce visibility of watermark, the neighborhood should have a similar shape with flows or blood vessels. In this work, we choose a rectangular block to embed a watermark bit in the vessel, with two orientations of the block parallel with and perpendicular to the local flow direction. The dimension of the block is designed to be adaptive to the local flow width. The intensity of each pixel inside the block is updated according to the value of the watermark bit. That is, if the bit is 1, the intensities of pixels will be increased, otherwise, decreased.

A simple approach of updating each pixel's intensity by the same constant value would make the watermark visible, due to the abrupt intensity change from blocks to un-marked areas on the flows. To reduce such changes, we can make the intensity change smoothly on blocks and from blocks to adjacent areas on the flow. One implement, as did in our work, is to make the intensity change of pixels in the block following a Gaussian template. Here, the intensity change at the center of block is maximal, and it decreases gradually toward the boundary. The two-dimensional



(a)

(b)

Figure 22: Performance comparison over the four local ordering algorithms and two global ordering algorithms for the two training images.

Gaussian kernel of size $b_x \times b_y$ can be expressed by $\Delta I(x, y) = \alpha \exp(-\frac{x^2}{2\sigma^2})$, where $\sigma = 0.3b_x$. The kernel is then translated and rotated so that the x-axis orientation of the kernel is parallel with the flow direction, and the kernel is centered at the determined location.

The block size, or the Gaussian kernel size, need to be configured to balance the tradeoff between the invisibility and recovery accuracy of the watermark. Small block size results in the rapid intensity change on the block, making it distinct from unmarked areas on the same flow. Large block size allows the smooth intensity change on the block, at the cost of lower watermark embedding rate (the ratio of watermark size to the size of host image) or even interference to the adjacent blocks.

In addition to the visual performance, we also need to consider the effect of block size on the recovery performance of watermark, where the recovery performance is defined as the percentile of correctly estimated watermark bits. We tested various block sizes with b_x ranging from d to $4d$, and b_y varying from 1 pixel, $0.5d$, to d, where d is the local flow width. The experiments on the two training images, using different α values and the two watermark recovery techniques (that will be introduced shortly), show that the maximal recovery performance is achieved when $b_x = 3d$ and $b_y = d$. Hence, such configuration of the block size is used in the following experiments.

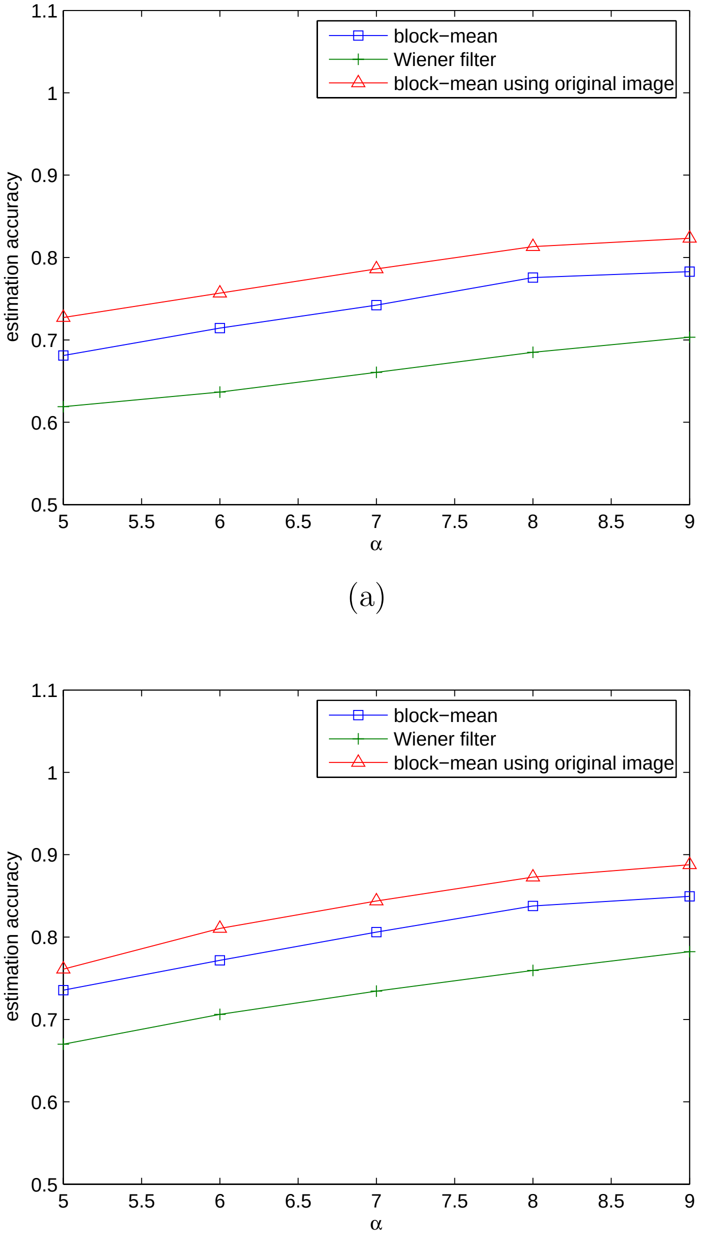
Watermark detection follows a process very similar to that of watermark embedding, except for the step of recovering/detecting the watermark in each synchronized block. Here, k_2 is used to regenerate the expected watermark bit sequence, and k_3 is used to locate blocks in each selected flow. A simple majority rule can be used to detect the embedded watermark bit on each block when the original image is available. That is, for each watermarked block, the watermark bit is estimated to be 1 (0) when the intensities of the majority of pixels in the block of the testing image are larger (lower) than those in the original one.

The design of a blind detector, i.e., detecting watermark bits without using the original image, is more complicated. The key issue here is how to estimate certain properties of the original image, so that it can be used by the watermark recovery algorithm reliably. Here, we propose a simple intensity estimation technique, block-mean, and compare its performance with the well known Wiener filter that is widely used in the watermark literature [101][102]. Wiener filter is based on the principle of minimizing the mean square errors between the restored images and true images, and it is applied to an entire image without considering the differences between background and foreground information in our experiments.

The block-mean algorithm is based on the observation that in the green channel, the pixel intensities along the vessel cross section is approximately a Gaussian distribution, and the distribution does not change rapidly along the flow direction [5][22][24]. As a result, we can estimate the flow intensities on the flows (of the original image) from the testing image by taking the average of the intensity of all blocks centered at every pixel of the centerline.

We compared the watermark recovery performance of the two estimation techniques for the two images in Figure 18, after they are distorted by the different distortions listed in Table V. Results on distorted images are given in Figure 23, where the recovery performance is defined as the average percentile of correctly estimated bits when the flow ordering is free of any error. The watermark recovery performance using the block-mean technique on the original image is also presented here as a comparison reference.

From the results shown in Figure 23, one can see that for the two blind detectors, the block-mean algorithm outperforms Weiner filter. This is attributed to the fact that block-mean uses the information on the same flow to estimate the block intensity, while Wiener filter simply uses the entire image for restoration, without differentiation



(a)

(b)

Figure 23: Performance comparison of the watermark recovery schemes based on block-mean and Wiener filter.

between the foreground (blood vessels) and background. In addition, the performance of the mean-block algorithm based on the testing image is very close to that based on original images. Therefore, we adopt the block-mean technique for the rest of discussion due to its better performance and much lower computing overhead.

Obviously, the estimation accuracy increases with the α value, but at the cost of increased watermark visibility. When $\alpha = 9$, the watermark bits become noticeable to naked eyes, see Figure 24(b). After some experiments, we chose the block-mean based watermark estimation method with $\alpha = 6$, by which the watermark is nearly invisible, see Figure 24(c), and the watermark recovery performance is approximately 77% for the rest of discussion.

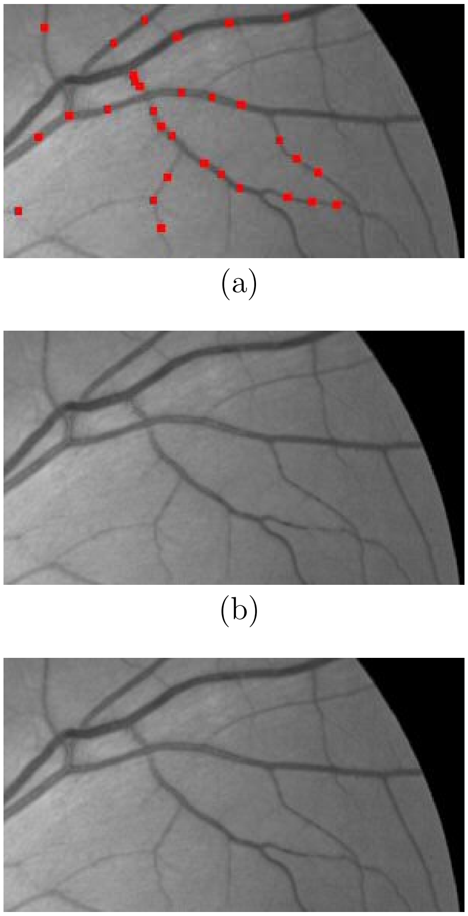
The final authentication decision is made by comparison of the detected watermark bit sequence with the original one. If the similarity value is greater than a chosen threshold, the testing image is considered to be authentic. Otherwise, inauthentic. The similarity between two watermarks is measured by their correlation coefficient [85][81].

$$s(w, \hat{w}) = \frac{w \cdot \hat{w}}{\sqrt{\sum_i w_i^2 \sum_i \hat{w}_i^2}} \tag{4.5}$$

D. Experimental Results

We have implemented and tested our scheme in Matlab that runs on Microsoft Windows XP. The two sample images in Figure 18 were used as the training images. In addition to the optimized parameters mentioned earlier, the other key parameters are $n = 50$, $m = 3$. That is, fifty flows are selected for watermarking, each of which contains three watermark bits.

The authenticity of a watermarked image is measured by using the likelihood ratio test (LRT), because it has the lowest probability in decision error [100]. Let c_1



(a)

(b)

(c)

Figure 24: Watermark embedding: (a) the red squares present the locations where the watermark bits are embedded. (b) The watermarked image with $\alpha = 9$. (c) The watermarked image with $\alpha = 6$.

and c_2 denote the inauthentic and authentic classes, respectively. If $P(c_1|s) > P(c_2|s)$, the image is considered to be inauthentic; otherwise authentic, where $P(c_i|s)$ is the posterior probability of similarity for c_i. We used a training process to optimize the decision threshold for the s values as follows.

One-hundred randomly generated watermarks were applied to each of the two training images, and distortions were applied to the two training images, before and after watermarking. The training set of the inauthentic class (c_1) consists of s values generated by the watermark recovery routine on the distorted, un-watermarked images. The training set of the authentic class (c_1) consists of the s values generated by the watermark recovery routine on the distorted, watermarked images.

The histograms of the s values for the two classes are shown in Figure 25. Noting that overlapping of c_1 and c_2 indicates that the decision error is inevitable for any threshold value of s. Assuming that c_1 and c_2 have an equal prior probability, and let the distributions of s for c_1 and c_2 be approximated by normal distributions, with parameters estimated as $\mu_1 = 0.33$,$\sigma_1 = 0.075$, and $\mu_2 = 0.58$,$\sigma_2 = 0.116$, respectively. By Bayes' rule, we have

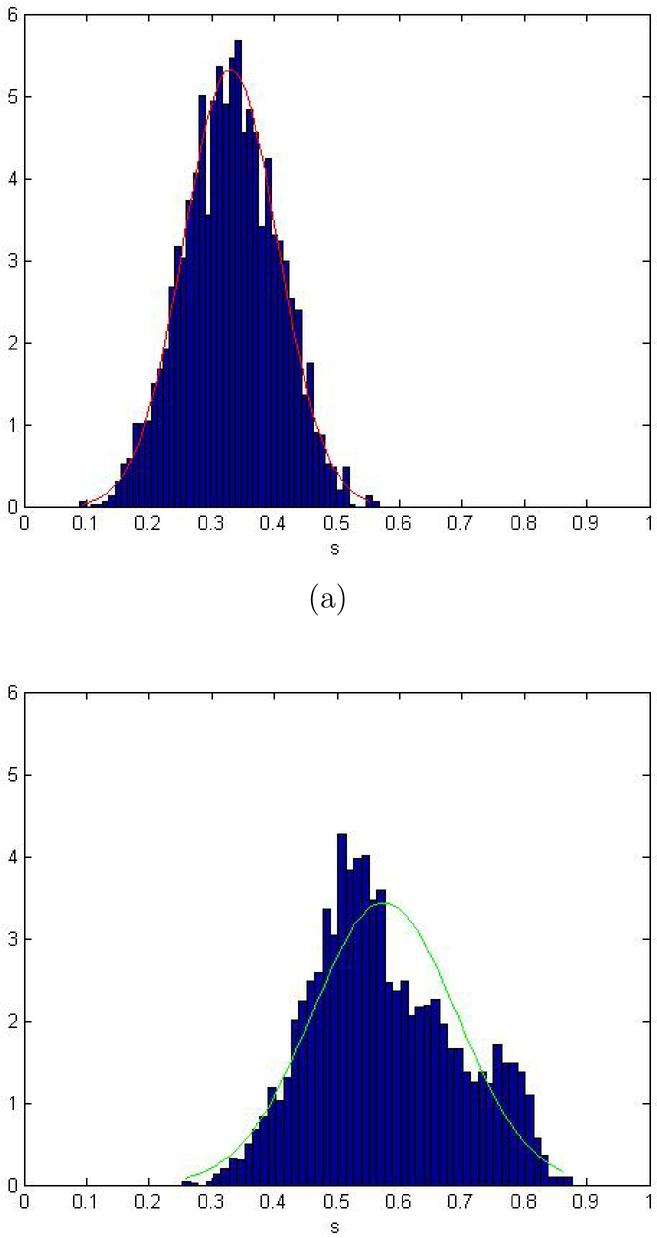
$$P(c_i|s) = \frac{P(s|c_i)P(c_i)}{P(s)} \tag{4.6}$$

Thus, the LRT decision rule becomes: if

$$\frac{1}{\sigma_1} \cdot \exp[-\frac{(s-\mu_1)^2}{2\sigma_1^2}] > \frac{1}{\sigma_2} \cdot \exp[-\frac{(s-\mu_2)^2}{2\sigma_2^2}], \tag{4.7}$$

the testing image is decided to be inauthentic, otherwise it is decided to be authentic.

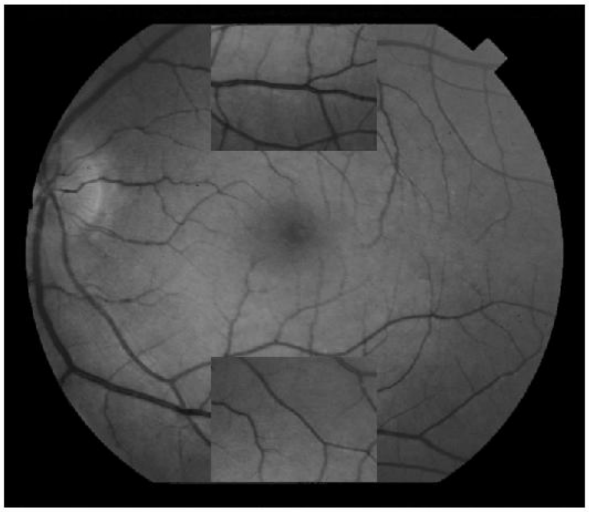
Due to the absence of a generic model to characterize manipulation, we simulate two simple cases of manipulation to test our algorithm. As shown in Figure 26, the two blocks whose areas make up 19% of the entire image are altered by moving and removing operations. In the first case, two blocks in a testing image are exchanged.



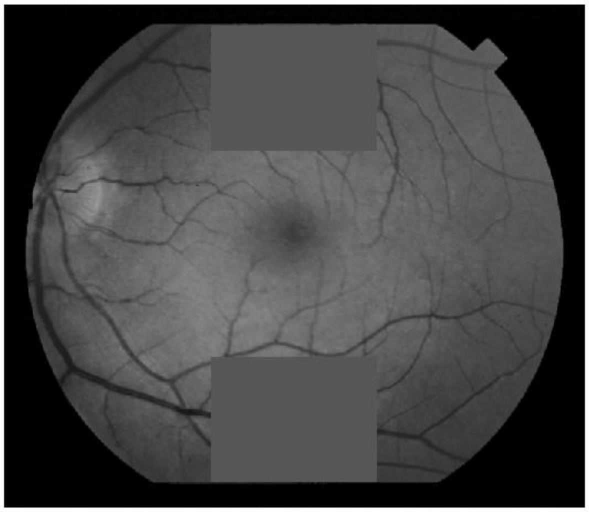
(a)

(b)

Figure 25: (a) Histogram of the similarity value of the in-authentication class and (b) histogram of the similarity value of the authentication class, where the two curves are the approximated normal distributions.



(a)



(b)

Figure 26: Two manipulation examples for a testing image: (a) two blocks in the image are exchanged; (b) two blocks in the image are replaced by the uniform regions with mean intensity of the entire image.

In the second case, the two blocks in the image are replaced by the uniform regions with mean intensity of the entire image.

Eight additional images of the retinal image database [7] were used to test the performance of our scheme. The testing image set contains both healthy (pathology-free) images and pathological ones, with various vasculature distribution and density. The Matlab computation time for watermark embedding or detection of an image is about one minute on a PC (2GHz Pentium 4 processor and 512 MB RAM). Table VII gives the authentication results on the testing images against the distortions mentioned earlier, and the two simulated manipulations. On average, the decision accuracy is 89% for all the tested images. Low-pass filtering, JPEG compression and rotation have less effects on accuracy of authentication. Impulse noise and size scaling have the most significant performance impacts, mainly due to loss of accuracy in segmentation of blood vessels. The two positive-false decisions on images 3 and 8, after the “manipulation 1” effect, are due to the relatively small number of vessels within the manipulated areas, so that only a few watermark bits are embedded to the area.

Table VII: Authentication results of the testing images against distortions and manipulations. "Yes" means that an image is decided to be authentic, and "No" means that an image is decided to be inauthentic.

Distortion	001	081	082	162	235	236	239	255	Ground truth
Salt-pepper noise 1	Yes	**No**	Yes	**No**	Yes	Yes	Yes	**No**	Yes
Salt-pepper noise 2	Yes	Yes	Yes	Yes	Yes	Yes	**No**	**No**	Yes
Gaussian noise 1	Yes	Yes	Yes	**No**	Yes	Yes	Yes	Yes	Yes
Gaussian noise 2	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Mean filtering	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Median filtering	Yes	Yes	Yes	**No**	Yes	Yes	Yes	Yes	Yes
Gaussian filtering	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
JPEG90	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
JPEG80	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
JPEG70	Yes	Yes	Yes	Yes	Yes	Yes	Yes	**No**	Yes
JPEG60	Yes	Yes	Yes	Yes	Yes	Yes	Yes	**No**	Yes
downsize 5%	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
downsize 10%	Yes	Yes	Yes	Yes	Yes	Yes	**No**	Yes	Yes
downsize 15%	**No**	Yes	Yes	Yes	Yes	**No**	Yes	Yes	Yes
downsize 20%	Yes	Yes	**No**	Yes	Yes	Yes	**No**	Yes	Yes
Rotation 10	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Rotation 20	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Rotation 30	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Rotation 40	Yes	**No**	Yes	Yes	Yes	Yes	Yes	**No**	Yes
Manipulation 1	No	No	**Yes**	No	No	No	No	**Yes**	No
Manipulation 2	No	No	No	No	No	No	No	No	No

CHAPTER V

CONCLUSION

In this dissertation, I have investigated three major problems in medical image processing. As a specific application, I studied the problems in segmentation, registration, and watermarking of retinal images. The segmentation results provide a good basis on which the registration and watermarking can be performed. The registration of two or more images taken on the same retina provides a complete view of the vascular network. The ROI-based watermarking scheme helps protect the image from malicious manipulation without affecting analysis-ROI.

For the segmentation problem, I proposed an adaptive scheme for blood vessel detection in color retinal images. The large and small vessels are treated as two types of image objects and are characterized by different features extracted from the pre-processed images. The vessel detection is performed by tracing along the vascular networks with forward prediction and backward verification. The average performance of our method over twenty tested images is 84.3% for true positive rate (TPR), and 3.9% for false positive rate (FPR). For normal images, the TPRs range from 80% to 91%, and their corresponding FPRs range from 2.8% to 5.5%. For abnormal images, the TPRs range from 73.8% to 86.5%, and the FPRs range from 2.1% to 5.3%, respectively. Small vessels take up 42% of overall vessel pixels, where 75% of small vessels were captured by our method.

For retinal image registration, speed and accuracy are major concerns. The performance of registration can be affected by many factors such as motions, illumination, and pathological changes, etc. To address this problem, I developed a fast registration method that employs the feature points extracted from the coarse vascular maps. It takes only 15 seconds to match two images, thus our method can be employed in real-time applications such as laser surgery. The experiments on a number of retinal images with good and poor qualities show that our method is fast and effective.

For the problem of image watermarking, I proposed a new ROI-based watermarking scheme for image authentication based on linear patterns such as the blood vessels in medical images. Several flow-ordering algorithms are proposed, and compared for their performances. A simple watermark recovery method was developed, whose performance was shown to be better than the existing techniques. The experiment results show that our scheme can prove the image authenticity with high probabilities under difference distortions.

During study of these three problems, geometric models are developed to characterize the objects of different shapes in retinal images. The proposed robust segmentation techniques can differentiate the ROIs from noise and lesion in the images with good and poor illuminations. The registration scheme meets the needs of the robustness and efficiency by using the coarsely extracted vascular network as landmarks. It can relate the structures in different images and help detect the laser scars in the post-surgery images. The ROI-based watermarking scheme uses extracted blood vessels as reference locations for embedding and detecting watermark. The effect of watermarking and distortions on the ROI segmentation and watermark synchronization were carefully studied. The authenticity of image content can be verified by comparing the recovered watermark with the original one. The proposed algorithms in these three problems, although tested only on retinal images, can be extended to other applications. Our study is expected to set a foundation to help development of computer-aided diagnosis of diabetic retinopathy and other ophthalmologic diseases.

REFERENCES

[1] N. J. Wareham, "Cost-effectiveness of alternate methods for diabetic retinopathy screening," *Diabetes Care*, vol. 16, page 844, 1993.

[2] J. C. Javitt, J. K. Canner, and A. Canner, "Cost effectiveness of current approaches to the control of retinopathy in type I diabetics," *Ophthalmology*, vol. 96, pp. 255–264, 1989.

[3] L. Ballerini, "An automatic system for the analysis of vascular lesions in retinal images," in Proc. *IEEE Nuclear Science Symposium*, vol. 3, Seattle, WA, Oct. 1999, pp. 1598–1602.

[4] Y. Wang, and S. C. Lee, "A fast method for automated detection of blood vessels in retinal images," in *Proc. of the Thirty-First Asilomar Conference on Signals, Systems & Computers*, vol. 2, Pacific Grove, CA, Nov. 1997, pp. 1700–1704.

[5] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, and M. Goldbaum, "Detection of blood vessels in retinal images using two-dimensional matched filters," *IEEE Trans. on Medical Imaging*, vol. 8, no. 3, pp. 263–269, 1989.

[6] M.E. Martinez-Perez, A.D. Highes, A.V. Stanton, S.A. Thorn, S.A, N. Chapman, A.A. Bharath, and K.H. Parker, "Retinal vascular tree morphology: a semi-automatic quantification," *IEEE Trans. on Biomedical Engineering*, vol. 49, no. 8, pp. 912–917, Aug. 2002.

[7] A. Hoover, V. Kouznetsova, and M. Goldbaum, "Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response," *IEEE Trans. on Medical Imaging*, vol. 19, no. 3, pp. 203–210, 2000.

[8] P. Jasiobedzki, D. McLeod, and C. Taylor, "Detection of non-perfused zones in retinal images," in *Proc. of the Computer-Based Medical Systems: Fourth Annual IEEE Symposium*, Baltimore, MD, 1991, pp. 162–169.

[9] F. Zana and J.-C. Klein, "Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation," *IEEE Trans. on Image Processing*, vol. 10, no. 7, pp. 1010–1019, 2001.

[10] H. Li and O. Chutatape, "Fundus image features extraction," in *Proc. IEEE Int. Conf. Engineering in Medicine and Biology*, vol. 4, Chicago, July 2000, pp. 3071–3073.

[11] A. Pinz, S. Bernogger, P. Datlinger, and A. Kruger, "Mapping the human retina," *IEEE Trans. on Medical Imaging*, vol. 17, no. 4, pp. 606–620, Aug. 1998.

[12] N. H. Solouma, A. Youssef, Y. A. Badr, and Y. M. Kadah, "A new real-time retinal tracking system for image-guided laser treatment," *IEEE Trans. on Biomedical Engineering*, vol. 49 No. 9, pp. 1059–1067, Sept. 2002.

[13] X. Jiang, and D. Mojon, "Adaptive local thresholding by verification-based multithreshold probing with application to vessel detection in retinal images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25 no. 1, pp. 131–137, Jan. 2003.

[14] R. Nekovei, and Y. Sun, "Back-propagation network and its configuration for blood vessel detection in angiograms," *IEEE Trans. on Neural Networks*, vol. 6, no. 1, pp. 64–72, Jan. 1995.

[15] J. Staal, M. D. Abramoff, M. Niemeijer, M. A. Viergever, and B. V. Ginneken,

"Ridge based vessel segmentation in color images of the retina," *IEEE Trans. on Medical Imaging*, vol. 23, no. 4, pp. 501–509, April 2004.

[16] A. Sim and E. de Ves , "Segmentation of macular fluorescein angiographies. A statistical approach," *Pattern Recognition*, vol. 34, no. 4, pp. 795–809, April 2001.

[17] G. Tascini, G. Passerini, P. Puliti, and P. Zingaretti, "Retina vascular network recognition," in *Proc. of SPIE Conference on Image Processing*, vol. 1898, Newport Beach, CA, 1993, pp. 322–329.

[18] L. Zhou, M. Rzeszotarski, L. Singerman, and J. Chokreff, "The detection and quantification of retinopathy using digital angiograms," *IEEE Trans. on Medical Imaging*, vol. 13, no. 4, pp. 619–626, 1994.

[19] H. Shen, B. Roysam, C.V. Stewart, J.N. Turner, and H.L. Tanenbaum, "Optimal scheduling of tracing computations for real-time vascular landmark extraction from retinal fundus images," *IEEE Trans. on Information Technology in Biomedicine*, vol. 5, no. 1, pp. 77–91, March 2001.

[20] Y. Sun, "Automated identification of vessel contours in coronary arteriograms by an adaptive tracking algorithm," *IEEE Trans. on Medical Imaging*, vol. 8, no. 1, pp. 78–88, March 1989.

[21] A. Can, H. Shen, J. N. Turner, H. L. Tanenbaum, and B. Roysam, "Rapid automated tracing and feature extraction from retinal fundus images using direct exploratory algorithms," *IEEE Trans. on Information Technology in Biomedicine*, vol. 3, no. 2, pp. 125–138, June 1999.

[22] O. Chutatape, L. Zhang, and S. M. Krishnan, "Retinal blood vessel detection

and tracking by matched Gaussian and Kalman filters," in *Proc. of the 20th International Conference of IEEE Engineering in Medicine and Biology Society*, vol. 6, Hong Kong, 1998, pp. 3144–3149.

[23] X. Gao, A. Bharath, A. Stanton, A. Hughes, N. Chapman, and S. Thom, "A method of vessel tracking for vessel diameter measurement on retinal images," in *Proc. of the 2001 International Conference on Image Processing*, vol. 2, Thessaloniki, Greece, Oct. 2001, pp. 881–884.

[24] L. Gang, O. Chutatape, and S. Krishnan, "Detection and measurement of retinal vessels in fundus images using amplitude modified second-order Gaussian filter," *IEEE Trans. on Biomedical Engineering*, vol. 49, no. 2, pp. 168–172, 2002.

[25] Y. A. Tolias, and S. M. Panas, "A fuzzy vessel tracking algorithm for retinal images based on fuzzy clustering," *IEEE Trans. on Medical Imaging*, vol. 17, no. 2, pp. 263–273, April 1998.

[26] B. Kochner, D. Schuhmann, M. Michaelis, G. Mann, and K. Englmeier, "Course tracking and contour extraction of retinal vessels from color fundus photographs: most efficient use of steerable filters for model-based image analysis," in *Proc. of the SPIE Conference on Image Processing*, vol. 3338, San Diego, CA, 1998, pp. 755–761.

[27] M. Goldbaum. (Nov. 2000). STARE: structured analysis of the retina. [Online]. Available: http://www.ces.clemson.edu/ ahoover/stare

[28] A. Hoover, and M. Goldbaum, "Locating the optic nerve in a retinal image using the fuzzy convergence of the blood vessels," *IEEE Trans. on Medical Imaging*, vol. 22, no. 8, pp. 951–958, Aug. 2003.

[29] J.A. Stark, “Adaptive image contrast enhancement using generalizations of histogram equalization,” *IEEE Trans. on Image Processing*, vol. 9, no. 5, pp. 889–896, May 2000.

[30] W. M. Morrow, R. B. Paranjape, R. M. Rangayyan, and J. E. L. Desautels, “Region-based contrast enhancement of mammograms,” *IEEE Trans. on Medical Imaging*, vol. 11, no. 3, pp. 392–406, 1992.

[31] D. Gabor, “Theory of communication,” *Journal of the Institute of Electrical Engineers*, vol. 93, pp. 429–457, 1946.

[32] J.G. Daugman, “Uncertainty relation for resolution in space, spatial frequency and orientation optimized by two-dimensional visual cortical filters,” *Journal of the Optical Society of America*, vol. 2, pp. 1160–1169, July 1985.

[33] D. Dunn and W.E. Higgins, “Optimal gabor filter for texture segmentation,” *IEEE Trans. on Image Processing*, vol. 4, no. 7, pp. 947–964, July 1995.

[34] A.K. Jain, and F. Farrakhina, “Unsupervised texture segmentation using Gabor filters,” *Pattern Recognition*, vol. 24, no. 2, pp. 1167–1186, 1991.

[35] Z.-Q. Liu, J. Cai, and R. Buse, *Handwriting Recognition: Soft Computing and Probabilistic Approaches*. Berlin: Springer, 2003.

[36] D. Kincaid and W. Cheney, *Numerical Analysis: Mathematics of Scientific Computing*, 2nd edition, Pacific Grove, CA: Brooks Cole, 1996.

[37] V. Barnett and T. Lewis, *Outliers in Statistical Data*, 3rd edition, New York: John Wiley, 1994.

[38] F. Grubbs, “Procedures for detecting outlying observations in samples,” *Technometrics*, vol. 11, no. 1, pp. 1–21, Feb. 1969.

[39] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. T. H. Romeny, J. B. Zimmerman, and K. Zuiderveld, "Adaptive histogram equalization and its variations," *Computer Vision, Graphics, and Image Processing*, vol. 39, pp. 355–368, 1987.

[40] L. G. Brown, "A survey of image registration techniques," *ACM Comput. Surveys*, vol. 24, no. 4, pp. 325–375, Dec. 1992.

[41] N. Ritter, R. Owens, J. Cooper, R. H. Eikelboom, and P. P. van Saarloos, "Registration of stereo and temporal images of the retina," *IEEE Trans. on Medical Imaging*, vol. 18, no. 5, pp. 404–418, May 1999.

[42] A. V. Cideciyan, "Registration of ocular fundus images: an algorithm using cross-correlation of triple invariant image descriptors," *IEEE Engineering in Medicine and Biology Magazine*, vol. 14, no. 1, pp. 52–58, 1995.

[43] A. Can, C.V. Stewart, B. Roysam, and H.L. Tanenbaum, "A feature-based, robust, hierarchical algorithm for registering pairs of images of the curved human retina," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 347–364, March 2002.

[44] F. Laliberte, L. Gagnon, and Y. Sheng, "Registration and fusion of retinal images - an evaluation study," *IEEE Trans. on Medical Imaging*, vol. 22, no. 5, pp. 661–673, May 2003.

[45] C. Heneghan, P. Maguire, N. Ryan, and P. de Chazal, "Retinal image registration using control points," in *Proc. of the 2002 IEEE International Symposium on Biomedical Imaging*, Washington, DC, July 2002, pp. 349–352.

[46] D. E. Becker, A. Can, J. N. Turner, H. L. Tanenbaum, and B. Roysam, "Image processing algorithms for retinal montage synthesis, mapping, and real-time location determination," *IEEE Trans. on Biomedical Engineering*, vol. 45, pp. 105–118, 1998.

[47] G. K. Matsopoulos, N. A. Mouravliansky, K. K. Delibasis, and K. S. Nikita, "Automatic retinal image registration scheme using global optimization techniques," *IEEE Trans. on Information Technology in Biomedicine*, vol. 3, no. 1 , pp. 47–60, March 1999.

[48] C. V. Stewart, C.-L. Tsai, and B. Roysam, "The dual-bootstrap iterative closet point algorithm with application to retinal image registration," *IEEE Trans. on Medical Imaging*, vol. 22, no. 11, pp. 1379–1394, Nov. 2003.

[49] J. Park, J.M. Keller, P.D. Gader, and R.A. Schuchard, "Hough-based registration of retinal images," in *Proc. of the IEEE International Conference on Systems, Man, and Cybernetics*, vol. 5, San Diego, CA, 1998, pp. 4550–4555.

[50] D. Lloret, J. Serrat, A. M. Lopez, A. Soler, and J. J. Villaneuva, "Retinal image registration using creases as anatomical landmarks," in *Proc. of the 15th International Conference on Pattern Recognition*, vol. 3, Barcelona, Spain, Sept. 2000, pp. 203–206.

[51] W. Hart and M. Goldbaum, "Registering retinal images using automatically selected control point pairs," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Austin, TX, 1994, pp. 576–581.

[52] P. Jasiobedzki, "Registration of retinal images using adaptive adjacency graphs," in *Proceedings of the Sixth Annual IEEE Symposium on Computer-Based Medical Systems*, Ann Arbor, MI, June 1993, pp. 40–45.

[53] A. M. Mendonca, A. Campilho, and J. Nunes, “A new similarity criterion for retinal image registration,” in *Proc. of the IEEE International Conference Image Processing*, vol. 3, Austin, TX, Nov. 1994, pp. 696–700.

[54] J. B. A. Maintz and M. A. Viergever, “A survey of medical image registration,” *Med. Image Anal.*, vol. 2, no. 1, pp. 1–36, 1998.

[55] G. Lin, C. V. Stewart, B. Roysam, K. Fritzsche, G. Yang, and H. L. Tanenbaum, “Predictive scheduling algorithms for real-time feature extraction and spatial referencing: application to retinal image sequences,” *IEEE Trans. on Biomedical Engineering*, vol. 51, no. 1, pp. 115–125, 2004.

[56] H. Shen, C. V. Stewart, B. Roysam, G. Lin, and H. L. Tanenbaum, “Frame-rate spatial referencing based on invariant indexing and alignment with application to online retinal image registration,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 3, pp. 379–384, March 2003.

[57] G. K. Matsopoulos, P. A. Asvestas, N. A. Mouravliansky, and K. K. Delibasis, “Multimodal registration of retinal images using self organizing maps,” *IEEE Trans. on Medical Imaging*, vol. 23, no. 12, pp. 1557–1563, December 2004.

[58] S. Zhang and K.S. Fu, “A thinning algorithm for discrete binary images,” in *Proc. of the International Conference on Computers and Application*, Beijing, China, 1984, pp. 879–886.

[59] J. E. Gentle, *Numerical Linear Algebra for Applications in Statistics*, Berlin: Springer-Verlag, 1998.

[60] F. Zana and J. C. Klein, “A multimodal registration algorithm of eye fundus images using vessels detection and hough transform,” *IEEE Trans. on Medical*

Imaging, vol. 18, no. 5, pp. 419–428, May 1999.

[61] M.J. Black and P. Anandan, “The robust estimation of multiple motions: parametric and piecewise-smooth flow-fields,” *Computer Vision Image Process*, vol. 63, no. 1, pp. 75–104, 1996.

[62] K. Fukunaga and J. M. Mantock, “Nonparametric data reduction,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, no. 1, pp. 115–118, 1984.

[63] S. H. Lai, “Robust image matching under partial occlusion and spatially varying illumination change,” *Computer Vision and Image Understanding*, vol. 78, pp. 84–98, 2000.

[64] G. Li, “Robust regression,” in *Exploring Data Tables, Trends, and Shapes*, D. C. Hoaglin, F. Mosteller and J. W. Tukey, Eds. New York: Wiley, 1985, pp. 281-343.

[65] P. J. Rousseeuw, “Least median of squares regression,” *Journal of the American Statistical Association*, vol. 79, no. 388, pp. 871–880, 1984.

[66] P. J. Rousseeuw, and K.V. Driessen, “A fast algorithm for the minimum covariance determinant estimator,” *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999.

[67] W. Hsu, P. M. Pallawala, L. L. Mong, and A. E. Kah-Guan, “The role of domain knowledge in the detection of retinal hard exudates,” in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, Kauai, Hawaii, Dec. 2001, pp. 246–251.

[68] Z. Bensbeh, L. D. Cohen, G. Mimoun, and G. Coscas, “A new approach of geodesic reconstruction for drusen segmentation in eye fundus images,” *IEEE*

Trans. on Medical Imaging, vol. 20, no. 12, pp. 1321–1333, Dec. 2001.

[69] T. Walter, J. C. Klein, P. Massin, and A. Erginay, "A contribution of image processing to the diagnosis of diabetic retinopathy – detection of exudates in color fundus images of the human retina," *IEEE Trans. on Medical Imaging*, vol. 21, no. 10, pp. 1236–1243, Oct. 2002.

[70] M. D. Swanson, M. Kobayashi, and A. H. Tewfik, "Multimedia data-embedding and watermarking technoligies," *Proceedings of the IEEE*, vol. 86, no. 6, pp. 1064–1087, June 1998.

[71] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding - a survey," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062–1078, July 1999.

[72] F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1079–1107, July 1999.

[73] G. C. Langelaar, I. Setyawan, and R. L. Lagendijk, "Watermarking digital image and video data. A state-of-the-art overview," *IEEE Signal Processing Magazine*, vol. 17, no. 5, pp. 20–46, Sep. 2000.

[74] B. B. Zhu and M. D. Swanson, "Multimedia authentication and watermarking," in *Multimedia Information Retrieval and Management*, D. Feng, W.C. Siu and H.J. Zhang, Eds., Berlin: Springer, pp. 148–177, 2003.

[75] S. Walton, "Information authentication for a slippery new age," *Dr. Dobbs Journal*, vol. 20, no. 4, pp. 18–26, April 1995.

[76] R. van Schyndel, A. Tirkel, and C. Osborne, "A digital watermark," in *Proceedings of the IEEE International Conference on Image Processing*, vol. 2, Austin, TX, Nov. 1994, pp. 86–90.

[77] H. Yuan, and X. P. Zhang, "Fragile watermark based on the Gaussian mixture model in the wavelet domain for image authentication," in *Proc. of the International Conference on Image Processing*, vol. 1, Barcelona, Sept. 2003, pp. 505–508.

[78] M. Alghoniemy and A. H. Tewfik, "Geometric invariance in image watermarking," *IEEE Trans. on Image Processing*, vol. 13, no. 2, pp. 145–153, Feb 2004.

[79] J. J. K. O'Ruanaidh and T. Oun, "Rotation, scale and translation invariant digital image watermarking," in *Proc. of the IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Oct. 1997, pp. 536–539.

[80] M. Kutter, S.K. Bhattacharjee, and T. Ebeahimi, "Towards second generation watermarking schemes," in *Proc. of the 1999 International Conference on Image Processing*, vol. 1, Kobe, Japan, Oct. 1999, pp. 320–323.

[81] I. J. Cox, J. Killian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Processing*, vol. 6, pp. 1673–1687, Dec. 1997.

[82] D. Kundur and D. Hatzinakos, "Digital watermarking using multiresolution wavelet decomposition," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing 1998 (ICASSP 98)*, vol. 5, Seattle, WA, May 1998, pp. 2969–2972.

[83] M. Kutter, "Watermarking resisting to translation, rotation, and scaling," in *Proc. of SPIE: Multimedia Systems and Applications*, vol. 3528, Boston, MA, Nov. 1998, pp. 423–431.

[84] A. Nikolaidis and I. Pitas, "Region-based image watermarking," *IEEE Trans. on Image Processing*, vol. 10, no. 11, pp. 1726–1740, Nov. 2001.

[85] P. Bas, J. M. Chassery, and B. Macq, “Geometrically invariant watermarking using feature points,” *IEEE Trans. on Image Processing*, vol. 11, no. 9, pp. 1014–1028, Sep. 2002.

[86] P.C. Su, H.J. Wang, and C.C.J. Kuo, “Digital watermarking in regions of interest,” in *Proc. of the IS&T Image Processing / Image Quality / Image Capture Systems (PICS)*, vol. 2, Savannah, Georgia, April 1999, pp. 295–300.

[87] Y. Zhao, P. Campisi and D. Kundur, “Dual domain watermarking for authentication and compression of cultural heritage images,” *IEEE Trans. on Image Processing*, vol. 13, no. 3, pp. 430–448, March 2004.

[88] X. Q. Zhou, H. K. Huang, and S. L. Lou, “Authenticity and integrity of digital mammography images,” *IEEE Trans. on Medical Imaging*, vol. 20, no. 8, pp. 784–791, Aug. 2001.

[89] A. Wakatani, “Digital waterwarking for ROI medical images by using compressed signature image,” in *Proceedings of the 35th Hawaiian International Conference on System Sciences*, Big Island, HI, 2002, pp. 2043–2048.

[90] N. Nikolaidis and I. Pitas, “Copyright protection of images using robust digital signatures,” in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP ’96)*, vol. 4, Atlanta, May 1996, pp. 2168–2171.

[91] X.-G. Xia, C. G. Boncelet, and G. R. Arc, “A multiresolution watermark for digital images,” in *Proc. International Conference on Image Processing* , vol. 1, Santa Barbara, CA, Oct. 1997, pp. 548–551.

[92] A. Piva, M. Barni, F. Bartolini, and V. Capellini, “DCT-based watermark recovering without resorting to the uncorrupted original image,” in *Proc. IEEE*

Int. Conf. Image Processing (ICIP'97), vol. 1, Santa Barbara, CA, Oct. 1997, pp. 520–523.

[93] F. Deguillaume, S. Voloshynovskiy, and T. Pun, "A method for the estimation and recovering from general affine transforms in digital watermarking applications," in *Proc. SPIE, Security and Watermarking of Multimedia Contents IV*, San Jose, CA, 2002, pp. 313–322.

[94] S. Craver, N. Memon, B.-L. Yeo, and M. M. Yeung, "Resolving rightful ownerships with invisible watermarking techniques: limitations, attacks, and implications," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 573–586, May 1998.

[95] H.S. Stone, "Analysis of attacks on image watermarks with randomized coefficients," NEC Research Inst., Technical Report, Princeton, NJ, May 1996.

[96] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital Watermarking*, London: Academic Press, 2002.

[97] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd ed. Chichester, UK: John Wiley & Sons Ltd, 2000.

[98] J.-M. Chassery and A. Montanvert, *Gomtrie Discrte en Analyze D'images*, Paris: Herms, 1991.

[99] E. Bertin, S. Marchand-Maillet, and J.-M. Chassery, *Optimization in Voronoi Diagrams*, Norwell, MA: Kluwer, 1994.

[100] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd edition, New York: John Wiley & Sons Inc., 2001.

[101] A.D. Hillery, and R.T. Chin, "Iterative Wiener filters for image restoration," *IEEE Trans. on Signal Processing*, vol. 39 , no. 8, pp. 1892–1899, Aug. 1991.

[102] J. Guo, D. Zhang, and P.-f. Shi, "Self-synchronizing watermarking scheme for an arbitrarily shaped object," *Pattern Recognition*, vol. 36, pp. 2737–2741, 2003.

VITA

Di Wu received his M.S. degree in mathematics in 1995, and B.S. degree in mathematics in 1992, both from Nankai University, Tianjin, China. From 1995 to 1998, he worked with the China Council for Promotion of International Trade for developing software and database management systems, among others.

His areas of interest include image processing, pattern recognition, robust estimation, traffic modelling, and numerical analysis. His permanent address is:
13-1-1202 Yangfangdian Rd.
Beijing, 100038
P.R. China