VILNIUS UNIVERSITY

FACULTY OF MATHEMATICS AND INFORMATICS

SOFTWARE ENGINEERING STUDY PROGRAM

INVESTIGATION OF EYE FUNDUS BLOOD VESSEL SEGMENTATION USING AUTOENCODERS

AUTOENKODERIŲ TAIKYMAS TIKSLU ATPAŽINTI AKIES DUGNO KRAUJAGYSLES

Master's thesis

Author: Aleksandr, Shirvinskii _____

Thesis supervisor: Povilas, Treigys, assoc. prof. _____

Reviewer: Linas, Petkevičius, assoc. prof. _____

Vilnius, 2022

Summary

Progress of compact high definition digital retinal cameras allowed us to reveal structural and functional information about the human retina in a harmless and non-invasive way. Eye diseases cause noticeable pathological changes, so segmentation of vessels in fundus images is of great importance. In this research work I developed a new retinal vessel segmentation method that achieves state-of-the-art performance in several metrics and with threshold optimization the model showed comparable results to well researched methods on cross datasets experiments on the segmentation task.

CONTENTS

1. INTRODUCTION

Nowadays researchers seek to develop automated reasoning systems that could assist ophthalmologists, primary level physicians or optometrists in the eye diseases screening process. Manual segmentation of retinal vessels is really difficult and time-consuming; it also requires special training of a specialist. Right now, to diagnose some serious disease you must go to hospital and ophthalmologists must use non-mobile and expensive devices to segment blood vessels. And when people come for examinations usually the disease is in the late stages and it is difficult to treat. Mobile technologies like hand-held eye fundus cameras enable the medical personnel to do screening at the general practitioner, take images and upload them on the online database for specialist evaluation, which makes the diagnostic process much faster. Also, the doctor can do all of this at home for those low mobile patients who cannot get to the hospital.

The condition of the vascular network of the human eye is an essential diagnostic factor in ophthalmology. Eye diseases cause noticeable pathological changes, so segmentation of vessels in fundus images is of great importance. Over the decades, research has been carried out on receipt recognition technologies and algorithms.

Goal of this work

Automatic blood vessel segmentation is an important task for the diagnosis and the treatment of different ocular diseases. With the advent of portable fundus cameras, it has become important to segment the retinal vessels as accurately as possible and make a diagnosis outside of hospitals for patients with limited mobility. The resulting photographs can be very different, but the algorithm must accurately segment the vessels, regardless of the photograph. This master thesis aims to develop improvements of an autoencoder based fundus blood vessel segmentation algorithm. The research consists of analysis on present blood vessel segmentation methods and suggested approaches.

Tasks

— Investigate literature of the field.
— Develop a blood vessel segmentation autoencoder based model.
— Test and analysis of the developed model.
— Cross-datasets experiment on developed model.

1.1. Retinal Fundus Imaging

The retina is the inner shell of the eye, which is the peripheral part of the visual analyzer; contains photoreceptor cells that provide perception and conversion of electromagnetic radiation in the visible part of the spectrum into nerve impulses, and also provides their primary processing Blood vessels in retina can be detected directly non-invasive using special camera.

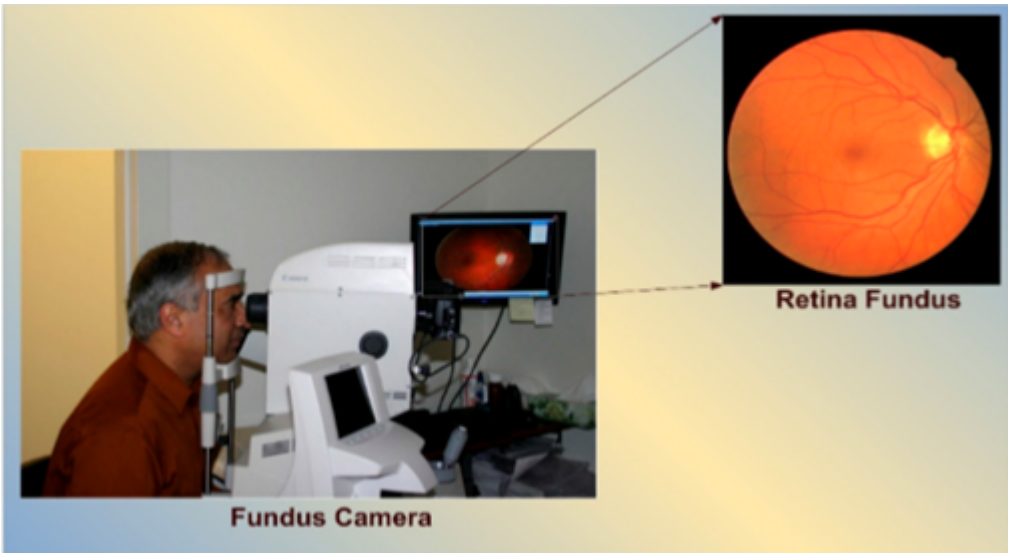

Figure 1 – Retinal fundus camera [AEE18].

For the starting point of vessel segmentation understanding the 2018 paper Retinal Vessels Segmentation Techniques and Algorithms: A Survey [AEE18] was chosen. The fundus camera shown in Figure 1 is a sophisticated optical system for retina [Kol95] photography. The fundus camera consists of a specialized low-power microscope with an attached camera and special light for retina illumination. It is designed to capture the inner surface of the eye, which includes the retina, optic disc, macula and posterior pole. There are three models that are usually used for retina photography. Color photography when the retina is illuminated with white light and the image is in full color. When shooting without red light, the contrast of vessels and other structures increases, and the light for visualization is filtered to remove red. Fluorescence angiograms were obtained using a dye tracking method. Special fluorescent dyes are injected into the bloodstream, and then, when the retina is illuminated with blue light, and the dyes emit light, a photograph is taken, as shown in Figure 2.

To diagnose diseases of the retina, the condition of the blood vessels of the retina is examined. In many cases, the vascular structure of the retina has a low contrast in relation to its background. Therefore, the diagnosis of retinal diseases becomes a complex task, and it becomes necessary to use an appropriate image segmentation technique to accurately determine the vascular structure of the retina, as this leads to an accurate diagnosis.

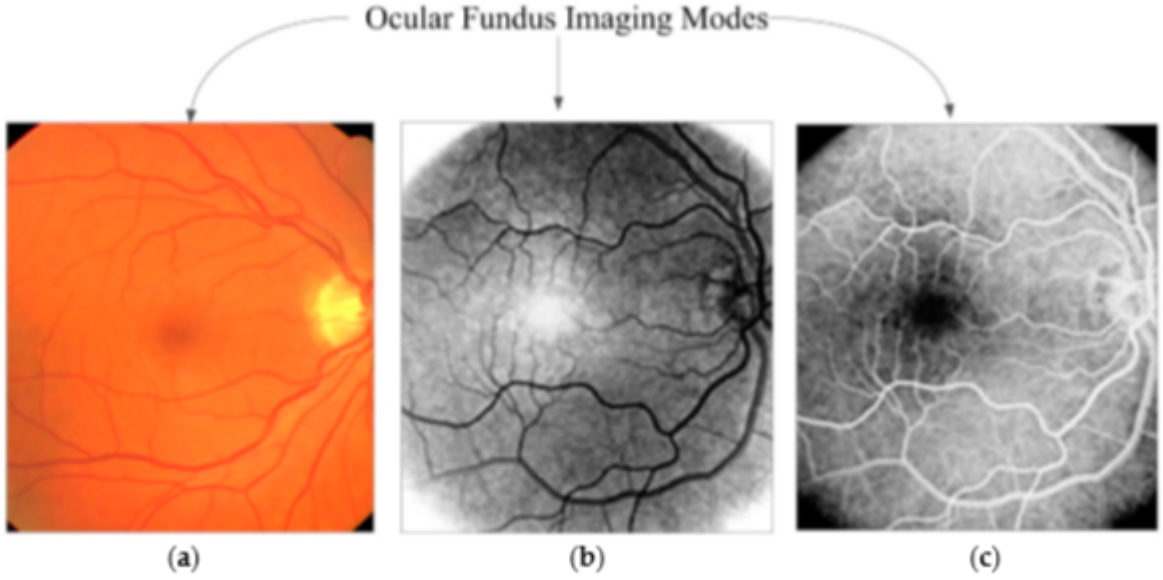

Figure 2 – Imaging modes of ocular fundus photography: (a) full color retinal fundus image; (b)monochromatic (filtered) retinal fundus image; (c)fluorescence angiogram retinal fundus image [AEE18].

The white round or oval 3 mm in diameter area in the center of the retina is the optic nerve. The retinal vasculature consists of arteries and veins, which are elongated elements, and their tributaries are visible in the image of the retina. The width of the vessels in the image depends on the actual width of the vessel and the resolution of the image, it can range from one to twenty pixels. Blood vessels fill the entire area of the retina, except “macula” or fovea [Kol95] the oval shape located in the center of the area and lies directly to the left of the optic disc. It can be expected that the vessels are connected and form a double tree structure in the retina. Intersection and branching of the vessel can complicate the profile model also local level of blood vessels intensity, the shape and size can vary drastically. Lack of image contrast, image intensity drift, and image noise create serious problems for segmenting blood vessels.

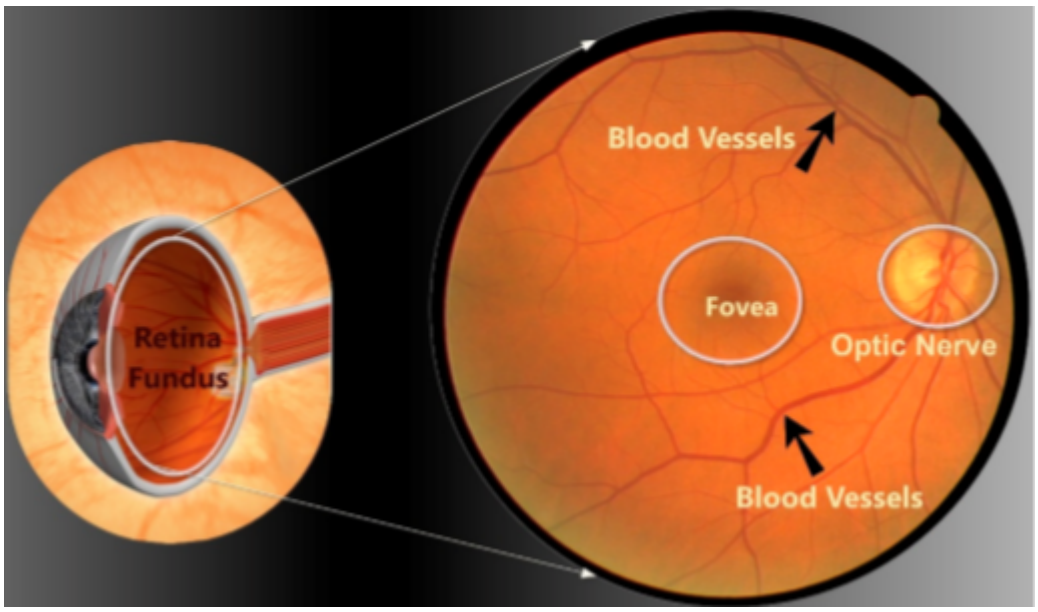

Figure 3 – Retina fundus as seen through fundus camera [AEE18].

1.1.1. Challenges of retinal vessel segmentation

Diagnosis of retinal diseases is a difficult task, and the use of an appropriate image segmentation technique becomes a necessity for highly accurate determination of the structure of the vascular retina since this leads to accurate diagnosis. Identification and retinal vascular retrieval face many problems. The width of the reference vessels can range from one to twenty pixels in the image and can have a wide range of color intensities, as shown in Figure 4, so identification technology with high flexibility is needed.In addition, the identification of vessels in pathological images of the retina faces a dilemma between accurate segmentation of the vascular structure and false segmentation near non-vascular structures (fovea, optic nerve) and pathologies (hemorrhages, microaneurysms and others).

In summary, the vascular structure of the retina within normal or abnormal images of the retina has low contrast compared to the background of the retina. In addition, non-vascular structures of the retina have high contrast compared to other backgrounds, but are fuzzy compared to pathological structures. Lesions of the optic nerve head and exudate are typical examples.

1.1.2. Datasets

There are several well-known retinal images databases. But most of the retinal vessel segmentation methodologies are trained and tested on three databases:

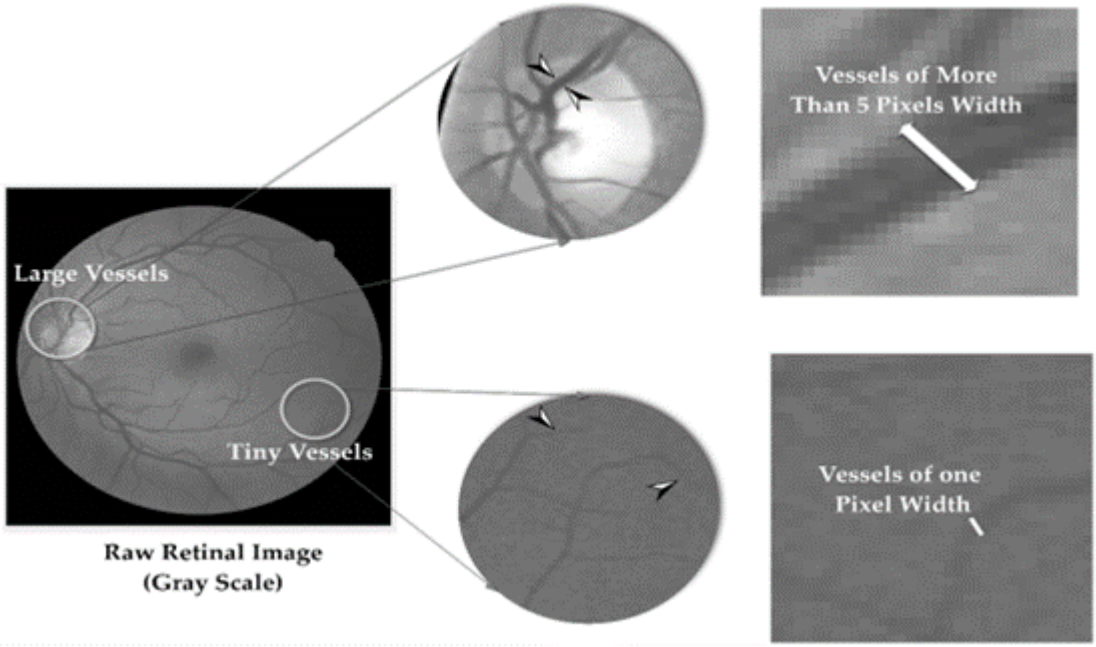

Figure 4 – Pixel width variation of retinal vessels (in pixels) [AEE18].

DRIVE, CHASE and STARE. There are different reasons for the popularity of these datasets: the good resolution of the retinal fundus images in CHASE and STARE datasets and consistently good quality and contrast in DRIVE dataset therefore better segmentation results. Also they are freely available manually labeled ground truth images. However, some researchers use other, less common datasets.

The Digital Retinal Images for Vessel Extraction (DRIVE) dataset [Sta+04] is a retinal vessel segmentation dataset. It consists of 40 color JPEG images of the fundus; including 7 cases of abnormal pathology. The images were obtained as part of the screening program for diabetic retinopathy in the Netherlands. The images were taken with a non-mydriatic 3CCD Canon CR5 camera with a FOV of 45 degrees. The resolution of each image is 584*565 pixels with eight bits per color channel (3 channels).

A set of 40 images was equally divided into 20 images for the training set and 20 images for the test set. Within both sets, for each image, there is a circular field of view (FOV) mask approximately 540 pixels in diameter. Within the training set, one manual segmentation was applied for each image by an expert ophthalmologist. Within the test set, two manual segmentations by two different observers were ap-

plied for each image, where the first segmentation of the observer was taken as valid for performance evaluation.

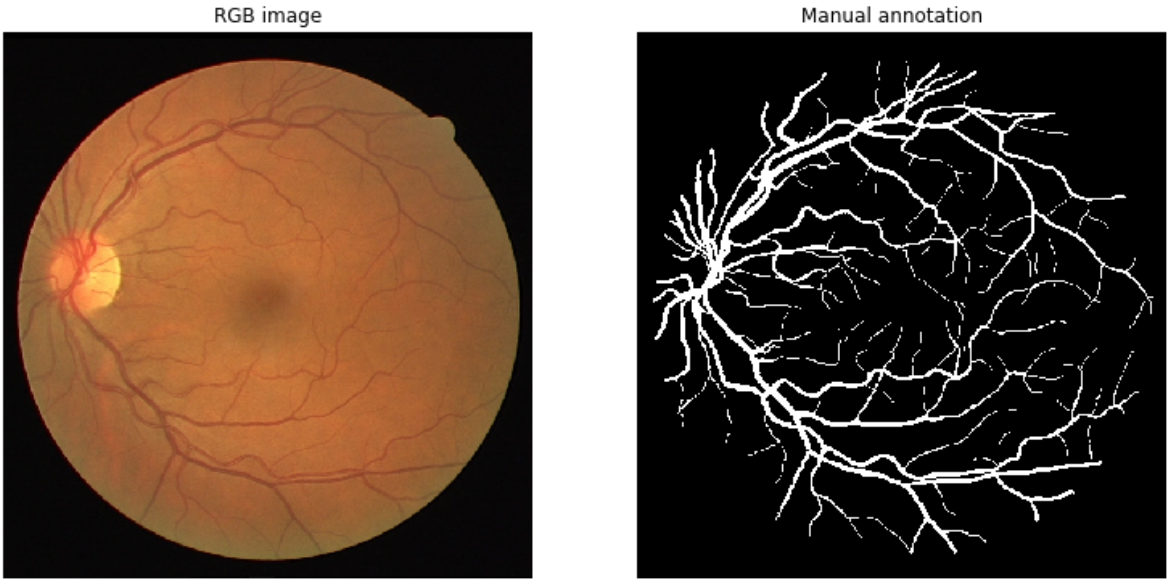

Figure 5 – The ”01” entry of the test set in the DRIVE database: retinal image, manual segmentation.

The Structured Analysis of the Retina (STARE) [HKG00] is a retinal dataset that can be used for retina segmentation and classification, optic nerve segmentation. The full set consists of nearly 400 raw images and a list of diagnosis codes and diagnoses for each image.

Expert annotations of the features visible in each image are written to text files. In total, experts were asked for 44 possible properties during data collection, and then reduced to 39 values during coding. Blood vessel segmentation work, including 40 hand-labeled images. Artery/vein labeling on 10 images by two independent experts.

CHASE [CG+09] is a retinal vessel segmentation dataset. The dataset contains 28 color retina images of 14 school children with the resolution of 999×960 pixels. Two professionals independently annotate each image.

Comparing these datasets reveals a serious discrepancy in the data: the CHASE-DB and STARE datasets are high resolution with backlighting and poor image contrast, while the DRIVE datasets are consistently good quality and contrast but low resolution. The developed model should work stably on both of these options.

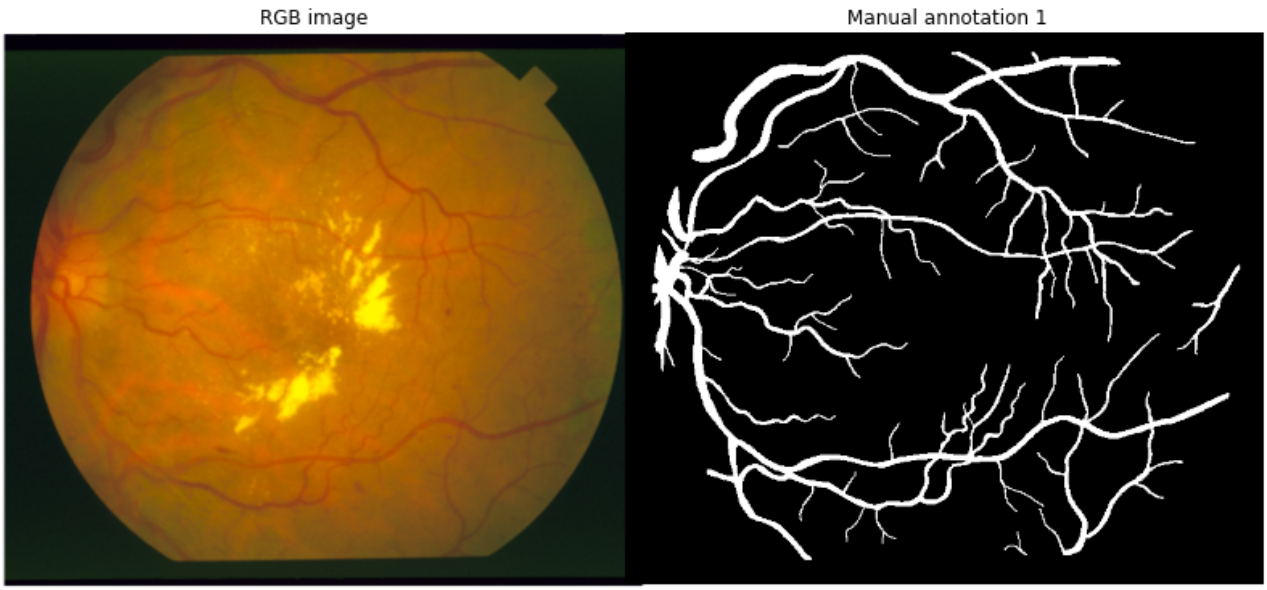

Figure 6 – The ”01” entry of the dataset in the STARE database: retinal image, manual segmentation.

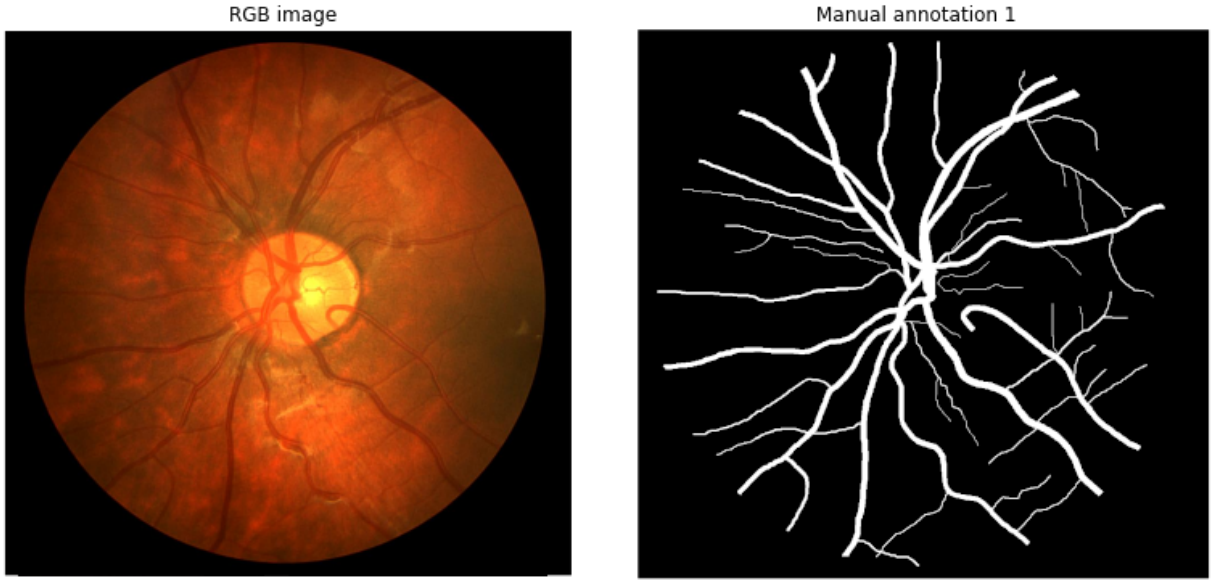

Figure 7 – The ”01” entry of the dataset in the CHASE database: retinal image, manual segmentation.

1.2. Neural networks and auto-encoders

Machine learning studies methods for constructing algorithms that do not directly solve problems, but are capable of learning. Today there are many problems that can be solved using machine learning, such as clustering, classification, regression and others.

1.2.1. Neural networks

Neural networks - a simplified model of a biological neural network, which is a collection of artificial neurons that interact with each other. The basic principles of the operation of neural networks were described back in 1943 by Warren McCallock and Walter Pitt. In 1957, neurophysiologist Frank Rosenblatt developed the first neural network [FS99], and in 2010, large amounts of training data opened up the possibility of using neural networks for machine learning.

Currently, neural networks are used in numerous areas of machine learning and solve problems of varying complexity.

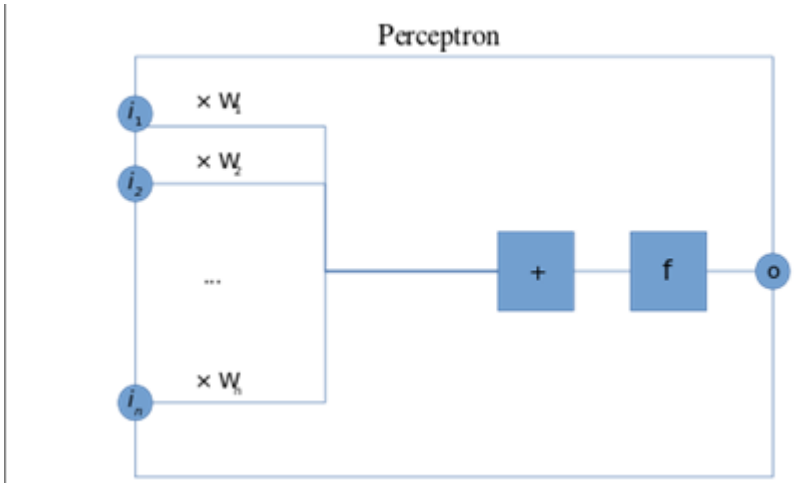

Figure 8 – Artificial neuron circuit.

$$o = f(\sum_{k=1}^{n} i_k \cdot W_k)$$

As can be seen in Figure 8, the neuron has n inputs xi, each of which has a weight wi, which is multiplied by the signal passing through the connection. After that, the weighted signals $w_i x_i$ are sent to the adder, which aggregates all the signals into a weighted sum. This amount is also called net. Thus,

$$net = \sum_{i=1}^{n} w_i x_i = w^T x$$

It is just pointless to transfer the weighted net sum to the output - the neuron must somehow process it and form an adequate output signal. For these purposes, use the activation function, which converts the weighted sum into some num-

ber, which will be the output of the neuron. The activation function is denoted by $\varphi(net))$. Thus, the output of an artificial neuron is $\varphi(net)$.

1.2.2. Neural network training

Neural network training - the search for such a set of weights, in which the input signal after passing through the network is converted to the output we need.

This definition of "neural network training" also corresponds to biological neural networks. Our brain consists of a huge number of neural networks connected to each other, each of which individually consists of neurons of the same type (with the same activation function). Our brain is trained through a change in synapses - elements that enhance or weaken the input signal.

If you train the network using only one input signal, then the network simply "remembers the correct answer", and as soon as we give a slightly changed signal, instead of the correct answer we get nonsense. We expect from the network the ability to generalize some signs and solve a problem on various input data. It is for this purpose that training samples are created.

A training sample is a finite set of input signals (sometimes together with the correct output signals) by which the network is trained. After training the network, that is, when the network produces the correct results for all input signals from the training sample, it can be used in practice. However, before immediately using a neural network, they usually assess the quality of its work on the so-called test sample.

A test sample is a finite set of input signals (sometimes together with the correct output signals), which evaluate the quality of the network. The training of the neural network itself can be divided into two approaches: supervised training and unsupervised training. In the first case, the weights change so that the network responses are minimally different from the ready-made correct answers, and in the second case, the network independently classifies the input signals.

A validation dataset is a collection of example data used to calculate the error and select the best model. To prevent overfitting when any classification parameter requires tuning, it is necessary to have an assertive dataset in addition to the training and test datasets. The test set operates in a hybrid way: it is the training data that is used for the test, but is neither part of the low-level training nor part of the final test.

1.2.3. Hyperparameters

Hyperparameters is a parameter that is not tunable during model training. A neural network is used to automate feature selection, but some parameters are manually configured.

Learning rate is a very important hyperparameter. If the learning rate is too low, then even after training the neural network for a long time, it will be far from optimal results. On the other hand, if the learning rate is too high, then the network will respond very quickly.

The activation function is one of the most powerful tools that affects the force attributed to neural networks. In part, it determines which neurons will be activated, in other words, and what information will be transmitted to subsequent layers.

Without activation functions, deep networks lose much of their learning ability. The non-linearity of these functions is responsible for increasing the degree of freedom, which allows generalization of high-dimensional problems in lower dimensions.

The loss function is at the center of the neural network. It is used to calculate the error between real and received responses. Our global goal is to minimize this error. Thus, the loss function effectively brings neural network training closer to this goal.

The loss function measures ”how good” the neural network is for a given training set and expected responses. It can also depend on variables such as weights and biases.The loss function is one-dimensional and not a vector, as it estimates how well the neural network is performing as a whole.

To select hyperparameters, it is needed to divide the dataset into three parts:

— Training dataset to train the model.
— Validation dataset for calculating the error and choosing the best model.
— Test dataset to test the selected model.

The model can over fit on the validation dataset. A test dataset is used to detect overfitting.

1.2.4. Metrics to evaluate capability

The capability of retinal segmentation algorithms to extract the retinal vasculature structure is evaluated by many metrics.

An intuitive, obvious and almost unused metric is **accuracy** - the proportion of correct answers of the algorithm. This metric is useless in problems with unequal classes. **Precision** is the proportion of correct answers of the model within the class - this is the proportion of objects that really belong to this class relative to all objects that the system assigned to this class. **Recall** is the percentage of true positive classifications. The completeness shows what proportion of objects that really belong to the positive class, we predicted correctly. **Specificity** is defined as the proportion of actual negatives, which got predicted as the negative.

$$Sensitivity(Recall) = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

$$Precision = \frac{TP}{TP + FP}$$

Precision and recall do not depend, unlike accuracy, on the ratio of classes and therefore are applicable in conditions of unbalanced samples. Often in real practice, the task is to find the optimal balance between these two metrics. It is clear that the higher the accuracy and recall, the better. But in real life, maximum accuracy and completeness are not achievable at the same time, and a certain balance has to be sought. Therefore, we would like to have a certain metric that would combine information about the accuracy and completeness of our algorithm. In this case, it will be easier to make a decision about which implementation to choose. The F-measure is just such a metric.

$$F1 = \frac{2}{Recall^{-1} + Precision^{-1}} = \frac{TP}{TP\ + (FP + FN)/2}$$

The performance curve is used to analyze the behavior of classifiers at various thresholds. Allows you to consider all threshold values for a given classifier. Shows the proportion of false positives (FPR) compared to the proportion of true positives (TPR). One way to compare classifiers involves measuring the area under the curve

(AUC). A perfect classifier will have an area under the ROC curve (ROC-AUC) of 1, while a purely random classifier will have an area of 0.5.

1.2.5. Loss functions.

Paper [Tag+20] described and compared different loss functions. Designing new loss functions for retinal segmentation can improve segmentation performance.

Cross Entropy is the most used loss function for the segmentation image tasks. This loss function compares each pixel individually with the ground truth image. For the case of binary segmentation, let $P(Y = 0) = p$ and $P(Y = 1) = 1 - p$. The predictions are given by the logistic/sigmoid function $P(\hat{Y} = 0) = \frac{1}{1+e^{-x}} = \hat{p}$ and $P(\hat{Y} = 1) = 1 - \frac{1}{1+e^{-x}} = 1 - \hat{p}$ where x is output of network. Then cross entropy (CE) can defined as:

$$CE(p,\, \hat{p}) \,=\, -(p\, log(\hat{p}) \,+\, (1 \,-\, p)\, log(1 \,-\, \hat{p})).$$

Equation for the multi-class segmentation can be written as:

$$CE \,=\, -\sum_{classes} p\, log\, \hat{p}$$

Weighted Cross Entropy can solve the main problem of cross-entropy loss: unbalanced representation in the image. Classes that have bigger representation can dominate the training. WCE was defined as:

$$WCE(p,\, \hat{p}) \,=\, -(\beta p\, log(\hat{p}) \,+\, (1 \,-\, p)\, log(1 \,-\, \hat{p}))$$

If there is a need to decrease the number of false positives: β should be smaller than 1 and to decrease the number of false negatives: β should be larger than 1.

Focal Loss can reduce overfit from easy examples so that during training the neural network focuses more on the difficult examples. The term $(1 \,-\, \hat{p})\gamma$ was added to the cross-entropy loss as:

$$FL(p, \hat{p}) \,=\, -\,(\alpha(1 \,-\, \hat{p})^{\,\gamma}\, p\, log(\hat{p}) \,+ (1 \,-\, \alpha)\hat{p}^{\,\gamma}\, (1 \,-\, p)\, log(1 \,- \hat{p}))$$

Dice coefficient is similar to the F1 score and measure ranges from 0 to 1, where a Dice coefficient of 1 means a perfect prediction from the model. The Dice coefficient (DC) is calculated as, where X is predicted and Y is ground truth:

$$DC = \frac{TP}{T\,P + F\,P + F\,N} = \frac{2|X \cap Y|}{|X| + |Y|}$$

We can define a Dice loss (DL) function. $p \in \{0,\ 1\}^n$ and $0 \leqslant p.$ where $p \in 0,\ 1^n$ and $0 \leqslant \hat{p} \leqslant 1.pisthegroundtruth$ and $\hat{p}$ is the predicted segmentation.

$$DL(p,\ \hat{p}) = \frac{2 < p, \hat{p} >}{||p||_1 + ||\hat{p}||_1}$$

Tversky loss is a generalization of the dice lost and controls the level of FP and FN by giving them weights. With $\beta = 0.5$ it is just a dice lost function.

$$TL(p,\ \hat{p}) = \frac{< p, \hat{p} >}{< p, \hat{p} > +\beta < 1 - p, \hat{p} > +(1 - \beta) < p, 1 - \hat{p} >}$$

Exponential Logarithmic Loss is a weighted sum of the exponential logarithmic Dice loss and the weighted exponential cross-entropy loss. This function can improve the segmentation accuracy on small structures for tasks where there is a large variability among the sizes of the objects to be segmented.

$$L = w_{eld}\ E\ [(-ln(D_i))^{\gamma D}\ ] + w_{wece} E[(-ln(p_l(x)))^{\gamma CE}]$$

X - pixel position, I - predicted label, l - ground truth label and the D_i - smoothed Dice loss.

The Lovasz-Softmax Loss is a smooth extension of the discrete Jaccard loss, it can be applied for the multi-class segmentation task. ΔJc (·) is the convex closure of the submodular Jaccard loss, · is a tight convex closure and polynomial time computable, C - all the classes, and Jc - Jaccard index and m(c) - the vector of errors for class c respectively.

$$L_{LovaszSoftmax} = \frac{1}{|C|} \sum_{c \subseteq C} \overline{\Delta Jc}(m(c))$$

Combo Loss discussed using solo overlap-based loss functions as regularizations along with a weighted cross entropy to explicitly handle input and output imbalance.

$$Combo\ Loss\ =\ \alpha(-\frac{1}{N}\sum_{i=1}^{N}\beta(t_i-\ln p_i)+(1-\beta)[(1-t_i)\ln(1-p_i)])+ \\ (1-\alpha)\sum_{i=1}^{K}-\frac{2\sum_{i=1}^{N}p_i t_i+S}{\sum_{i=1}^{N}p_i+\sum_{i=1}^{N}t_i+S} \tag{1}$$

α - the amount of Dice term contribution in the loss function L, and $\beta \in [0,\ 1]$ - model penalization for false positives/negatives: when β is set to a value smaller than 0.5, FP are penalized more than FN as the term $(1-t_i)\ ln\ (1-\ p_i)$ is weighted more heavily, and vice versa. S - unity constant to prevent division by 0.

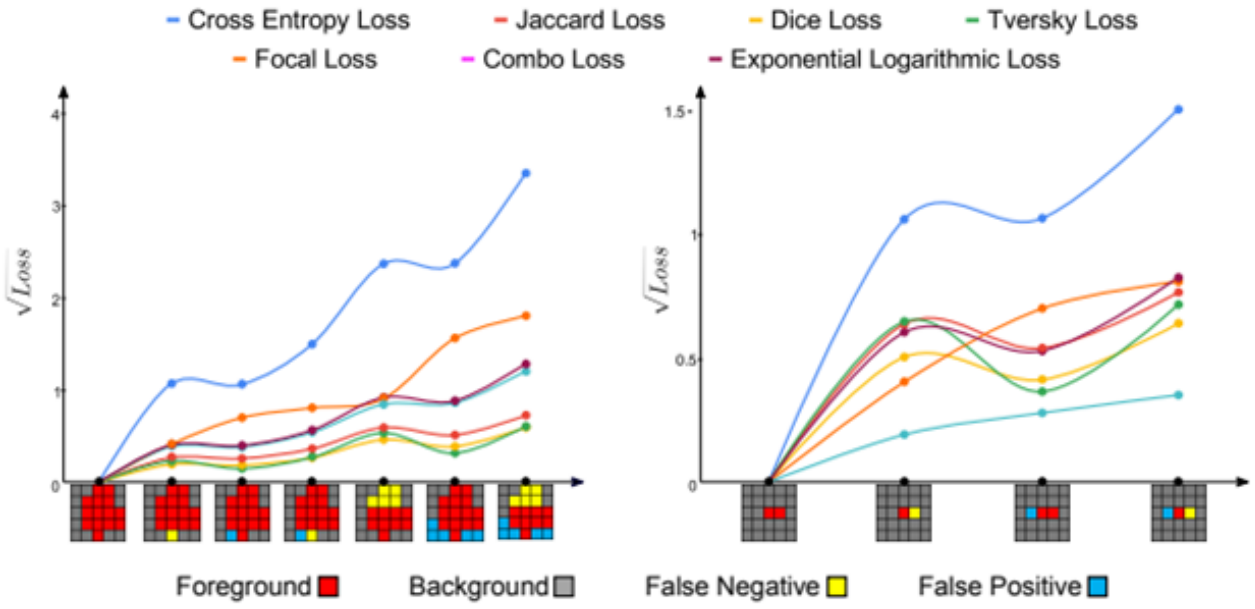

Figure 9 – A comparison of seven loss functions for different extends of overlaps for a large (left) and a small (right) object [Tag+20].

Figure 9 visualizes the behavior of different loss functions for segmenting large (left plot) and small objects (right plot). As more FP and FN are predicted the loss function value should monotonically increase. As shown on the right plot combo loss and focal loss penalize monotonically more for larger errors. This property

might increase segmentation perforce so combo loss and focal loss will be tested for the research alongside standard Cross Entropy and Weighted Cross Entropy.

1.2.6. Auto-encoders

Auto-encoder (AE) [KM91] is a special architecture of artificial neural networks that allows you to apply unsupervised learning when using the backpropagation method. The simplest autoencoder architecture is a feed-forward network, without feedback, most similar to a perceptron and containing an input layer, an intermediate layer, and an output layer. Unlike a perceptron, the output layer of an autoencoder must contain as many neurons as the input layer. The autoencoder can be used for preliminary investigation, for example, when there are too few tokenized pairs when it comes to recovery. Or to prevent data scale for early development. Or when you just need to learn how to acquire the properties of a useful input signal.

The autoencoder consists of two parts: encoder g and decoder f. The encoder converts the input signal into its representation (code): $h = g(x)$, and the decoder restores the signal according to its code: $x = f(h)$.

The autoencoder, by changing f and g, seeks to learn the identical function $x = f(g(x))$, minimizing some error function. $L(x, f(g(x)))$

At the same time, the families of functions of the encoder g and decoder f are somehow limited so that the autoencoder is forced to select the most important properties of the signal.

The autoencoder can be used for pre-training, for example, when there is a classification task, and there are too few labeled pairs. Or to downsize the data for later visualization. Or when you just need to learn to distinguish the useful properties of the input signal.

A denoising AE was proposed in 2010 in [Vin+10] paper for reconstruction of the input from the data with added noise. To achieve that AE layers are stacked on top of each other, this structure is shown in Figure 10 called Stacked Autoencoder Neural Networks (SAEs). In this structure outputs of each level are connected to the inputs of the next layer.

It is an auto-encoder architecture that has the encoder part to extract the features followed by the decoder part that reconstructs the image again to the same input dimension and generates the segmented image. Skip connections between the

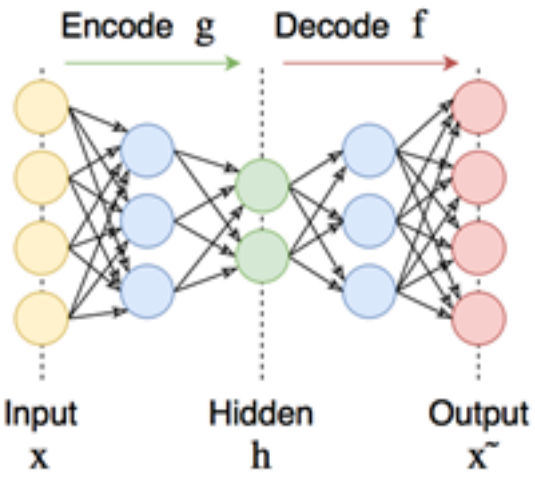

Figure 10 – Deep auto-encoder (AE) Ref: [Habr]

encoder and the decoder fuse some features from the encoder with their matched features from the decoder. The skip connection layers preserve the features details and help to transfer rich information from the encoder features and fuse them with the decoder features to better segment the vessels, especially the tiny ones.

1.3. Other types of neural networks that can be used for retina vessels segmentation

Generative Adversarial Nets (GAN) are a machine learning algorithm that is part of the family of generative models and is built on a combination of two neural networks: the generative model G, generates candidates, and the discriminative model D, which estimates the probability that the candidate came from training data, not generated by model G. Training for model G consists in maximizing the probability of error of the discriminator D.

A convolutional neural network (CNN) a special architecture of deep learning neural networks, initially aimed at efficient image recognition but right now can be used for other pattern recognitions.

In a convolutional neural network, the outputs of intermediate layers form a matrix (image) or a set of matrices (multiple image layers). So, for example, three image layers (R-, G-, B-channels of the image) can be fed to the input of a convolutional neural network. The main types of layers in a convolutional neural network are convolutional layers, pooling layers, and fully connected layers.

Table 1 – Advantages and disadvantages of discussed segmentation algorithms.

Algorithm	Advantages	Disadvantages
CNN	Well researched type of neural networks	Performance is much lower compare other algorithms
GAN	The best performance	Complex model structure that require powerful hardware to learn and execute
Auto-encoder	Well researched type of neural networks. Relatively fast to learn and execute models	

2. MAIN MATERIAL SECTIONS

2.1. Related works

The segmentation task can be solved by many methods and algorithms for specific cases and situations. This paper only deals with machine learning methods, or rather, methods based on autoencoders. There are a lot of different approaches in vessels segmentation using generative adversarial networks or fully convolutional neural networks, but in this work result will be compared to other autoencoders based solutions, because those solutions is relatively fast to train with not worse segmentation performance: 4-5 hours on consumer graphics card compare to days to train GAN based model on the professional graphics card. Also the resulting model for the autoencoder based model is much smaller in size and complexity compared to GAN based models. Convolutional neural networks based models were popular several years ago in the vessel segmentation field, but right now their performance is much lower compared to autoencoders based solutions.

2.1.1. UNet

U-net [RFB15] is a convolutional neural network that was created in 2015 for biomedical image segmentation. The novelty of this neural network is that it can accurately segment images with a limited amount of training data. It is a really useful property in medical imaging where big annotated datasets are rare. U-net is common in all kinds of medical fields where segmentation is needed from retina photography to brain and liver image.

The network architecture is illustrated in Figure 11. There are two parts of the network: the convolutional part (left) and unfolding part (right). There are 5 different steps in the network and 23 convolutional layers in total:

- 3x3 convolutions - unpadded convolutions with Rectified linear units (ReLU).
- 2x2 max pooling operation with stride 2 for downsampling, double the number of features.
- 2x2 convolution - up-convolution, halves the number of features.
- A concatenation with the correspondingly cropped feature map from the contracting path.
- 1x1 convolution is for mapping a 64-component feature vector to the number of classes of segmentation.

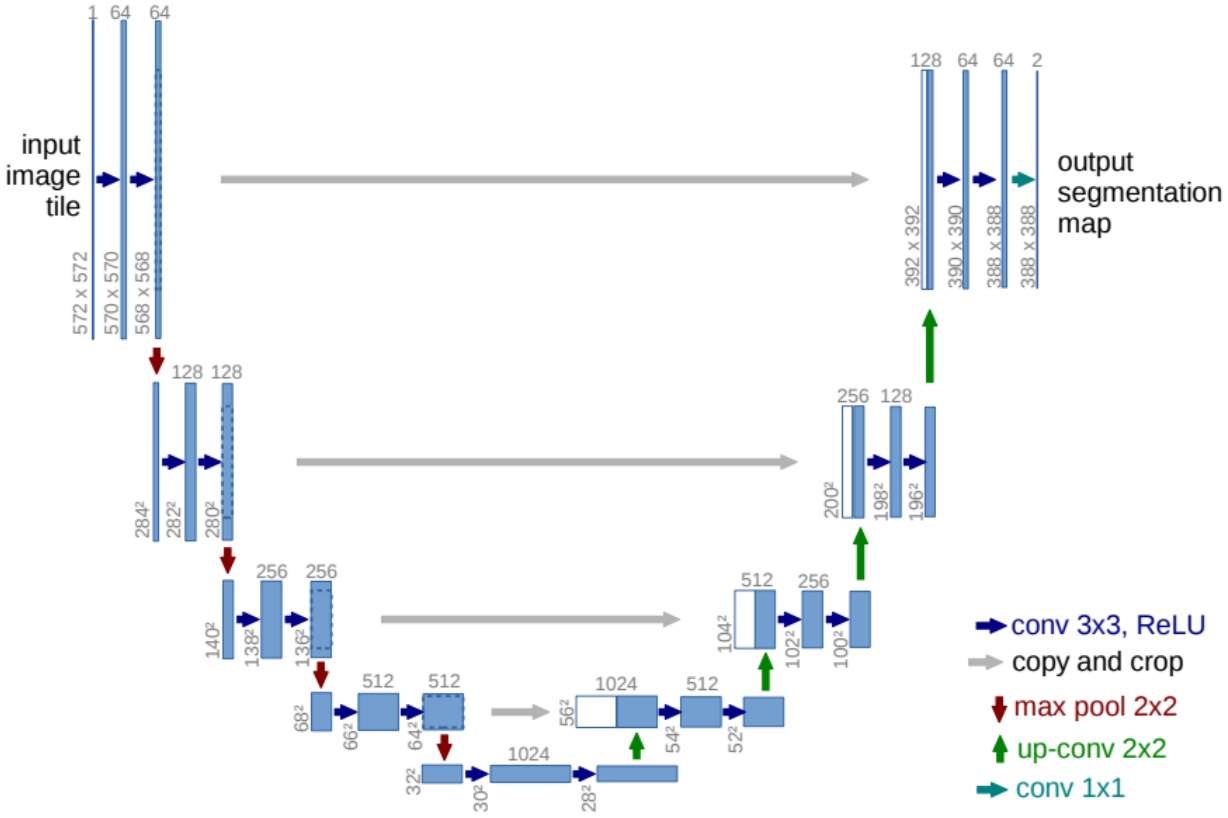

Figure 11 – U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations Ref: [RFB15]

2.1.2. UNet based methods and performance increasing strategies

Nest U-net and patch-learning combine was proposed in paper [WZY21], it increased fine retinal vessel segmentation performance. Special extraction strategies were designed to effectively generate massive training samples containing fine retinal vessels. Nest U-net model was designed as an image segmentation network model, which directly fast-forwards high-resolution feature maps from the encoder to the decoder network. This model was trained by the k-fold cross-validation strategy, and testing patches were predicted, and the segmentation result was reconstructed by the sequential reconstruction strategy.

In paper [Cha+20] was presented a Channel Attention Residual U-Net (CAR-UNet) for retinal vessel segmentation of funds images. CAR-UNet considers the relationship between the feature channels, so a novel channel attention mechanism

is introduced to strengthen the network's discriminative capability. Firstly they introduced a Modified Efficient Channel Attention (MECA) modified from the recently proposed Efficient Channel Attention (ECA). Also they integrated MECA into Double Residual Block (DRB) to construct the contracting path and expansive path of the network. In addition, they applied MECA to "skip connections", assign weights to feature maps from the contracting path, instead of equally copying to the corresponding expansive path.

In the article [Alo+18], a recurrent convolutional neural network (RCNN, RU-Net) and recurrent residual convolutional neural network (RRCNN, R2U-Net) was proposed. Those two networks are both based on the U-Net model. There are several advantages in these architectures for segmentation tasks. First, the residual unit helps in teaching deep architecture. Second, the accumulation of features using repeated residual convolutional layers provides a better representation of features for segmentation tasks. Third, these architectures have the same number of parameters and better segmentation performance on retina segmentation.

The researcher suggests a Study Group Learning (SGL) approach in their [ZYS21] article to increase the resilience of models trained on noisy labels in limited datasets. It was influenced by knowledge distillation techniques and the K-fold cross-validation approach. Also researchers proposed a unique enhancer module in a model that generates the enhancement map. Using the source image and producing a 3-channel output with a higher contrast level. Researchers compared learnt enhancement of retinal pictures to other baseline approaches such as Histogram Equalization (HE), Contrast Limited Adaptive Histogram Equalization (CLAHE) [Piz+87], and Retinex [Zha+15] in their article. Traditional approaches such as HE, CLAHE, and Retinex are unable to achieve a consistent contrast level both locally and regionally. The chopped patches are from either the brighter or darker portions of the picture, making it difficult for the inspector to analyze the vessel pixels effectively. The learnt map in the fifth column, on the other hand, has a higher contrast and intensity level, boosting vessel information and making it easier for physicians to identify dark spots. The vessel sections are highlighted while the textures are preserved. Visual examination or labeling can also be done with the enhanced photos. In the DRIVE and CHASE DB1 datasets, their results show that the suggested technique enhances vessel segmentation performance, even when the training labels are noisy.

Table 2 – Comparison of discussed algorithms.

Algorithm	Augmentation	Preprocessing	Model	Loss function
Channel Attention Residual U-Net	Random rotation and horizontal, vertical and diagonal flips	-	CAR-UNet + Modified Efficient Chanel Attention	Binary cross entropy
Nest U-net and patch-learning	-	image conversion and data nor-malization, random extraction strategy	Nest U-net	Categorical cross-entropy
R2U-Net	Data whitening, rotation, translation, and scaling	Mean subtraction and normalized according to the standard deviation	Residual RCNN based U-Net model	Binary cross entropy
SGL-Retinal-Vessel-Segmentation	Deeply unsupervised learned enhancement	Study Group Learning	Concatenated UNet consisting of an enhancement module and a segmentation module	pixel-wise binary cross entropy loss

All of the related articles was focused on improving performance results on the same dataset as model was trained. There is a lack of a sufficient number of modern research papers with cross datasets experiments therefore not it is impossible to determine how models will work on real world data. For this reason, the studied topic is highly relevant and of great practical importance in medicine. Also there is no experiments with different loss functions that can substantially improve model performance.

Table 3 – Advantages and disadvantages of discussed models.

Model	Advantages	Disadvantages
Channel Attention Residual U-Net	Channel Attention block for improving the performance of CNN. Channel Attention Double Residual Block to prevent overfitting.	Complex model structure without meaningful performance improvement
Nest U-net and patch-learning	Promising performance improvements using patch-leaning	Impossible to use general structure of blood vessel tree from fundus image
R2U-Net	Lower number of parameters when compared to other methods	Path learning. Longer process of model learning.
SGL-Retinal-Vessel-Segmentation	State-of-the-art performance on the CHASE dataset. Proposed learning scheme can boost the DICE score.	Vessel label erasing should emulate annotators errors, but not clear from the work why this needed

This section describes the problem of recognizing the retinal vessels of fundus, autoencoders, neural networks and the formulation of the segmentation problem. Fundus datasets were analyzed, which revealed serious inconsistency in data. Additionally, a review of articles related to this work was done. The disadvantages of these methods are noted, which lead to the need to search for new solutions. For this reason, the studied topic is highly relevant and of great practical importance in medicine.

2.1.3. Artery/vein Classification

Classification of retinal vessels into arteries and veins is a key step for diagnosis of several eye diseases. For example glaucoma has been associated with the visible changes of retinal arteries [Orl+18]. Classification of arteries or veins is a hard task even for trained ophthalmologists. There are a lot of different approaches for the Artery/vein Classification task: graph based, neural networks, color properties based etc. But almost every work is including 3 main steps:

— Retinal vascular tree is extracted from the input image.
— Location of the optic disc is identified and removed because it is almost impossible to classify vessels inside the disk.
— Vessels are finally classified into arteries and veins.

In paper [Gal+19] a Fully-Convolutional Neural Network used for A/V segmentation task, where the task is to classify every pixel in the entire image of an artery, vein, or background. This is a different task compared to the A/V classification task. They used encoder-decoder architecture of the U-NET. Performance classification of vessel pixels with use of the proposed method was on par with all current state-of-the-art techniques other than graph-based approach [Est+15]. However, in testing, the authors use a model trained on only one dataset for all experiments, while competitors restrained their models. This testing approach is of great importance in the real world because it is important to have consistent performance regardless of the state of the photo. However, this method can be improved using a more powerful deep architecture, along with domain match regularization techniques to prevent overfitting.

The paper was published [Li+20b] in 2020. They used a U-net based model to classify arteries/veins. In addition to that they used post-processing to address errors around crossing and branching points, this significantly boosted their performance and because of that experimental results showed that their method can achieve the state-of-the-art performance on STAIR and DRIVE databases.

2.2. Developed model methodology

Standard UNet is a convolutional auto-encoder based on CNN where the output layer must contain as many neurons as the input layer to restore the original image size. First, images are downsampled and the resolution is reduced, and then the opposite occurs, upsampling with increasing image resolution. Skip connections are common in UNets, and they concatenate or add activation volumes from the downstream sampling path to the upstream sampling path to recover better resolution information and enable gradient flow during training. The UNet model φ can be parameterized by the number of downscaling and upscaling of the image k, and the number of filters applied at each level, f_k. Only 3 × 3 filters are considered, and

the number of filters is doubled each time k increases. The UNet can be completely defined by two numbers (k, f_0). After each convolutional layer, batch-Norm layers are added, and there are standard skip connections for each block.

2.2.1. Stacked UNets

The idea is that passing an image through multiple UNets can improve segmentation performance. In the beginning image x is passed through the standard UNet $\phi_1(x)$, then the output is concatenated to the original image, and passed again through a second UNet. Then again output $\phi_2(x)$ is concatenated to x, and passed again through the third UNet $\phi_3(x)$. Each previous UNet generates vessel segmentation, which is then used by the next as an attention map to focus more on areas of the image containing vessels. Last autoencoder is generating prediction based on image and most accurate attention map from the previous steps. Additional layers of UNet increases capability of fine vessel detection and can be represented as:

$$\Phi^n(x) = \phi^n(x, \phi^{n-1}((x, \phi^1(x))))$$

The loss is calculated for the final output and combined linearly with the calculated loss for the remaining outputs:

$$L(\Phi^n(x), y) = \sum_{i=0}^{n} L(\phi^i, y)$$

The number of stacked UNets is a hyper-parameter that should be chosen based on performance metric and size of the model.

2.2.2. Image enhancer

This model did not use pre-processing to save all image information. The goal is to process the image by increasing its contrast and highlighting the vessel tree in the enhanced image and evaluate the vessel segmentation map associated with the original vessel segmentation image. Resulting image contains maximum image content, including vessel structures and retinal background. This can help clinicians validate segmentation results and better explain the model.

The original image goes through the UNet model. A 3-channel output is then extracted that can visualize the vessel tree with a higher level of contrast.

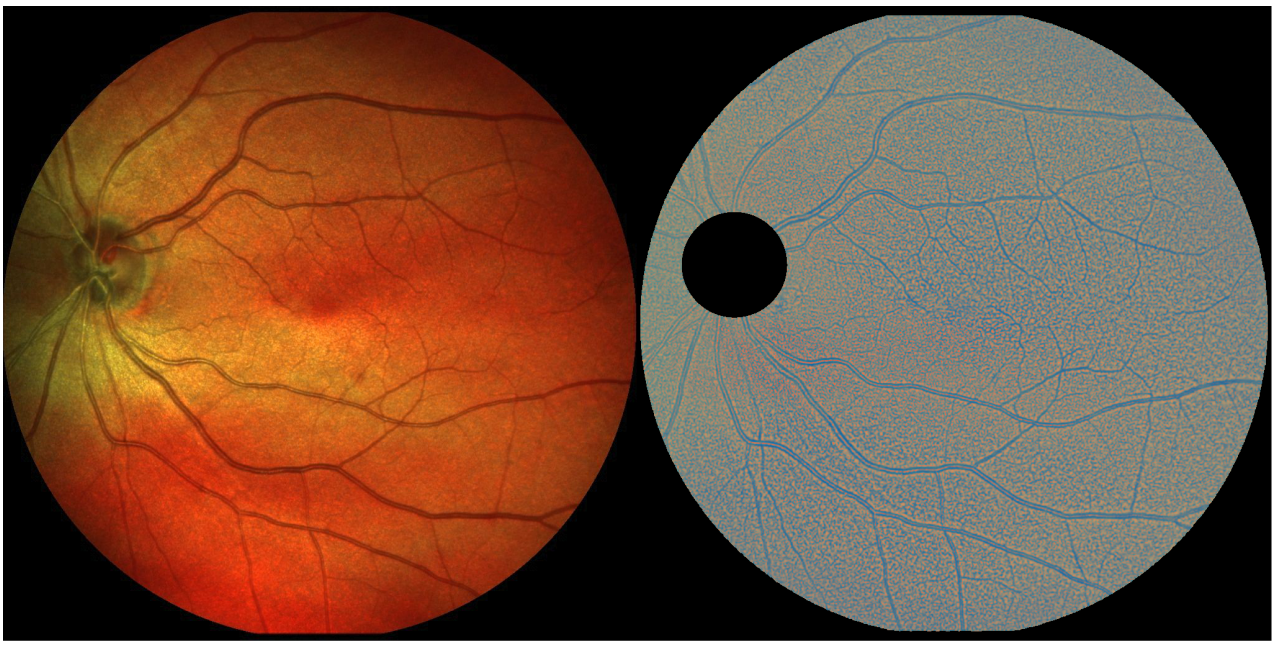

Figure 12 – The learned map demonstrates a great contrast and intensity, enhancing the vessel information for a better identifiable visualization for clinicians.

2.2.3. Augmentation

Data augmentation is a technique that you can use to expand your training set. The main way to do this is to create modified versions of objects from existing data. Particularly for images, augmentations include operations such as translations, flips, scaling, and many others. By artificially enlarging the samples, the representatives of the sample is improved, which helps machine learning algorithms to better describe the data space.

It is important to note that augmentations must produce plausible specimens. For example, horizontally rotating an image of a dog makes sense because the photo could have been taken from either the right or the left. However, flipping the same photo vertically is unlikely to make sense, given that the trained model is unlikely to see a photo of an upside-down dog.

The data transformation approach significantly improves the results, especially for tasks in which the training sample is not large enough. In the current work, the problem of a small dataset is present, so various augmentations related to rotations and mappings were used. Photographs of the main bottom can be rotated by any number of degrees, since the location of the macula depends on where the patient is looking at the moment of the photo and the photo of which eye was taken, right or

left. For this work augmentation of rotation of image by every 10 degrees was used. Examples can be seen in Figure 13.

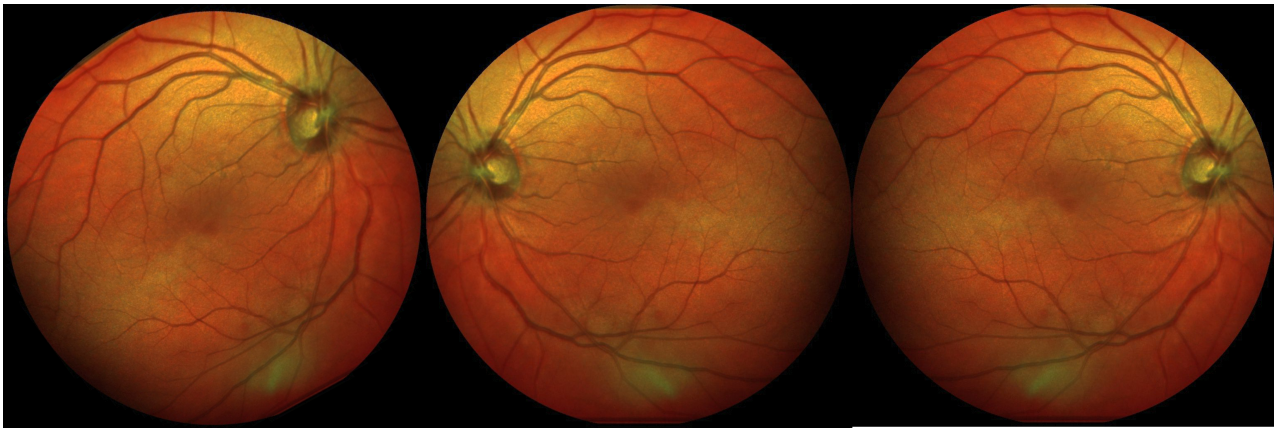

Figure 13 – Examples of augmented images from the CHASE Dataset using rotations.

Color of the picture can't be changed because this can affect the structure of the vessel tree in the retina and create impossible structure that will decrease segmentation capability.

2.2.4. Threshold optimisation

Model output is the probability set of being a vessel for each pixel. Threshold for probability can be chosen based on the best performance in terms of Accuracy, F1 score or other metric on train dataset. To calculate optimal threshold differentiation of the function should be calculated. This function is composed of two functions: image(threshold) - function is calculating binary image based on original probability image and threshold, f(image) - calculating performance metric based on binary image. Differentiation of this composition can be calculated using central-difference method:

$$y'_j = \frac{(y_{j+1} - y_{j-1})}{2 * \delta x}$$

Assuming that this function is monotonically increasing and then monotonically decreasing after the optimal threshold. So to find an optimal threshold the point where differentiation of the function is zero. This method can be used to improve performance results when testing on different than train datasets. New threshold can

be calculated based on subset of dataset images, it can dramatically increase model performance.

2.2.5. K-fold cross-validation

Cross validation is a resampling technique for testing machine learning models on a small set of data. The process has only one argument, k, which specifies how many groups this data collection should be split into. As a result, k-fold cross-validation is a common name for the process.

Cross validation is mostly used in applied machine learning to verify a machine learning model's credentials on wasted data. That is, utilize a test sample to see how the model will perform overall when it is used to generate predictions on data that was not used during training.

The general procedure is as follows:

— Divide the dataset into k-groups.

— For each unique sample: take a group as a testing dataset, take the rest of the groups as a sample of the training data, prepare a model on training samples and evaluate it on a test sample, keep the model evaluation and discard the model.

— Summarize the quality parameters of the model using the sample estimation models.

It's worth noting that each observation in the data sample is allocated to a different group and stays in that group throughout the procedure. This means that each sample is utilized once in the set and k-1 times to train the model.

Any data preparation prior to fitting the model was done on a sample of training data generated using cross-validation in a loop, rather than on a larger dataset. This holds true for any hyperparameter adjustment as well. Data leakage and an optimistic assessment of the model's quality can occur if these actions are not performed in a loop. The results of the k-fold cross-validation findings are frequently summarized with the average overall quality of the model.

After this procedure there are estimated segmentation labels $\tilde{I_{ck}}$ of G_k , where

$$\tilde{I_{ck}} = M_k(G_k), k[1, K]$$

Then pseudo label was obtained using all the predicted estimated segmentation labels:

$$\tilde{I}_c = \cup_{k=1}^{K} \tilde{I_{ck}}$$

Finally, the model M is trained by optimizing the obtained pseudo label set $\tilde{I}_c$ plus the ground truth vessel labels I_c:

$$L = Loss(\hat{I}_c, I_c) + Loss(\hat{I}_c, \tilde{I}_c)$$

where $\hat{I}_c = M(G)$, Loss is the chosen loss function.

3. EXPERIMENTAL RESULTS

3.1. Description of experiments

The loss is backpropagated and minimized using Adam's optimization method.. The learning rate initial value is set to $\lambda = 10^{-2}$, and annealed cyclically according to the law of cosines until it reaches $\lambda = 10^{-8}$. The training batch size is 6, and a total of 25 epochs are trained for each dataset. The model was trained with randomly cropped 256 × 256 patches with applied data augmentation techniques described in the previous part.

3.2. Loss functions

For the experiments several loss functions were chosen and tested: standard Cross Entropy and Weighted Cross Entropy as a baseline, Combo Loss and Focal Loss.

Table 4 – Models with different Loss function performance

Loss function	DRIVE		CHASE-DB	
	AUC	DICE	AUC	DICE
Cross Entropy	0.9889	0.8316	0.9918	0.8265
Weighted Cross Entropy	0.9890	0.8320	0.9920	0.8271
Focal Loss	**0.9891**	**0.8325**	**0.9922**	**0.8275**
Combo Loss	0.9890	0.8322	0.9917	0.8260

Focal Loss for the small objects penalizes monotonically more for larger errors and shows the best performance across the board, therefore it will be used in all experiments. Combo Loss showed disappointing performance even with tweaked α and β parameters; its performance was on par with Weighted Cross Entropy Loss on DRIVE dataset and even lower on CHASE dataset.

3.3. K-fold cross-validation

K-fold cross-validation learning scheme overall improves the robustness of a model output. A higher sensitivity means that the model is able to segment more thin vessels and edge pixels.

Table 5 – Results of different fold number K on DRIVE

k	Sensitivity	Specificity	DICE	Accuracy	AUC
1	0.8451	0.9820	0.8287	0.9698	0.9881
2	0.8454	0.9819	0.8329	0.9701	0.9886
4	0.8465	0.9825	0.8333	0.9704	0.9889
8	0.8481	0.9826	0.8336	0.9705	0.9890

3.4. Stacked UNets

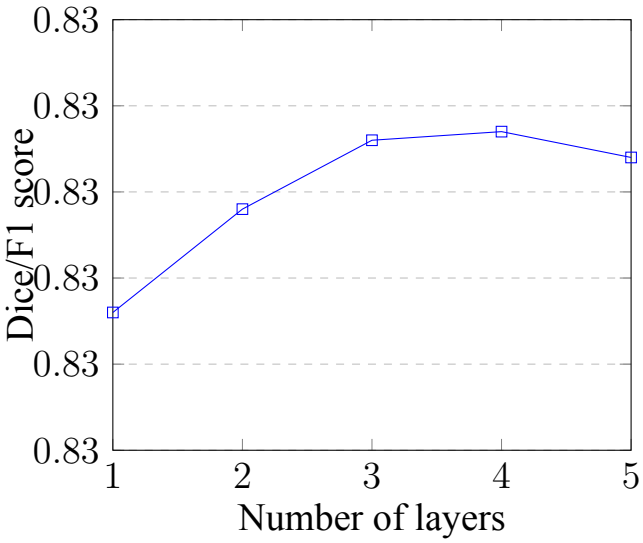

Figure 14 – Dice/F1 performance dependence of number of layers

Table 6 – Results of different number of stacked UNet layers on DRIVE

n	Sensitivity	Specificity	DICE	Accuracy	AUC
1	0.8380	0.9834	0.8316	0.9705	0.9886
2	0.8481	0.9827	0.8328	0.9705	0.9889
3	0.8481	0.9826	0.8336	0.9705	0.9890
4	0.8481	0.9820	0.8337	0.9705	0.9890
5	0.8480	0.9812	0.8332	0.9705	0.9889

From this graph 14 we can conclude that the optimal number of stacked UNet layers is 3, because it is an optimal number of layers for DICE score metric and at the same time Specificity is not much lower compared to baseline. Further increase of this number didn't increase DICE performance of the model and time to train model became unreasonably high as well as Specificity became much lower compared to the baseline.

3.5. Model trained on DRIVE dataset

The Digital Retinal Images for Vessel Extraction (DRIVE) dataset consists of 40 images of size 584*565 with eight bits per color channel (3 channels). The train/-validation/test splits for DRIVE are provided by the authors and the ground truth manual annotations are given. The set of 40 images is divided into 20 images for the testing set and 20 images for the training set.

Table 7 – Comparison with other baseline methods on DRIVE dataset

Method and Year	Sensitivity	Specificity	DICE	Accuracy	AUC
R2U-Net [Alo+19] 2018	0.7792	0.9813	0.8171	0.9556	0.9784
LadderNet [Zhu18] 2018	0.7856	0.9810	0.8202	0.9561	0.9793
Dual E-UNet [WQH19] 2019	0.7940	0.9816	0.8270	0.9567	0.9772
IterNet [Li+20a] 2020	0.7791	0.9831	0.8218	0.9574	0.9813
SA-UNet [Guo+20] 2020	0.8212	0.9840	0.8263	0.9698	0.9864
BEFD-UNet [Zha+20] 2020	0.8215	0.9845	0.8267	0.9701	0.9867
SGL [ZYS21] 2021	0.8380	**0.9834**	0.8316	**0.9705**	0.9886
My model	**0.8480**	0.9828	**0.8339**	**0.9704**	**0.9893**
stderr $\times 10^{-3}$	3	0.4	0.9	0.3	0.7

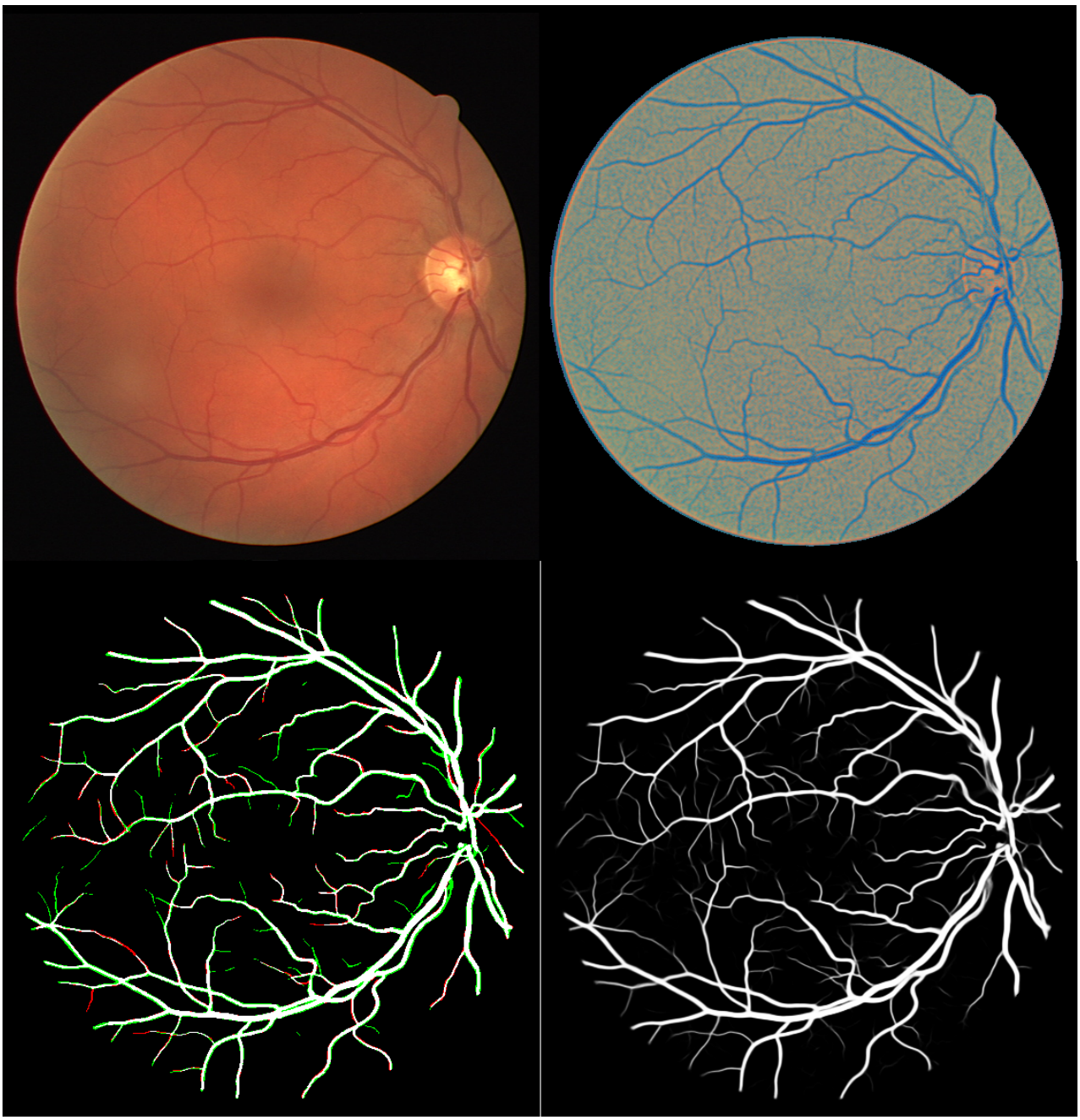

Figure 15 – Inference results on one test DRIVE example. Clinicians may only label some salient vessels while ignoring the ambiguous ones. Bottom left picture shows the predicted vessel map: red pixels are false negative and green is false positive. Bottom right picture shows the probability map that was generated by the model. All of those images as a whole can be used by clinicians to drastically improve segmentation results.

3.6. Model trained on CHASE dataset

CHASE DB1 dataset consists of 28 retinal images of size 999 × 960. There is no official split into test and train sets. It was decided to use the most commonly

used split. The first 20 images are used for training, and the remaining 8 images for testing.

Table 8 – Comparison with other baseline methods on CHASE dataset.

Method and Year	Sensitivity	Specificity	DICE	Accuracy	AUC
R2U-Net [Alo+19] 2018	0.7756	0.9712	0.7928	0.9634	0.9815
LadderNet [Zhu18] 2018	0.7978	0.9818	0.8031	0.9656	0.9839
Dual E-UNet [WQH19] 2019	0.8074	0.9821	0.8037	0.9661	0.9812
IterNet [Li+20a] 2020	0.7969	0.9881	0.8072	0.9760	0.9899
SA-UNet [Guo+20] 2020	0.8573	0.9835	0.8153	0.9755	0.9905
SGL [ZYS21] 2021	**0.8690**	0.9843	0.8271	0.9771	0.9920
My model	0.8570	**0.9857**	**0.8290**	**0.9774**	**0.9921**
stderr $\times 10^{-3}$	5	0.3	1	0.9	0.4

Each result is reported as an average over 5 runs for DRIVE and CHASE DB datasets along with the standard errors.

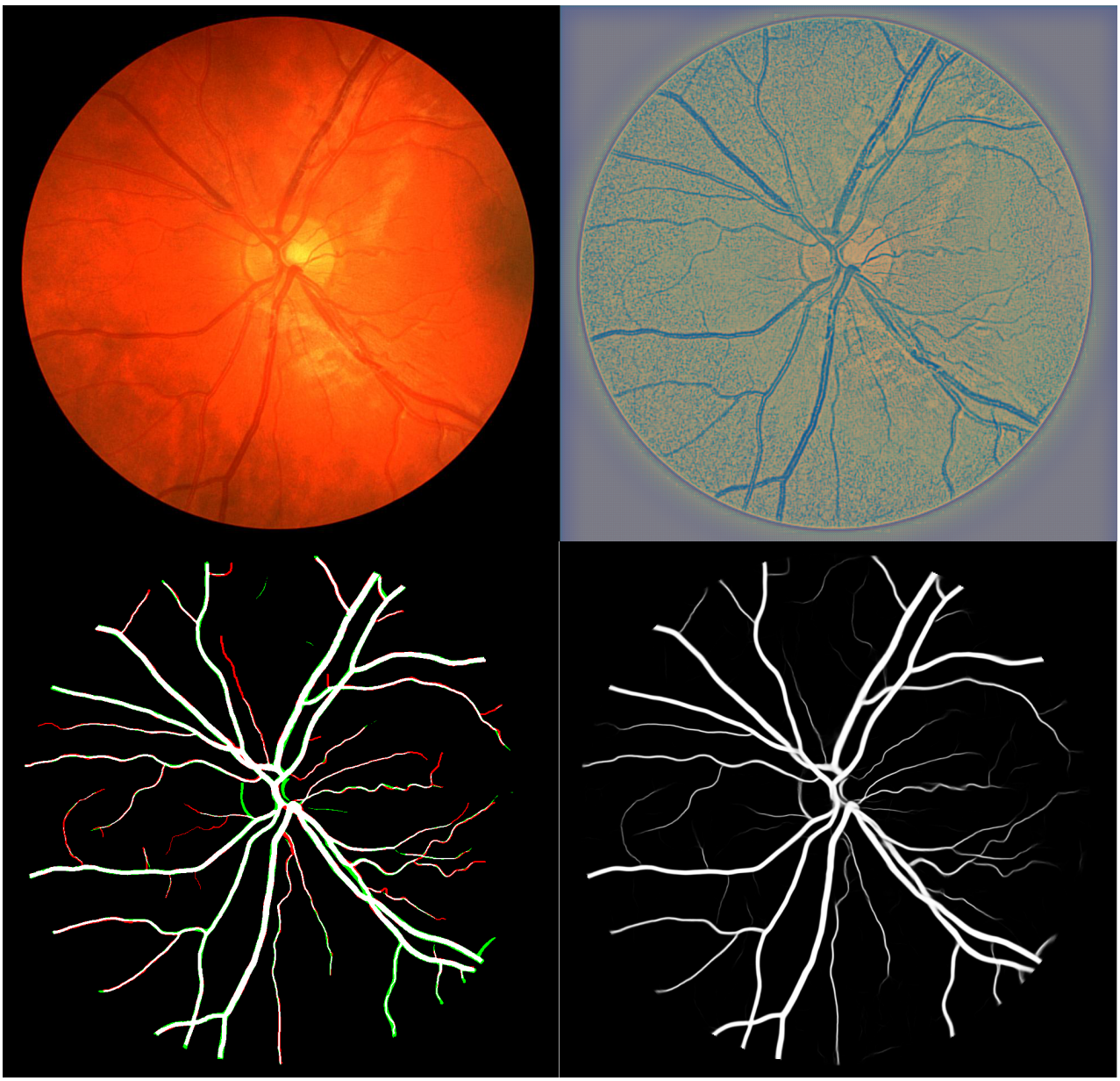

Figure 16 – Inference results on one test CHASE example. Red pixels on the left segmented is false negative and green is false positive.

3.7. Cross-dataset experiments and threshold optimisation

To prove the practicality of the model in real life, it should be tested on data other than training. Actual retinal photographs can vary greatly in terms of resolution of lighting conditions and image quality.The models trained on DRIVE and CHASE datasets were selected and generated predictions for different than training datasets.

The threshold optimization technique was applied for each model and dataset in an attempt to close the performance gap. From the data set without training, 5

images were selected, on which threshold optimization was performed, and then all other images were tested with the best optimized threshold.

The results are present in the table 9, threshold optimization technique increased performance in terms of DICE score for cross dataset experiments. This means that models with a calibrated threshold are better at vessel segmentation for new data.

This also means that the developed model is able to segment very different retinal blood vessels, but performance may suffer if threshold calibration has not been performed.

Table 9 – Cross dataset experiments DICE score results with and without threshold optimisation technique(TOT).

	DRIVE	CHASE	STARE
Trained dataset			
DRIVE		0.3553	0.3905
DRIVE with TOT		**0.5308**	**0.6857**
CHASE	0.7570		0.8144
CHASE with TOT	**0.7949**		**0.8191**

From the Figure 17 we can see that the number of false negative segmented pixel is much lower compare to model with the original threshold and almost all of the false positive pixels here is out liners of existing vessels therefore this CHASE based vessel segmentation model showed good cross dataset performance.

From the Figure 18 we can see that the model trained on the DRIVE dataset has overfit on the trained data. Even with threshold calibration the model couldn't segment vessels near the nerve, the number of false negative segmented pixels is much lower compared to the model with the original threshold, but also there are a lot of non segmented vessels and a lot of false positive vessels that can confuse practitioners. These results can be interpreted as models trained on the DRIVE dataset have bad adaptability and only show good results on the high quality retina images with good lighting. Models trained in the CHASE dataset showed much better adaptability even on vastly different DRIVE dataset after threshold optimisation.

The STARE dataset is much closer to the CHASE dataset in terms of light quality and picture resolution compared to DRIVE dataset. Therefore threshold op-

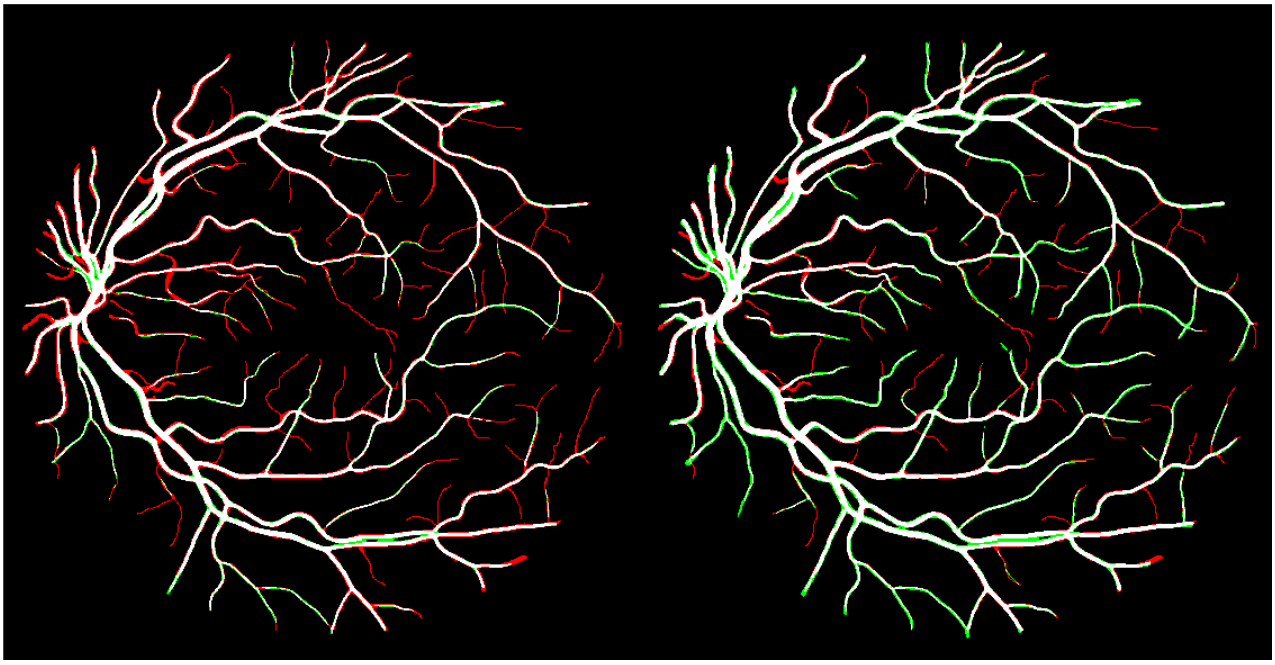

Figure 17 – Model trained on CHASE dataset. Inference results on one test DRIVE example with threshold optimisation on the left and without on the right. Red pixels is false negative and green is false positive.

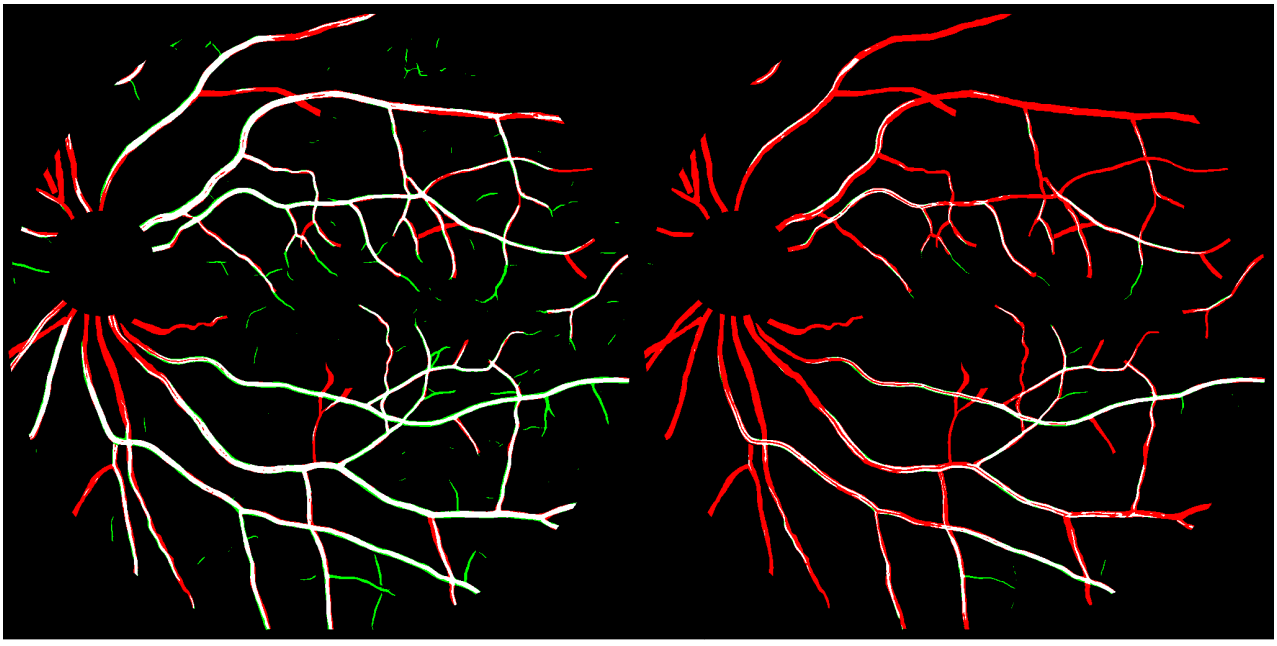

Figure 18 – Model trained on DRIVE dataset. Inference results on one test STARE example with threshold optimization on the right and without on the left. Red pixels is false negative and green is false positive.

timization technique showed not a big difference in performance in terms of DICE coefficient compared to the original model. But when the model was trained on the DRIVE dataset threshold optimization technique showed big improvements in

terms of DICE coefficient with AUC performance nearly identical compared to the original model.

3.8. Artery/Vein segmentation

Artery/Vein segmentation task is to classify every pixel in the entire image of an artery, vein, or background. This is a different task compared to A/V classification task, where a vessel tree is available and then the classification should be performed for each vessel pixel of the tree among two classes. To account for the greater complexity of the task, the number of training cycles was doubled, and training was done with 4 classes considering invalid pixels, as it proved to be useful for this task [Hem+19]. The results are presented for the artery/vein segmentation problem.

Table 10 – Performance comparison for the artery/vein segmentation task. Performance is reported on the entire image domain.

Method and Year	DICE	Accuracy	AUC
Baseline model	0.9671	0.9671	0.9064
My model	**0.9678**	**0.9681**	**0.9068**

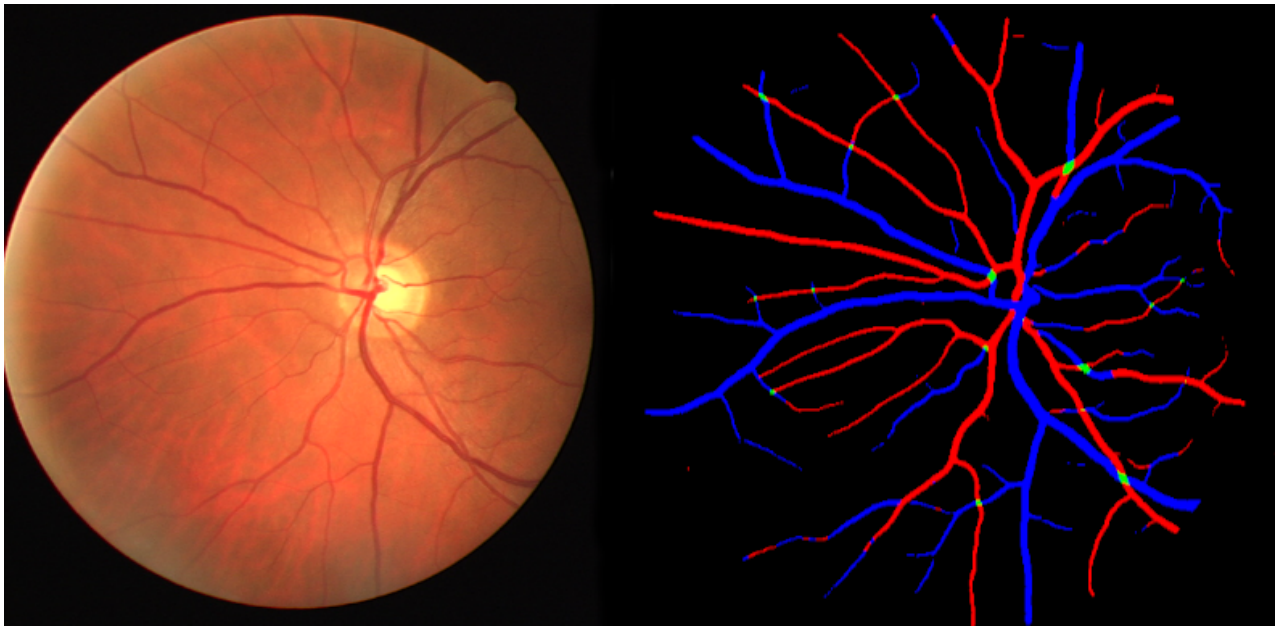

Figure 19 – Inference results of Artery/Vein segmentation on one test DRIVE example. Arteries are colored red while veins are colored blue and uncertain pixels are colored green.

Table 10 shows the results of the developed model, compared with baseline SGL [ZYS21] model that was modified for artery/vein segmentation task. Model

was trained on the DRIVE dataset and showed improvements in all metrics. That means that modifications introduced to the developed model are also increasing performance in the artery vein classification task. Some qualitative results of the model trained on DRIVE are shown in Figure 19.

CONCLUSION

Summarizing the results of this work, the main stages were listed of the study and the results obtained. Based on the results of the literature review, in order to achieve the goal, the task was set of developing and implementing a model for the retinal vessel segmentation based on autoencoder neural networks because those solutions is relatively fast to train and segment blood vessels using trained model along with with competitive segmentation performance compare to other segmentation methods.

The analysis of the DRIVE, CHASE-DB and STARE datasets revealed serious inconsistency in data: CHASE-DB and STARE datasets has high resolution with background illumination and poor contrast images and DRIVE is opposite: consistent good quality and contrast, but low resolution.

A set of methods was introduced for improving the model performance such as stacked UNet method, k-fold cross validation technique, augmentation by rotation of retina image and using Focal loss function with the highest performance, which improved the results of previous researchers on tested DRIVE and CHASE datasets. DICE on the DRIVE dataset score showed the most significant performance advantage: 0.8339 ± 0.0009 compared to the best model with 0.8316. All other metrics namely sensitivity, specificity, accuracy and area under the error curve showed similar performance compared to the state-of-the-art SGL model along with the standard errors.

With threshold optimization introduced, the developed model showed comparable results to well researched methods on cross datasets experiments. Model trained on the DRIVE dataset with threshold optimization technique has DICE score 0.7949 and 0.9774 AUC, meanwhile R2U-Net has lower 0.7928 DICE and 0.9784 AUC.

A new method of stacking autoencoders was successfully implemented and tested. As well as threshold optimisation technique that greatly improved cross dataset performance results on both variants of datasets.

Artery/Vein segmentation problem was also mentioned in this thesis. The best segmentation model was modified for the artery/vein segmentation task. The model was trained on the DRIVE dataset with 4 classes segmentation considering invalid pixels, as it proved to be useful for this task. The model showed improvements

of 0.9671 DICE, 0.9681 Accuracy and 0.9068 AUC metrics compared to baseline model 0.9671 DICE, 0.9671 Accuracy and 0.9064 AUC that was also modified for artery/vein segmentation task. That means that modifications introduced to the developed model are also increasing performance in the artery vein classification task. These are promising results however require more thorough research.

For further research, it would be worth exploring experiments with datasets and the performance of the model in real-life conditions in more detail as well as investigating how improvement segmentation performance from the developed model can improve classification.

REFERENCES

[AEE18] Jasem Almotiri, Khaled Elleithy, and Abdelrahman Elleithy. — "Retinal Vessels Segmentation Techniques and Algorithms: A Survey." — In: *Applied Sciences* vol. 8, no. 2 (2018), p. 155.

[Alo+18] Alom et al. — "Nuclei Segmentation with Recurrent Residual Convolutional Neural Networks based U-Net (R2U-Net)." — In: *NAECON 2018 - IEEE National Aerospace and Electronics Conference* (2018), p. 228–233.

[Alo+19] Md. Zahangir Alom et al. — "Recurrent residual u-net for medical image segmentation." — In: *Journal of Medical Imaging* 6(1) (2019).

[CG+09] Owen CG et al. — "Measuring Retinal Vessel Tortuosity in 10-Year-Old Children: Validation of the Computer-Assisted Image Analysis of the Retina (CAIAR) Program." — In: *Investigative Opthalmology* vol. 50, no. 5 (2009), p. 2004.

[Cha+20] Guo Changlu et al. — "Channel Attention Residual U-Net for Retinal Vessel Segmentation." — In: *ArXiv* abs/2004.03702 (2020).

[Est+15] R. Estrada et al. — "Retinal Artery-Vein Classification via Topology Estimation." — In: *IEEE Transactions on Medical Imaging* (2015), p. 2518–2534.

[FS99] Yoav Freund and Robert Schapire. — "Large margin classification using the perceptron algorithm." — In: *Machine Learning* 3 (1999), p. 277–296.

[Gal+19] Adrian Galdran et al. — "Uncertainty-Aware Artery/Vein Classification on Retinal Images." — In: *International Symposium on Biomedical Imaging* 16th (2019), p. 556–560.

[Guo+20] Changlu Guo et al. — "Sa-UNet: Spatial attention u-net for retinal vessel segmentation." — In: (2020), p. 1236–1242.

[Hem+19] Ruben Hemelings et al. — "Artery–vein segmentation in fundus images using a fully convolutional network." — In: *Computerized Medical Imaging and Graphics* (2019).

[HKG00] A.D. Hoover, Valentina Kouznetsova, and Michael Goldbaum. — "Locating Blood Vessels in Retinal Images by Piecewise Threshold Probing of a Matched Filter Response." — In: *IEEE Transactions on Medical Imaging* vol. 19, no. 3 (2000), p. 203–210.

[KM91] Kramer and A. Mark. — "Nonlinear principal component analysis using autoassociative neural networks." — In: *AIChE Journal* 37 (2) (1991), p. 233–243.

[Kol95] Nelson R Kolb H Fernandez E. — "Anatomy of the Retina. In Webvision: The Organization of the Retina and Visual System." — In: *University of Utah Health Sciences Center: Salt Lake City, UT, USA* (1995).

[Li+20a] Liangzhi Li et al. — "Iternet: Retinal image segmentation utilizing structural redundancy in vessel networks." — In: *Winter Conference on Applications of Computer Vision.* (2020), p. 3656–3665.

[Li+20b] Liangzhi Li et al. — "Joint Learning of Vessel Segmentation and Artery/Vein Classification with Post-processing." — In: *Proceedings of Machine Learning Research* (2020), p. 1–14.

[Orl+18] J. I. Orlando et al. — "Towards a Glaucoma Risk Index Based on Simulated Hemodynamics from Fundus Images." — In: *In Medical Image Computing and Computer Assisted Intervention – MICCAI* (2018), p. 65–73.

[Piz+87] S.M. Pizer et al. — "Adaptive histogram equalization and its variations." — In: *Computer vision, graphics, and image processing* 39(3) (1987), p. 355–368.

[RFB15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. — "U-Net: Convolutional Networks for Biomedical Image Segmentation." — In: *LNCS* (2015), p. 234–241.

[Sta+04] Joes Staal et al. — "Ridge-Based Vessel Segmentation in Color Images of the Retina." — In: *IEEE Transactions on Medical Imaging* vol. 23, no. 4 (2004), p. 501–509.

[Tag+20] Saed Taghanaki et al. — “Deep Semantic Segmentation of Natural and Medical Images: a Review.” — In: *Artificial Intelligence Review* (2020).

[Vin+10] Pascal Vincent et al. — “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion.” — In: *J. Mach. Learn. Res.* vol. 11 (2010), p. 3371–3408.

[WQH19] Bo Wang, Shuang Qiu, and Huiguang He. — “Dual encoding u-net for retinal vessel segmentation.” — In: *International Conference on Medical Image Computing and Computer-Assisted Intervention.* (2019), p. 84–92.

[WZY21] Chang Wang, Zongya Zhao, and Yu Yi. — “Fine retinal vessel segmentation by combining Nest U-net and patch-learning.” — In: *Soft Comput* 25 (2021), p. 1–14.

[Zha+15] Y. Zhao et al. — “Retinal vessel segmentation: An efficient graph cut approach with retinex and local phase.” — In: *PloS one* 10(4) (2015).

[Zha+20] Mo Zhang et al. — “Befd: Boundary enhancement and feature denoising for vessel segmentation.” — In: *In: International Conference on Medical Image Computing and Computer-Assisted Intervention.* (2020), p. 775–785.

[Zhu18] Juntang Zhuang. — “Laddernet: Multi-path networks based on u-net for medical image segmentation.” — In: *arXiv preprint arXiv* 1810 (2018).

[ZYS21] Yuqian Zhou, Hanchao Yu, and Humphrey Shi. — “Study Group Learning: Improving Retinal Vessel Segmentation Trained with Noisy Labels.” — In: *arXiv preprint arXiv:2103.03451* (2021).