
Retina blood vessel segmentation based on U-Net

Libo Xu
libox@kth.se

Ruihao Zhu
ruihao@kth.se

Abstract

The project aims to segment the blood vessels in digital retinal images. The main method is to train a deep convolutional neural network, i.e. U-net proposed by Ronneberger et al.[1] on the DRIVE database, which are preprocessed with Z-score and min-max normalization, contrast limited adaptive histogram equalization and gamma correction. Given the small number of retina images in both training and testing data set, we also extract many small patches from the original images to augment the data set. When our best-trained model predicts on the testing data set, the algorithm we applied outperforms some other techniques in terms of the area under the receiver operating characteristic curve (AUC) of 0.9770 and accuracy of the classification of background and the blood vessel pixels of 0.9537.

1 Introduction

The assessment of the geometry and the distribution of retinal blood vessels in human eyes has been widely applied to diagnosis, screening, treatment, and evaluation of various cardiovascular and ophthalmologic diseases. For example, diabetes could be found by automatic detection and analysis of the retinal vasculature[2] and diabetic retinopathy is the main cause of adults aged 20-74 becoming blind[3]. In order to prevent the server result of losing vision, early intervention such as regular examination of the fundus images is necessary[4]. Since it is time-consuming and labor-intensive for ophthalmologists or related specialists to do the analysis of retinal images traditionally, the deep learning based segmentation method is introduced to speed the process and improve the accuracy of recognition of blood vessels.

The main problem we work on in this project is a medical image segmentation task, which is to extract the blood vessels in the digital retinal images. Essentially, this is a binary classification task because each pixel in the images is either a pixel of a blood vessel or a pixel of the background. Hence, the method of using the convolutional neural networks is considered to be state-of-the-art to solve the problem accurately. In this project, the architecture of the convolutional neural network is built up according to the U-net architecture. It is widely agreed in deep learning field that usually the good performance of the neural networks on the corresponding task requires large numbers of annotated training samples in the data set. However, thousands of training images are not available in most biomedical imaging tasks. For instance, only 20 images with corresponding ground truth segmentation are available in this project. Therefore, instead of training directly on the given data set, a small window is set up to slide on each image to extract a local region, i.e. patch to create a training data set with a large number of samples[1]. Based on the data augmentation techniques, the U-net could take advantage of a few annotated images and performs well on the segmentation task. Except for the patch extraction, the original images are also preprocessed with some other techniques such as gamma correction to enhance the images.

To make the model perform better on predicting, the training model is also tested with many different hyper parameters settings. As for the evaluation of the trained model on predicting the segmentation of unlabelled retinal images, the confusion matrix of the segmentation result is applied to calculate the accuracy, sensitivity, precision, specificity and F1-score. What's more, the area under the receiver

operating characteristic curve of 0.9770 is achieved with the trained model, which indicates this segmentation technique outperforms many previous segmentation methods on the DRIVE database.

2 Related work

There are several related works based on DRIVE database to segment blood vessels in retina fundus images. Soares et al. classify each image pixel based on the feature vectors, which are composed of the pixel's intensity and two-dimensional Gabor wavelet transform responses. During the classification process, they combine Bayesian classifier with Gaussian mixture models to achieve fast classification and model complex decision surfaces[5]. The result of their method on DRIVE is AUC = 0.9614, Accuracy = 0.94660.

Azzopardi et al. propose a new filter called B-COSFIRE, which can selectively respond to vessels. In their segmentation process, they configure both symmetric and asymmetric B-COSFIRE filters and combine the responses together[6]. They have a better result as AUC = 0.9614, Accuracy = 0.9442.

QiaoLiang et al. remodel the segmentation as a problem of cross-modality data transformation from retinal image to vessel map. Normally the network will output a single label of the center pixel, but their method can output the label map of all pixels for a given image patch[7]. Their result is AUC = 0.9738, Accuracy = 0.9527.

Roychowdhury et al. divide segmentation into three stages. They first extract two binary images from both high-pass filter and morphologically reconstructed enhanced images, then use Gaussian mixture model classifier to classify all the remaining pixels. In the end, they combine the major blood vessels and classified vessel pixels together. Compared with other methods, their algorithm requires less training data and segmentation time[8]. Their result is AUC = 0.962, ACC = 0.9519.

3 Data

The images in the DRIVE database were obtained from a diabetic retinopathy screening program in the Netherlands[9]. There are 40 images of retina in total, which are equally divided into a training and a test data set. Each image has a resolution of 584×565 pixels with 3 channels. Besides, each image has a corresponding mask image which represents the field of view, i.e. the region of interest and a segmentation image of its vasculature which is manually marked by specialists. Figure 1 shows a sample of an original image (only transformed from RGB image to grayscale image), its corresponding ground truth and mask.

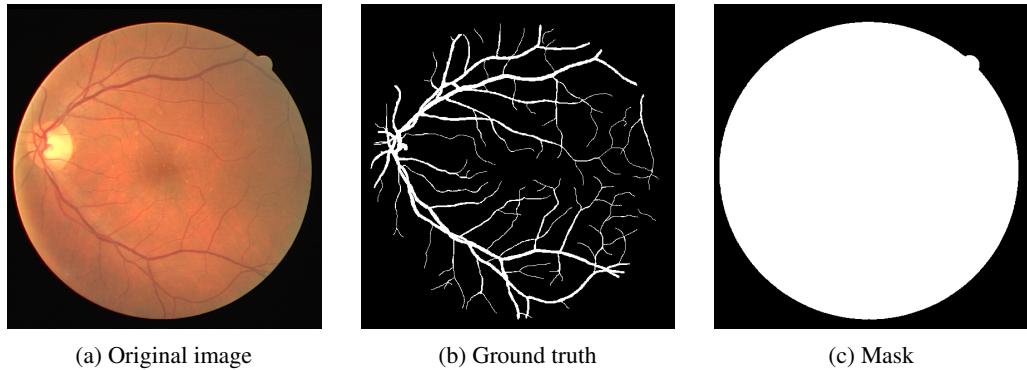


Figure 1: A sample of the dataset

As aforementioned, small patches (48×48 pixels) are extracted by the sliding window in order to increase the numbers of samples in both training and test data sets (before the extraction, there are some preprocess methods which will be discussed in Chapter 4.1). Firstly, for each image in the training set, we randomly select 5000 center pixels inside the range of image size and then take the local region of each center as a patch. In this way, a data set of 100000 48×48 pixels images is created for training. Among these patches, 90% of them would be used for training and the rest of them would be used for validation. However, for each patch in the test data set, the prediction

would also be a small patch displaying the segmentation blood vessels, from which we can't clearly visualize the effect of segmentation of a whole image by human eyes. Hence, instead of random extraction, the patches are extracted in order and consecutively with a stride of 5 in both height and width of images, in which way it would be easier and more efficient to assemble the segmentation patches to full images correctly.

4 Methods

4.1 Preprocessing methods

Some preprocessing techniques are implemented on the training and test original images before they are divided into patches. Firstly, since the most important feature our algorithm wants to extract is the gradient of intensity, which does not rely much on color information, the original RGB images are transformed to grayscale images in order to speed up the calculation process.

Secondly, Z-score normalization, i.e. standardization which is calculated by subtracting the population mean of pixel intensity from the original images and dividing the difference by the standard deviation leading to data centralization is implemented[10] .

$$\frac{\text{value} - \min}{\max - \min} \quad (1)$$

Thirdly, the intensity of all pixels in the images are normalized to interval [0,1] by min-max normalization, which would benefit the evaluation process[11] .

$$\frac{\text{value} - \mu}{\sigma} \quad (2)$$

Next, contrast limited adaptive histogram equalization (CLAHE) is applied to the images. It limits the amplification to suppress the over-amplifying noise in relatively homogeneous regions of an image[12]. Ordinary histogram equalization uses the same transformation derived from the image histogram to transform all pixels. But there is an issue when the image contains regions, which can be regarded as outliers, that are much lighter or darker than most of the image. The issue is that the contrast in those regions will not be sufficiently enhanced. Adaptive histogram equalization (AHE) can be used to fix this issue by transforming each pixel with a transformation function derived from a neighborhood region. However, AHE may cause noise to be amplified in near-constant regions[13]. Contrast Limited AHE (CLAHE) is an improved version of adaptive histogram equalization In CLAHE the contrast amplification is limited in order to reduce this problem of noise amplification. The contrast amplification of a pixel value is given by the slope of the transformation function, which is proportional to the slope of the neighborhood cumulative distribution function (CDF) and to the value of the histogram at that pixel value. CLAHE limits the amplification by clipping the histogram at a predefined value before computing the CDF. This limits the slope of the CDF and avoids the spike amplification[14].

Finally, gamma correction is used on the images. Gamma correction is usually applied to perform non-linear or inverse calculations on the luminance or tristimulus value of light in the film or imaging system. The simplest Gamma correction function is as below:

$$V_{out} = A V_{in}^{\gamma} \quad (3)$$

4.2 Architecture of U-net

As the Figure 3 shows, the architecture of this network looks like a 'U' shape. The network has three sections: contraction, bottleneck, and expansion section. This architecture contains 4 convolutional blocks at the contraction path, 4 convolutional blocks at the expansion path and one convolutional layer at the bottleneck. Except for the layer displayed in Figure 3, batch normalization layers and dropout layers with 0.2 are also added to the network to mitigate overfitting and make network more robust to different initialization schemes and learning rates[15]. Since there are many same layers in the architecture, we divide the convolutional blocks into 3 types to develop (details could be found in the file 'Unet.py' in the github link in the end of the report), instead of implementing the architecture into program layer by layer. In this way, U-net could be built up by the implementation of 2 loops for the same type of blocks, which makes it possible to easily control the depth of the U-net for possible future tasks and research (the change of depth is not discussed in this project).

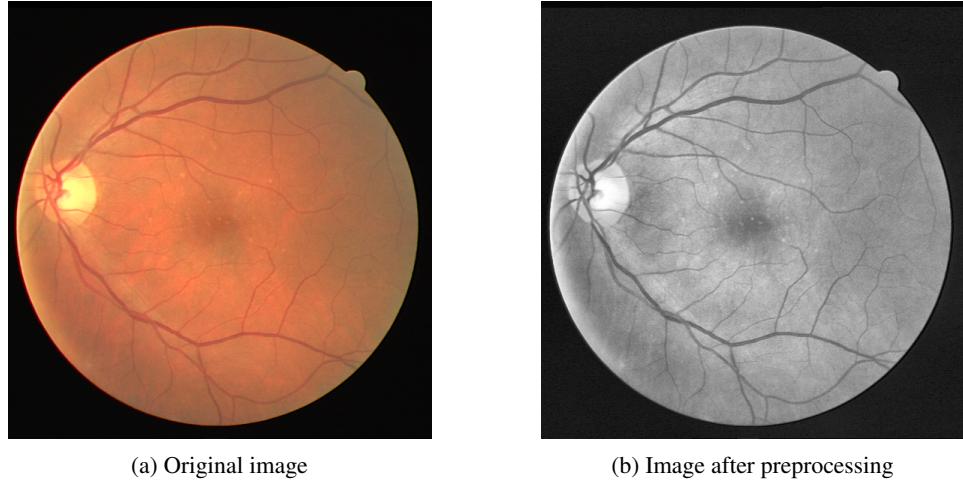


Figure 2: A sample of preprocessing

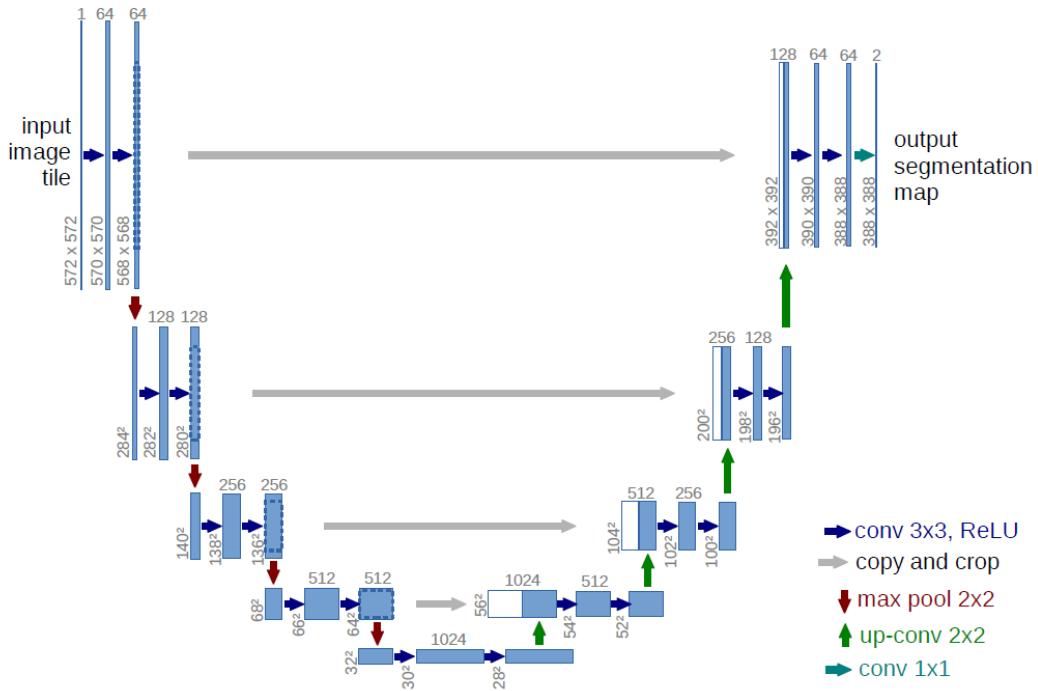


Figure 3: U-net Architecture[1]

4.3 Parameter setting of the model

Firstly, the number of epochs is set to 150. The batch size is set to 64 to mitigate the oscillations of the validation loss. Besides, stochastic gradient descent (SGD) is chosen to be the optimizer, with the decay of learning rate. If the learning rate is set too high, the update step of the parameters would be very large, which may lead to the convergence to the local optimum or the loss would begin to increase. On the other hand, if the learning rate is set too low, the loss would drop very slowly which means more epochs is required to converge. Hence, a decaying learning rate strategy is applied. In the setting of SGD function from keras, the initial learning rate is set to 0.1 and the decay rate is the division of initial learning rate by the total number of epochs which is 0.1/150 in this case. Besides, momentum is set to 0.8 which could speed up the learning process by affecting the direction of current gradient.

4.4 Training strategy

Two training strategies from the callbacks are used in this project. The first one is ‘ModelCheckpoint’, which is set to monitor the value of validation loss in every epoch. When the validation loss after this epoch is smaller than the previous one, the checkpointer would save the weights to the specified path. In this way, the best weights would be saved when the training process is over. The other strategy is ‘EarlyStopping’, which will terminate the training if the validation loss does not improve in the given number of epochs. With the early stopping strategy, the training process could stop if the model does not perform well or the algorithm converges early.

5 Experiments

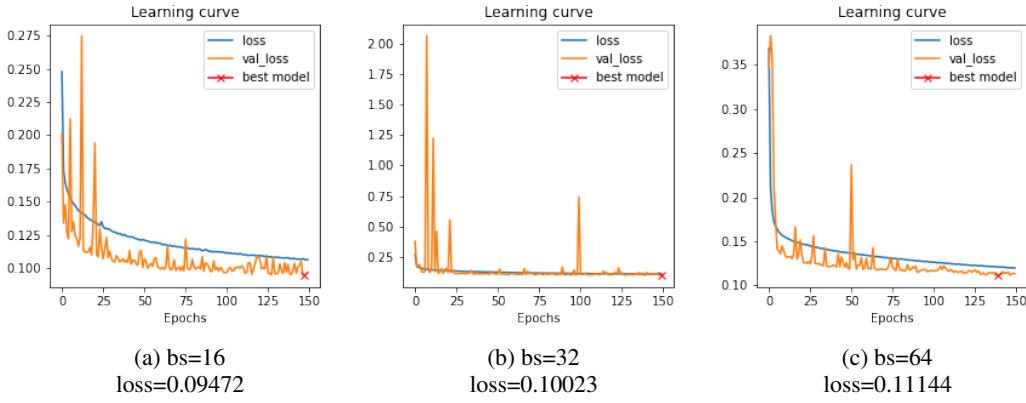


Figure 4: Testing results of batch size with lr=0.01(no decay),N_epochs=150

From Figure 4, we could see that the model with batch size of 16 performs best according to the value of loss. However, the algorithm in this model has not converged within 150 epochs. What's more, comparing to the curve and value of loss of bs=16, the algorithm with batch size of 64 converged at a suboptimum.

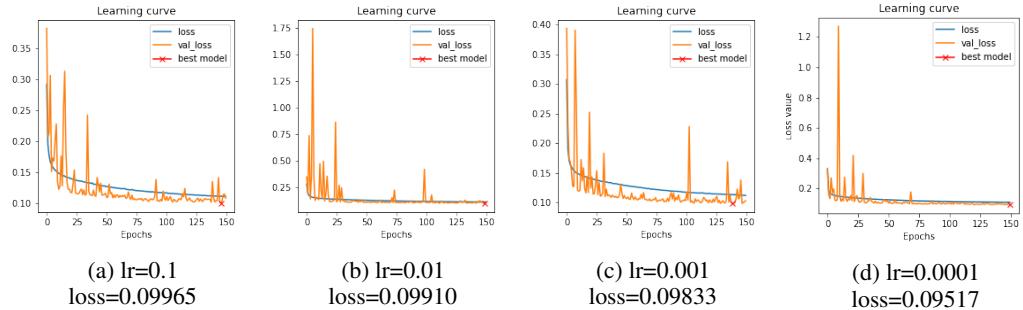


Figure 5: Testing results of learning rate with batch size=32, N_epochs=150

From Figure 5, we could see that the model with learning rate of 0.0001 performs best in terms of the value of loss. However, there is huge oscillations in the value of loss in this model. The results are kind of confused because the loss function curves of models of different learning rate settings does not varies in the way they should be. Hence, we implement another experiment with batch size of 16 and learning rate of 0.0001, which turns out that the algorithm is overfitting in the early stage.

Given the results in Figure 4 and Figure 5, we decide to apply the aforementioned decaying strategy to the setting of learning rate. Since the training process is time-consuming and smaller batch size may lead to oscillations in the value of loss according to the previous experiences, the batch size is set to 64 to speed up the training process. The result of this experiment is presented in Figure 6.

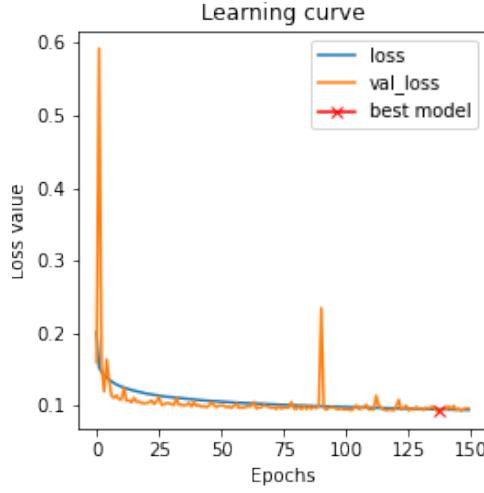


Figure 6: Best performance with loss=0.09242
n_epochs=150, batch size=64, lr=0.1(with decay strategy)

From Figure 6 we can see that the decay of learning rate at early stage is not enough because the value of loss drops very quickly. But the curve is relatively smooth compared to the previous results and the lowest loss achieved is lower than before. Due to the time constraint, the experiments of smaller batch size with the decaying strategy or a different parameter setting of the decaying strategy are not attempted. This trained model is applied to predict on the test data set.

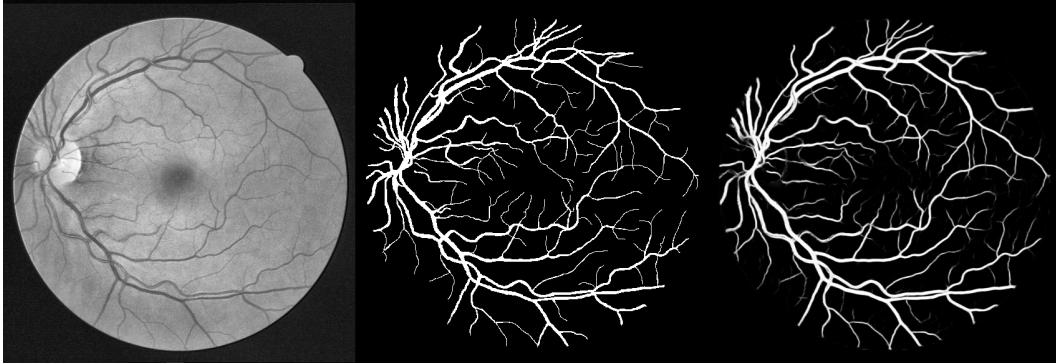


Figure 7: From left to right: original image(preprocessed), ground choose, segmentation result

Table 1: Predicting performance of the best model

Metrics	Value
accuracy	0.9537
recall	0.7243
precision	0.8917
specificity	0.9872
F1	0.7993
AUC	0.9770

Figure 7 shows the segmentation result of our best model and table 1 shows the metrics of the performance of the model on DRIVE database. Comparing the segmentation result to the ground truth image, we could observe that the thick blood vessels are segmented well while the thin vessels appears to be darker in algorithm-segmented image, which indicate the future imporvement could

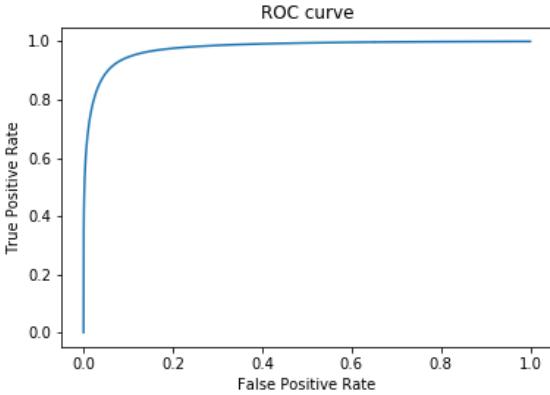


Figure 8: ROC curve

Table 2: Performance comparison

Name	AUC	Accuracy
Qiaoliang et al. [7]	0.9527	0.9738
Marin et al. [16]	0.9588	0.9452
Azzopardi et al. [6]	0.9614	0.9442
Soares et al. [5]	0.9614	0.9466
Roychowdhury, S [8]	0.9620	0.9519
Ricci and Perfetti[17]	0.9633	0.9595
Proposed method	0.9770	0.9537

be focus on the thin edge part of the vessels (All the segmentation result images could be found in Appendix). Figure 8 shows the ROC curve. Table 2 shows the comparison with the some previously published methods on DRIVE database. From the data we could see that the value of accuracy of our method does not rank the first as the AUC does in this table. However, from the analysis of ground truth images in which all pixels have either intensity of 1 (blood vessel) or intensity of 0 (background), we could conclude that only approximately 10% of all pixels have intensity of 1, which indicates that there is a unbalanced issue in this data set. Therefore, the metric of accuracy can not evaluate the performance of the model very well. Hence, our DL-based method outperforms all of these prior techniques in terms of the value of AUC.

6 Conclusion

The project proposes a supervised segmentation method to extract the blood vessels in retinal images by taking advantage of the U-net, which is one of the most efficient segmentation networks. Although the hyper-parameter setting of our best-performance model has not been tuned to its best way aforementioned in Chapter 6, this methods still performs better on DRIVE database than many previous methods, which could be attribute to the high efficiency of U-net. As for the future works, the decaying strategy of learning rate could be researched further to improve the performance. Besides, the depth of U-net could possibly be changed to fit different segmentation task. What's more, many biomedical images are acquired in 3D form, which indicates that the implementation of 3D U-net could be an intriguing project for us to work on in the future. All the code could be found on the github link([click here](#)).

References

- [1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [2] Jack J Kanski and Brad Bowling. *Clinical ophthalmology: a systematic approach*. Elsevier Health Sciences, 2011.
- [3] Ryan Lee, Tien Y Wong, and Charumathi Sabanayagam. Epidemiology of diabetic retinopathy, diabetic macular edema and related vision loss. *Eye and vision*, 2(1):1–25, 2015.
- [4] Sue J Lee, Catherine A McCarty, Hugh R Taylor, and Jill E Keeffe. Costs of mobile screening for diabetic retinopathy: a practical framework for rural populations. *Australian Journal of Rural Health*, 9(4):186–192, 2001.
- [5] J. V. B. Soares, J. J. G. Leandro, R. M. Cesar, H. F. Jelinek, and M. J. Cree. Retinal vessel segmentation using the 2-d gabor wavelet and supervised classification. *IEEE Transactions on Medical Imaging*, 25(9):1214–1222, 2006.
- [6] George Azzopardi, Nicola Strisciuglio, Mario Vento, and Nicolai Petkov. Trainable cosfire filters for vessel delineation with application to retinal images. *Medical Image Analysis*, 19(1):46–57, 2015.
- [7] Q. Li, B. Feng, L. Xie, P. Liang, H. Zhang, and T. Wang. A cross-modality learning approach for vessel segmentation in retinal images. *IEEE Transactions on Medical Imaging*, 35(1):109–118, 2016.
- [8] S. Roychowdhury, D. D. Koozekanani, and K. K. Parhi. Blood vessel segmentation of fundus images by major vessel extraction and subimage classification. *IEEE Journal of Biomedical and Health Informatics*, 19(3):1118–1128, 2015.
- [9] <https://drive.grand-challenge.org/>.
- [10] <https://www.investopedia.com/terms/z/zscore.asp>.
- [11] <https://www.codecademy.com/articles/normalization>.
- [12] Stephen M Pizer. Contrast-limited adaptive histogram equalization: Speed and effectiveness stephen m. pizer, r. eugene johnston, james p. ericksen, bonnie c. yankaskas, keith e. muller medical image display research group. In *Proceedings of the First Conference on Visualization in Biomedical Computing, Atlanta, Georgia*, volume 337, 1990.
- [13] David J Ketcham, Roger W Lowe, and J William Weber. Image enhancement techniques for cockpit displays. Report, HUGHES AIRCRAFT CO CULVER CITY CA DISPLAY SYSTEMS LAB, 1974.
- [14] Stephen M Pizer, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.
- [15] Wikipedia. <https://www.investopedia.com/terms/z/zscore.asp>.
- [16] Diego Marín, Arturo Aquino, Manuel Emilio Gegúndez-Arias, and José Manuel Bravo. A new supervised method for blood vessel segmentation in retinal images by using gray-level and moment invariants-based features. *IEEE Transactions on medical imaging*, 30(1):146–158, 2010.
- [17] Elisa Ricci and Renzo Perfetti. Retinal blood vessel segmentation using line operators and support vector classification. *IEEE transactions on medical imaging*, 26(10):1357–1365, 2007.

Appendix

Segmentation results of all 20 images in test data set

