

# Chapter 3

## Supervised Retinal Blood Vessel Segmentation

### 3.1 Introduction

The recent advancements in automated disease diagnostics have been a tremendous help to our society and our over-loaded medical experts. Many factors such as high-quality image acquisition, quick data transfer, immense storage capacities, improved computing facilities, and the use of artificial intelligence have upgraded the level of medical image analyses and automated diagnostics. The process of diagnosis using medical images includes the study of various imaging modalities to understand the morphology of normal and abnormal features, their localization, and segmentation. The introduction of deep learning to medical image analysis has helped extract high-level features from the images, which are otherwise overlooked. This deep pattern recognition can be well exploited only if we have abundant data or we have ways to resolve the data scarcity in the medical field.

As discussed in chapter 2, the retinal blood vessel extraction has been an exhaustively explored research area with quite an enriching literature. These segmentation methods are both unsupervised and supervised. The previous chapter discusses a survey on the conventional image processing based, unsupervised methods in detail. These methods use techniques like matched filtering [6], [8], morphology [18], [22], active con-

tour [30], vessel modeling [29], etc. to enhance and segment the vessel pixels. On the other hand, unsupervised ML-methods are based on manual or automatic feature extraction, and an iterative model tuning [45], [35]. In the case of unsupervised methods, though there is no requirement of prior knowledge about the ground truth, yet these methods are observed to be less generic. In other words, these methods may perform less in case of an unseen new datasets which is entirely different from the ones used during experimentation. Another class of vessel segmentation involves supervised methods. These methods can be divided in two categories: first, the conventional ML-based methods [13], [24], [32] etc., which need explicit handcrafted feature engineering, and second, the automated deep learning methods [33], [40], etc. in which machine itself is responsible for pattern recognition. In both the cases, data labeling is crucial.

With emerging computing and data storage facilities, the above mentioned deep learning architectures have gained a lot of interest of research community. These methods are supervised with multiple convolution layers, which automatically learn the relevant deep features from the input data. First came the Convolutional Neural Network (CNN) based methods which use convolution layers for feature extraction and then either an ML-based model [36], [37] or fully connected layers [33] are used, in the end, for deciding the final vessel probability map. CNN with fully connected layers are proved to be faster and more precise than CNN with ML models; however, a comparatively larger amount of data is required in the former case to avoid any overfitting. To mitigate this requirement of extensive pixel-level annotations, Generative adversarial networks (GAN) based retinal vessel extraction are also proposed [40]. The emergence of Fully Convolutional Networks (FCN's), where deconvolutional layers replace the fully connected layers, has been revolutionary research. These networks provide an end-to-end framework for image segmentation. Most of the state-of-the-art methods, specially in the field of semantic segmentation, are based on these models [71], [72], [73], [38] etc. U-net [71] is one of the most widely accepted FCN in the field of medical image segmentation. It is an encoder-decoder-based model with skip connections. Several research groups have experimented with this architecture to improve retinal blood vessel extraction. Zhang et al. [34] used an edge-aware mechanism and additional labels

to segment boundary vessel pixels with the U-net model. The idea was to pay more attention to the edge pixels. Jin et al. [74] proposed DUNet in which the internal modules of U-net are replaced by deformable convolution and RoI pooling for dynamic adjustments according to the input. Li et al. [72] and Kim et al. [76] proposed iterative U-net models for vessel segmentation. Although all these models work on a common backbone and have performed well in extracting retinal blood vessels, yet none of the models have focused particularly on segmentation of vessels lying in difficult regions of fundus.

Though retinal blood vessel segmentation has been a well-exploited research problem, some issues need to be addressed to improve the automated diagnosis based on extracted vessel maps. Most of the state-of-the-art algorithms focus on extracting the vessel maps with high accuracy, which leads to the loss of thin vessel ends, especially those located in low-contrast regions. Segmentation of these thin vessel ends is inevitable for diagnostic purposes. Moreover, the extraction of precise vessel width, which continuously varies in the fundus plane, is another crucial parameter that needs attention.

In this work, we propose vessel segmentation algorithm-I and II, from which we reach our final proposed algorithm-III. We start with algorithm-I, in which we apply wavelet transform to enhance the vessel pixels and feed the enhanced gray-scale map to a deep model for fine-tuning. Here, we cover fine-to-coarse vessel width with multi-scaling. Next, we modify this algorithm by adding Gaussian matched filtering and texture-suppression steps to pre-process the input fundus image before feeding it to the deep model for vessel extraction. Here, along with multi-scaling, we use fine texture suppression to enhance vessels in low contrast regions. We term this updated vessel segmentation as algorithm-II. We use random patch-based training and data augmentation to handle the data scarcity in both of these methods.

However, till now, we have been choosing fixed scales for multi-scaling. Also, random patches are not efficient in training the model to extract complete vessel structure. Thus, we evolve into our final algorithm-III. Here, we proceed by targeting the spatially varying vessel-width with multiple scales, where the selection of best fitting scales

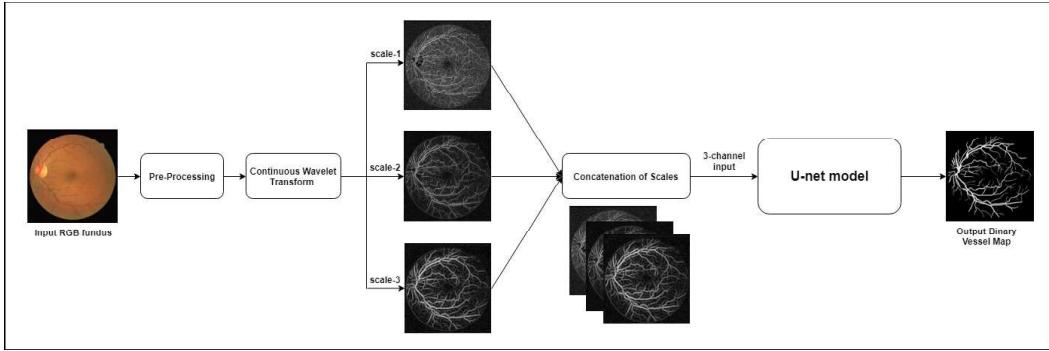


Figure 3.1: *Block diagram of the proposed vessel extraction using algorithm-I.* Here, we process input color fundus images to extract corresponding three channel images, with the magnitude of real wavelet transform coefficients corresponding to different scales in each channel. These images are used to train U-net for vessel extraction.

and their optimized combinations are learned as per the input resolution. We use 2-D continuous Gabor wavelet transform to evaluate the real, imaginary, and absolute coefficient maps at the most significant scales. Such adaptive multi-scale analysis helps in preparing a generic vessel segmentation method. The weighted combination of these coefficients is learned using a U-net-based deep model. Moreover, in order to understand various challenging regions of the fundus, a novel characteristic patch-based training procedure is adopted where we extract the training patches belonging to difficult areas like boundaries of OD, near-pathological lesions, low contrast regions, etc. These patches are used to train our model, making the learning of vessel structures better, especially in the aforementioned error-prone regions of the retina.

## 3.2 Algorithm I

In this method, we process a color fundus image to generate a new three-channel image, with the magnitude of real wavelet transform coefficients corresponding to different scales in each channel. These scales are chosen such that all kinds of vessel thicknesses (thin-to-thick) are covered in this range. This vessel enhanced, multi-scale, three-channel input image is fed to the U-net model [71]. The proposed algorithm boosts the performance of the basic U-net model for vessel segmentation significantly. Figure 3.1 illustrates the block diagram of the proposed vessel segmentation algorithm-I.

### 3.2.1 Pre-processing

In pre-processing, we take the input color fundus image and extract the RoI in a similar manner as described in chapter 2. Again, we use only the green channel of extracted RoI as it presents the best contrast between vessel foreground and non-vessel background [8]. In addition to this, CLAHE is used to improve the contrast of the green channel image to obtain an enhanced image. For further processing, we invert the RoI of contrast-enhanced image.

At this stage, we have a gray-scale, vessel-enhanced fundus image. The algorithm further progresses in two steps: 1. Wavelet-based fine-to-coarse vessel enhancement, 2. Training of U-net model using these vessel enhanced images for a more sensitive and generic retinal vessel extraction.

### 3.2.2 Wavelet-based Vessel Enhancement

As observed in chapter 2, the magnitude of the maximum real response best presents the vessel pixels while using the Gabor analyzing wavelet. The diameter of blood vessels (thickness) is variable. Vessels are thicker at the origin and thinner at the ends. We experiment with the magnitude of real response at different scales ( $a = 1, 2, 3, 4, 5$ ) and it is observed that almost all the vessel thicknesses are covered using just three scales,  $a = 2, 3, 4$  (here, for two fixed datasets: DRIVE and STARE). Though these scales cover almost all the vessels, we still have many vessel-like false edges that need to be removed. This concatenated, vessel-enhanced image is now used to train the standard U-net model for further fine-tuning and suppression of false edges. Figure 3.2 shows the real response at three different scales and their concatenation for a sample fundus image.

### 3.2.3 Supervised Fine-Tuning

U-net model is a symmetric, contracting-expanding deep learning architecture that was specifically developed for medical image segmentation [71]. It has convolution layers followed by the ReLU activation layer, max-pooling layers, and up-sampling layers.

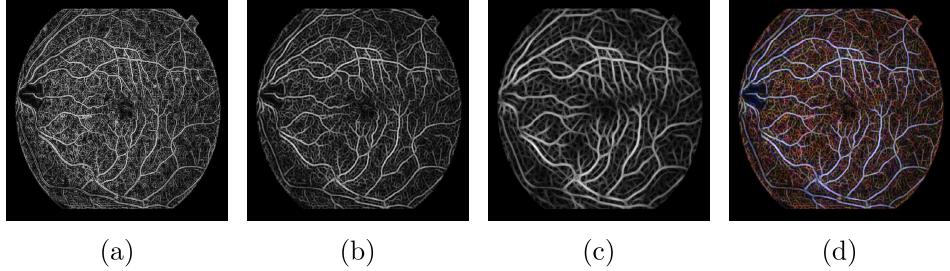


Figure 3.2: *Magnitude of maximum real response at different scales for image ‘im0077’ of STARE data-set : (a)  $a=2$ , (b)  $a=3$ , (c)  $a=4$ , (d) Final concatenated image shown in RGB colorspace.*

High-resolution information is passed directly, from the contracting path to expanding path. This architecture is developed so that it is very good at capturing image context along with precise localization.

At this stage, we have a three-channel image with enhanced vessel-like patterns at three different scales. We now train the U-net model mentioned above using this vessel-enhanced image. Previously, researchers have trained the same model for retinal blood vessel extraction but using RGB fundus images [71], [74]. Training this model with a pre-processed, vessel-enhanced image leads to the model’s better and easier learning of precise vessel features. The model performance is significantly improved in this case. Here, we perform a random patch-based training of the deep model.

### 3.3 Algorithm II

After covering the varying vessel width in algorithm-I, we target the problem of vessel segmentation in low-contrast regions by proposing algorithm-II. This algorithm works at multiple ‘levels’. It first enhances all the vessel-like features, and then vessel-like background texture is suppressed while preserving the strong vessel edges at different levels. These preserved edges at different levels are combined by extracting the maximum value at each pixel. This handcrafted feature image trains a deep U-net model, which gives the final segmented vessel map. Figure 3.3 presents the block diagram of the proposed vessel segmentation algorithm-II.

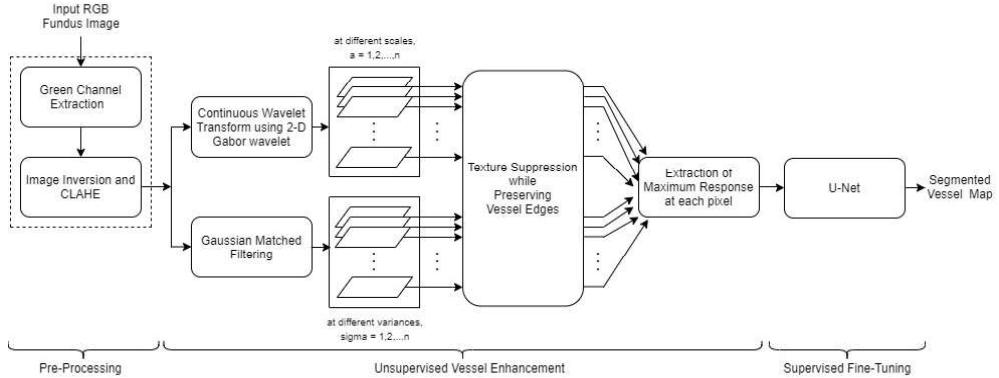


Figure 3.3: *Block diagram of the proposed vessel extraction using algorithm-II.* Here, we propose two parallel pipelines, CWT and Gaussian matched filtering to enhance vessels at multiple levels. The texture suppression preserving the vessel edges, is done at each level and we extract the common maximum response for each pixel. U-net is further used for fine-tuning.

### 3.3.1 Pre-processing

We again apply some simple pre-processing steps to prepare the input color fundus image for the proposed algorithm. We take the color fundus image and extract its green channel, which has the best contrast between vessels and background [8]. This image is further inverted, and contrast-enhanced using CLAHE [44].

### 3.3.2 Multi-level Vessel Enhancement

This step includes 2-D Gabor Wavelet Analysis at multiple scales ( $a$ ), and Gaussian Matched filtering at multiple spreads ( $\sigma$ ). We use the term ‘level’, to combine denote different values of  $a$  and  $\sigma$ . The resulting images at each ‘level’ are processed using the Local Laplacian filter. In the end, out of multi-level responses for each pixel, the maximum response value is retained.

#### 2-D Gabor Wavelet Analysis

This is a wavelet-based vessel enhancement step, the same as in algorithm-I. Here, we use the Gabor mother wavelet to enhance the oriented vessel structures of varying thickness at different scales values  $a$ . Along with the vessels, the texture of the background with a similar intensity profile is also enhanced. Figure 3.4 (a), (b) and (c)

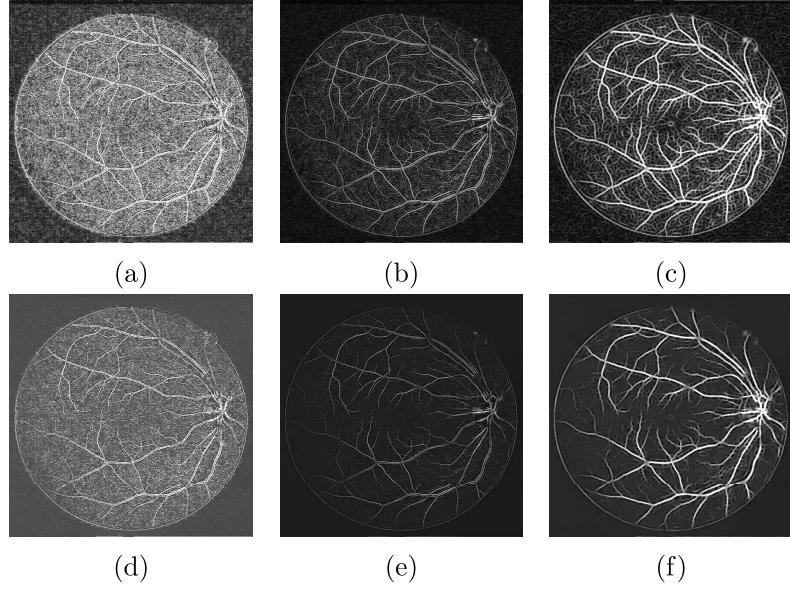


Figure 3.4: Vessel Enhancement using Gabor Wavelet at different scales : (a)  $a=1$  , (b)  $a=2$  (c)  $a=3$ ; Texture Removal at different scales : (d)  $a=1$ , (e)  $a=2$ , (f)  $a=3$

present the vessel enhancement using Gabor wavelet at three scales.

### Gaussian Matched Filtering

Assuming the cross-sectional vessel intensity profile to be Gaussian [6], we apply the Gaussian-shaped 2-D rotating kernels to the pre-processed fundus image. A 2-D Gaussian kernel is defined as [6],

$$K_\sigma(x, y) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \forall |x| \leq 3\sigma, |y| \leq L/2 \quad (3.1)$$

Here,  $L$  denotes the length of the kernel for which the vessel is assumed to have fixed orientation,  $\sigma$  represents the spread of the Gaussian curve, which covers the thickness of the vessel. This kernel assumes vessel length along the y-axis and its spread (thickness) along the x-axis ( $\pm 3\sigma$ ). Positive values of kernel denote that in our algorithm, after pre-processing, vessels are brighter than the background.

In this work, we have generated multiple Gaussian kernels corresponding to different values of  $\sigma$  for different vessel thicknesses. Each kernel is rotated at angle  $\theta$  covering all possible orientations from 0-180.  $K_\sigma^\theta(x, y)$  denotes a 2-D Gaussian kernel with spread

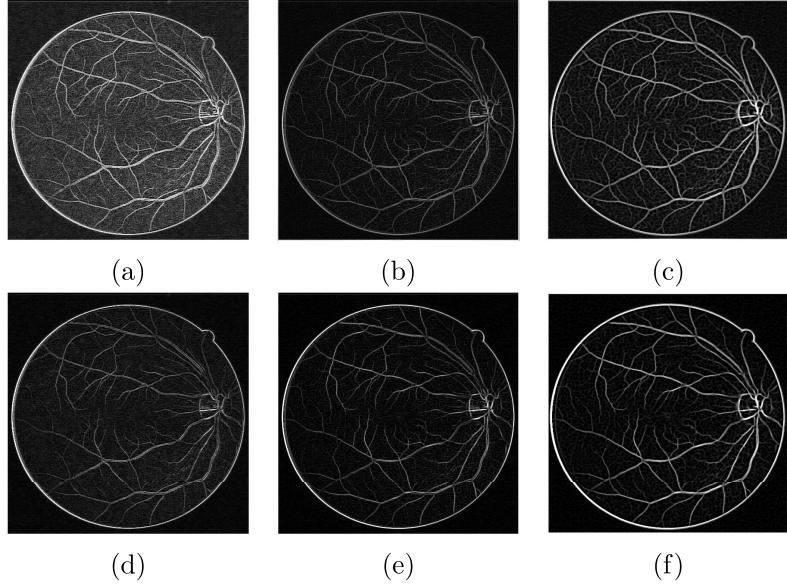


Figure 3.5: Gaussian Matched Filtering at different spreads : (a)  $\sigma=1$  , (b)  $\sigma=2$  (c)  $\sigma=3$ ; Corresponding Texture Removal at each variance : (d)  $\sigma=1$  , (e)  $\sigma=2$  (f)  $\sigma=3$

$\sigma$ , tilted at orientation  $\theta$ ,

$$K_\sigma^\theta(x, y) = \text{Rot}_\theta\{K_\sigma(x, y)\}, \forall(x, y) \quad (3.2)$$

Here,  $\text{Rot}_\theta\{\cdot\}$  denotes the operator which rotates the input kernel by angle  $\theta$  in anti-clockwise direction. For a particular value of  $\sigma$ , this kernel is matched with the pre-processed image  $I_{CLAH}(x, y)$  at all orientations  $\theta$ . The maximum response over all the angles is retained in  $G_\sigma(x, y)$  as,

$$G_\sigma(x, y) = \max_\theta[MFR\{I_{CLAH}(x, y), K_\sigma^\theta(x, y)\}], \forall(x, y) \quad (3.3)$$

Here,  $MFR\{\cdot\}$  denotes the matched filtering operator, which matches the rotating 2-D kernel with the pre-processed image. Different values of spreads  $\sigma$ , help in enhancing oriented vessels of different thicknesses. Along with these vessels, the texture of the background with a similar intensity profile is also enhanced. Fig. 3.5 (a), (b), and (c) denote the vessel enhanced images corresponding to spread values  $\sigma=1, 2$ , and  $3$ , respectively, used in the proposed work.

### Texture Suppression while Preserving Vessel Edges

At this stage, we have multiple responses at different ‘levels’ corresponding to different values of  $a$  and  $\sigma$ . These images have enhanced vessels along with many vessel-like background textures. Now, we apply edge-preserving, Local Laplacian filter [25] to suppress this vessel-like texture at each ‘level’.

Local Laplacian Filter (LLF) decomposes the input image into its corresponding Gaussian pyramids. For each pixel value in the pyramid, the input image is remapped using a point-wise function to build a Laplacian pyramid. For our purpose, an ‘S-shaped’ transformation is used to boost the vessel pixels and suppress the non-vessel pixels. This process is repeated for each pixel over all the decomposition until the output pyramid is filled to give the final response. This helps in suppressing fine vessel-like texture while preserving the strong vessel edges. Let  $LLF\{\cdot\}$  denotes the Local Laplacian Filter operator,  $S_a(x, y)$  and  $S_\sigma(x, y)$  denote the texture-smoothened output images at each level then,

$$S_a(x, y) = LLF\{R_a(x, y)\}, \forall(x, y) \quad (3.4)$$

$$S_\sigma(x, y) = LLF\{G_\sigma(x, y)\}, \forall(x, y) \quad (3.5)$$

Figure 3.4 (d), (e), and (f) show the resulting images after texture suppression at different scales  $a=1, 2$ , and  $3$ , respectively. Similarly, Fig. 3.5 (d), (e) and (f) present the images after texture suppression at  $\sigma=1, 2$  and  $3$ , respectively. The resulting smooth images at all the levels are combined by retaining the maximum value at each pixel  $(x, y)$  as  $M(x, y)$ , i.e.,

$$M(x, y) = \max\{S_a(x, y), S_\sigma(x, y)\}, \forall(x, y) \quad (3.6)$$

#### 3.3.3 Supervised Fine-Tuning

At this stage, we have an exhaustively enhanced image with suppressed texture. We feed this image to train a baseline U-net model to fine-tune our segmentation process.

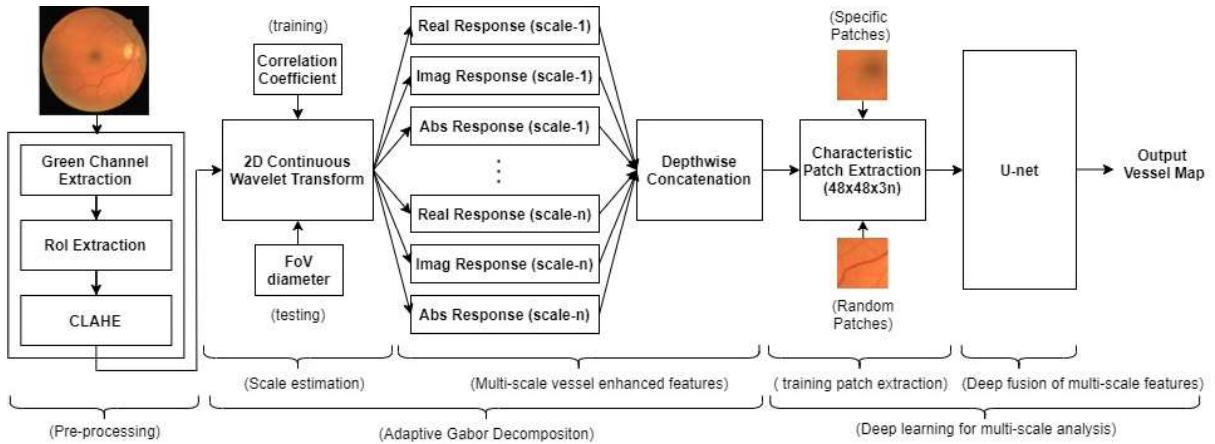


Figure 3.6: *Pipeline of the proposed Vessel Extraction using Deep-fusion of Multi-scale Features.* The pipeline is composed of three major steps: Pre-processing, Adaptive Gabor Decomposition, and Deep learning for multi-scale analysis.

Researchers have used the same model or its modifications using RGB fundus images [71], [74]. Training this model with a handcrafted, unsupervised vessel enhanced image leads to much better performance. The fine vessels and the vessel boundary pixels are already enhanced, which will help in the model learning at these low-contrast, challenging regions. We use random patches to train the model.

### 3.4 Algorithm III

After experimenting with various image processing based pre-processing steps and fine-tuning the vessel map with U-net model, our method evolved into algorithm-III. This method proposes a supervised retinal blood vessel segmentation algorithm, which uses multi-scale information to target different vessel width. Further, a U-net based deep network learns the weighted fusion of this multi-scale information to extract an optimized vessel map. As the range of vessel width depends on the resolution of input, thus, an adaptive scale-estimation is done in the beginning to generalize the algorithm. Moreover, we implement a novel ‘characteristic patch based training’, to learn the vessel structures lying in various difficult regions of fundus image. Figure 3.6 illustrates the proposed vessel segmentation algorithm as a block diagram.

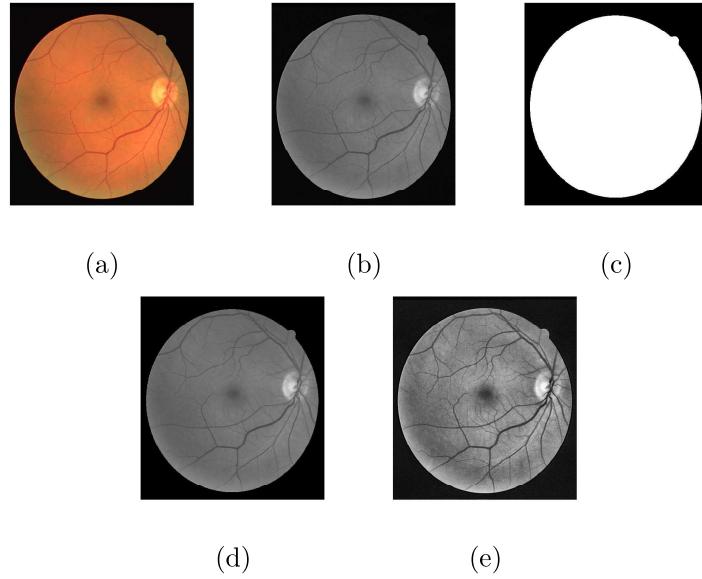


Figure 3.7: *Illustration of pre-processing of input data* : (a) Input RGB image ‘16\\_test’ from DRIVE, (b) Green channel, (c) Binary FoV mask, (d) Extracted RoI and (e) CLAHE enhanced image.

### 3.4.1 Pre-processing

The pre-processing again uses the green channel and RoI extraction followed by applying the CLAHE algorithm for contrast enhancement. All of these steps are discussed in detail in the previous algorithms. Figure 3.7 helps in visualizing the images corresponding to these pre-processing steps.

### 3.4.2 Adaptive Multi-scale Analysis

The spatially variable width of blood vessels leads us to analyse the problem of vessel segmentation at multiple scales using wavelet transforms. Moreover, fundus images coming from different sources have different ranges of vessel thickness. Estimation of best fitting scales for each input image is a crucial parameter for automated vessel segmentation. Here, we propose Gabor decomposition with adaptive scale estimation to cover fine-to-coarse vessel width, irrespective of the resolution of input image.

## Gabor Decomposition

Again in the proposed work, we use 2-D Gabor wavelet as our analysing (mother) wavelet due to its extreme sensitivity to directional features [23]. We use the inverse of pre-processed image to evaluate the CWT coefficients. The inverse image has bright vessel pixels in dark background which is the preferred way to visualise the area of interest. It is denoted as  $I'_c(x, y)$ ,

$$I'_c(x, y) = 1 - I_c(x, y), \forall (x, y) \quad (3.7)$$

Being complex in nature, the Gabor wavelet produces complex coefficients in the obtained CWT of image. We analyse these coefficients at different scales to extract the vessel information. As our vessel specific feature maps, we use the magnitude of real, imaginary and absolute responses, maximized over all possible orientations  $\theta$ , at different scales ‘ $a$ ’ denoted by,  $R_a^\psi(x, y)$ ,  $I_a^\psi(x, y)$  and  $A_a^\psi(x, y)$ , respectively, defined as,

$$\begin{aligned} R_a^\psi(x, y) &= \left| \max_{\theta} [Re\{I^\psi(x, y, \theta, a)\}] \right| \\ I_a^\psi(x, y) &= \left| \max_{\theta} [Im\{I^\psi(x, y, \theta, a)\}] \right| \\ A_a^\psi(x, y) &= \left| \max_{\theta} [Abs\{I^\psi(x, y, \theta, a)\}] \right| \end{aligned} \quad (3.8)$$

Now, we have three types of vessel-enhanced feature maps, at each scale. It is observed that, the vessels are blurry (losing boundary pixels) in the absolute response and only partly enhanced in imaginary response. Whereas, the real response is observed to intensify many sharp, fine vessel-ends, which are either lost or very poorly enhanced in other two responses (as discussed in algorithm-I). However, all the three responses enhance an overall vessel-structure, in a coarse manner. Thus, in-spite of discarding any response, we concatenate the real, imaginary and absolute coefficients at multiple scales to create a multi-channel, vessel enhanced input to train our deep model. This model, further, learns to give the best weightage to each of the feature.

## Scale Estimation

As already discussed, the images coming from different sources have different resolutions and therefore, the range of vessel width varies. To develop an adaptive multi-scale vessel extraction algorithm, we need to estimate the scaling parameter of wavelet functions based on the resolution of input image. In this regard, we evaluate the magnitude of real response at a range of scales ( $a \in [1, 10]$ ) for estimation of most significant scales. Here, we use only the real response as it better enhances the fine-to-coarse vessel map in comparison to the absolute and imaginary responses. In order to estimate the above mentioned scales, we find the correlation between the real coefficient maps at each scale and the corresponding ground truth. At scale  $a$ , the evaluated correlation coefficient is denoted by  $C_r(a)$ . Let,  $A(x, y) = R_a^\psi(x, y) - R_{mean}$  and  $B(x, y) = G(x, y) - G_{mean}$ , then, we evaluate the 2-dimensional correlation coefficient, denoted by  $C_r(a)$  as follows,

$$C_r(a) = \frac{\sum \sum A(x, y)B(x, y)}{\sqrt{(\sum \sum A^2(x, y))(\sum \sum B^2(x, y))}} \quad (3.9)$$

Here,  $G(x, y)$  is the corresponding ground truth image,  $R_{mean}$  and  $G_{mean}$  denote the 2-dimensional average value of real response and ground truth of the same input image, respectively. With the available ground truth, we find the most suitable scales for different databases. Figure 3.8 presents a bar graph to depict the average variation

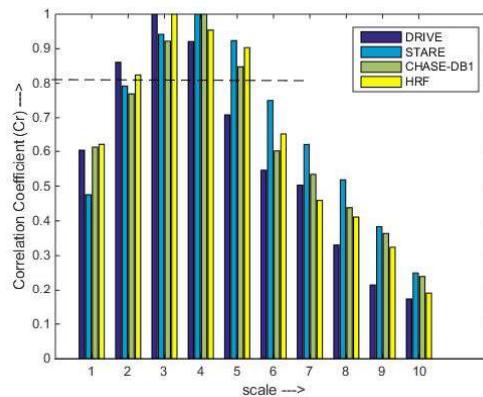


Figure 3.8: Bar graph between the correlation coefficient and scale parameter for different databases. It is clear that signal's energy is focused in only a few scales, which can be considered as the most significant ones.

Table 3.1: Fundus image datasets used in the proposed algorithm

S.No.	Dataset	Mean FoV diameter ( $d_{FoV}$ )	Most Significant Scales	Range of $d_{FoV}$
1	DRIVE [11]	526	2, 3, 4	$d_{FoV} < 550$
2	STARE [8]	593	3, 4, 5	$550 < d_{FoV} < 1100$
3	CHASE_DB1 [13]	908	3, 4, 5	$550 < d_{FoV} < 1100$
4	HRF [48]	2795	2, 3, 4, 5	$d_{FoV} > 1100$

of 2-D correlation coefficient,  $C_r(a)$  with changing scale values  $a$  for training images of DRIVE, STARE, CHASE\_DB-1 and HRF datasets. It is clearly observed that, the major portion of signal's energy is focused in only a few scales, which can be considered for signal's reconstruction. Thus, out of the given range, we consider the scales corresponding to the maximum energy as the most significant ones. This experiment with the four databases, helps in generalising the scaling parameter. Table 3.1 tells the most significant scales for each database. In order to generalize the process of estimation of the significant scales, we relate these scales to the mean FoV diameter ( $d_{FoV}$ ) of the fundus images. Assuming the FoV to be a circle, we evaluate the mean FoV diameter ( $d_{FoV}$ ) for the fundus image as,

$$d_{FoV(x,y)} = 2 \times \sqrt{\frac{Ar(x,y)}{\pi}} \quad (3.10)$$

where,  $Ar(\cdot)$  is the area of binary FoV mask, evaluated morphologically. In this manner, we calculate the mean FoV diameter of each database ( $d_{FoV}$ ). Now, we create a lookup table to relate the estimated scales to a range of  $d_{FoV}$ , corresponding to each database. We use uniform intervals to define these ranges. While testing, first we evaluate the binary FoV mask and its  $d_{FoV}$ , then, according to the matching range of diameter, the corresponding significant scales are used for multi-scale vessel extraction. Table 3.1 gives the values of mean FoV diameter, most significant scales and corresponding range of  $d_{FoV}$  for each dataset.

Deciding the scaling parameter with the variations in input, brings adaptivity to the system. Once the most significant scales corresponding to the given input are decided, we evaluate the CWT at these scales and concatenate each type of response i.e. real, imaginary and absolute response at different scales in different channels. This multi-channel image is now used to train our deep model.

### 3.4.3 Deep learning for Multi-scale Analysis

From the previous step, we have multi-scale vessel enhanced information concatenated as feature maps. To learn the best combination of these features, we train a deep model which learns their weighted fusion and further helps in fine tuning of blood vessel extraction. In the proposed work, as the base-line model, we use U-net architecture [71] as it is quite successful in the semantic segmentation, specially for the medical images.

#### Characteristic Patch Extraction

In this work, we propose a novel ‘characteristic patch extraction’ for training of the deep model. These characteristic patches are a mix of ‘specific’ and ‘random’ patches. The specific patches are supposed to target the challenging, error prone regions of fundus, whereas, the random patches facilitate the understanding of major vessel structure. As per the best of our knowledge, people have used only random patches for training [69], [71], etc. which may lose vessels lying in challenging regions of fundus. Training using such characteristic patches has a stronger potential to prepare a robust vessel segmentation model. 50% of the total number of patches per image, are extracted from the specific regions and rest 50%, randomly inside the FoV of fundus image. In specific patch extraction, we focus on four most challenging regions of the fundus image, which are as follows,

- **Category-1 : Partially inside OD**, to avoid false detection of OD boundary as vessels. We apply thresholding on the red channel of the raw input fundus image. It gives us the bright portions of retina, which includes OD, some pathological areas and other non-homogeneous bright backgrounds. Using Hough transform we detect an approximate OD boundary. Now, we select random pixels out of these boundary pixels as centre-points to make category-1 patches.
- **Category-2 : Near bright illuminations**, to discriminate between pathology and vessels; and to avoid loss of thin vessel pixels in the bright illuminating backgrounds. We exclude the OD boundary from the bright portions segmented for category-1 patches and extract the second set of specific patches.

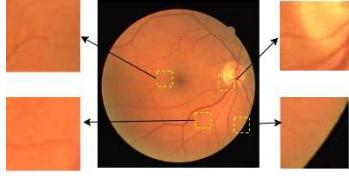


Figure 3.9: *Specific patch extraction from four challenging regions of fundus image: partially inside OD, near bright illuminations, partially inside FoV, and poorly illuminated regions.*

- **Category-3 : Partially inside FoV**, to understand the difference between the boundary of fundus and vessels. Here, we use the extracted binary FoV mask to sample the boundary pixels for patch extraction.
- **Category-4 : Poorly illuminated regions**, to learn the segmentation of low-contrast vessels in dark regions. To extract the patches from less-illuminated regions, we calculate the mean intensity of green channel of raw input image,  $I_g(x, y)$ , denoted by ( $I_g^{mean}$ ). The patches of the image with intensity values less than the  $I_g^{mean}$  value are considered as required specific patches.

The above discussed regions of retina are the ones where the chances of false vessel detection and loss of vessels are quite common. Specific patch extraction facilitates the model to better understand the vessel structure in such error prone fundus backgrounds. Fig 3.9 illustrates the patch extraction from specific challenging regions of a sample retinal image. Moreover, in order to understand the overall vessel structure, we also perform a random patch extraction for remaining 50% of the total number of patches. These patches are extracted randomly inside the FoV to make the learning better. The use of multiple patches from a single image resolve the issue of data-scarcity. Furthermore, we use augmentation techniques like varying the brightness, applying horizontal and vertical flips, to avoid the overfitting of our deep model.

### Network Model for Deep-Fusion

In the base-line U-net architecture, we modify the first  $3 \times 3$  convolutional layer to  $1 \times 1$  convolutional layer for fusing the input multi-scale feature maps. This reduces the number of trainable parameters and combines the input channels linearly via end-to-end

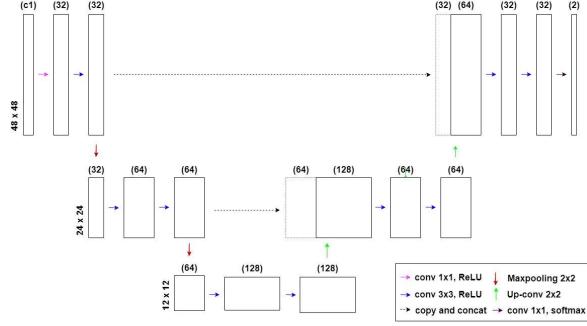


Figure 3.10: *The proposed U-net architecture for vessel segmentation using wavelet based multi-scale feature maps.*

training. The proposed model is an encoder-decoder network with skip connections. After the first layer of  $1 \times 1$  convolution with ReLU activation function, it uses multiple  $3 \times 3$  convolutional layers with the same activation function. For downsampling,  $2 \times 2$  max-pooling layers with a stride of 2 are used. Upsampling, again, uses  $2 \times 2$  convolutional layers. In the end,  $1 \times 1$  convolutional layer followed by the softmax activation function is used. Figure 3.10 illustrates the proposed U-net architecture for retinal blood vessel extraction. The model is an end-to-end framework, which learns from the annotations corresponding to each pixel. Here, we use Stochastic Gradient Descent (SGD) as optimizer and binary cross-entropy as the loss function while training, which is given as,

$$Loss = -\{y_k \log(q_k) + (1 - y_k) \log(1 - q_k)\} \quad (3.11)$$

where,  $y_k$  denotes the binary correctness indicator of predicted label corresponding to pixel  $k$ , i.e.  $y_k = 0$  or 1 for each wrong or correct prediction, respectively.  $q_k$  denotes the probability of prediction that pixel  $k$  is vessel pixel.

**Parameter analysis.** The proposed model has less number of trainable parameters than the standard U-net due to difference in the first layer. In case of standard U-net, first layer is a  $3 \times 3$  convolutional layer with say,  $f_1$  input and  $f_2$  output channels. Thus, number of trainable parameters are  $f_1 \times f_2 \times 9$  whereas that in case of a  $1 \times 1$  convolutional layer with same number of input and output channels is  $f_1 \times f_2 \times 1$ . Thus, there is 9 times reduction in the number of trainable parameters at the first layer.

Table 3.2: Step-wise contribution of the proposed method (DRIVE)

Method	Acc	Sen	Spe	F1	AUC
RGB input + random patches	0.955	0.782	0.981	0.817	0.975
Multiscale input + random patches	0.951	0.841	0.967	0.817	0.980
<b>Proposed</b> (Multiscale input + characteristic patches)	<b>0.965</b>	<b>0.872</b>	<b>0.974</b>	<b>0.818</b>	<b>0.984</b>

## 3.5 Experiments and Discussions

### 3.5.1 Datasets

In this chapter, we again use the same four open-source databases: DRIVE [11], STARE [8], CHASE\_DB-1 [13], HRF [48]. These datasets come from different sources, thus are different from each other in terms of resolution, image format, size and color. Experimenting with such a variety of database helps in developing a robust algorithm. Table 1.1 gives the specifications of these databases in brief.

### 3.5.2 Evaluation metrics

As discussed in chapter 2, the vessel extraction is a case of semantic segmentation with major class-imbalance issue. Thus, to fairly evaluate such classification problem, we need some appropriate performance metrics. In this chapter, for evaluation and comparison of the proposed work with other existing algorithms, we use five performance evaluation metrics : Accuracy (*Acc*), Sensitivity (*Sen*), Specificity (*Spe*), F-1 score (*F1*), and Area Under (*AUC*) the Receiver Operating Characteristic curve (ROC). These metrics are discussed in chapter 2, in detail.

### 3.5.3 Discussion

In case of DRIVE dataset, there is a given training-testing split of images, we use the same split in our experiments. For the STARE and CHASE\_DB-1 datasets, we follow the split as is done in the paper [46]. For HRF dataset, we decide this split as 80-20 for training and testing sets.

For pre-processing of data, we extract the green channel of input fundus image, obtain the corresponding binary FoV mask and extract the RoI. Next, the RoI is

Table 3.3: Performance comparison of proposed algorithm on DRIVE

<b>Method</b>	<b>Acc</b>	<b>Sen</b>	<b>Spe</b>	<b>F1</b>	<b>AUC</b>
Liskowski et al. [33]	0.949	0.776	0.977	NA	0.972
Orlando et al. [41]	NA	0.790	0.968	0.786	0.951
Yan et al. [50]	0.954	0.765	0.982	NA	0.975
Ronneberger et al. [71]	0.955	0.782	0.981	0.817	0.975
Jin et al. [74]	0.956	0.786	0.980	0.819	0.978
Li et al. [39]	0.957	0.773	0.984	0.820	0.982
<b>Proposed method</b>	<b>0.965</b>	<b>0.872</b>	<b>0.974</b>	<b>0.818</b>	<b>0.984</b>

Table 3.4: Performance comparison of proposed algorithm on STARE

<b>Method</b>	<b>Acc</b>	<b>Sen</b>	<b>Spe</b>	<b>F1</b>	<b>AUC</b>
Liskowski et al. [33]	0.957	0.787	0.975	NA	0.978
Orlando et al. [41]	NA	0.768	0.974	0.764	0.971
Yan et al. [50]	0.961	0.758	0.985	NA	0.980
Ronneberger et al. [71]	0.964	0.668	0.991	0.759	0.971
Jin et al. [74]	0.964	0.681	0.990	0.763	0.976
Li et al. [39]	0.970	0.771	0.989	0.815	0.988
<b>Proposed method</b>	<b>0.964</b>	<b>0.819</b>	<b>0.978</b>	<b>0.802</b>	<b>0.978</b>

contrast enhanced using CLAHE algorithm. For scale estimation, we use the observed binary FoV mask and calculate its diameter,  $d_{FoV}$ . This diameter is used to predict the significant scales (using Table 3.1). After pre-processing and deciding the scaling parameter, we evaluate the 2-D CWT of pre-processed image using Gabor wavelet at the estimated scales. The enhanced vessel maps corresponding to real, imaginary and absolute coefficients at all the estimated scales are concatenated as separate channels. The ‘n’ number of scales produce ‘3n’ number of channels. This multi-channel feature map is fed to the U-net model via an initial  $1 \times 1$  convolution layer, which learns to fuse the multi-scale input.

To train the model, we perform the proposed ‘characteristic patch extraction’ on the obtained feature maps. We extract 50% patches from the specific challenging regions of fundus image. Remaining 50% patches are extracted randomly, but inside the FoV. The patch-size is an important parameter to decide on the performance of the proposed method. Using a big patch will need more pooling layers, which in turn affects the model’s localization. On the other hand, using a small patch may lead us into losing the context. In order to achieve a good localization and context, we take equal size patches

Table 3.5: Performance comparison of proposed algorithm on CHASE\_DB-1

Method	Acc	Sen	Spe	F1	AUC
Orlando et al. [41]	NA	0.728	0.971	0.733	0.948
Ronneberger et al. [71]	0.964	0.784	0.982	0.799	0.981
Jin et al. [74]	0.964	0.786	0.982	0.800	0.983
Li et al. [39]	0.965	0.797	0.982	0.807	0.985
<b>Proposed method</b>	<b>0.971</b>	<b>0.799</b>	<b>0.983</b>	<b>0.786</b>	<b>0.981</b>

of size  $48 \times 48$  for DRIVE and STARE datasets [71]. For the other two datasets, which are approximately twice and four times of the DRIVE and STARE datasets, we take multiples of 2 and 4 for patch-size. To resolve the data constraints and thereby to avoid any possible overfitting of the deep model, we use data augmentation by varying the brightness and applying horizontal, vertical flips. We keep model’s hyper-parameters same for all the datasets, in order to avoid fine tuning with changing input. This approach is adopted to develop a generic model for vessel segmentation. Moreover, the proposed approach is compatible with datasets of different resolutions, thereby, we have avoided any information losses which generally occur due to data-resizing.

To analyse the efficacy and contributions of the proposed adaptive multi-scaling and characteristic patch-based approach, we evaluate the performance measures for training the same baseline deep model with : random patches of RGB input fundus images (i.e. without multi-scaling and without specific patches); random patches with multi-scaling (i.e. with multi-scaling and without specific patches) and the proposed characteristic patches with multi-scaling. Table 3.2 presents the values of performance measures corresponding to each case. It can be seen that the values of sensitivity and AUC metrics increase with multi-scaling and they further improve with the characteristic patch-based training. As discussed in chapter 2, better sensitivity means less  $FN$ , which means better vessel-extraction in error prone fundus background. Hence, it can be said that the proposed adaptive multi-scaling and use of characteristic patches has contributed in improving the overall performance of the base-line U-net model.

Subsequently, we compare the performance of proposed vessel segmentation approach with some state-of-the-art methods. Table 3.3, 3.4, 3.5 and 3.6 present the performance comparison of the proposed work on DRIVE, STARE, CHASE\_DB-1 and

HRF datasets, respectively. It includes five metrics : Acc, Sen, Spe, F1-score and AUC under ROC curve. It can be seen that the proposed method has performed quite well, specially in terms of sensitivity measure. The training of deep model using characteristic patches at multiple-scales has helped in improving the vessel extraction in challenging regions of fundus. In terms of computation cost, with a Xeon W-2195, 2.30GHz processor, the computational time for the proposed pipeline was 0.63 s, 0.69 s, 0.84 s, and 0.92 s for DRIVE, STARE, CHASE\_DB-1, and HRF datasets, respectively.

Along-with the intra-database testing, we also evaluate the algorithm's performance in cross-database environment. The databases corresponding to similar FoV diameters,  $d_{FoV}$ , will have same range of significant scales, thus, we cross-test them. Table 3.7 gives three such cases and their performance. As the STARE and CHASE\_DB-1 databases share same range of scales, the cross-testing performance is better as compared to that in the case of STARE and DRIVE databases. Overall, we observe a promising performance of the proposed algorithm in cross-database testing.

Further, we plot the Receiver Operating Characteristic curve for the four databases, shown in Figure 3.11. It shows the trade-off between the sensitivity and specificity measures. The best  $AUC$  is observed for DRIVE database with a value of 0.984. The minimum is for STARE, 0.978. Only slight deviation of curves from best to worst performance over four different databases, prove the stable and robust performance of the proposed method.

After numerical analyses, we present the visual analyses of the proposed vessel segmentation algorithm. Figure 3.12, 3.13, 3.14 and 3.15 illustrate a qualitative comparison of vessel segmentation results on sample images taken from each of the four databases. Each figure comprises of the input RGB image, the ground truth, our predicted vessel map, along-with two closest performing state-of-the-art vessel maps.

Table 3.6: Performance comparison of proposed algorithm on HRF

<b>Method</b>	<b>Acc</b>	<b>Sen</b>	<b>Spe</b>	<b>F1</b>	<b>AUC</b>
Odstrcilik et al. [48]	0.949	0.779	0.965	0.732	0.968
Orlando et al. [41]	NA	0.787	0.958	0.716	0.952
Yan et al. [50]	0.943	0.788	0.959	NA	NA
<b>Proposed method</b>	<b>0.969</b>	<b>0.825</b>	<b>0.984</b>	<b>0.831</b>	<b>0.980</b>

Table 3.7: Cross-database performance of proposed algorithm

Test data	Train data	Acc	Sen	Spe	F1	AUC
STARE	CHASE_DB-1	0.937	0.765	0.982	0.759	0.956
CHASE_DB-1	STARE	0.954	0.823	0.964	0.789	0.970
STARE	DRIVE	0.931	0.812	0.940	0.769	0.940

Moreover, in order to focus on the performance of the proposed algorithm in the challenging regions, we have shown two zoomed blocks per image. Fig. 3.12 and 3.13 focus on the segmentation of vessels in low-contrast and less illuminated backgrounds. In Fig. 3.14, vessel segmentation in bright backgrounds (inside OD) and dark backgrounds is observed. Fig. 3.15 presents the proposed vessel segmentation in low-contrast regions and near FoV boundary. The proposed algorithm is seen to successfully segment the fine and continuous thick vessels in various difficult regions of the retina whereas the other algorithms are losing on the fine vessels.

## 3.6 Chapter Summary

This chapter approaches the problem of vessel segmentation in a supervised manner. Here, we start with multi-scale pre-processing of fundus images and evolve from su-

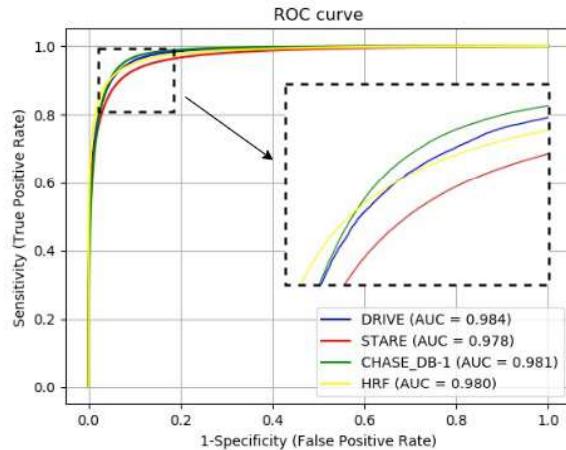


Figure 3.11: *ROC plot for the four different databases. Curves are deviating very slightly which prove the consistent thus robust performance of the proposed algorithm.*

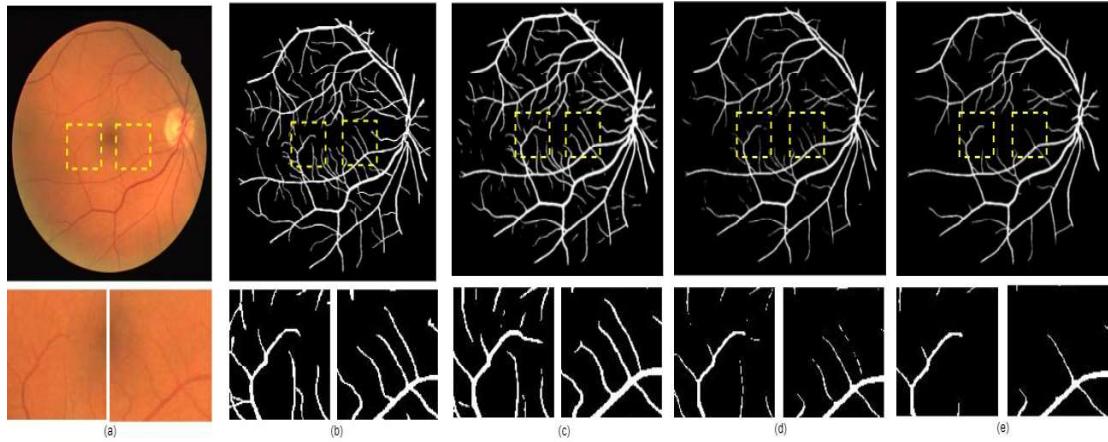


Figure 3.12: Qualitative comparison of vessel segmentation on DRIVE database : (a) RGB image ‘16\\_test’, (b) Ground truth, (c) Our predicted vessel map, (d) Vessel map predicted by Orlando et al. [41], (e) Vessel map predicted by Li et al. [39]

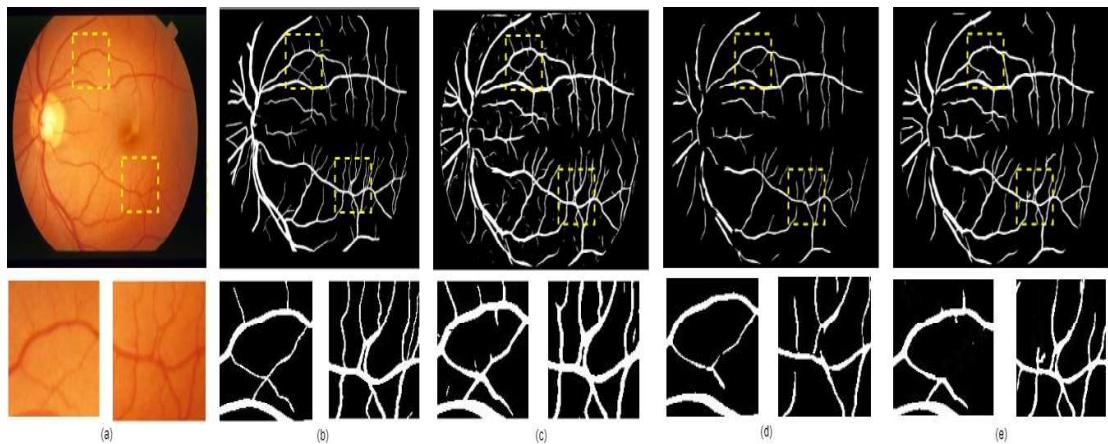


Figure 3.13: Qualitative comparison of vessel segmentation on STARE database : (a) RGB image ‘im65’, (b) Ground truth, (c) Our predicted vessel map, (d) Vessel map predicted by Liskowski [33], (e) Vessel map predicted by Li et al. [39].

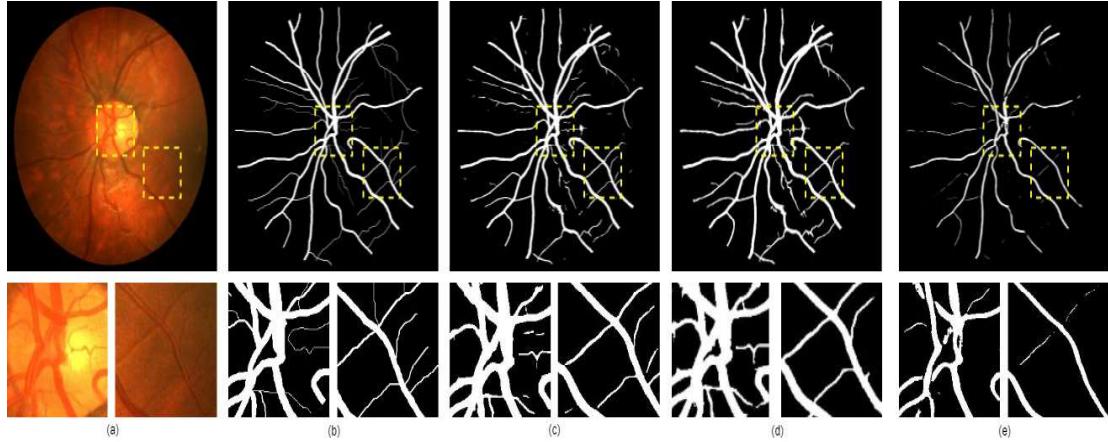


Figure 3.14: Qualitative comparison of vessel segmentation on CHASE\_DB-1 database : (a) RGB image ‘Image\_01L’, (b) Ground truth, (c) Our predicted vessel map, (d) Vessel map predicted by Jin et al. [74], (e) Vessel map predicted by Li et al. [39].

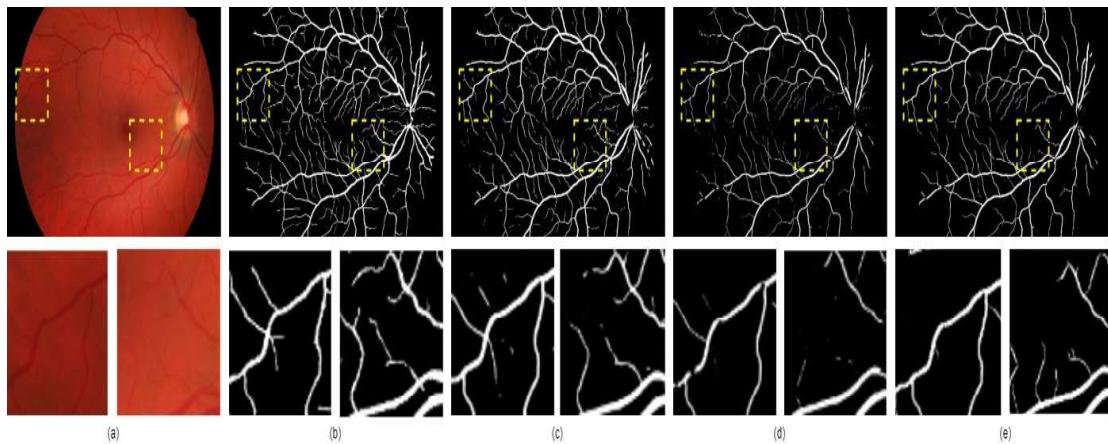


Figure 3.15: Qualitative comparison of vessel segmentation on HRF database : (a) RGB image ‘11\_H’, (b) Ground truth, (c) Our predicted vessel map, (d) Vessel map predicted by Yan et al. [50], (e) Vessel map predicted by Li et al. [39].

pervised algorithm-I to II, into our final vessel segmentation algorithm-III. The final algorithm presents a promising retinal blood vessel extraction by targeting the two associated major challenges : continuously varying thickness of vessels and in-homogeneous retinal background. To cover fine-to-coarse vessel thickness, we do adaptive scale estimation for multi-scale feature extraction. This adaptive multi-scale analysis leads into a generic vessel segmentation. Further, a deep-model learns the weighted combination of extracted multi-scale features using ‘random’ and ‘specific’ termed as characteristic patches. Training using these characteristic patches helps the model learn vessel structure in non-homogeneous retinal background. Various evaluations of the proposed algorithm have shown significant improvements in the state-of-the-art vessel extraction.