Segmentation of Fundus Images using U-net Algorithm

Gaurav, Ravikesh Yadav, and Sharad Dubey
Cluster Innovation Centre
University of Delhi

Developing methods to segment eyes in medical images, study and analyze it remains a significant challenge. Since the structure of the eye being complex, there can be possibilities where the various eye diseases can be caused by several reasons. Retina is the outer lining of the human eye where the image formation takes place. Any threat to retina causes severe eye defects and may lead to complete blindness. During a defect the retina gets distorted. Diabetic retinopathy is one of several common eye diseases, but is the most common cause of vision impairment and blindness among working-age adults in the United States. From 2010 to 2050, the number of Americans with diabetic retinopathy is expected to nearly double, from 7.7 million to 14.6 million. We aim to identify this disease using fundus images. It is caused by changes in the blood vessels of the retina. In some people with diabetic retinopathy, blood vessels may swell and leak fluid. In other people, abnormal new blood vessels grow on the surface of the retina. A healthy retina is necessary for good vision. So, we'll segment the blood vessels from fundus images and check whether their shape and size is normal or not.

I. INTRODUCTION

To study retina a retinal examination is done. The images of retina are taken through either fundus photography or Optical coherence Tomography (OCT). Fundus photography provides a color or red-free image of the retina. It is primarily digital.[1]This has many advantages compared with its predecessor, color photographic film. Digital retinal imaging provides rapidly acquired, high-resolution, reproducible images that are available immediately and easily amenable to image enhancement.

In today's world with such a high dependency on machines, It is important that these machines can perform all sort of tasks. And this dependency is rapidly increasing. When it comes to machines checking for the eye diseases, It has some drawbacks too, Human can't trust on a machine completely. Fundus photography is most often used for disease documentation and clinical studies, with potential use for telemedicine and patient education. Types of fundus photography include standard view and wide field. Fundus photography is a valuable clinical tool for evaluating progression of retinopathy in individual patients and in participants in clinical trials. Photography is used in clinical practice to document the status of retinopathy and effects of treatment.[3]

Specialized fundus cameras that consist of a microscope attached to a flash enabled camera are used in fundus photography. Fundus images are then analyzed by ophthalmologists who look for certain patterns and defects in the image to predict diseases [6],[7]. There are many problems in this system. World is short of highly qualified ophthalmologists. Due to this people have to wait for long before starting medications. We have started our study with the examination of fundus images. This report aims to identify the disease with the sophisticated image processing techniques. We are trying to extract blood vessels and other features from the fundus images. These are then sent to a classifier and then the classifier decides based on the learning.

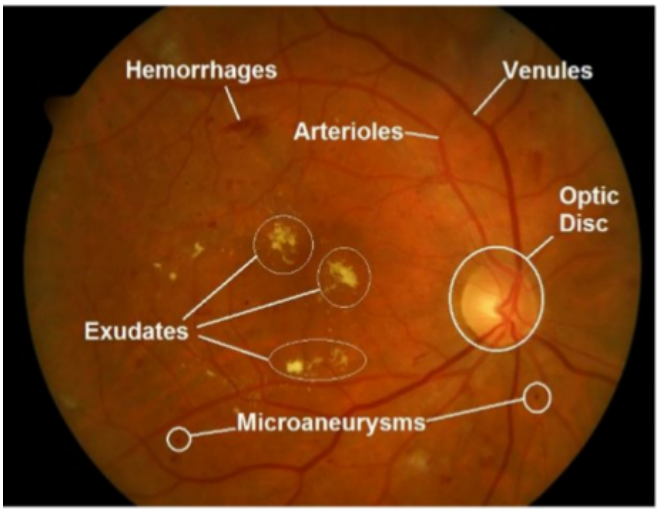


FIG. 1. An overview of Fundus image

II. LITERATURE REVIEW AND RELATED WORK

We have used several tools and techniques for segmenting the blood vessels. First we converted image in grayscale, Then After contrast enhancement using CLAHE[2], We removed the green channel. After it we sliced the images in order to get a greater accuracy and finally applied U-net for segmenting the vessels[4], [5],[7].These methods/Tools are discussed here. Also we used the STARE data set[9]containing only 40 images, Out of these, 20 were used for training and 20 were for testing. We sliced each image into 32 equal parts, so that there were total 1280 images(640 train and 640 test)

1. Contrast enhancement techniques

In digital image processing, contrast enhancement is a very important factor if we want to extract the features or segment the images, There are several techniques available for this, HE, AHE, CLAHE, Rayleigh CLAHE etc.[1]. We have used the three of these techniques and found that the CLAHE works best. The input and output of these tools are shown below. In the figure above,

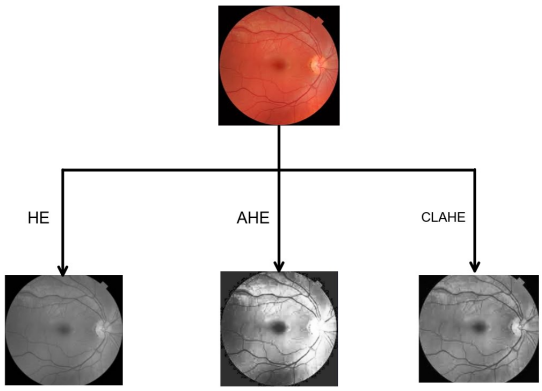


FIG. 2. Comparison of different contrast enhancement techniques

we can see that contrast is maximum and noise is minimum in the CLAHE image. AHE is similar to CLAHE but having more noise, hence non acceptable. CLAHE stands for Contrast Limited Adaptive Histogram Equalization. It is a method for retinal vessels enhancement as shown in the images. AHE adds some noise in the homogeneous pixels, therefore, to avoid this, we used CLAHE. In CLAHE R and B channel of the images are excluded and CLAHE is applied only on the Green channel of the image.[1] AN enhanced G channel is then obtained. Finally, R, B and enhanced G channel are merged to give the output image after removal of the noise that is added in the process.

2. Thresholding

Our main objective is to extract the features of the image which has to be followed by the classification.In the simplest terms, Thresholding is a segmentation technique to make the segmentation more robust. Thresholding can be defined as a process of dividing an image into two (or more) classes of pixels. It is used where we need to establish the difference between the layers of an image. So as to get a thresholded picture, for the most part, we convert the first picture into a gray scale picture and afterward apply the thresholding strategy. This technique is otherwise called Binarization as we convert the picture into a binarized structure, for example on the off chance that the estimation of a pixel is lesser than the limit esteem, convert it to 0(Black). In the event that the estimation of a pixel is more noteworthy than the edge esteem, convert it to 1(White) or the other way around.There are several types of thresholding. We, in this report particularly are interested in Adaptive threshloding.

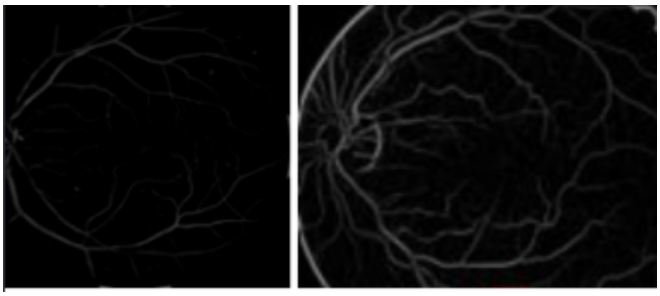


FIG. 3. Difference in thresholding: Image on left is normally thresholded while right one is thresholded adaptively(Mean

In Adaptive thresholding,input image is converted into a gray scale or binary image on the basis of threshold for each pixel. A threshold has to be calculated. If the pixel value is below the threshold it is set to the background value, otherwise it assumes the foreground value.Different threshold values are used for different areas due to difference in background illumination[11]. If we denote our input image as $g(x)$, thresholding can be defined as an operation that involves a test such that:

$$g(x) = \begin{cases} -1 & p(x) \leq 0 \\ 1 & p(x) > 0 \end{cases}$$

Here, $p(x)$ is the testing function. Simply If a pixel is labeled 1, It'll correspond to the light object and if it is labeled 0, it'll correspond to the dark object. Since, we are using Adaptive local thresholds, these are defined as follow:

$$(t(x)) = (p(x)) - (g(average, variance)) \qquad (1)$$

where g(average, variance) stands for a function of "average" and "variance." A simple that represents Thresholding is shown below, This was plotted in Python. At

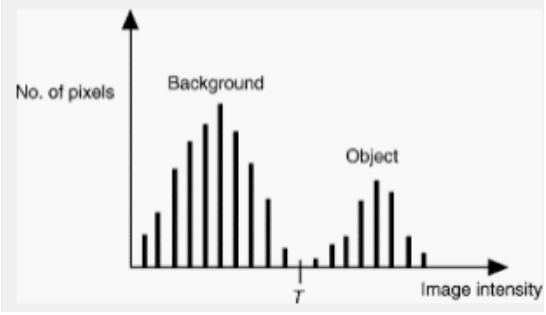


FIG. 4. Histogram showing thresholding

several times, A small change in the threshold can change

the accuracy by a much bigger factor. Thus, We'll look to make the most accuracy out of our thresholding algorithm. And we'll modify it by inserting a more generalized funtion, thus making it more robust. This will also lead to less Noise and clearer image.

3. U-Net

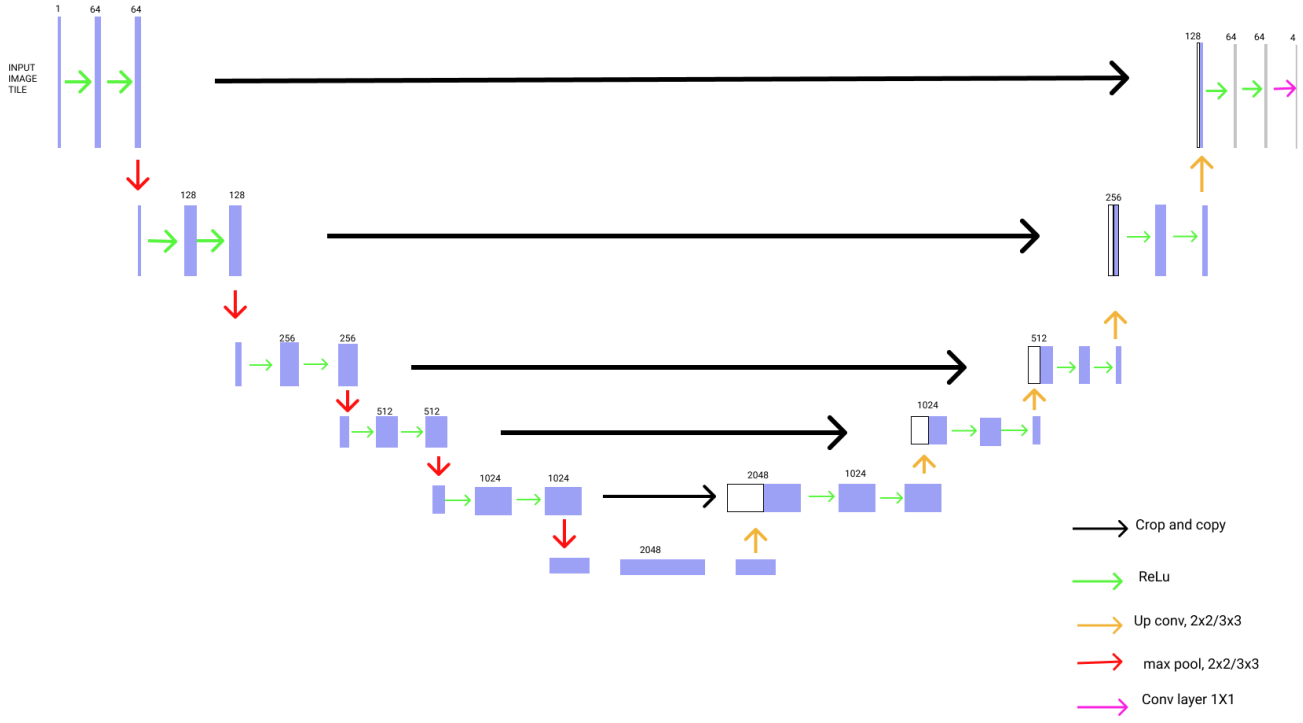
Vision in one of the most important sense that humans possess, although when it comes to identify complex objects, with a lot of restrictions, like reflection, angle of perception, light etc. We rarely fail. Machines are also becoming able to identify objects after the mergence of Artificial Intelligence and particularly CNNs. Although Convolutional Neural Networks gave decent results in easier image segmentation problems but it hasn't made any good progress on complex ones. And that's where the need of U-net has risen like never before.It was especially designed for medical images segmentation, but It is now used in all sort of applications. We should first discuss the Idea behind U-net and then it's working. CNN works excellent in the classification tasks but when it comes to segmentation, CNNS are not highly reliable. Because the idea behind the working of the CNN is to learn from the feature mapping and to exploit it for more feature mapping. But in image segmentation we not only need to convert feature map into a vector but also reconstruct an image from this vector. This is a mammoth task because it's a lot tougher to convert a vector into an image than vice versa. The whole idea of UNet is revolved around this problem. While converting an image into a vector, we already learned the feature mapping of the image so we can also use the same mapping to convert it again to image. This is the background on which U-net works. The feature maps that are used for contraction are used for expansion from a vector to a segmented image. Hence, the core structure of the image will not be deformed.

4. Training and loss calculation

Two things are used to train the Network, the input images and their corresponding segmentation maps with the help of stochastic gradient Descent. The output image is smaller than input due to unpadded convolutions. We favor large input tiles to minimize the overhead.
U- net uses the novel loss weighting scheme for each pixel. So that the weight has higher magnitude at the border than inside. This loss weighting scheme helped the U-Net model segment cells in biomedical images in a discontinuous fashion such that individual cells may be easily identified. As a matter of first importance pixel-wise softmax applied on the resultant picture which is trailed by cross-entropy loss function. So we are ordering every pixel into one of the classes. The thought is that even in division each pixel need to lie in some class and we simply need to ensure that they do. So we simply changed over a division issue into a multiclass grouping one and it performed very well when contrasted with the conventional loss functions.

III. METHODOLOGY

As discussed above, We started off with removing the green channel from the image. We removed the green channel from image array in python and merged it after it got enhanced. The following figure shows the effect of doing this on our image. It can be clearly seen that classification would not be possible with image of low contrast.
The main interest for this model is the design of the U-net. We changed the structure of U-net to some extent(from the original one) so that it can increase the efficiency. A detailed architecture of U-net is shown in Fig.5. U-net is designed in such a way that it works with very few training images and yields more precise segmentations. The main idea of this is to supplement a usual contracting network by successive layers, where pooling operators are replaced by upsampling operators. Hence, these layers increase the resolution of the output[11]. The architecture looks like a 'U' which justifies its name. This architecture consists of three sections: The contraction, The bottleneck, and the expansion section. Each block takes an input applies two 3X3 convolution layers followed by a 2X2 max pooling with stride 4 for down sampling. This max pooling depends totally on the image structure and can be 3X3 in some of the datasets. The number of kernels or feature maps after each block doubles so that architecture can learn the complex structures effectively. The bottom most layer mediates between the contraction layer and the expansion layer. It uses two 3X3 CNN layers followed by 2X2(or 3X3, depends on the image structure) up convolution layer. In the middle of this architecture lies in the expansion section. Similar to contraction layer, it also consists of several expansion blocks. Each block passes the input to two 3X3 CNN layers followed by a 4X4 up sampling layer. Also after each block number of feature maps used by convolutional layer get half to maintain symmetry. However, every time the input is also get appended by feature maps of the corresponding contraction layer. This action would ensure that the features that are learned while contracting the image will be used to reconstruct it. The number of expansion blocks is as same as the number of contraction block. After that, the resultant mapping passes through another 3X3 CNN layer with the number of feature maps equal to the number of segments desired. At the final layer a 1x1 convolution is used to map each 128- component feature vector to the desired number of classes. In total the net- work has 24 convolutional layers.[7]



To allow a seamless tiling of the output segmentation map, it is important to select the input tile size such that all 2x2 max-pooling operations are applied to a layer with an even x- and y-size.
It is quite important to enhance the contrast of the image. Thus, We have applied CLAHE here. The contrast amplification was limited here, So this noise is reduced and we got a much clear picture with higher contrast. CLAHE does not store the chromatic information of the image. So, using Rayleigh CLAHE is also an option, But since we are working with Gray scale images it's not required. The image got completely transformed after CLAHE. Figure above shows the image obtained after CLAHE. It is quite important to enhance the contrast of the image. Thus, We have applied CLAHE here. It is quite important to enhance the contrast of the image. Thus, We have applied CLAHE here. The contrast amplification was limited here, So this noise is reduced and we got a much clear picture with higher contrast. CLAHE does not store the chromatic information of the image. So, using Rayleigh CLAHE is also an option, But since we are working with Gray scale images it's not required. The image got completely transformed after CLAHE. Figure below Shows the image obtained after CLAHE. After applying CLAHE, Our next step is feasible area selection. This step includes the construction of several shape models based on 40 fundus images.
float Each 2-D segmentation image has its own X and Y pixel spacing. These dimensions have to be uniformized for all the images to aggregate all of them in a single model. After a quadratic interpolation, all 40 segmentations have the same spacing dimensions of X, Y, which is one millimeter pixel size. These values allowed imaging the real dimensions of the eye and processing further any new data. After doing this, we'll segment the images. We needto define a couple of terms before starting segmentation. The bounding box, denoted by BB, is used as a reference on each set of segmentations and it is the number of pixels that falls outside the box when we make our images uniform.

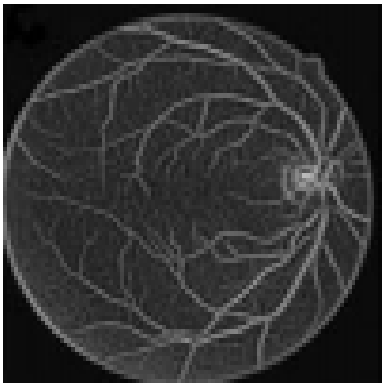


FIG. 5. Output image from CLAHE

The BB provides an approximate position of blood vessels inside the eye. We crop this Area on the dataset p(j) belongs to the set {1, p }. Then we compute the intensity histogram to find the 2-D image , where the largest plane

of the eye. Thanks to the BB, a lot of pixels belong to the eye and have similar intensities, therefore the slice with the largest eye surface has an intensity histogram with the highest peak. We'll plot these intensity histograms in the later sections. Then, by uniformizing pixel spacing, we can observe that the images of the different shapes have an increasing resolution according to the size of the original images. This allows us to choose the shape model for the segmentation of the patient images P(j) belongs to the set {1, p }. After doing this all, we found that the data available with us is not sufficient to get a decent accuracy.Thus, we decided to divide each of the images of the dataset into 16 images via slicing the original images. We now have 640 images altogether.Next, we focus on the pixels that are outside the BB, if these pixel's matrix value D is such that:

$$\mu - k(\delta)\delta <= D <= \mu + k(\delta)\delta \tag{2}$$

Then, we can safely consider these otherwise these pixels won't create a great effect on the final result, if they fall outside the described range. We say this as our final image. A result of this process is shown below: After getting this image we are ready to process it with thresholding. We are going for a bit modified Adaptive mean Thresholding. We have discussed that equation(1) describes the adaptive thresholding. We want to make a little change to this to make out thresholding more robust. This method uses the local image statistics in a linear combination of mean, standard deviation and 'variance'. Thresholding Function is now defined as:

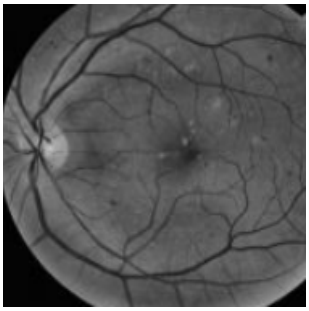


FIG. 6. Area selection

$$(t(x)) = f(x) - \mu_s(x) - K.\sigma_s(x) \tag{3}$$

where

$$\mu_s(x) = 1/N \times \sum_{x=o}^{s} f(x) \tag{4}$$

$$And\sigma_s(x) = \sqrt{(1/N-1) \times \sum_{x=o}^{s} .(f(x) - \mu_s(x))^2} \tag{5}$$

are the mean and the standard deviation, respectively, in a b · b neighborhood region S centered at x, $N = b \times b$, where b is a constant. This was extremely necessary in order to remove the noise and for a clearer picture containing blood vessels only. The output of these techniques are compared below: Finally, We move on to our Aim, i.e. Prediction of Diabetic Retinopathy. Now, we have noise free thresholded image. The training of the neural network is performed on sub-images (patches) of the pre-processed full images. Each patch, of dimension 48x48, is obtained by randomly selecting By slicing each image into 32 identical images. A set of 6000 patches is obtained by randomly extracting 300 patches in each of the 20 STARE training images. Although the patches overlap, i.e. different patches may contain same part of the original images, no further data augmentation is performed. The first 70% of the dataset is used for training (4200 patches), while the last 30% is used for validation (1800 patches). Finally, After we detected The DR, We measured our accuracy by varying no. of images and also compared it with some of the methods that were there previously. Following flowchart shows all the processes that are used in this report in a nutshell:

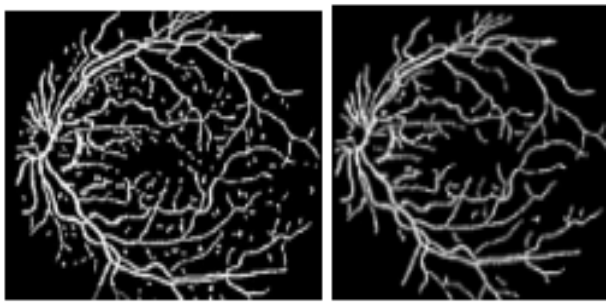


FIG. 7. Difference of Adaptive mean thresholding and thresholding after introducing variance: Left one is normal AMT, While right one is the output of abovementioned process

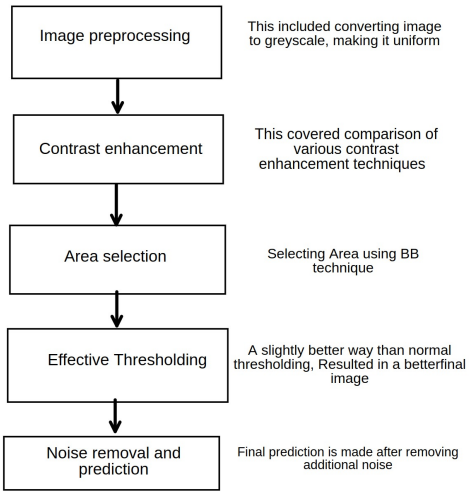


FIG. 8. Flowchart showing different processes

IV. RESULTS AND DISCUSSIONS

Testing is performed with the 300 images of the STARE testing dataset, using the gold standard as ground truth. Only the pixels belonging to the FOV are considered. The FOV is identified with the masks included in the STARE dataset. In order to improve the performance, the vessel probability of each pixel is obtained by averaging multiple predictions. With a stride of 5 pixels in both height and width, multiple consecutive overlapping patches are extracted in each testing image. Then, for each pixel, the vessel probability is obtained by averaging probabilities over all the predicted patches covering the pixel. We got the following Measures for our performance:

- Accuracy: 95.63
- Area under the ROC curve: 0.9721
- sensitivity:73%

Accuracy on the basis on no. of images in the STARE dataset:

No. of images in the dataset	Accuracy(in %)
80	63
160	83
320	87
640	91
1280	94.3
2560	95.3
3000	95.6

We also Ran our model on a couple of Dataset Apart from STARE, And got the following measures as outcomes(3000 images were taken from each of these datasets for training and testing purposes):

Dataset	Accuracy	AUC
DRIVE	93.654	0.9673
JSIEC	85	0.9056
IDRiD	81	0.8671

Since Most of the authors have expressed the efficiency in terms of AUC(area under curve). We have compared our AUC with some of the Authors. Our average AUC on STARE dataset was 0.9721 and on the DRIVE dataset, it was 0.9673. The low accuracy on the JSIEC and IDRiD dataset was mainly due to difference in the structure of the images. The JSIEC dataset has full images of the eye, therefore, Our preprocessing techniques were not very efficient over this. Similar thing happened with the images in IDRiD dataset. Here, due to difference in the structure of the image, Area selection was abrupt. This lead to formation of non-uniform patches and hence resulting in less efficiency.

Following table compared our AUC(DRIVE dataset) with some authors:

Authors	AUC
Soares et al [7]	0.9614
Azzopardi et al. [9]	0.9614
Osareh et al [10]	0.9650
Roychowdhury et al. [5]	0.9670
Proposed Method	0.9721

A further study on this report can lead to the identification of Haemorrhages, Exudates, calculation of Fractal Dimension etc. which can result in identification of many Eye diseases.Also, This adaptive thresholding can increase the efficiency of segmentation by a great factor, which will increase the efficiency of the model as overall.

V. CONCLUSION

Our image processing techniques have been very reliable and consistent. We have been successful in identifying blood vessels with a great accuracy. Result of blood vessels extraction works better than the cases we have discussed. A binary classifier system has been designed for diabetic retinopathy retinal defect has been developed and tested.

We have tried our model on different datasets and got the best accuracy with STARE dataset, followed by DRIVE dataset. The accuracy in JSIEC and IDRiD datasets were kow because of different alignment of the images.

VI. REFERENCES

1. Imran, Azhar, et al. “Comparative Analysis of Vessel Segmentation Techniques in Retinal Images.” IEEE Access, vol. 7, 2019, pp. 114862–114887.
2. Nidhi, S., Navdeep S., Blood Vessel Contrast Enhancement Techniques for Retinal Images. International Journal of Advanced Research in Computer Science, vol. 8, 2017
3. Kitrungrotsakul, T., Han, X.-H., Iwamoto, Y., Lin, L., Foruzan, A. H., Xiong, W., and Chen, Y.-W. (2019). VesselNet: A deep convolutional neural network with multi pathways for robust hepatic vessel segmentation. Computerized Medical Imaging and Graphics, 75, 74–83. doi: 10.1016/j.compmedimag.2019.05.002
4. Ronneberger, Olaf, et al. “U-Net: Convolutional Networks for Biomedical Image Segmentation.” Lecture Notes in Computer Science Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, 2015, pp. 234–241., doi:10.1007/978-3-319-24574-4-28.
5. Roychowdhury et al., “Blood vessel segmentation of fundus images by major vessel extraction and subimage classification,” Biomedical and Health Informatics, IEEE Journal of, vol. 19, no. 3, pp. 1118–1128, 2015.

6. Imran, A., Li, J., Pei, Y., Yang, J.-J., and Wang, Q. (2019). Comparative Analysis of Vessel Segmentation Techniques in Retinal Images. IEEE Access, 7, 114862–114887. doi: 10.1109/access.2019.2935912
7. Soares et al., "Retinal vessel segmentation using the 2-d Gabor wavelet and supervised classification," Medical Imaging, IEEE Transactions on, vol. 25, no. 9, pp. 1214–1222, 2006.
8. Li, L., Verma, M., Nakashima, Y., Nagahara, H., and Kawasaki, R. (2020). IterNet: Retinal Image Segmentation Utilizing Structural Redundancy in Vessel Networks. 2020 IEEE Winter Conference on Applications of Computer Vision (WACV). doi: 10.1109/wacv45572.2020.9093621
9. Azzopardi et al., "Trainable cosfire filters for vessel delineation with application to retinal images," Medical image analysis, vol. 19, no. 1, pp. 46–57, 2015.
10. Osareh et al., "Automatic blood vessel segmentation in color images of retina," Iran. J. Sci. Technol. Trans. B: Engineering, vol. 33, no. B2, pp. 191–206, 2009.
11. Sirpa et. al., "Efficient Detection of Retina Blood Vessels Using Proficient Morphological Algorithms", International Research Journal of Engineering and Technology, Vol. 3, 2016.
12. Akbar, S., Sharif, M., Akram, M. U., Saba, T., Mahmood, T., and Kolivand, M. (2019). Automated techniques for blood vessels segmentation through fundus retinal images: A review. Microscopy Research and Technique, 82(2), 153–170. doi: 10.1002/jemt.23172
13. Pathan, S., Kumar, P., Pai, R., and Bhandary, S. V. (2020). Automated detection of optic disc contours in fundus images using decision tree classifier. Biocybernetics and Biomedical Engineering, 40(1), 52–64. doi: 10.1016/j.bbe.2019.11.003
14. Cao, L., and Li, H. (2020). Enhancement of blurry retinal image based on non-uniform contrast stretching and intensity transfer. Medical Biological Engineering Computing, 58(3), 483–496. doi: 10.1007/s11517-019-02106-7