Project Report

Group "Caravan"
Jingwen Zhang - z5155096
Kuo Shi - z5168805
Boxuan Hu - z5145438
Yan Wang - z5133879
Jianghang Cheng - z5172640

Abstract - **In this project, we are aiming to implement segmentation and classification techniques in two tasks, to segment retinal lesions of abnormal conditions associated with diabetic retinopathy and to segment blood vessels in retinal images, implementing by different pre-processing methods and deep learning models.**

Keywords – U-net, Segnet, FCN, Images augmentation, Resize image, CLAHE, Receiver operating characteristic, Keras, OpenCV.

INTRODUCTION

Computer vision software is changing industries and making the lives of consumers not only easier but also more interesting during the past few decades.

Computer vision works in three basic steps: acquiring an image, processing the image, understanding the image. Today's AI systems can go a step further and take actions based on an understanding of the image. There are many types of computer vision that are used in different ways: image segmentation, object detection, facial recognition, edge detection, pattern detection, image classification, feature matching. Simple applications of computer vision may only use one of these techniques, but more advanced uses, like computer vision for self-driving cars, rely on multiple techniques to accomplish their goal.

Apart from autonomous vehicles, Computer vision in practice today also includes agriculture, real-time sports tracking, manufacturing and moreover, healthcare. Since 90 percent of all medical data is image based there is a plethora of uses for computer vision in medicine. From enabling new medical diagnostic methods to analyze X-rays, mammography and other scans to monitoring patients to identify problems earlier and assist with surgery, computer vision has been widely used for predictive analytics and therapy.

In this project, we are aiming to implement segmentation and classification techniques in two tasks, to segment retinal lesions of abnormal conditions associated with diabetic retinopathy and to segment blood vessels in retinal images in order to get a general understanding of how computer vision applied in real life and helped human address medical problems. The data includes original retinal images for training, with a wide range of supervised and unsupervised segmentation algorithms and appropriate pre- and post-processing techniques, along with other retinal images and the corresponding ground-truth segmentation masks for testing and evaluating for such as accuracy based on evaluation metric.

LITERATURE SURVEY

Image Processing

Mathematical Morphology

Mathematical morphology [2] is a nonlinear tool for analysis of size, shape and inner structure of objects using set theory. It is used for de-noising, edge or contrast enhancement and background and foreground separation. The theory of mathematical morphology is built on two basic image processing operators: the dilation and the erosion.

Contrast Limited Adaptive Histogram Equalization (CLAHE)

Contrast limited adaptive histogram [2] is implemented to improve the visibility level of foggy image, increase overall image quality. In CLAHE, the contrast amplification in the vicinity of a given pixel value is given by the slope of the transformation function.

Segmentation Techniques

U-Net

The u-net [1] is convolutional network architecture for fast and precise segmentation of images. The main idea is to supplement a usual contracting network by successive layers, where pooling operations are replaced by up-sampling operators. Hence these layers increase the resolution of the output. In addition, a successive convolutional layer can then learn to assemble a precise output based on this information. W. Xianchenga et al. achieved 0.979 accuracy on DRIVE dataset by using U-Net convolutional network [4].

Residual neural network

A residual neural network (ResNet) is an artificial neural network (ANN) of a kind that builds on constructs known

from pyramidal cells in the cerebral cortex. Residual neural networks do this by utilizing skip connections, or shortcuts to jump over some layers. Typical ResNet models are implemented with double- or triple- layer skips that contain nonlinearities (ReLU) and batch normalization in between. He et al. proved that Resnet can achieve a desirable accuracy on image segmentation tasks [5].

Recursive Region Growing Segmentation (RRGS) Algorithm

The algorithm [3] was used for hard exudates detection. The basis of RRGS is the identification of similar pixels within a region to determine the location of a boundary. Fully automated computer algorithms were able to detect hard exudates and HMA.

Moat Operator

Moat operator [3] was used to optimize recognition of haemorrhages and microaneurysms (HMA), sharpening the edges of the red lesions against the red-orange background. Applying the Moat Operator to the RRGS generated image and thresholding was used to classify the image into HMA and non-HMA regions. Using the same method to produce the HMA mask and overlaid the original image to get the result of haemorrhages and microaneurysms recognition.

Adaptive Threshold Algorithm

This threshold technique [2] is used for the separation of background and foreground images. Hidden Markov Model (HMM) and morphological operators are implemented for pre-processing. The image smoothing is achieved by median filter and the HMM is implemented at second stage for predicting vessel pixel. The main idea in this algorithm is that each pixel is compared to an average of the surrounding pixels.

FCN

Fully convolutional network (FCN) [6] indicates that the neural network is composed of convolutional layers without any fully-connected layers or MLP usually found at the end of the network. A CNN with fully connected layers is just as end-to-end learnable as a fully convolutional one. The difference between FCN and U-net is that FCN uses deconvolution instead of upsampling.

METHOD

Software

Language: Python
Libraries: Keras, NumPy, Scikit-learn, matplotlib, OpenCV, skimage

Hardware: Free GPU on Google Colab

Colab is a free cloud service based on Jupyter Notebooks for machine learning education and research, which is a Google internal research tool. Colab has 12-hour limit for a continuous assignment of VM.
Type of GPU: 12GB NVIDIA Tesla K80 GPU

Dataset

Task 1

Indian Diabetic Retinopathy Image Dataset (IDRiD)

The Indian Diabetic Retinopathy Image Dataset is organized as a challenge workshop, aiming to evaluate algorithms for automated detection and grading of diabetic retinopathy and diabetic macular edema using retinal fundus images.
The fundus images in IDRiD were captured by a retinal specialist at an eye clinic located in Nanded, Maharashtra, India. From a total of 516 images in the dataset, our project used 54 original retinal images for training and 27 original retinal images for testing, corresponding with ground-truth segmentation masks for each lesion type in each image. Along with appropriate augmentation methods, the actual data we use would be larger.

Task 2

Digital Retinal Images for Vessel Extraction (DRIVE)

The DRIVE database has been established to enable comparative studies on segmentation of blood vessels in retinal images, which can be further utilized for the diagnosis, screening, treatment, and evaluation of various cardiovascular and ophthalmologic diseases such as diabetes.
The images for the DRIVE database were obtained from a diabetic retinopathy screening program in The Netherlands. The database consisted of 400 diabetic subjects between 25-90 years of age. In project, 20 original retinal images in TIF format are used for training and other 20 images for testing, corresponding with ground-truth segmentation masks for both training and test. And a black background around the retinal image is given as a mask for each image.

Models

Pre-process of the retinal images

In order to improve the accuracy of the model, firstly we pre-processed the images. Here we used a local contrast enhancement algorithm which can make the features in the image more visible. According to Sinthanayothin el al. [3], We did the following operations for the original images in our dataset:

Firstly, we convert the image in RGB colour space to an HSV colour space, then we run the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm [2] on the V channel. After that, the image is converted back to RGB space.

After performing the above series of operations, we obtained the results shown in Fig. 1b.

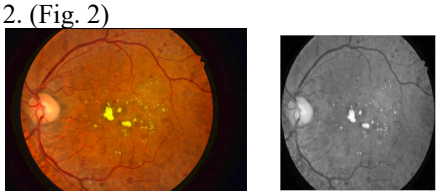
a b

Fig. 1: a is the original retinal image in our dataset, b is the processed image after our pre-processing algorithm.

From the result shown in Fig. 1, we can clearly see that after the original image is passed through our preprocessing algorithm, the local contrast is enhanced, and the various detailed features in the image become clearer.

Resize Images

Considering that the size of our images is very large, if we directly put these images with such high resolution into the neural network for training, there will be a problem of insufficient memory. Hence, here we resized the images before putting them into our neural network. Considering the structure of the neural networks we used for this task, the shape of the input image should be a square with only one channel, and its side length should be divisible by 32. In our dataset, the size of images is 4288x2848*3 (Fig. 2a). Therefore, we first convert the preprocessed color image into a grayscale image. In order to save the features in the image as much as possible, we first trim the image, cut it into squares as much as possible, and then use the reshape (resampling using pixel area relation) operation to convert the image to a resolution of 512*512. (Fig. 2)



a b

Fig. 2: a is the pre-processed image, b is the image after the resize operation.

Images Augmentation

Since there are only 54 images in our training set. Obviously, the size of training data is not enough. Therefore, we use the data augmentation operation here, using the operations of zooming in, zooming out, rotating, shifting, flipping, etc. to augment the data. The part of augmentation result for Fig. 2b are shown in Fig. 3.

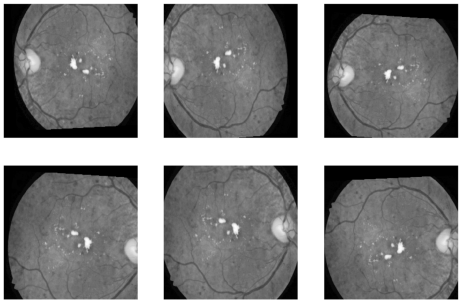


Fig. 3: Result of data augmentation

Training

Here we tried 3 different models for this project. In task 1, we used U-net and Segnet. In task2, in addition to the above two models, we also implement another model called FCN.

We took image with resolution 512*512*1 as input. The output size is also 512*512*1. In the output result, each pixel is given a value of 0-1 by the U-net neural network. If the pixel is closer to 1, the more likely the pixel is the feature point we need to find, and the closer the pixel is to 0, the more likely that it is the background point.

Because task1 and task2 add up to a total of five different features, we train each of the different features separately using our chosen model.

Model 1: U-net

The structure of U-net is shown in Fig. 4.

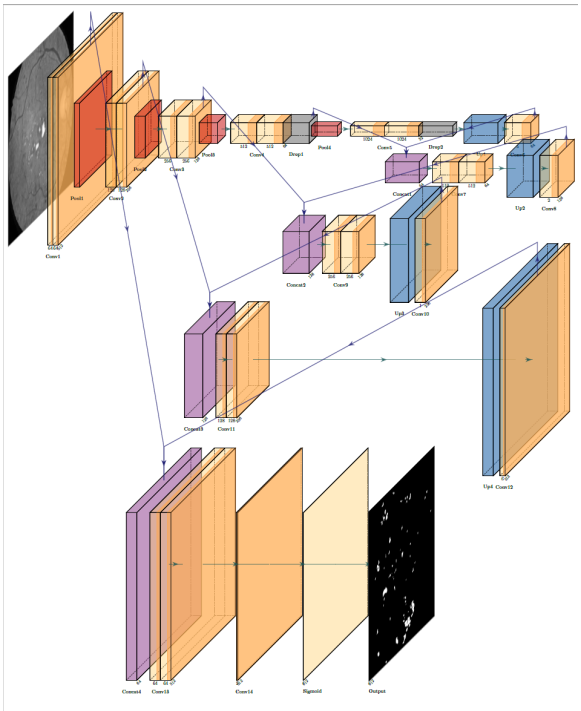


Fig. 4: Structure of U-net (plot by Latex)

For U-net, we use the following parameters for training.
Task 1: Mini-batch equals to 5, 54 times of training for each epoch, 100 epochs in total. Here we use binary cross entropy as the loss function, Adam optimizer as the optimization method. Learning rate is set to 0.0001.
Task 2: Mini-batch equals to 5, 1000 times of training for each epoch, 150 epochs in total. Here we use binary cross entropy as the loss function, Adam optimizer as the optimization method. Learning rate is set to 0.0001.

In the encoding part, U-net uses convolution 2D followed by relu and maxpooling to downsize the input image and extract main features. In the decoding part, U-net uses upsampling followed by concolution 2D and concatenate the current layer with previous layer.

Model 2: Segnet

The structure of Segnet is shown in Fig. 5.

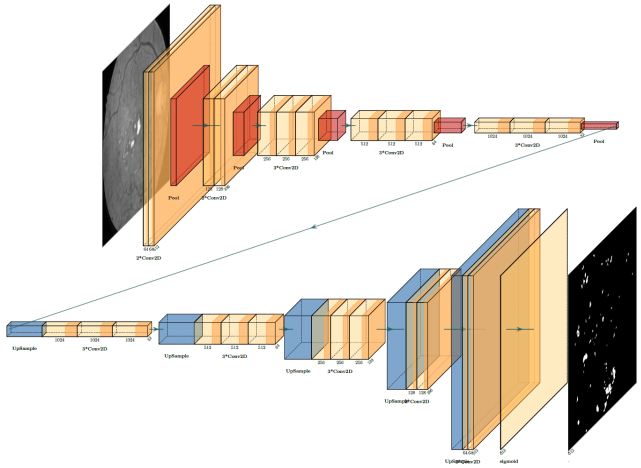


Fig.5: Structure of Segnet (plot by Latex)

For Segnet, we use the following parameters for training.
Task 1: Mini-batch equals to 5, 1000 times of training for each epoch, 20 epochs in total. Here we use binary cross entropy as the loss function, Adam optimizer as the optimization method. Learning rate is set to 0.0001.
Task 2: Mini-batch equals to 5, 1000 times of training for each epoch, 50 epochs in total. Here we use binary cross entropy as the loss function, Adam optimizer as the optimization method. Learning rate is set to 0.001.

In the encoding part, same as U-net, Segnet uses convolution 2D followed by relu and maxpooling to downsize the input image and extract main features. In the decoding part, unlike U-net, Segnet only uses upsampling to get the final output. As can be seen form the network structure, there is no fully connected layers and hence it is only convolutional.

Model 3: FCN

The structure of FCN is shown in Fig. 6.

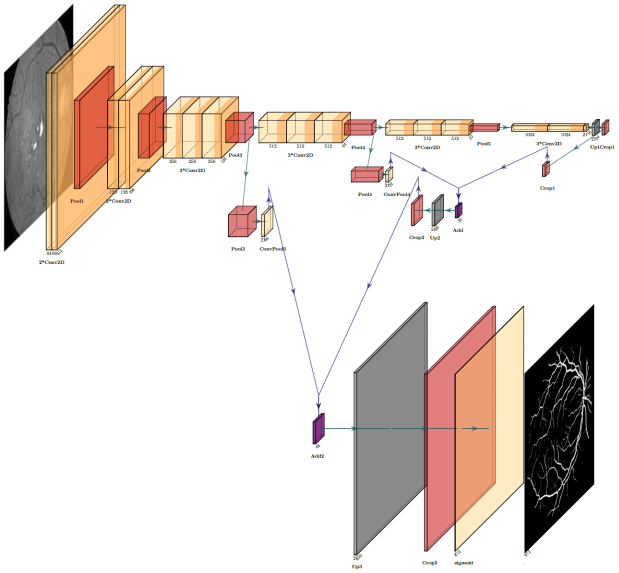


Fig.6: Structure of FCN (plot by Latex)

For FCN, we use the following parameters for training.
Task 2: Mini-batch equals to 5, 1000 times of training for each epoch, 160 epochs in total. Here we use binary cross entropy as the loss function, Stochastic gradient descent optimizer as the optimization method. Learning rate is set to 0.0001.

In the encoding part, same as U-net, FCN uses convolution 2D followed by relu and maxpooling to downsize the input image and extract main features. In the decoding part, FCN uses up convolution 2D, which is also called transposed convolution to do the upsampling work. But unlike upsampling, transposed convolution contains parameters. Then it uses add to combine with previous layer.

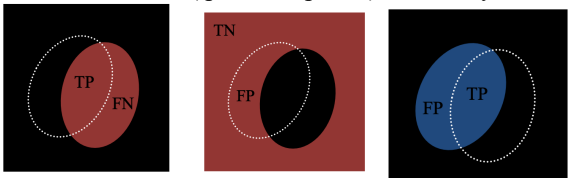
EXPERIMENTAL SETUP

Evaluation metrics

Evaluation matrices are used for analyze the performance of the algorithm. The evaluation matrix is calculated after applying the training model. We compare the result with the ground-truth mask and use sklearn.metrics.confusion_matrix to calculate tn, fp, fn, tp, which is shown in Fig. 7. Applying the value to the equation respectively,

Recall = Sensitivity = tp / (tp + fn): Fraction of the true object that is correctly segmented.
Specificity = tn / (tn + fp): Fraction of the true background that is correctly segmented.
Precision = tp / (tp + fp): Fraction of the segmented object that is correctly segmented.
F_measure = 2 * (Recall * Precision) / (Recall + Precision): Harmonic mean of precision and recall.

Jaccard similarity coefficient = tp / (fp + tp + fn): Fraction of the union of the segmented object and the true object that is correctly segmented.
Dice similarity coefficient = 2 * tp / (fp + 2 * tp + fn): Fraction of the segmented object set joined with the true object set that is correctly segmented.
accuracy = (tp + tn) / (tp + tn + fp + fn): Accuracy.



a b c

Fig. 7: Pixel classification

Receiver operating characteristic (ROC) analysis is also used in this project to compare the performance of different methods.

RESULTS

Task 1

The result segmentation of four types of diseases in retinal image are shown in Fig. 8, 9. Followed by the result of evaluation in Fig. 10.

Task 2

The result segmentation of blood vessels in retinal image are shown in Fig. 10. Followed by the result of evaluation in Fig. 11.

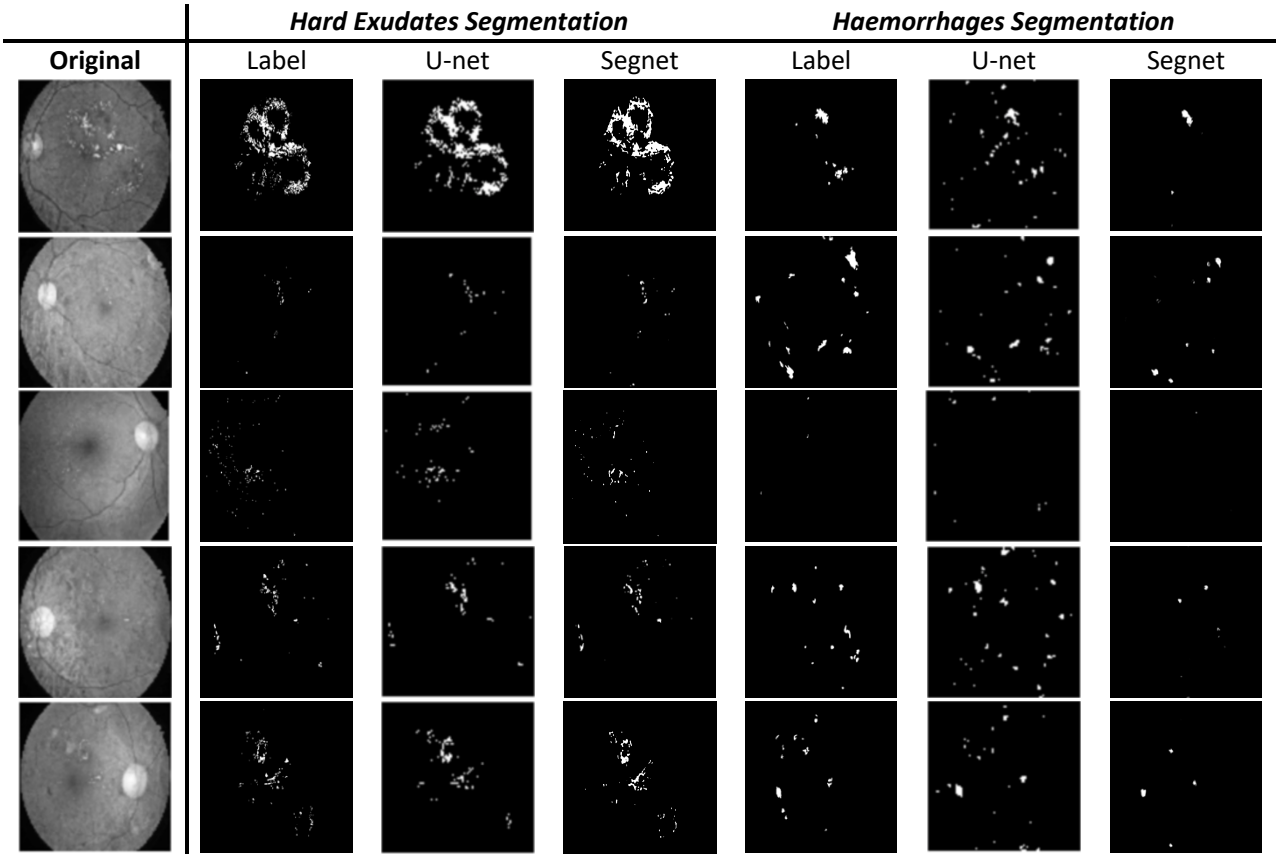


Fig. 8: Result of data augmentation

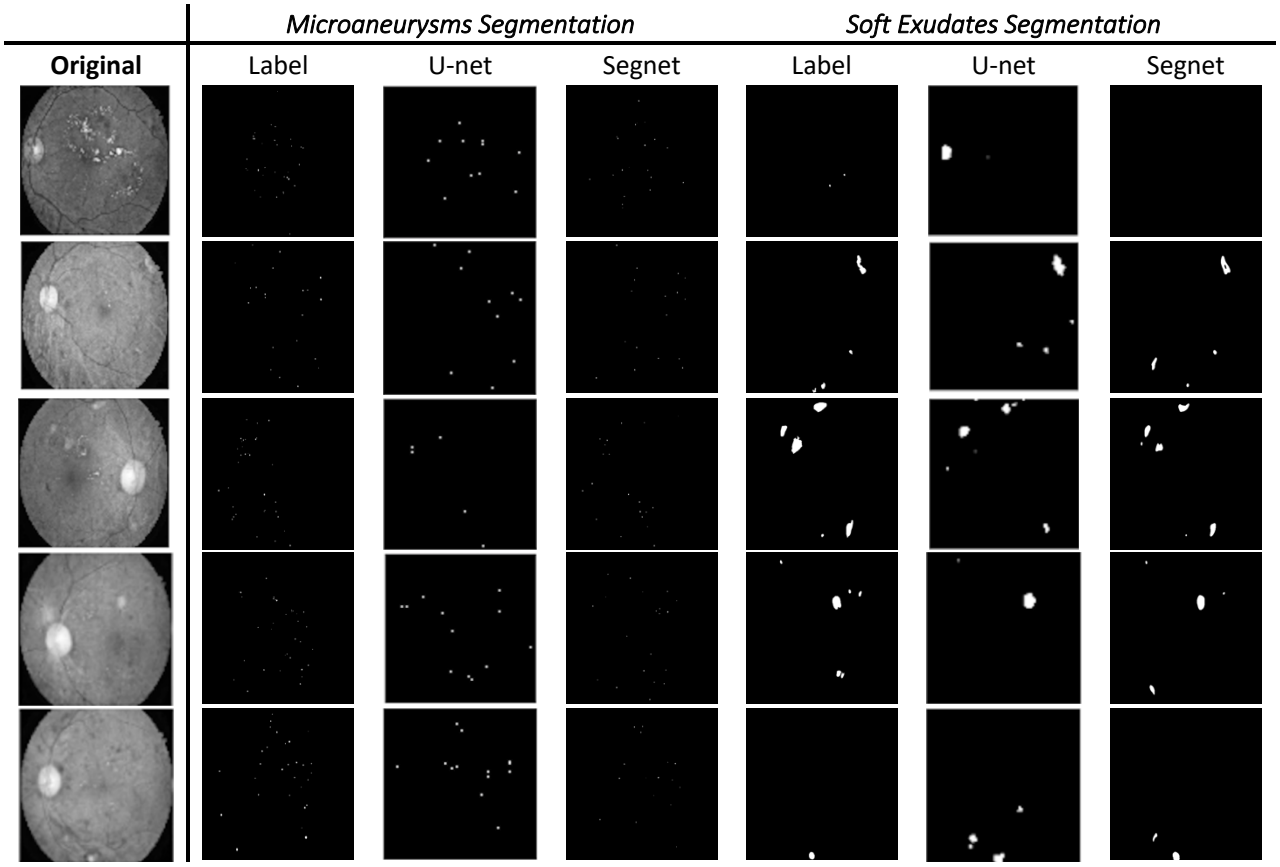


Fig. 9: Result of data augmentation

	Hard Exudates Segmentation		Haemorrhages Segmentation		Microaneurysms Segmentation		Soft Exudates Segmentation	
	U-net	Segnet	U-net	Segnet	U-net	Segnet	U-net	Segnet
Sensitivity	0.7846	0.5970	0.3654	0.1254	0.2436	0.2153	0.5298	0.6220
Specificity	0.9960	0.9942	0.9922	0.9991	0.9991	0.9997	0.9978	0.9990
Precision	0.7948	0.5345	0.4204	0.6267	0.3674	0.3929	0.5363	0.6905
Recall	0.7846	0.5970	0.3654	0.1264	0.2436	0.2153	0.5298	0.6220
F-measure	0.7897	0.5641	0.3910	0.2104	0.2918	0.2782	0.5330	0.6545
Jaccard similarity coefficient	0.6525	0.3928	0.2430	0.1176	0.1708	0.1616	0.3634	0.4864
Dice similarity coefficient	0.7897	0.5641	0.3910	0.2104	0.2918	0.2782	0.5330	0.6545
Total accuracy	0.9921	0.9900	0.9827	0.9899	0.9975	0.9989	0.9955	0.9978

Fig. 10: Evaluation results

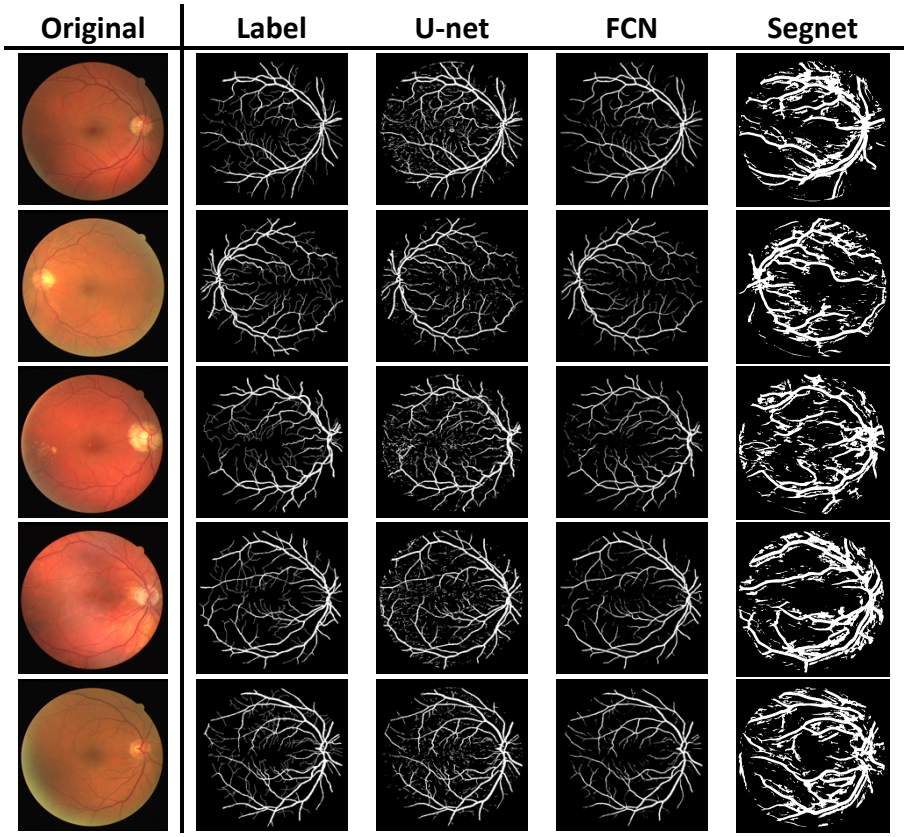


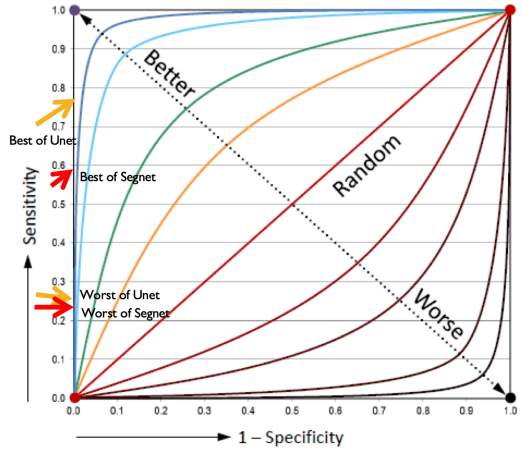
Fig. 11: Result of data augmentation

	U-net	*Segnet*	*FCN*
Sensitivity	0.6670	0.7216	0.7526
Specificity	0.9844	0.8358	0.9839
Precision	0.8409	0.2962	0.8170
Recall	0.6670	0.7216	0.7526
F-measure	0.7439	0.4200	0.7835
Jaccard similarity coefficient	0.5922	0.2658	0.6441
Dice similarity coefficient	0.7439	0.4200	0.7835
Total accuracy	0.9494	0.8258	0.9636

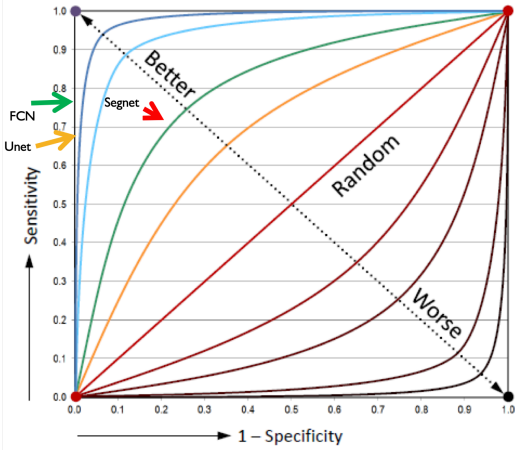
Fig. 12: Evaluation results

DISCUSSION AND CONCLUSION

The Receiver operating characteristic (ROC) for both task 1 and task 2 are shown in Fig. 13.



a



b

Fig. 13: The Receiver operating characteristic (ROC). a is for task 1, b is for task 2.

In Task 1, the overall accuracy of U-net and Segnet is very close. Compare their performance by ROC analysis in Fig. 13a, U-net is a better method.

Since most of the pixels in this segmentation task is the background, the true-positive rate becomes very important, which means the ability to recognize different diseases. Compare to the Sensitivity of two methods, U-net usually performs better than Segnet. As a result, Segnet misclassified more true object pixels than U-net.

In Task 2, ROC analysis in Fig. 13b shows that the performance of the models: FCN > U-net > Segnet.
In the evaluation results (Fig. 12):
Accuracy:
FCN > U-net > Segnet
Sensitivity (TP rate):
FCN (75%) > Segnet (72%) > U-net (66%)
Specificity (TN rate):
U-net (98%) > FCN (98%) > Segnet (83%)

As can be seen from the evaluation results (Fig. 12), although Segnet's sensitivity is greater than U-net's, U-net clearly has a better result shown in Fig. 11. Since the precision shows that U-net (84%) is greater than Segnet (29%), which means every 3-4 pixels Segnet classified as positive, only one is correct. That makes Segnet looks very coarse and perform worse than U-net.

In three models, FCN have 41,619,060 parameters, U-net have 31,031,685 parameters and Segnet have 7,817,541 parameters in total, corresponding with the calculated accuracy shows that FCN > U-net > Segnet. Since FCN have the most parameters than other two models, it is convinced that FCN is the most likely model to learn the segmentation pattern.

In addition, in models decoding part, Segnet does not use previous encoding samples to build correlation from earlier layers. By contrast, U-net and FCN use previous encoding layers when decoding, the difference is U-net uses concatenate, FCN uses add to relate previous layers.

We think concatenate is a better way to relate previous information, however the main reason that FCN is better than U-net is because when doing decoding, U-net uses upsampling but FCN uses convolution 2D transpose, which is an inverse version of convolution. Since convolution 2D transpose describes a different form of convolution, it needs parameters to learn how to map features, but the upsampling that U-net uses does not need parameters therefore it cannot learn like FCN. As a result, FCN can learn more features from the decoding part, which makes FCN a better model.

Contribution of Group Members

Jingwen Zhang: write report, presentation, implement task 2
Kuo Shi: write report, implement task 1&2
Boxuan Hu: write report, implement task 1
Yan Wang: find reference
Jianghang Cheng: find reference

REFERENCES

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. " U-Net: Convolutional Networks for Biomedical Image Segmentation" In Computer Science Department and BIOSS Centre for Biological Signalling Studies, University of Freiburg, Germany.

[2] Salamat, Nadeem, Malik M. Saad Missen, and Aqsa Rashid. "Diabetic retinopathy techniques in retinal images: a review." *Artificial intelligence in medicine* (2018).

[3] Sinthanayothin, Chanjira, James F. Boyce, Tom H. Williamson, Helen L. Cook, Evelyn Mensah, Shantanu Lal, and David Usher. "Automated detection of diabetic retinopathy on digital fundus images." Diabetic medicine 19, no. 2 (2002): 105-112.

[4] W. Xianchenga *et al.*, "Retina Blood Vessel Segmentation Using A U-Net Based Convolutional Neural Network," 2018.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition,* 2016, pp. 770-778

[6] Long, J., Shelhame, E., & Darrell, T. (n.d.), "Fully Convolutional Networks for Semantic Segmentation." In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2015.