ABSTRACT
Identifying the vascular bifurcations and crossovers in the retinal image is helpful for predicting many cardiovascular diseases and can be used as biometric features and for image registration. In this paper, we propose an efficient method to detect vascular bifurcations and crossovers based on the vessel geometrical features. We segment the blood vessels from the color retinal RGB image, and apply the morphological thinning operation to find the vessel centerline. Applying a filter on this centreline image we detect the potential bifurcation and crossover points. The geometrical and topological properties of the blood vessels passing through these points are utilized to identify these points as the vessel bifurcations and crossovers. We evaluate our method against manually measured bifurcation and crossover points by an expert, and achieved the detection accuracy of 95.82%.

Index Terms— Retinal fundus image, Vessel bifurcation and crossover, Texture classification, Gabor energy filter bank, FCM clustering, A* search.

1. INTRODUCTION
Retinal vascular bifurcation or branching and crossover points are special landmarks (i.e., features) for predicting cardiovascular diseases, for image analysis and as a biometrics application. The changes in the retinal vasculature can be observed by the modification of bifurcation and crossover points which may be an indication about an illness in progress. For example, early diagnosis of hypertension is possible by measuring changes in selected vascular branching and crossover regions. These points are also used as valuable features, for instance, the landmarks for image registration, mosaicing and change detection [1, 2]. Vascular patterns based on these bifurcations and crossovers can also be used for personal identification and biometrics application in most reliable way [3].

The analysis of geometry and network properties of branching structure is not new but retinal blood vessel tree geometry and topology has not been widely studied by means of image processing. Previous methods for vessel bifurcation and crossover points detection can be categorized in two classes. The Model based approach which is exemplified as vectorization or exploratory tracing is much faster, more adaptive and more practical for implementation. However, most of the model based techniques developed are used in image registration and matching purposes, and they suffer from accuracy (missing landmarks) [4, 2, 5]. The geometrical feature based approach requires extensive pixel processing, and generally relies on segmentation, followed by skeletonisation and branch point analysis. Such techniques are more accurate in terms of landmark location and also useful for providing information to diagnose diseases, personal identification and biometrics features [6, 1, 3, 7].

Perez et al. [6] proposed a geometric feature based technique for detecting vascular bifurcation and crossover points (landmarks) based on skeletonized vascular network. They characterize the bifurcation and crossover points by the total number of intersections using a fixed-size circular window centered on the candidate landmark points. This process extracts the morphological features to identify the landmarks of retinal vascular tree. However, it fails when two true bifurcation points are very close and when two vessels cross at a very small acute angle so that the two candidate bifurcation points fall outside the circular window. This process also not prune to false identifying landmarks if another vessel passing through this fixed size window. Jung et al. [3] proposed cross perpendicular structure and Y-type structure with eight connected neighbors for vascular landmarks extraction. Bevilacqua et al. [1] and Chawimaluang et al. [7] also proposed similar techniques to detect landmarks based on some specific formation of vessel centrelines with their eight connectivity. But all these methods tend to detect landmarks wrongly if there is any impulse noise or remaining pixels after skeletonisation (because of abrupt change of the width of any vessel) within this window. The processes also fail to detect landmarks with different formation or pixels orientation which are not defined.

In this paper we propose a novel approach to detect the vessel bifurcation and crossover points based on the geometri-
2. PROPOSED METHOD

We propose a method for blood vessel bifurcation and crossover detection based on vessel geometrical and topological features. We choose this method as it has higher accuracy than the model based method. In addition, we can perform more comprehensive analysis on the potential bifurcation or crossover points. The process is based on vessel centerline image which we obtain from the vessel segmented retinal image. At first we segment the blood vessels from original retinal RGB images using color space transformation, texture analysis and Fuzzy C-Means Clustering (FCM). Following that we apply the morphological thinning operation to find the centerline of the blood vessels. We apply a rotational invariant mask which scan the centerline image to obtain the potential vessel bifurcation and crossover points (for some specific pixel formations). For each of this potential landmark, we find the number of blood vessels which pass through this. The number of these blood vessels is the key factor for identifying this point as a vascular bifurcation or crossover.

To find the number of vessels (i.e., centerline pixel) we map a circular region of that potential bifurcation or crossover position based on this junction (a point where more then one vessel pass through width). We measure the junction width based on the edge image and this potential landmark point. The edge image is obtained by applying Sobel edge detector to the segmented image. We consider the width of this junction point as a multiplication factor to obtain the radius of a circle, which find the vessel centerline pixels through its circumference. If a pixel position on this circle’s circumference satisfies the criteria to be a vessel centerline, we check its connectivity with the potential bifurcation or crossover point by A* search algorithm. In this way we can eliminate the detected pixels from other vessel centerlines which is not belong to that landmark position. If a pixel position passes the connectivity check then we count it as a vessel for that potential landmark which latter will be considered for identifying the vessel bifurcation or crossover. If we detect more than one potential landmark pixel within the distance of the junction width, we then check the slope and acute angle of the lines which are belong to the potential bifurcation or crossover. Finally, we classify a potential vessel bifurcation or crossover point as an actual landmark based on the number of vessels.

3. VESSEL SEGMENTATION

We use the RGB transformed perceptually uniform color spaces i.e., Gaussian’s first two components $\tilde{E}$ and $\tilde{E}_\lambda$ and $L^*a^*b^*$ for texture analysis. We analyze the texture of each color channel image based on Gabor filter and construct texture feature vector based on these texture images and classify the vessel pixel using FCM clustering technique and obtain the vessel segmented output image [8].

3.1. Color space transformation and preprocessing

Generally image data is given in RGB space (because of the availability of data produced by the camera apparatus). We convert the RGB image into Gaussian and $L^*a^*b^*$ which are the perceptually uniform color spaces. These are also the true representation of color spaces, difference as perceived by human and effective in rotation invariant color texture analysis. The definition of $L^*a^*b^*$ and Gaussian are based on an intermediate system, known as the CIE XYZ space (ITU-Rec 709). The derivations are shown in [8, 9, 10].

![Fig. 1. Original RGB image and Gaussian transformed first and second component image (left to right).](image)

3.2. Texture Feature Extraction

Texture generally describes second order property of surfaces and scenes, measured over image intensities. A Gabor filter has weak responses along all orientations on the smooth (background) surface. On the other hand, when it positioned on a linear pattern object (like a vessel) the Gabor filter pro-
3.2.1. Gabor Filter

An input image $I(x, y)$, $(x, y) \in \Omega$ where $\Omega$ is the set of image points, is convolved with a 2D Gabor function $g(x, y)$, $(x, y) \in \omega$, to obtain a Gabor feature image $r(x, y)$ (Gabor filter response) as follows [12]

$$r(x, y) = \int_{\Omega} I(\xi, \eta) g(x - \xi, y - \eta) d\xi d\eta$$

(1)

We use the following family of 2D Gabor functions to model the spatial summation properties of an image [12]

$$g_{\xi, \eta, \lambda, \Theta, \phi}(x, y) = \exp\left(-\frac{x'^2 + y'^2}{\sigma^2}\right) \cos(2\pi \lambda x' + \phi)$$

$$x' = (x - \xi) \cos \Theta - (y - \eta) \sin \Theta$$

$$y' = (x - \xi) \sin \Theta + (y - \eta) \cos \Theta$$

(2)

where the arguments $x$ and $y$ specify the position of a light impulse in the visual field and $\xi, \eta, \sigma, \gamma, \lambda, \Theta, \phi$ are parameters. The pair $(\xi, \eta)$ specifies the center of a receptive field in image coordinates. The standard deviation $\sigma$ of the Gaussian factor determines the size of the receptive field. Its eccentricity is determined by the parameter $\gamma$ called the spatial aspect ratio. The parameter $\lambda$ is the wavelength of the cosine factor which determines the preferred spatial frequency $\frac{1}{\lambda}$ of the receptive field function $g_{\xi, \eta, \lambda, \Theta, \phi}(x, y)$. The parameter $\Theta$ specifies the orientation of the normal to the parallel excitatory and inhibitory stripe zones - this normal is the axis $x'$ in (5).

Finally, the parameter $\phi \in (-\pi, \pi)$, which is a phase offset argument of the harmonic factor $\cos(2\pi \frac{x'}{\lambda} + \phi)$, determines the symmetry of the function $g_{\xi, \eta, \lambda, \Theta, \phi}(x, y)$.

3.2.2. Gabor Energy Features

A set of textures was obtained based on the use of Gabor filters (1). The filter results of the phase pairs were combined, yielding the Gabor energy quantity [12]:

$$E_{\xi, \eta, \lambda, \Theta, \phi} = \sqrt{r^2_{\xi, \eta, \lambda, \Theta, \phi} + r^2_{\xi, \eta, \lambda, \pi/2}}$$

(3)

where $r^2_{\xi, \eta, \lambda, \Theta, \phi}$ and $r^2_{\xi, \eta, \lambda, \pi/2}$ are the outputs of the symmetric and antisymmetric filters. We used Gabor energy filters with twenty-four equidistant preferred orientations ($\Theta = 0, 15, 30, ... , 345$) and three preferred spatial frequencies ($\lambda = 6, 7, 8$). In this way an appropriate coverage was performed of the spatial frequency domain.

We constructed an image (Fig. 2) on each color channel which was used for histogram analysis to determine the cluster number. From these images we constructed twelve element length feature vector for each pixel in each retinal image to classify them into vessel and non-vessel using the FCM clustering algorithm.

3.3. Texture Classification and Image Segmentation

The FCM is a data clustering technique where in each data point belongs to a cluster to some degree that is specified by a membership grade. Let $X = x_1, x_2, ..., x_N$ where $x \in R^N$ present a given set of feature data. The objective of the FCM clustering algorithm is to minimize the FCM cost function formulated as [13]

$$J(U, V) = \sum_{j=1}^{C} \sum_{i=1}^{N} (U_{ij})^m ||x_i - v_j||^2$$

(4)

$V = \{v_1, v_2, ..., v_C\}$ are the cluster centers. $U = (U_{ij})_{N \times C}$ is fuzzy partition matrix, in which each member is between the data vector $x_i$ and the cluster $j$.

We used the Matlab Fuzzy Logic Toolbox for clustering 253440 vectors (the size of the retinal image is 512x495) in length twelve for each retinal image. We produced the binary image considering the cluster central intensity value which identifies the blood vessels only (Fig 3).

3.4. Experimental Results

Using the DRIVE database [14] we applied our method for vessel segmentation. For performance evaluation five images produced by our method with the hand leveled Ground Truth images. We achieved an overall 84.37% sensitivity and 99.61% specificity which is very promising in this domain.

4. VESSEL CENTERLINE DETECTION

We apply the skeletonisation operation on the vessel segmented binary image to detect the vessel centerlines. Skeletonisation.
5. VASCULAR BIFURCATION AND CROSSOVER DETECTION

We detect the bifurcations and crossovers based on the centerline image. We apply a rotational invariant mask on the centerline image to find the potential bifurcation and crossover points. For each of these points we find the geometric features of the vessels from the centerline image and edge image, and finally identify the bifurcation and crossover points. The processes are described in the following sections.

5.1. Potential Location Identification

We apply the rotational invariant masks to pick the potential bifurcation and crossover points. The mask is a three by three window which positioned on a pixel as its center and filter this pixel as a potential bifurcation if it has a specific formation of pixels within this eight neighboring connectivity. Fig. 4 depicts the masks which are implemented on rotating from $0^\circ$ to $360^\circ$. We scan the whole binary centerline image with these masks and pick the pixel positions which satisfy the criteria of formation or matching with these masks and select these points as potential vascular bifurcations or crossovers. After that we check for the vessels which pass through these potential bifurcation and crossover points.

5.2. Vascular tree Detection

Applying a mask to detect the vessel bifurcation and crossover (like some other methods) would be an easiest and simplest method. However, very often it may lead us to a misclassification of the vascular bifurcations or crossovers. Fig. 5 portrays a circumstances while the masks pick a vessel pixel as a bifurcation point which is actually an additional pixel due to the abrupt changes of the vascular width. Therefore, we need to analyze the vascular tree for each potential bifurcation or crossover point to identify it properly.

The analysis of the vascular tree starts with the measurement of vessel centerline position which enables us to count the number number of vessels source from or sink into this bifurcation or crossover point. We find the centerline point of a vessel from a certain distance of that bifurcation or crossover point. The distance we measure based on the width of that junction point. We introduce a method to measure the width of any vessel cross-section based on the vessel edge and centerline pixel. We select the distance as $1.5 \times \text{width}$ as we have observed that there is always a continuation of length at least three times width of the junction point from which a vessel sources. We consider a circular distance around the junction and if any centerline pixel is found we check its connectivity with this junction point (for vessel identification). In this way we can discard any centerline pixel or vessel from counting into this junction point.

5.2.1. Vessel Edge Detection

We apply the Sobel operator on the segmented images to detect the edge (Fig. 6). The Sobel operator performs a 2-D spatial gradient measurement on an image (horizontality and verticality) and so emphasizes regions of high spatial gradient that correspond to edges. Typically it is used to find the approximate absolute gradient magnitude at each point in an input greyscale image. Further illustration can be found in [16].
5.2.2. Junction width measurement

After obtaining the vessel edge image we apply the junction width measurement method. For each potential bifurcation or crossover pixel position we map the edge image to find the vessel width for that cross-section. First we pick that potential landmark pixel from the vessel centerline image. Then we apply a mask considering this landmark pixel as its center to find the width for that cross-section. The width is the shortest distance of all pixel pairs from the opposite edges which are passing through this landmark pixel. For searching all the pixel positions inside the mask, we calculate the pixel positions by shifting from one to the size of the mask and rotating each position from 0 to 180 degree at the same time. The mask size is taken as $2 \times (\text{widest junction})$ based on the observation of any of these edge images. For increasing the rotation angle we find the step size (depending on the size of the mask) which is less than $180/(\text{mask length})$. Hence, we can access every cell in the mask using this angle.

For each obtained position we search the edge image gray scale value to check whether it is an edge pixel or not. Once we find an edge pixel we then find its mirror by shifting the angle of 180 degree and increasing the distance from one to the maximum size of the mask (Figure 7). In this way we produce a rotational invariant mask and pick all the potential pixel pairs to find the width or diameter of that cross sectional area.

$$\begin{align*}
x_1 &= x' + r \times \cos \theta \\
y_1 &= y' + r \times \sin \theta
\end{align*}$$

where the $(x', y')$ is the vessel centerline pixel and $r = 1$ to $(\text{mask - size})/2$ and $\theta = 0$ to $180^\circ$. For any pixel position if the gray scale value in the edge image is 255 (white or edge pixel) then we find the pixel $(x' + 2, y')$ in the opposite edge (mirror of this pixel) considering $\theta = (\theta + 180^\circ)$ and varying $r$.

After applying this operation we obtain the pairs of pixels which are on the opposite edge (as line end points) and which imaginary lines pass through the centerline pixels. From these pixel pairs we find the minimum distance using Euclidian distance, where $\text{Dist} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$. In this way, we can measure the width for junction points.

**Input:** landmark, centerline image, radius  
**Output:** vessel centerline pixel

*Considering the landmark pixel $(cx, cy)$ as the center of a circle and radius $r$*

**begin**

for $\text{start} ← 0$ to 1 do

for $\text{rad} ← -\text{radius to radius}$ do

for $\text{loop} ← 0$ to 1 do

if $\text{start} == 0$ then

$x = cx + rad$

if $\text{loop} == 0$ then

$y = cy - dy$

else

$y = cy + dy$

end

else

$y = cy + rad$

if $\text{loop} == 0$ then

$x = cx + dy$

else

$x = cx + dy$

end

end

Check $(x,y)$ and its neighborhood pixels in the Centerline image and save it in a LIST if it is a vessel centerline pixel

**end**

**end**

**Algorithm 1:** Find the centerline pixels for a potential landmark searching on a circular region.
5.2.3. Finding Vessel Centerline Points

After obtaining the junction width we consider it as a circle radius to find the vessel centerline points. We consider the circle equation, \( x^2 + y^2 = r^2 \) where \((x, y)\) is the set of all points, the circle passes through on the x-y coordinate system with radius \(r\). We consider the potential landmark position as the center of the circle and implement algorithm 1 to find vessel centerline.

Each centerline pixel selected by this method is checked for the connectivity with the potential bifurcation or crossover point. This centerline pixel is considered as the part of a blood vessel for this potential vessel bifurcation or crossover point, if it passes the connectivity test. This check is performed to filter out any vessel pixel belong to other vessel bifurcation or crossover position. The following figure (Fig. 8) depicts the detected landmarks and circular region around them showing the searching positions for vessel centerline pixels. For some special orientations of a vessel centerline, the pixel positions we map by the circle miss the centreline pixels. Therefore, we check the eight neighborhood connectivity for each calculated pixel position by Algorithm 1.

![Fig. 8. Circular region searching for detecting vascular tree (i.e., centerline pixel).](image)

5.2.4. Connectivity Checking

We apply \(A^*\) search algorithm to check the connectivity between the vessel centerline pixel and the potential vessel bifurcation or crossover pixel (Algorithm 2). We accept the vessel belongs to the potential landmark if the centerline pixel and landmark pixel are connected. Therefore, we overcome the possibility of picking another vessel centerline pixel which are passing through the neighbor of this landmark position and within this circular region. \(A^*\) is a graph/tree search algorithm that finds a path from a given initial node to a given goal node (or one passing a given goal test). \(A^*\) is complete in the sense that it will always find a solution if there is one. \(A^*\) also takes already traveled distance into account which makes this algorithm complete and optimal [17].

To make the search automatic we start searching from the landmark position which stops after finding the vessel centerline position or certain iterations. We check the eight neighborhood connectivity for that landmark position and find the neighboring pixels. We then calculate the euclidian distance between those neighboring pixels and the vessel centerline pixels. Therefore, we assign the the pixel which has the minimum distance to start the search. The algorithm must stop after a certain number of iteration if it fails to find the target centerline pixel and assign the second one in order and so on.

5.3. Estimation of Vessel Bifurcations and Crossovers

For each bifurcation or crossover point we consider the number of vessel centerline pixels (also the indicator of the number of vessels) which passes the connectivity check. After calculating all these possibilities for a landmark position, if the number of centerline pixels for this landmark is three, it is a vessel bifurcation or branching and if the number is four it is a vessel crossover.

**Input:** Centerline im, Centerline pixel, landmark pixel  
**Output:** Connectivity

```
begin
1. Find the neighboring pixels of the landmark
2. Calculate the Euclidian distance between these neighboring pixels and centerline pixel
3. Sort the pixels in ascending based on the distance and put in a LIST
end
```

```
begin
start the search from the first pixel in the LIST
while not the target centerline pixel is found do
    Proceed the searching through eight neighborhood connectivity
    Each time a centerline pixel is found, a value assigned for track indication and check for the target pixel
    if the target centerline pixel is found then
        return a value for positive connectivity and break the while loop
    else if certain no of iteration reaches and target pixel not found then
        consider the next pixel from the LIST until its end and go back to the beginning of the current section
    else if certain no of iteration reaches and target pixel not found and LIST is empty then
        return a value for non-connectivity and break the while loop
    else
        continue search
end
end
```

**Algorithm 2:** Connectivity checking for blood vessel estimation by \(A^*\) search algorithm

Generally, in a large vessel crossover we detect two bifurcation or crossover points instead of one crossover (Fig
9). This problem is also mentioned at [6] which was unresolved. We address this problem by considering the width of that junction. The junction points which are within this width distance are selected for merging if they satisfy the following criteria. We measure the slope of each line segment which are belong to these junction points. Based on the slope of each line segment, we apply the perceptual grouping procedure for these landmark positions. We calculate the acute angle between the opposite line segments (which has the closest slope values) with slope $m_1$ and $m_2$, which is, 
\[ \phi = \arctan \left( \frac{m_2 - m_1}{1 + m_1 m_2} \right). \]
We observe that this angle is always more than $120^\circ$ (for the same vessel) and we consider this as a threshold value. Therefore, we merge the landmarks which are within the distance of a landmark junctions’ width and satisfy the angular measurement.

![Fig. 9. A vessel crossover, initially detected as two crossovers.](image)

6. EXPERIMENTAL RESULTS AND DISCUSSIONS

We used DRIVE database [14] and applied our method to produce the output images indicating the vessel bifurcation and crossover points (JPs). It took approximately 9.49 minutes on MATLAB 7.1 to produce each output image (Fig. 10) on a 2.79 GHz Pentium(R) 4 CPU with 1.5 GB of RAM (manual grading required approximately an hour for each image and a very exhaustive process). To evaluate the performance of our method, we compared the manually graded images by an expert to the output images produced by our method (Table 1). Figure 11 depicts the key features for the evaluation procedure.

First we compare the JPs (position wise) in the images obtained by our method with the manually graded images. Column 4 gives the number of JPs which were selected by both methods as true positives or agreed. The JPs that were missed by our method are categorized as initial error or missed shown in column 5. For any junction point that was picked by the automatic method and was not manually graded, we did further analysis with the expert (post processing). If the expert concurred, we added this JP to find the total number of JPs and obtained the final number of JPs for manual method. The number of false detections (spurious JPs) was always zero (another proficiency of our method). The final error was calculated as follows: \[ \text{Error} = \frac{\text{JPs missed} + \text{spurious JPs}}{\text{Manual JPs} + \text{JPs missed in manual after post processing}}. \]
In our experiment, we had a maximum error of 6.36% and minimum error of 2.97% in any image. The reason of this error is the failure of segmentation for some minor blood vessels. Therefore, we are investigating on further improvement of the segmentation method based on the texture and edge information from the color retinal image. We considered five images for evaluation and received an average error of 4.18%. This gives an accuracy of 95.82%, which is of high acceptable standard.

![Fig. 10. An output retinal image showing vessel bifurcations and crossovers.](image)

![Fig. 11. The evaluation procedure of JPs.](image)

7. CONCLUSION

In this paper we proposed a novel method for vessel bifurcation and crossover detection. We used the geometrical and topological properties of retinal blood vessels for identifying the vessel bifurcation and crossover points. We provided a full map of the junction points on the vascular network which are ready to be used in diagnosis of cardiovascular diseases, image registration or biometrics application. Our method is automatic and also equally capable of identifying the false bifurcation and crossover points. As evidenced by the experimental results our method is able to detect bifurcation and crossover points with average accuracy of 95.82% which is very efficient. Currently, we are working in developing a vascular network model based on these results.
Table 1. Evaluation of Bifurcation and Crossover Points (JPs)

<table>
<thead>
<tr>
<th>Image</th>
<th>Detected JPs no.</th>
<th>Initial assessment</th>
<th>Post Analysis</th>
<th>Total error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Agreed (no.)</td>
<td>Missed in auto.</td>
<td>Missed</td>
<td>Final no.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>in auto.</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>100</td>
<td>98</td>
<td>2</td>
<td>102</td>
</tr>
<tr>
<td>2</td>
<td>99</td>
<td>96</td>
<td>0</td>
<td>99</td>
</tr>
<tr>
<td>3</td>
<td>104</td>
<td>102</td>
<td>3</td>
<td>107</td>
</tr>
<tr>
<td>4</td>
<td>101</td>
<td>98</td>
<td>0</td>
<td>101</td>
</tr>
<tr>
<td>5</td>
<td>110</td>
<td>103</td>
<td>0</td>
<td>110</td>
</tr>
</tbody>
</table>

8. REFERENCES


