Automation of Diabetic Lesions and blood vessels using Unet and Gaussian Pyramids

Shashank Reddy Boosi*, David Sison*, Harmanpreet Singh*, Aravind Murugesan*, Allen kombasseril*
* School of Computing
University of New South Wales
{z5222766, z5019783, z5228917, z5175965, z5232188}@student.unsw.edu.au

Abstract—Task 1 - **Diabetic retinopathy, known as diabetic eye disease, is a medical condition which damages the retina due to diabetes mellitus. This report is an attempt to solve the problem where we use Deep Learning to segment the four categories of lesions in a retinal image. We present a network and training strategy that relies on the strong use of data augmentation to use the available annotated samples efficiently. The architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. The model used is a fine tuned U-Net which is an enhanced encoder-decoder CNN model and we attained Jaccard Similarity of 0.3093 and a Dice Similarity of 0.3838 respectively.**

Task 2 - **The segmentation of blood vessels can be used for various retinal imaging. We present a report of our study of a Resolution Hierarchy based method. The method reduces computational needs and could detect vessel of different thickness by using a Gaussian resolution hierarchy. The algorithm reported an accuracy of 87.5% while taking less time to run.**

Keywords: **Diabetic Retinopathy, Blood Vessels, UNet, Gaussian Pyramid.**

I. TASK 1

A. Introduction

Diabetic retinopathy, a chronic, progressive eye disease, has turned out to be one of the most common causes of vision impairment and blindness especially for working ages in the world today [1]. Retinal abnormal conditions are associated with lesions namely microaneurysms, haemorrhages, hard exudates and soft exudates. The diagnosis in the early stages of the disease is particularly important to receive proper treatment and reduce the chance of progression of the disease to advanced stages which can lead to permanent loss of vision in extreme cases [2]. The presence of these lesions is highly correlated with DR, so detection of their presence is crucial. Retinopathy detection system is usually accomplished by involving a well-trained physician manually detecting vascular abnormalities and structural changes of retina in the retinal fundus images. These results are closely observed by expert readers but they often inconsistent as the method is manual in nature [3]. So an automated method is necessary to identify these defects.

Most of the problems of computer vision of recent times have been solved with great accuracy with the help of modern deep learning algorithms, Convolutional Neural Networks (CNN's) to name one. Here we are using a tweaked version of the Fully connected convolutional neural network named UNET, which was developed by Olaf Ronneberger et al [4], which have shown impressive results for biomedical image segmentation problems with lower number of training images.

B. Related Works

Automatic diabetic retinopathy detection system can be achieved by various different algorithms and methods. Earlier solutions were based on hand-crafted feature extraction and standard machine learning algorithm for prediction [5]. C.Sinthanayothin used Recursive region growing segmentation algorithms along with the use of a new technique , named as 'Moat Operator' to detect defects such as haemorrhages, microaneurysms and hard exudates. They ended up with a sensitivity and specificity for exudate detection of 88.5% and 99.7% respectively . They could detect microaneurysms, haemorrhages with a sensitivity and specificity of 88.7% and 77.5% respectively.

Sheikh Muhammad Saiful Islam used Deep learning methods along with data preprocessing and data augmentation for Early Detection and Grading of Diabetic Retinopathy [3]. Their network achieved 98% sensitivity and more than 94% specificity in early-stage detection and on the challenging Kaggle EyePACS dataset [6].

There was also a challenge [7] hosted on Grand Challenges in Biomedical Imaging Platform.. One of the challenges in this competition was mainly to segment the lesions in a retinal image. For lesion segmentation task, most of the participating teams used U-Net architecture [4]. Unsurprisingly, the best approach for lesion segmentation in this competition used U-Net, with data augmentation boosting results significantly. It resulted in AUC of 0.8263, 0.9716, 0.9540 and 0.9883 for MA, HE, SE and EX respectively.

Olaf Ronneberger demonstrated the application of the U-NET to three different segmentation tasks [4]. One of it was a cell segmentation task in light microscopic images. This segmentation task is part of the ISBI cell tracking challenge 2014 and 2015 [8], [9]. Here they achieved an average IOU (Intersection over Union) of 92%.

This motivated us to use U-Net for the given task.

C. Methodology

Fig. 1 shows us an overview of the methodology pipeline. Our methodology consists of various steps: Image augmentation, Image pre-processing, and Modelling where we start with

preparing the dataset, augmenting using selective resampling, preprocessing, fine tuning the model, training and testing. The data-set is taken from IDRiD Grand Challenge [7]. We mainly used imgaug, opencv and pytorch [10] and pandas library for all the steps involved in the methodology. Section I-C1 to section I-C6a consist of detailed information of the steps.

1) Data Description: We have a total of 54 train images and 27 test images which is a multi class segmentation problem.

The input images contains the 3 channel retina and the label images contain 4 images of 1 channel diabetic lesions namely Microaneurysms, Haemorrhages, Hard Exudates and Soft Exudates per every input image.

2) Merge Segmentation masks: A single segmentation mask was generated from the available segmentation masks per image by stacking binary masks across an axis and taking the argmax per pixel. This resulted in integer encoding of the 4 categories of the lesions and the background.

- **0** - Background
- **1** - Haemorrhages
- **2** - Hard exudates
- **3** - Microaneurysms
- **4** - Soft Exudates

3) Image Augmentation: Data augmentation is essential to teach the network the desired invariance and robustness properties, when only few training samples are available. After generating the segmentation masks, it was found that there was a huge imbalance between the background data and the other categories. To counter this, using visual cues and masks' histograms, 16 images were selected to augment the data in the following way,

- **Crop** Images were cropped from the left and right sides to remove background.
- **Geometric** Randomly apply

  1) Rotation between 0^0 to 360^0
  2) Shear with angle between -20^0 and 20^0
  3) Flip horizontal or vertical
  4) Shift images between -25 and 25 pixel
  5) Zoom image on a random scale

- Finally, images(including the original) were partitioned into 4 parts and resized to 256x256

4) Image preprocessing: Due to different Lighting conditions and capture setup of the images , these images suffer a lot of exposure and lighting variation. To normalize these effects we subtracted our local average colour as shown in fig 2 first image. The result obtained was further processed using contrast limited adaptive histogram equalization(CLAHE) as shown in fig 2 second image. This is a special technique in which we increase the local contrast of the image without amplifying any noise in the image . This is a modified version of Adaptive Histogram equalization(AHE) where they limit amplification to not over amplify any noise like how CLAHE would. The resultant image obtained after these two steps are visibly enhanced to bring out the lesions to be segmented.
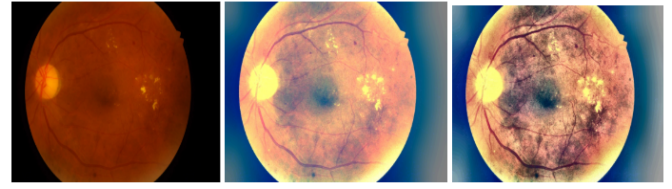

Fig. 2. Augmented Image to Normalization to CLAHE

5) UNet Modelling:

- After the preprocessing is done we feed the images to the customized pytorch IDRiD dataset which will return us a dictionary images of the dimension
  Images: (B, C, H, W) **Masks**: (B, H , W)
- We will split the dataset to train and validation with 20 percent being the validation.
- We have 940 train and 236 test images and then feed it to the customized UNet model.
- **Model Details**

  - **Optimizer** Adam
  - **Learning Rate** 0.01
  - **Loss** Dice Loss

U-Net [4] is a convolutional neural network that was developed for biomedical image segmentation which is a fully convolutional network. It is a U- shaped architecture which is famous for its upsampling part (decoder) and usage is the skip connection which are introduced in Resnet [11] from the downsampling part to the upsampling part which will help it to store the features from the down sampling as it helps to resize the image in an efficient manner. Fig 3 shows the traditional UNet architecture.
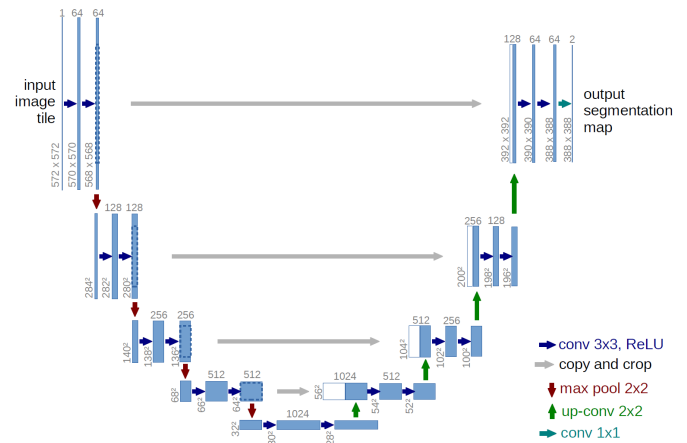

Fig. 3. UNet Architecture

As the input and the output dimensions of the UNet are not of the same dimension we fine tuned the UNet and customized it to return a dimension which is the same as input for us to compare with the label. Fig 4 custom Unet below is an example of the customized UNet architecture.
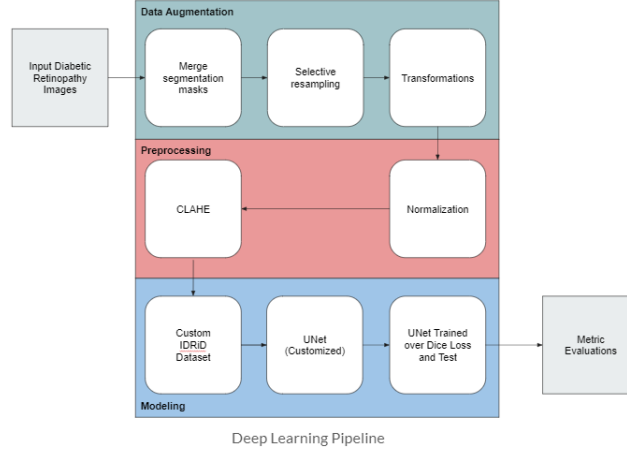
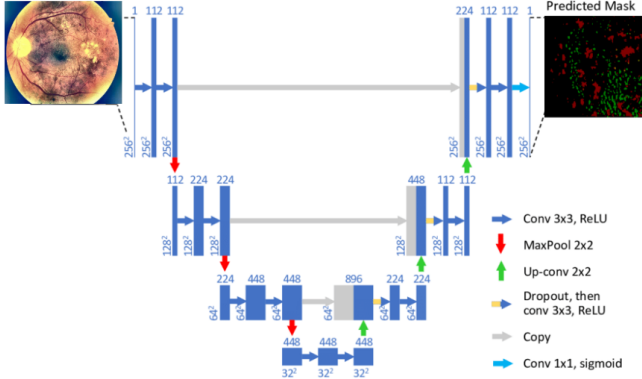Fig. 1. Deep Learning Pipeline Overview

Fig. 4. Customized UNet Architecture

6) Experiment Setup: The network architecture of our U-net is shown in Fig 3. In deep networks with many convolutional layers and different paths through the network, a good initialization of the weights is extremely important. Hence, we used Xavier initialization. It works because we want the variance of the input to be as the variance of the output.

Batch Normalization was added as it not only result in faster training but also helps to reduce the sensitivity to the initial starting weights.

Dropout was added at the bottleneck as it reduces the tendency for the model to overfit by providing a form of regularization.

The network was trained with Adam with a fixed schedule over 50 epochs. The learning rate for Adam was gradually changed from 0.0001 to 0.01 as the loss function was converging very slowly. To fit the model in our GPU memory (3 GB), we reduce the batch to a single image of size 256x256.

For hyper-parameter optimization, we used a random search over the respective ranges by running a few epochs (5 or 10) with cross validation.

To train the network, cross entropy loss function was used initially, but the results were pretty ordinary. We believe it

is because of the class imbalance present in the original input data. We experimented with weighted cross entropy but setting up the weights for this task turned out to be difficult with minimal to no improvement in results. Finally, a resampling strategy was devised to counter this and the loss function was changed to soft dice which is better at handling such cases.

a) Metrics: The combination of all the steps are evaluated under different performance metrics, including accuracy, accuracy per class, Jaccard Similarity Coefficient, Dice Similarity Coefficient for the multi class Segmentation. The metrics are defined as following,

$$JSC = \frac{|S \cap T|}{|S \cup T|} = \frac{|TP|}{|TP| + |FP| + |FN|} \tag{1}$$

$$DSC = 2\frac{|S \cap T|}{|S| + |T|} = \frac{2|TP|}{2|TP| + |FP| + |FN|} \tag{2}$$

$$Accuracy = \frac{1}{n}\sum_{i=1}^{n}\frac{|X_i \cap Y_i|}{|X_i \cup Y_i|} \tag{3}$$

where TP is the true positive, FP is the False Positive, FN is the False Negative, X_i is the set of predicted labels, Y_i is the set of true labels and n is the number of samples.

D. Evaluation and Results

Table I shows us the model performance in terms of metrics for UNet we have experimented. The metrics used for evaluation are overall accuracy, accuracy per class, Jaccard Similarity, Dice Similarity as shown in the section I-C6a. As we can see, the comparison between the train and test helped us understand the nature of how UNet is acting and predicting without over-fitting or under-fitting based on the metrics. The overall accuracy is just a measure to detect the whole image including the background but our goal is to detect the diabetic lesions without the background. So I used a metric called accuracy per class where every class used has a weight over the others. But the most famous metrics for semantic segmentation

is the Jaccard Similarity and Dice similarity which will not include the background while only calculating the intersection over the union. Comparison between other papers is not done because most of the papers use the full dataset if IDRiD and comparing those papers with our implementation would not be fair.

Metrics	Train	Test
Overall Accuracy	0.9460	0.9272
Accuracy per class	0.4570	0.4130
Jaccard Similarity	0.3431	0.3093
Dice Similarity	0.4253	0.3838

TABLE I
MODEL PERFORMANCE OF UNET

The main reason to get the Jaccard index of 0.3093 is because of the amount of data provided to us and the limited amount of features that can be obtained by the model. Another reason is the amount of background in every image is dominating the actual classes that we are detecting. Fig 5 shows how the metrics are acting for each epoch.
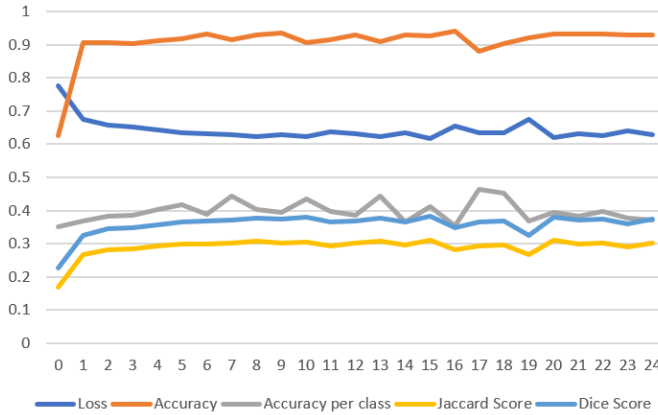

Fig. 5. Epoch vs Metrics Graph

Figure 6 and 7 are the examples for the images with good predictions and the bad predictions. The main difference noticed between the good and the bad predictions is that the images which has a good predictions contains a lot of diabetic lesions covered and for the bad predictions most of the image is covered with the background leading to the wrong prediction. We mitigated this effect by using a concept called selective resampling but removing this problem completely can only be done if we can increase our training data.
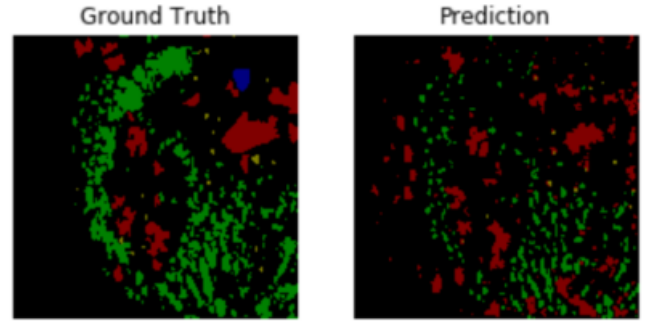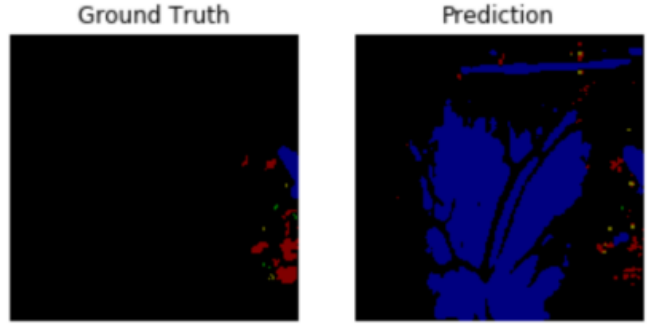

Fig. 6. Image with good prediction

Fig. 7. Image with bad prediction

E. Conclusion and Future Work

In this paper, the problem of summarizing UNet Deep Learning model which is used for retinal lesions associated with Diabetic Retinopathy in Augmented IDRiD datasets is discussed. The focus is mainly on using customized UNet to effectively segment the retinal lesions associzted with Diabetic Retinopathy [?] . For prediction we used an augmented dataset with 940 train images and 236 validation images and it is pre-processed to remove noise and contrasted with CLAHE. The outcome of the predictive results on the IDRiD data-set reveals the accuracy, accuracy per class, Jaccard Similarity Coefficient and Dice Similarity Coefficient with a score of 92.72, 41.30, 30.93, 38.38 % respectively. The second conclusion gives us an intuition of how Dice loss performed better than the other loss functions like Binary cross entropy without weights, pixel wise cross entropy using weights.

The proposed work can be enhanced by expanding its scope to using Transfer Learning with pre trained models like Resnet 50, Resnet 101 or Inception v3. Hyper parameter Tuning the weights for the data imbalance will also improve the Pixel wise Cross Entropy Loss Function and finally we can use different type of Segmentation models like Residual nets, FCN etc.,

II. TASK 2

A. Introduction

Eye disease is one of the leading causes of blindness and can often be prevented by early detection. In common practice this is done by opthamologists by visual inspection alone, through

the analysis of photos of the back of the eye taken using a fundus camera. Computer vision techniques can be used to speed up the process of diagnosis by trained professionals by automatically segmenting and tagging relevant regions of interest. In this task we aim to implement a segmentation algorithm to automatically and accurately segment the blood vessels in a fundus image.

Although this problem, like many others within the image segmentation and classification fields, can be solved by using a convolutional neural network (CNN), we have decided to implement a standard segmentation algorithm instead. This is in part motivated by the desire to avoid a similar implementation to Task 1, but also by the results from implementations described in the papers from our literature review. We have been providing a total of 20 training images and 20 testing images, but as our algorithm requires no training we instead use all of them for testing.

B. Related Works

Attila Budai, Georg Michelson, and Joachim Hornegger's paper "Multiscale Blood Vessel Segmentation in Retinal Fundus Images" [12] describes a relatively simple segmentation method that served as the primary inspiration for our implementation. Their method consists of three main stages: the creation of a gaussian pyramid for the input image, neighbourhood analysis which is performed an all levels of the pyramid, and finally binarization and fusion of each image to create the output image mask.

In the first stage, a gaussian resolution hierarchy consisting of 3 levels (in which the first level is the input image) is created. The green layer is the only layer analysed as it provides the largest contrast of blood vessel to background. Neighbourhood analysis is performed in each level of the hierarchy to detect vessels of different width, which results in a corresponding vessel enhanced image for each layer. The layers are then upscaled to the original resolution and hysteresis thresholding is applied before fusing using a pixel-wise OR operator to create the final output image mask.

The neighbourhood analysis described in this paper consists of computing the hessian matrix for each pixel. In this case, as our image is two-dimensional the hessian would be the 2x2 matrix of second order derivatives shown below. In calculus second order derivatives are often used to represent the curvature of a discrete function. The paper proposes the use of the eigenvalues of the hessian function to calculate the output vessel pixel as "they are reflecting the scale of the lowest curvature and the highest one in the neighbourhood" [12], using the formula below.

$$H(f) = \begin{pmatrix} \dfrac{\partial^2 f}{\partial x^2} & \dfrac{\partial^2 f}{\partial x \partial y} \\ \dfrac{\partial^2 f}{\partial x \partial y} & \dfrac{\partial^2 f}{\partial y^2} \end{pmatrix} \qquad P_{vessel} = 1 - a_l/a_h$$

Applying this algorithm in our experimentation to a single layer alone we found the results to be quite noisy, as shown Figure 8, though through visual inspection we could see vessel pixels correctly being identified. Attila Budai, Ruediger Bock, Andreas Maier, Joachim Hornegger, Georg Michelson's paper "Robust Vessel Segmentation in Fundus Images" [13], of which Budai and Michelson share authorship with the first paper, describes a similar but more robust implementation.
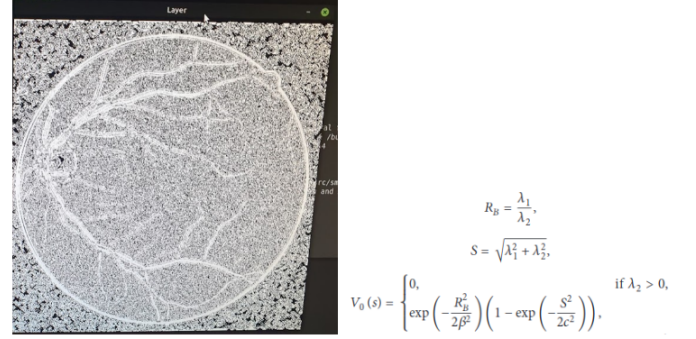

Fig. 8. Segmentation output without resolution hierarchy

In addition to a different "vesselness" function for calculating the value of each pixel on the output layer, pre-processing and post-processing steps are described. For pre-processing, histogram stretching and bilateral filtering was applied to the input layer to provide a larger contrast of colours for the algorithm to process, and to smooth out noise that could create potential false positives for vessel detection. After creation of each output layer, morphological post processing was conducted to fill gaps in the vessels and thin them out, and again after image combination to increase accuracy.

C. Methodology

The pipeline of our approach is defined in Fig 9 The processes are classified into pre-processing, vesselness extraction and post-processing which are explained below.

1) Image preprocessing: The input images for this experiment are digital color fundus photographs. Our analysis are restricted to green channel as this offers the highest contrast between vessels and background. Histogram stretching is performed on the green channel to increase contrast helping the algorithm to detect small changes. Further bilateral filtering is applied to denoise the image by smoothing intensity changes while preserving the boundaries.

2) Vesselness Extraction: Hessian analysis is performed on 3x3 windows and Hessian matrix is calculated for the neighbourhood window of each pixel. The eigenvalues of the Hessian matrix are calculated which signifies the lowest and highest curvature in the neighbourhood. The vesselness feature of the center pixel is calculated applying Frangi's formula.

The hessian analysis is applied on all layers in the resolution hierarchy. The outputs are resized to the original image size before pipelining them to post processing.
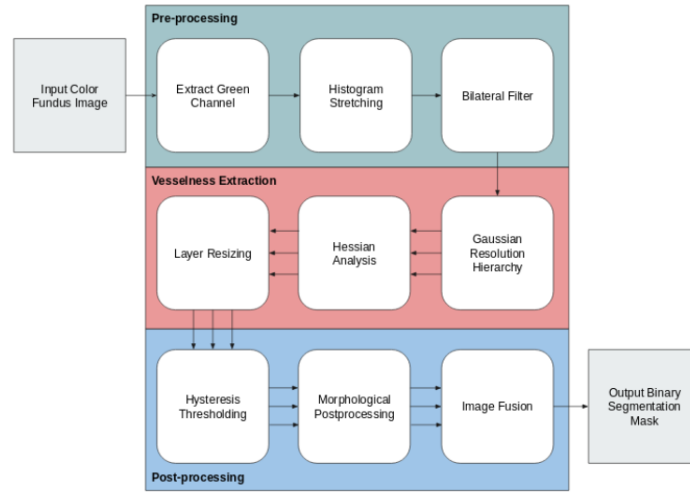
Fig. 9. Gaussian Pyramid Pipeline Overview

3) Image post-processing: After performing Vesselness Extraction, the resized image from each of the Gaussian Resolution Hierarchy layer is resized to the original image's size. We perform a thresholding technique, called Hysteresis thresholding proposed by Canny [14]. The hysteresis method uses two thresholding values to binarize the image unlike a simple thresholding method. The first threshold value, which is the higher of the two, is used to determine high contrast parts of the image. All pixels above the first threshold value is labeled as vessel pixels. Pixels between first and second threshold values are considered as vessel pixels only if it's connected in the image to a pixel labeled as vessel pixel by the higher threshold value. Hysteresis thresholding thus achieves better results when the fundus image has areas of low contrast where it's normally difficult using a simple thresholding to differentiate between vessel and noise. The thresholding has been performed using different values of thresholds for each layer.

Then we perform morphological operations, thinning and morphological closing on all the gaussian pyramid layers. Thinning is done to account for over segmentation of thin vessels caused usually in higher layers. So this operation is selectively applied only on higher layers. Morphological closing smoothens the boundaries and small gaps inside the vessels are filled while removing unwanted images.

The final segmented image is obtained by doing a bitwise OR operation on the thresholded image for all resolution hierarchy. Vessel detected in one of the resolution hierarchy therefore will appear in the final output.

D. Experimental Setup

Initial experimentation of Hessian analysis on the raw original images resulted in noisy segmentation. Preprocessing input images using histogram stretching and bilateral filtering helped by increasing contrast and reducing noise while preserving edges. Images in Fig 10, 11 and 12.
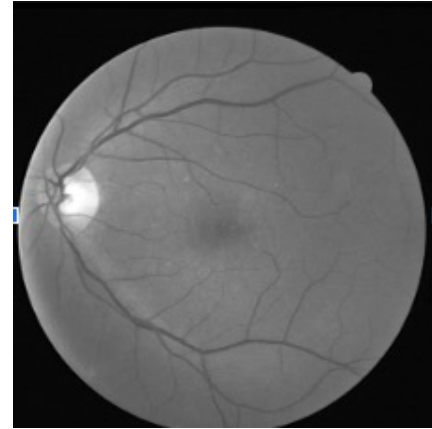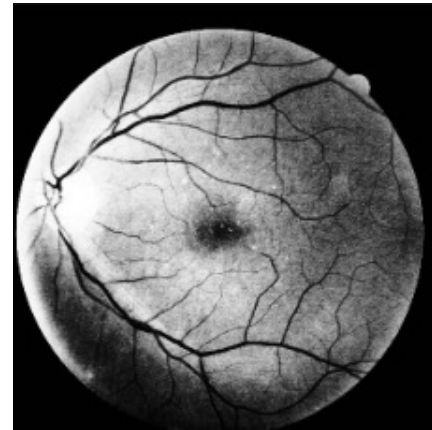

Fig. 10. Original image

Fig. 11. Histogram Stretching

Fig. 12. Bilateral Filter

Hessian analysis was done on 3x3 neighbourhoods in each of the resolution hierarchy layers. The effectiveness of our method relied on finding the right values for the constants in the Frangi's formula. We found that running analysis of all resolution hierarchy layers did not produce the best results. So unique values were used for each layers so that the larger layers in the pyramid captures more information while smaller layers capture the thicker vessels with more accuracy.

The idea of using different parameters for the layers also improved our hysteresis thresholding. After carefully tweaking we found the values in the table giving us the best results. Finding good values required some manual observation of output images from each resolution hierarchy layers. The values we used are given in Table II.

After combining the information found from all resolution hierarchy layers, a final morphological closing operation performed removed some of the gaps inside the segmented vessels.

E. Evaluation and Results

The metrics obtained on running on 20 images are shown in Table III. The manual segmentation provided were used as the ground truth to calculate true positives, true negatives, false positives, and false negatives.

Accuracy	87.49
Sensitivity	52.43
Specificity	90.8

TABLE III
ACCURACY METRICS

F. Conclusion and future works

A multiresolution method was implemented for segmenting blood vessels in fundus photographs. The method offers good accuracy and sensitivity with reduced computational complexities.

Accuracy gains with the method because lower resolution images could capture thick vessels better. The computational

demands were reduced as a 3x3 window could be effectively used with resolution hierarchy.

Future works involves reducing false positives by doing Specular Reflex Correction to account for bright specular reflex caused by camera flash in some fundus photographs. Tweaking the kernel for morphological post processing can further reduce noise.

REFERENCES

[1] N. G. Congdon, D. S. Friedman, and T. Lietman, "Important causes of visual impairment in the world today," *Jama*, vol. 290, no. 15, pp. 2057–2060, 2003.

[2] A. Melville, R. Richardson, J. Mason, A. McIntosh, C. O'Keeffe, J. Peters, and A. Hutchinson, "Complications of diabetes: screening for retinopathy and management of foot ulcers," *BMJ Quality & Safety*, vol. 9, no. 2, pp. 137–141, 2000.

[3] S. M. S. Islam, M. M. Hasan, and S. Abdullah, "Deep learning based early detection and grading of diabetic retinopathy using retinal fundus images," *arXiv preprint arXiv:1812.10595*, 2018.

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[5] N. Silberman, K. Ahrlich, R. Fergus, and L. Subramanian, "Case for automated detection of diabetic retinopathy," in *2010 AAAI Spring Symposium Series*, 2010.

[6] "Kaggle challenge," https://www.kaggle.com/c/diabetic-retinopathy-detection, accessed: 2019-11-16.

[7] P. Porwal, S. Pachade, M. Kokare, G. Deshmukh, J. Son, W. Bae, L. Liu, J. Wang, X. Liu, L. Gao *et al.*, "Idrid: Diabetic retinopathy–segmentation and grading challenge," *Medical image analysis*, vol. 59, p. 101561, 2020.

[8] "Cell tracking challenge," http://www.codesolorzano.com/celltrackingchallenge, accessed: 2019-11-14.

[9] "Segmentation challenge," http://brainiac2.mit.edu/isbi-challenge/, accessed: 2019-11-13.

[10] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.

[12] A. Budai, G. Michelson, and J. Hornegger, "Multiscale blood vessel segmentation in retinal fundus images." in *Bildverarbeitung für die Medizin*, 2010, pp. 261–265.

[13] A. Budai, R. Bock, A. Maier, J. Hornegger, and G. Michelson, "Robust vessel segmentation in fundus images," *International journal of biomedical imaging*, vol. 2013, 2013.

[14] J. Canny, "A computational approach to edge detection," *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679–698, 1986.

Thresholds	Highest hierarchy	Second hierarchy	Lowest hierarchy
First threshold	30	50	30
Second threshold	70	100	70

TABLE II
HYSTERESIS THRESHOLDING VALUES