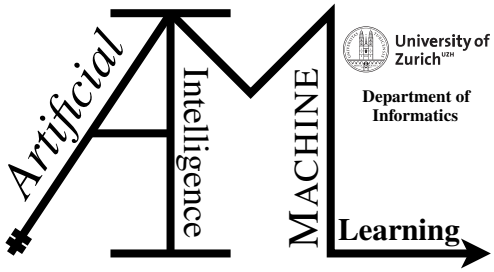
Semantic Segmentation of Weakly Labeled Retinal Images

Master Thesis

Xiao Tan
19-756-949

Submitted on
February 27 2023

Thesis Supervisor
Prof. Dr. Manuel Günther
Dr. André Anjos



Master Thesis

Author: Xiao Tan, xiao.tan@uzh.ch

Project period: 01.09.2022 - 01.03.2023

Artificial Intelligence and Machine Learning Group
Department of Informatics, University of Zurich

Acknowledgements

My deepest gratitude goes to my supervisor, Prof. Dr. Manuel Günther, for his guidance and support throughout the whole thesis. I would also like to thank Dr. André Anjos for giving me the opportunity to work on this topic and for his expertise. Last but not least, I would like to thank my beloved husband for his encouragement and faith in me.

Abstract

Semantic segmentation is an important task in computer vision. It performs pixel-level labeling with a set of object categories (e.g., human, car, tree, sky) for all image pixels; thus, it is generally a more demanding undertaking than whole-image classification, which predicts a single label for the entire image. Since Machine Learning is proposed, numerous supervised models have achieved very good performance in semantic segmentation tasks with reasonable computation costs. However, the performance of the supervised model is limited by the quality and amount of the labeled datasets, which are scarce and expensive to obtain. This work adapts a popular semi-supervised learning method, namely consistency learning, to the retinal vessel segmentation task. The main idea of this method is to minimize the differences between two predictions generated from two variants, which are produced by applying data augmentations to the same input, meanwhile, to maximize the agreement between the prediction and the ground truth. Because the distribution of pixels belonging to the vessels is sparse, limited data augmentations can be applied to the samples to produce the variants in this task. We figure out the basic data augmentations providing the best performance and test the model on four publicly available datasets. Our results suggest that our model can significantly improve the prediction performance on the labeled/unlabeled dataset pairs which have poor generalization ability in the supervised learning methods. For an unseen dataset, it is important to choose the labeled dataset used in training carefully. When the model is trained with a properly chosen labeled dataset, increasing the number of unlabeled datasets can improve its performance.

Contents

Chapter 1

Introduction

Semantic image segmentation is a key task in machine learning which aims to label each pixel of a target image with a corresponding class of what is being shown. For example, when Figure 1.1(a) is the task input, Figure 1.1(b) is expected as an output. It is a fundamental step in many computer vision tasks and is widely used in autonomous vehicles, robotic navigation, medical imaging, and diagnostics. In the early stage of machine learning, models were built to extract features from the input images. Then the feature vectors were sent into a classifier and according to the output of the classifier, every pixel was assigned a label. (Soares et al., 2006), (Marín et al., 2011) and other researchers proposed different methods to extract features, but the classifiers were similar.

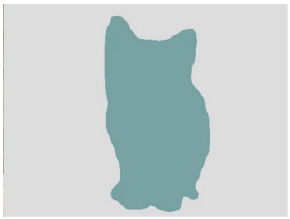
Later the Convolutional Neural Networks (CNN) took the lead and began to get first place by making big performance improvements. Many CNN models have been proposed to accomplish the task at a low cost such as U-NET (Ronneberger et al., 2015), DRIU (Maninis et al., 2016), and M2U-NET (Sandler et al., 2018). Usually, these models have a pre-trained network as their backbone, and then finetune the weights by the input datasets. The popular backbones such as AlexNet (Krizhevsky et al., 2012), VGG-16 (Simonyan and Zisserman, 2015), GoogLeNet (Szegedy et al., 2015), and ResNet (He et al., 2016) are trained on very large size image datasets and already become a well-known standard. However, these supervised methods suffer from limited annotated datasets because they require enough data to finetune the weights. Another limitation is that they lack the generalization ability to the other datasets.

In order to exploit the unlabeled data and improve the segmentation results on unlabeled data, semi-supervised learning methods were proposed to learn features from both labeled data and unlabeled data. Consistency learning (Tarvainen and Valpola, 2017) made promising improvements with a relatively small dataset for the classification task. Its main idea is to force the network to give consistent predictions to different variants of the same input. The variants are produced by applying data augmentation techniques to the input images. The prediction can be produced by any network that has proven feasible for the segmentation task. The network chosen is called the initialization network in our project. Based on that, (Perone and Cohen-Adad, 2018) adopted a consistency learning model named the Mean Teacher model to segment the Magnetic Resonance Imaging (MRI) pictures and achieved significant improvement. Our project is based on their work and we extend their model to the retinal vessel segmentation task.

The retinal images are captured by a fundus camera and the retinal vessels are the primary anatomical structures that are visible in retinal pictures. They are unique for each person and normally unchanged unless lesions occur. Segmentation of retinal vessels is a fundamental step in the further diagnosis of eye-related diseases. However, due to the complexity of the structure, it is difficult and time-consuming for experts to segment vessels manually, which means annotated datasets are scarce for retinal images. Therefore, it becomes an important task to segment vessels automatically and precisely on weakly labeled and unlabeled retinal images.

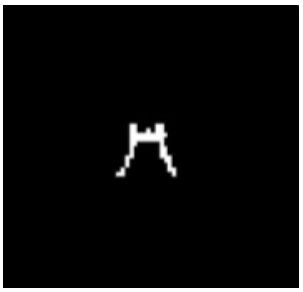
Although both retinal images and MRI are medical images, the tasks to segment them are very

(a) Normal segmentation input



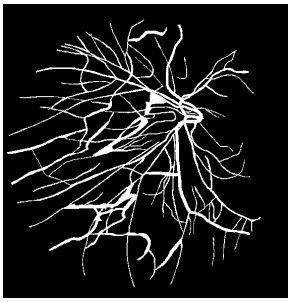
(b) Normal segmentation label

(c) MRI segmentation input



(d) MRI segmentation label

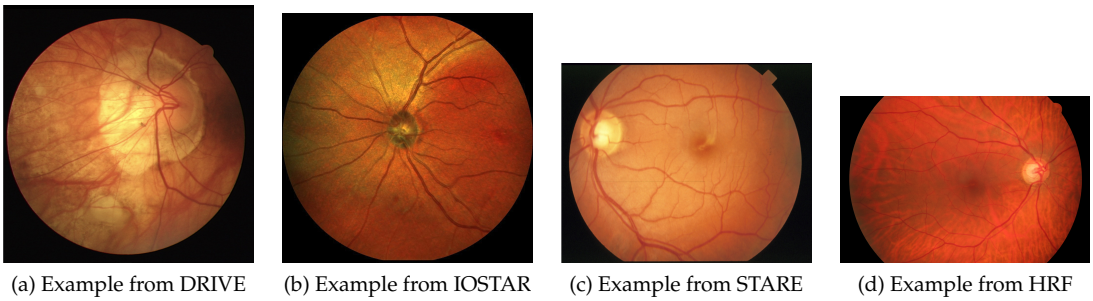
(e) Retinal vessel segmentation input



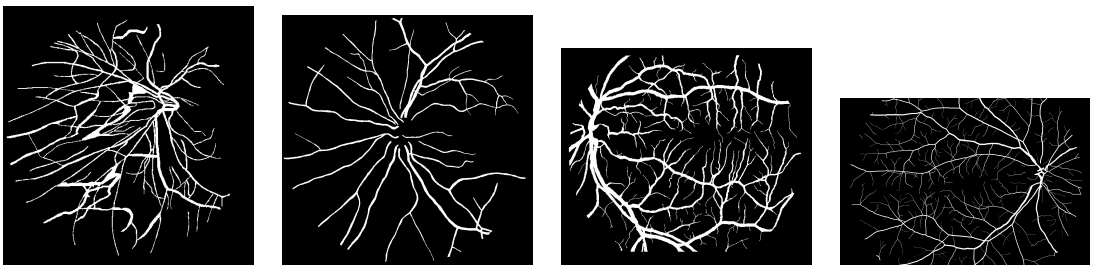
(f) Retinal vessel segmentation label

Figure 1.1: TASK COMPARISON. In every row, the left image is an input of a segmentation task and the right image is the corresponding label. (a) is an example of typical semantic segmentation task input, (c) is an input in (Perone and Cohen-Adad, 2018), and (e) is an input in our task. The distribution of true-labeled pixels is very different. In both (b) and (d), the true-labeled pixels locate in one region, while in (f), the true-labeled pixels cover almost the whole image.

different. The MRI images are grayscale (see Figure 1.1(c)) and their true-labeled pixels locate in a small area (as shown in 1.1(d)). But the inputs of the retinal task are in RGB (1.1(e)) and the true-labeled pixels are distributed all over the image (1.1(f)). In (Perone and Cohen-Adad, 2018), one dataset is split into the labeled dataset and the unlabeled dataset, so the generalization ability of their model was not scientifically tested. In our task, four different datasets are used to test our model. The examples of every dataset and their annotations are shown in Figure 1.2.



(a) Example from DRIVE (b) Example from IOSTAR (c) Example from STARE (d) Example from HRF



(e) Label of the example from DRIVE (f) Label of the example from IOSTAR (g) Label of the example from STARE (h) Label of the example from HRF

Figure 1.2: EXAMPLES FROM DATASETS USED IN THIS WORK. This figure shows examples from the four datasets and their corresponding annotations. It can be seen that they are different in quality and shape, and the annotations look dissimilar too.

This work extends the Mean Teacher model to the retinal vessel segmentation task and explores efficacious data augmentations to retinal images. The research questions this work focuses on are:

RQ1: What kind of data augmentation combination in the Mean Teacher model produces the best performance for retinal vessel segmentation?

RQ2: Will the choice of initialization network influence the performance of the Mean Teacher model?

RQ3: Will increasing the number of unlabeled datasets influence the performance of the Mean Teacher model on an unseen dataset?

In the experiments, we find that the Mean Teacher model can achieve a significant improvement in the labeled/unlabeled dataset pairs that can not be generalized well by the supervised learning model. Grayscale and Gaussian noise are proven the best-performance techniques out of the feasible data augmentation combinations. The performance of the model is also influenced by the choice of initialization network, and U-NET performs better than DRIU in our experiments.

When predicting an unseen dataset, increasing the number of unlabeled datasets can not always improve the performance of the model. The choice of the labeled dataset used to train the model plays an important role in its performance. The poor-quality labeled dataset can also produce good results with proper unlabeled datasets.

In the following part, Chapter 2 introduces related techniques for solving semantic segmentation tasks. Chapter 3 describes the Mean Teacher model. The content of Chapter 4 is a detailed description of the model's configuration and the experiments' setting up. Chapter 5 gives the details of the experiments' results and in Chapter 6 the results are analyzed and further unsupervised methods are discussed. Chapter 7 describes possible future work and concludes the work.

Chapter 2

Related Work

Semantic segmentation is a fundamental and rather challenging task in computer vision. Segmentation in medical images is even more challenging because of the complex structures and lack of labeled data. In this chapter, the related works in the field of semantic segmentation are introduced, including the general supervised methods and some important methods in semi-supervised and unsupervised learning.

2.1 Supervised Learning

In the very beginning, semantic segmentation was unsupervised and image preprocessing was usually applied first in those methods. They assumed that the objects to be segmented within a scene can be characterized by some similar features such as gray level intensity, color, or texture. So the algorithms tried to extract targets from the background according to these features. (Beucher et al., 1990) segmented road by the watershed method which adjusted the contrast of the image. The method was applied to the medical field even earlier. (Berns and Berns, 1982) used the combination between median filter, local Histogram, morphology filter, and binary with the watershed method to track living cells. In a task of retinal vessel segmentation (Zana and Klein, 2001), the images were preprocessed to normalize the background and enhance the vessels by noise reduction. Then they used a thresholding method to segment the vessels. However, such techniques are limited to the generalization of pathological structures for example the cataract, different resolutions of the images, and positions of optic discs.

When machine learning came into the picture, the performance of semantic segmentation models was improved quickly. A bunch of supervised models came up. For example, (Soares et al., 2006) extracted feature vectors from the pixel's intensity and then put the vector into a classifier to produce a probability distribution. Another example neural network (Marín et al., 2011) accomplished the task by classifying each pixel after it was represented by a 7-D vector. The vector was composed of gray-level features and moment invariants-based features. Other works, such as the Gradient Boosting framework (Becker et al., 2013) and conditional random field model (Orlando et al., 2017a), proposed different models to extract features. However, they focused more on feature extraction than on classifiers, which limited their performance.

In recent years, Convolutional Neural Networks (CNN) made outstanding improvements in semantic segmentation. CNN made it possible to handle raw data without any manually engineered features. CNN is a specialized type of artificial neural network. It uses a mathematical operation called convolution in place of general matrix multiplication in at least one of the layers (Goodfellow et al., 2016) to process pixel data in image recognition and processing.

The standard CNN approach for semantic segmentation is supervised. Training a CNN model from scratch is often not feasible because of the large number of the required dataset and the long

time for the model to converge. So it is often helpful to start with a pre-trained network and then finetune the weights. Certain networks have made such significant contributions to the field that they have become widely known standards. It is the case of AlexNet (Krizhevsky et al., 2012), VGG-16 (Simonyan and Zisserman, 2015), GoogLeNet (Szegedy et al., 2015), and ResNet (He et al., 2016). These networks were pre-trained for classification tasks but can be finetuned to perform semantic segmentation. Based on that, the method Regions with CNN feature (R-CNN) was proposed. Its main idea is to extract free-form regions from an image and describe them, followed by region-based classification. During testing, the region-based predictions were transformed into pixel predictions by labeling a pixel according to the highest-scoring region that contains it (Caesar et al., 2016). Then Faster R-CNN (Ren et al., 2015) improved the performance by sharing the convolution feature of the proposal region network with the detection network. Mask R-CNN((He et al., 2017)) extended Faster RCNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition.

Besides these region-based CNN methods, Fully Convolutional Network (FCN) was proposed by (Long et al., 2015). It was derived from CNN and recovered the spatial information from the downsampling layers by adding upsampling layers to standard CNN. Thus FCN can learn a mapping from pixels to pixels, without extracting the region proposals. However, the result of upsampling was blurred and smoothed so that the model was not sensitive enough to the details of the images. On the other hand, the classification of each pixel does not consider the relationship between pixels, leading to information loss in the training.

U-NET (Dong et al., 2017) extended the FCN architecture for biological microscopy images. It consisted of two parts, a downsampling part, and an upsampling part. The features got from the downsampling network were added to the upsampling part to improve the model's accuracy. In addition to U-NET, many other CNNs applied in medical image segmentation can already reach a very good performance with reasonable runtime and computation power, for example, HED (Xie and Tu, 2015), DRIU (Maninis et al., 2016), M2U-NET (Sandler et al., 2018), DUNet (Jin et al., 2019), and Lw-Net (Galdran et al., 2020a).

Data augmentation has proven effective in reducing overfitting and improving test performance for both natural and medical images (Zhang et al., 2017a). (Tajbakhsh et al., 2020) focused on medical images and summarized the traditional data augmentation techniques into three categories by the image property they try to manipulate: image quality, image appearance, and image layout. Image quality can be affected by sharpness, blurriness, and noise. Image appearance can be adjusted by image intensities meaning brightness, saturation, or contrast. Image layout can be changed by rotation, scaling, and deformation. For example, (Christ et al., 2016) applied Gaussian noise to CT scans. (Dong et al., 2017) employed random enhancement of brightness in 3D MR volumes to enrich the training set for brain tumor segmentation. The result of (Ronneberger et al., 2015) showed applying random elastic deformations to the training set plays a key role in the training when there are very few annotated images.

However, the supervised learning approaches require a large amount of labeled data, which is expensive and not always available, especially for medical images. Although data augmentation can help to partially alleviate the problem, these techniques are still not sufficient for weakly labeled data and even unlabeled data. Therefore, semi-supervised and unsupervised learning approaches attract more and more attention in the research of medical image segmentation.

2.2 Unsupervised Learning

The data augmentation techniques provide more approaches for researchers to explore unlabeled datasets. Contrastive learning is one of the most popular methods of unsupervised learning. The intuition of this technique is that different transformations of one image should have similar rep-

resentations and that these representations should be dissimilar from those of a different image. In this case, the loss function should maximize the agreement between instance embeddings from different augmentations of the same images. (Chen et al., 2020) implemented the idea and found the choice of data augmentation techniques and batch size were crucial to the performance. Big batch size and strong data augmentation improved the performance significantly but need more memory. (He et al., 2020) proposed a momentum contrastive learning method to solve this problem. The authors built a large and consistent dictionary on-the-fly that facilitated contrastive unsupervised learning. The dictionary was a memory bank of sample representations and was momentum updated so that it became more effective to deal with large-scale datasets.

However, the methods above focus on the image-level contrastive loss of the representations, which is more suitable for image classification. In a segmentation or object detection task, pixel-level prediction is desired. Therefore, the unsupervised model is not implemented in this work and a detailed explanation is discussed in Chapter 6.2.

2.3 Semi-supervised Learning

Nowadays, the dominant Semi-Supervised Learning (SSL) methods mainly include Pseudo labeling, adversarial training, and consistency training.

Pseudo labeling is to use the model trained by labeled data to predict the labels of unlabeled data and then to retrain the model with both labeled data and pseudo-labeled data (Lee et al., 2013). Recent techniques try to annotate the unlabeled data by not only the model trained by labeled data but also the ensemble segmentation model. Pseudo labeling suffers from the limitation of exploiting the unlabeled data to extract additional training features. If there is any bias when the model is trained with labeled data, the bias will be transferred to the unlabeled data. This may lead to worse performance.

Adversarial training takes another way to explore unlabeled data. In (Zhang et al., 2017b), a deep adversarial network (DAN) was applied to address a gland segmentation task. The discriminator was trained to distinguish labeled data and unlabeled data, and an evaluation network was trained to improve the performance in the segmentation task. More complex models, Generative Adversarial Networks(GAN) (Goodfellow et al., 2020) were proposed by extending the generic GAN frameworks to pixel-level predictions. In the GAN framework, the generator is trained to fool the discriminator by generating images that are similar to the real ones. The discriminator is trained to distinguish real images from fake ones (generated by the generator). For example, in (Zhao et al., 2018), the discriminator was trained to classify synthetic images which were made from a combination of unlabeled data and labeled ones by the generator. (Hung et al., 2018) proposed a novel GAN model with a generator to produce probability maps and a discriminator to distinguish the maps from the ground truth. The limitation of the GAN model is that it needs a large amount of high-quality annotated datasets to train a GAN model successfully. All three GANs mentioned above were trained on datasets with several thousand images. (Lahiri et al., 2017) applied GAN to retinal vessel segmentation with only one dataset. However, their result can not clearly show its performance when generalizing the model to the other dataset.

With the development of data augmentation techniques, applying them to learn the features of unlabeled data improve models' performance in the semantic segmentation task. Consistency learning also applies data augmentation to unlabeled datasets but in different directions from contrastive learning.

The main idea of consistency learning is to force the consistency of the predictions from the same network after the input image is perturbed slightly differently. The network produces two variants for the same input, and then produces two predictions of the variants. The loss for the network is a combination of consistency loss and supervised loss. The consistency loss is

computed on all the inputs, measuring the similarity of the two predictions. The supervised loss is computed only on the labeled data, measuring the similarity of the prediction of the model and the ground truth. By maximizing the loss, the network can learn from both the labeled data and the unlabeled data. In (Laine and Aila, 2017), the authors described two ways, Π-model and Temporal ensembling to solve image classification tasks. All inputs were augmented by different parameters and dropouts and then sent into the models twice. For the labeled data, the two predictions contributed to the consistency loss, and one of them was used to calculate the supervised loss together with the ground truth. The unlabeled data only contributed to the consistency loss. The whole network was evaluated by a loss which is a combination of the supervised loss and the weighted consistency loss. Based on their work, (Tarvainen and Valpola, 2017) built the Mean Teacher model and split one model into two networks to produce the two predictions, the teacher model and student model, respectively, to speed up the convergence of the loss function. The two networks were initialized the same at the beginning, and then the student model was evaluated by the combined loss and the teacher model was updated by the exponential moving average (EMA) weights of the student model. (Tarvainen and Valpola, 2017) applied their model in an image classification task. Later on, the Mean Teacher model was applied to the MRI segmentation task by (Cui et al., 2019) and (Perone and Cohen-Adad, 2018). The authors used different networks to initialize their models and different data augmentation techniques to perturb their datasets. Their results showed that the Mean Teacher model could achieve a better performance in generalization than the supervised model. However, the datasets they used were grayscale and the distribution of true-labeled pixels was relatively dense. Another problem in their work is that they split labeled data and unlabeled data from one dataset, so the generalization ability of the model was not tested. Whereas this work builds an adaptive Mean Teacher model for the retinal vessel segmentation task and tests the generalization ability of the model on four different datasets.

Chapter 3

Semi-supervised Learning

Lack of annotated data is a normal problem in deep learning, since high-quality labels require more experience and time, especially for images in the medical field. Semi-supervised learning is a promising solution to this problem. In this chapter, one of the most popular semi-supervised learning methods, consistency learning, is described in detail.

3.1 Π -model and Temporal ensembling

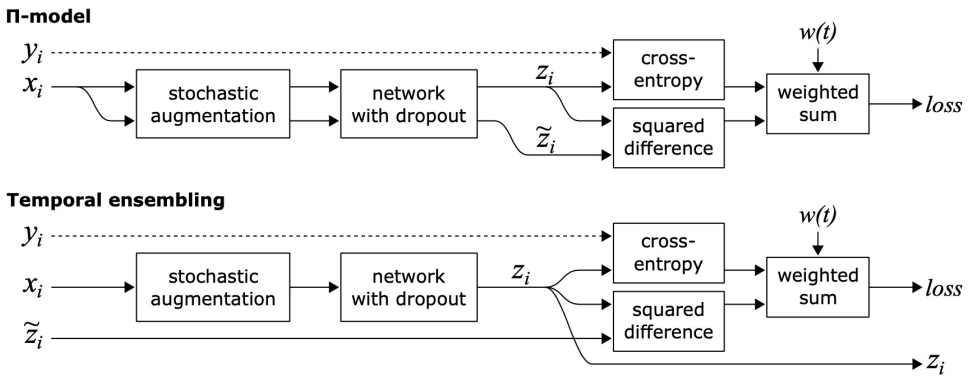


Figure 3.1: Π -MODEL AND TEMPORAL ENSEMBLING. (Laine and Aila, 2017)
In the Π -model, one sample x_i is sent into the network twice and two predictions z_i and $\tilde{z}_i$ are produced. y_i is the ground truth. In the Temporal ensembling, the network is updated once for each sample and produces only one prediction z_i. $\tilde{z}_i$ is updated by the EMA of z_i. Both networks are updated by the combined loss, which is the weighted sum of the consistency loss and the supervised loss (Equation (3.1)).

In the last decade, the idea becomes straightforward that the same network should give a consistent prediction even if the input is slightly perturbed. For an input x_i, two variants are produced after x_i is slightly perturbed. The network should be robust enough so that when the variants are sent into the network, the predictions z_i and $\tilde{z}_i$ produced are consistent. This works for both the labeled dataset and the unlabeled dataset. In this way, a network can learn features from both datasets at the same time. The idea was implemented by (Laine and Aila, 2017) for the

image classification task. They proposed two models and the Π -model's illustration is shown in the upper part of Figure 3.1. The predictions (z_i and $\tilde{z}_i$) of two different augmentations on one input x_i are forced to be consistent with each other. Then the entire network is evaluated by a combined loss of weighted consistency loss and supervised loss (Equation (3.1)):

$$\mathcal{L}_t = \mathcal{L}_{sup} + w(t) \cdot \mathcal{L}_{cons} \tag{3.1}$$

where $w(t)$ is a weighting function and t is the current epoch. The consistency weight function $w(t)$ starts from 0 and increases slowly to the maximum in the first T steps of training. The reason to set such a function is that z_i and $\tilde{z}_i$ may differ very much since the model has not learned from the dataset in the beginning. It is found very important that $w(t)$ is slow enough — otherwise, the network gets easily stuck in a degenerate solution where no meaningful classification of the data is obtained in (Laine and Aila, 2017)'s experiment. The maximum value for the consistency loss component is set to $\frac{w_{max} \cdot L}{M}$, where L is the number of labeled samples and M is the total number of training samples. $w(t)$ is defined as:

$$w(t) = \frac{w_{max} \cdot L}{M} \cdot e^{-5(1-\frac{t}{T})^2} \tag{3.2}$$

The supervised loss $\mathcal{L}_{sup}$ in their classification task is a cross-entropy loss, evaluated for labeled inputs only. The consistency loss $\mathcal{L}_{cons}$ is computed by taking the mean square difference between z_i and $\tilde{z}_i$ for all inputs. Every sample is sent into the network twice to produce two predictions so the network was updated twice per sample per epoch, which is very time-consuming.

In order to speed up the convergence of the network, (Laine and Aila, 2017) then improved the effectiveness of the Π -model by introducing the Temporal embeddings as shown at the bottom of Figure 3.1. Before the semi-supervised training, every input x_i is data augmented and sent into the network to produce a prediction $\tilde{z}_i$. During the semi-supervised training, every sample is perturbed slightly differently and sent into the network to produce a prediction z_i. Then $\tilde{z}_i$ will be updated by an exponential moving average (EMA) of z_i, which is defined in Equation (3.3):

$$\tilde{z}_{i(t)} = (\beta \tilde{z}_{i(t-1)} + (1-\beta) z_{i(t)}) / (1-\beta^t) \tag{3.3}$$

where β is a smoothing coefficient hyperparameter. The network is updated by the combined loss same as the Π -model, only once per input per epoch. In this way, the model speeds up to 2 times compared to the Π -model.

3.2 Mean Teacher Model

Because $\tilde{z}_i$ in the Temporal ensembling changes only once per epoch, the model becomes unwieldy when learning large datasets. In order to properly speed up the convergence of the network, the Mean Teacher model was proposed by (Tarvainen and Valpola, 2017). It consists of two networks with the same structure and the same parameters at the beginning. Same-category data augmentations with different parameters are applied to one input to produce two variants. They are sent into the two networks so z_i and $\tilde{z}_i$ are produced at the same time. $\tilde{z}_i$ is the prediction of the teacher model, which is maintained synchronously by the EMA parameters of the other network, the student model. z_i is the prediction of the student model. Let θ_t be the parameters of the student model and θ'_t be the parameters of the teacher model at epoch t, then θ_t is defined as:

$$\theta'_t = \beta \theta'_{t-1} + (1-\beta)\theta_t \tag{3.4}$$

β is set to 0.99 in the first 50 epochs and 0.999 thereafter. This is to speed up the convergence of z_i and $\tilde{z}_i$ at the beginning of training when they are far away from each other. Later when the two

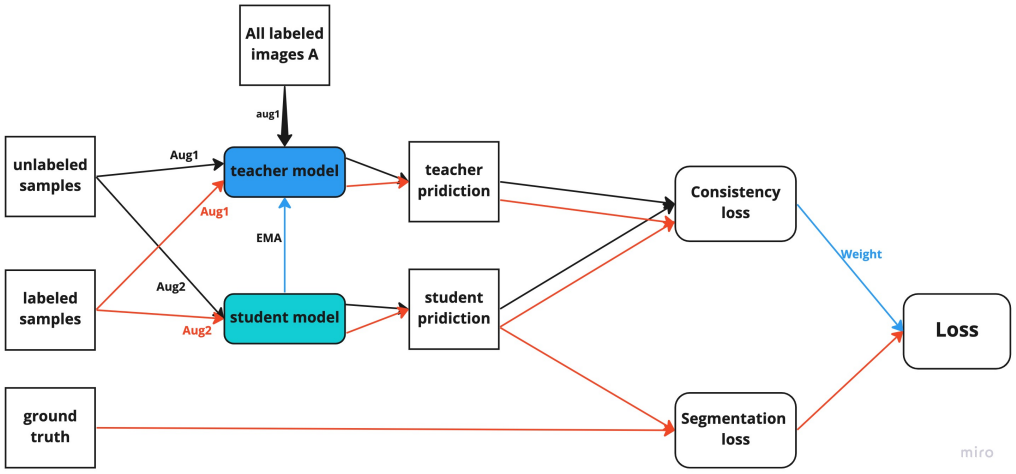


Figure 3.2: MEAN TEACHER MODEL ILLUSTRATION. Red lines represent the pipeline of the labeled data, and black lines represent the pipeline of the unlabeled data. The blue line is the consistency loss, and all inputs contribute to it. Since this is a segmentation task, the supervised loss is called segmentation loss here to easier distinguish our work from the prior works. Aug1 stands for the data augmentation techniques applied to the input sent into the teacher model and Aug2 stands for the data augmentations applied in the student model. They are the same techniques but with different parameters. The network used to initialize the teacher model and the student model is pre-trained by labeled data with data augmentation Aug1. The teacher model is updated by the EMA of the student model's parameters. The student model is updated by the combined loss. In the end, the teacher model is used to produce the final prediction.

models are already close enough to each other, it slows down the change of the teacher model to prevent big fluctuation. In this way, the effectiveness of the model is improved according to their experiments. The student model is updated normally by the combined loss shown in Equation (3.1). The Mean Teacher model in (Tarvainen and Valpola, 2017) was applied to an image classification task as well as in (Laine and Aila, 2017). Until 2018, (Perone and Cohen-Adad, 2018) applied this model to complete the image segmentation task successfully. However, the dataset in (Perone and Cohen-Adad, 2018) is a dataset from the MRI domain, which is different from our task (shown in Figure 1.1). Moreover, the unlabeled dataset and the labeled dataset in their experiments are split from the same dataset, so the model's generalization ability is not tested. In order to solve these problems, the Mean Teacher model is modified for our retinal vessel segmentation and described detailedly in the next section.

3.3 Adapted Mean Teacher Model

Based on the work of (Tarvainen and Valpola, 2017), an adapted Mean Teacher model is built for the retinal vessel segmentation task as shown in Figure 3.2. The main framework is the same as (Tarvainen and Valpola, 2017). The model consists of two same structured networks, namely the teacher model ($\mathcal{T}$) and the student model ($\mathcal{S}$). Let the network used to initialize the teacher model and student model be $\mathcal{N}$ and $\mathcal{N}$ can be any model that works for the task. To initialize the networks, $\mathcal{N}$ is trained first by the labeled dataset after data augmentation Aug1 and then both $\mathcal{S}$ and $\mathcal{T}$'s parameters are set as same as $\mathcal{N}$. During semi-supervised training, the teacher model keeps the same data augmentation Aug1, but the student model is fed with input after being applied data segmentation Aug2. Aug1 and Aug2 belong to the same data augmentation

technique, but adopt different parameters. The student model is evaluated by a combined loss and the teacher model's parameters are updated to the EMA of the student model's parameters which is the same as Equation (3.4).

In every epoch, the student model is updated by the combined loss $\mathcal{L}_t$:

$$\mathcal{L}_t = \mathcal{L}_{seg} + w(t) \cdot \mathcal{L}_{cons} \tag{3.5}$$

where $w(t)$ is the same as in (Perone and Cohen-Adad, 2018) and calculated by Equation (3.2). The ramp-up step T is set to 100 and w_{max} to 30 according to (Perone and Cohen-Adad, 2018). $\mathcal{L}_{seg}$ is the supervised loss and is defined as the same as the loss used in network $\mathcal{N}$'s training. The consistency loss is computed by the cross-entropy loss between the teacher model's prediction $\tilde{z}_i$ and the student model's prediction z_i. According to (Perone and Cohen-Adad, 2018), the Binary Cross-Entropy (BCE) loss achieves better performance than the mean squared error (MSE) loss. So $\mathcal{L}_{cons}$ can be formulated as:

$$\mathcal{L}_{cons} = -\frac{1}{B}\sum_{i=1}^{B} \tilde{z}_i \cdot \log(z_i) + (1 - \tilde{z}_i) \cdot \log(1 - z_i) \tag{3.6}$$

where B is the number of samples in the batch. When the training is finished, teacher model $\mathcal{T}$ is used to produce the final prediction.

Chapter 4

Experiment Setup

4.1 Mean Teacher Model Configuration

This section describes the adapted Mean Teacher model configuration in detail, including the initialization network choosing, metrics and training procedure.

4.1.1 Initialization Network $\mathcal{N}$

In the Mean Teacher model, the network $\mathcal{N}$ used to initialize the teacher model $\mathcal{T}$ and the student model $\mathcal{S}$ can be any network feasible for the task. In this project, the U-Net model (Ronneberger et al., 2015) is chosen as one of the initialization networks. The U-Net model is one of the most popular architectures in the field of medical image segmentation. It has a simple structure and can be easily implemented. The U-Net model is derived from FCN with skip connections. The structure of the U-Net model is shown in Figure 4.1. The U-Net model has two parts: encoder and decoder. The encoder is a modified VGG 16 (Simonyan and Zisserman, 2015). These encoder-decoder networks share a key similarity: skip connections, which combine deep, semantic, coarse-grained feature maps from the decoder sub-network with shallow, low-level, fine-grained feature maps from the encoder sub-network. The model has 23 layers in total, including 18 convolutional layers, 4 max-pooling layers, and 4 up-sampling layers. The convolutional layers are all followed by Rectified Linear Unit (ReLU) activation functions. This U-Net model has 25.9M parameters in total. The skip connections have proved effective in recovering fine-grained details of the target objects; generating segmentation masks with fine details even on complex backgrounds. With this structure, the U-Net model achieves good performance on small datasets.

Another model chosen to initialize $\mathcal{T}$ and $\mathcal{S}$ is the DRIU model (Maninis et al., 2016). The DRIU model is also a modified VGG 16 and the structure is shown in Figure 4.2. The DRIU model removes the last 3 fully connected layers of the VGG 16 network. The convolutional layers are all followed by ReLU activation functions. The 4 max-pooling layers in the architecture separate the network into five stages and a feature map is produced in every stage. The deeper the sample goes in the network, the coarser-grained the feature map is. Then task-specific "specialized" convolutional layers are connected to the final layer of each stage. Each specialized layer produces feature maps in K different channels, which are resized to the original image size and concatenated, creating a volume of fine-to-coarse feature maps. One last convolutional layer is appended which linearly combines the feature maps from the volume created by the specialized layers into a regressed result. The original network is built for two tasks, while in this project, only the left part for the vessel segmentation task is implemented. The DRIU model has 14.9M parameters in total. The DRIU model has fewer parameters than U-Net but similar performance. Both models have good performance in retinal vessel segmentation.

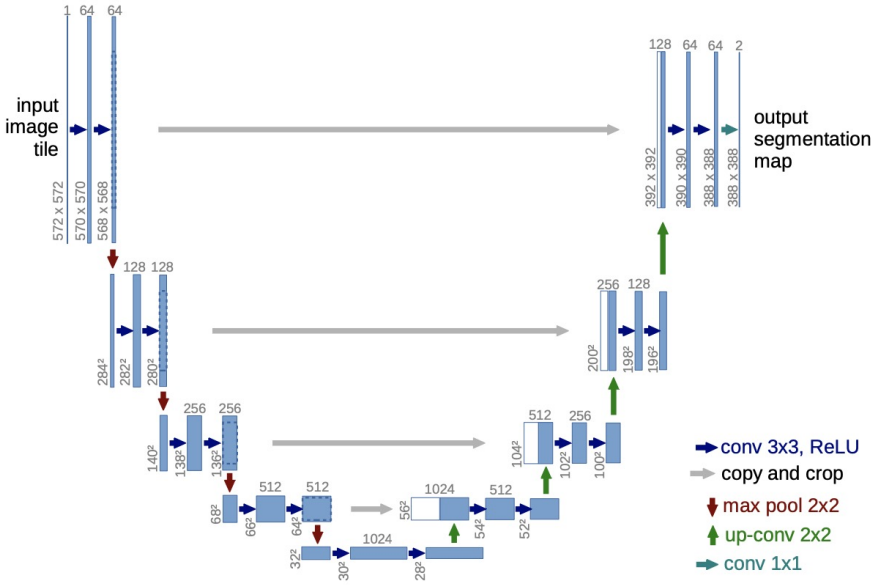


Figure 4.1: THE STRUCTURE OF THE U-NET MODEL. U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations (Ronneberger et al., 2015).

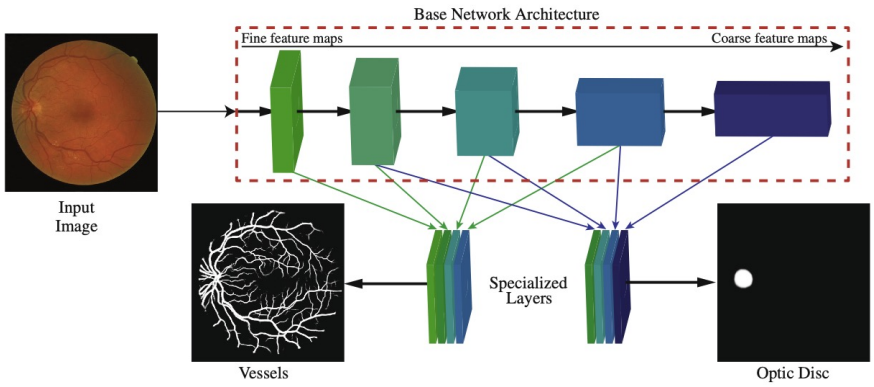


Figure 4.2: THE STRUCTURE OF THE DRIU MODEL. Feature maps from the Base Network (VGG 16) are extracted and sent into the specialized layers to perform blood vessel segmentation (left) and optic disc segmentation (right). In our implementation, only the left part is used (Maninis et al., 2016).

During training, a probability map is generated by the network after each epoch. The probability map is the same size as the original image. The pixel values in the probability map are the predicted probabilities of the pixels belonging to the vessel class. The ground truth labels are binary masks, where the pixels belonging to the vessel class are marked as 1 and the others are marked as 0. The probability together with the ground truth is then used to compute the loss function. The loss function used here is the Jaccard Loss $\mathcal{L}_{JBCE}$ (Iglovikov et al., 2018), a

combination of Binary Cross-Entropy loss $\mathcal{L}_{BCE}$ and the Jaccard Score $\mathcal{J}$ weighted by a factor α:

$$\mathcal{L}_{JBCE} = \alpha \cdot \mathcal{L}_{BCE} + (1 - \alpha) \cdot (1 - \mathcal{J}) \tag{4.1}$$

where $\alpha = 0.7$ is adopted as suggested by (Iglovikov et al., 2018) in this project. Let $p_{model}(y_i|x_i)$ be the value corresponding to the predicted probability of a pixel i belonging to the vessel class and y_i be the ground-truth binary value. The Binary Cross-Entropy loss $\mathcal{L}_{BCE}$ is defined as:

$$\mathcal{L}_{BCE} = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log(p_{model}(y_i|x_i)) + (1 - y_i) \cdot \log(1 - p_{model}(y_i|x_i)) \tag{4.2}$$

where N is the number of pixels in the image. Since the label of every pixel in the image is either 0 or 1, Binary Cross-Entropy loss makes sense here. In the latter part of Equation (4.1), $\mathcal{J}$ is the Jaccard Score. It is derived from the Jaccard Index, which is a measure of similarity between a finite number of sets. The index is also known as the intersection over union. It is defined as:

$$\mathcal{J}(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{4.3}$$

where A and B are two sets and $\mathcal{J}(A, B)$ measures the similarity between A and B. In order to have a differentiable loss function, Jaccard Index is generalized to probability prediction, which is defined as:

$$\mathcal{J} = \frac{1}{N}\sum_{i=1}^{N} \frac{y_i \cdot p_{model}(y_i|x_i)}{y_i + \hat{y}_i - y_i \cdot p_{model}(y_i|x_i)} \tag{4.4}$$

$\hat{y}_i \in \{0, 1\}$ is the predicted binary value of the pixel i. Jaccard Score turns the predictions into Jaccard and is differentiable so that can be optimized by gradient descent. The combination of equations (4.2) and (4.4) makes the final loss function (4.1).

4.1.2 Metrics and Evaluation

In this task, the data distribution is so imbalanced that the metric used here to evaluate the performances of three models is the F1 score (Dice coefficient), which is derived from precision and recall:

$$F1 = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{4.5}$$

where $precision = \frac{TP}{TP+FP}$ and $recall = \frac{TP}{TP+FN}$. TP is the number of pixels correctly predicted belonging to the vessel. FP is the number of pixels not belonging to the vessel but wrongly predicted to. FN is the number of pixels belonging to the vessel but wrongly predicted not. This metric makes a trade-off between precision and recall and gives a better view to evaluate the performance of a model in our task. The higher the F1 score is, the better the model performs.

After training, the parameters producing the lowest validation loss are used to produce the prediction of the test dataset. First, all probabilities and labels are accumulated across the training set, then an optimal threshold γ (maximizing the Dice score) is obtained by performing Area Under the ROC Curve (AUC) analysis. In the end, the pre-computed threshold γ is used to binarize test image segmentation. The F1 score is then computed for each test image and the average is reported as the final performance.

4.1.3 Training Procedure

The whole training includes two phases: pretraining the initialization network $\mathcal{N}$ with labeled data and semi-supervised training with both labeled data and unlabeled data. In the first phase,

because the numbers of samples in these datasets are relatively small, the validation set is not split out from the training set. Instead, a copy of the original train set is treated as the validation set. Then the training samples are augmented by the data augmentation Aug1 and sent into the chosen $\mathcal{N}$. The optimizer adopted is AdaBound (Luo et al., 2019). The learning rate is fixed to 0.001 as suggested by (Laibacher and Anjos, 2019). The learning rate is decreased by a factor of 0.1 after 500 epochs. The training is stopped after 600 epochs. The batch size of U-NET and DRIU models are 4 and 8, respectively, which is the maximum batch size that can be handled on our GPU. Since all batch samples can not be processed at the same time, gradient accumulation is applied. This method will divide one batch into some mini-batches with proper size in the capacity of the GPU and then accumulate the gradients of all mini-batches. The model parameters are updated only once after the whole batch is sent into the network. After training, the parameters leading to the lowest validation loss are used to initialize both the teacher model $\mathcal{T}$ and the student model $\mathcal{S}$.

In the semi-supervised training, one of four datasets is used as the labeled data and the others are unlabeled data. When one dataset is set to labeled data, the whole dataset is used as the labeled part of a training set. When it is chosen as unlabeled data, the original training set is used as the unlabeled part of the training set, and the original test set is seen as the whole model's test set. In every batch, at least one labeled sample is included. If the number of labeled samples is not enough for all batches, the labeled training set will be shuffled and used again. If the number of labeled samples that can be allocated to a batch is more than 1, the data loader tries to balance the number of labeled data and unlabeled data according to the ratio of their sizes. When the unlabeled samples are used up, the rest of the batch will be filled with the remaining labeled samples. The validation set is a copy of the labeled part of the training set. When samples are sent into the teacher model, Aug1 is applied. When samples are sent into the student model, Aug2 is applied.

The Mean Teacher model is trained with a learning rate of 0.0006 and the number of epochs is 350 as suggested in (Perone and Cohen-Adad, 2018). The learning rate is decreased by a factor of 0.1 after 300 epochs. The batch size and optimizer are set the same as in the initialization network training.

4.2 Experiment Setup

In order to answer the research questions proposed in Chapter 1, a series of experiments are set up to test the performance of the proposed model. In the following section, the datasets used in the experiments and a detailed description of the experimental setup are introduced. The task is

Dataset	Samples	Resolution	Train	Test
DRIVE	40	565×584	20	20
STARE	20	605×700	10	10
HRF	45	3504×2336	15	30
IOSTAR	30	1024×1024	20	10

Table 4.1: DATASETS USED IN THIS PROJECT.

to segment the blood vessels in the images. Examples of the original images and the ground truth are shown in Figure 1.2. The ground truth labels are binary masks, where the pixels belonging to the vessel class are marked as 1 and the others are marked as 0.

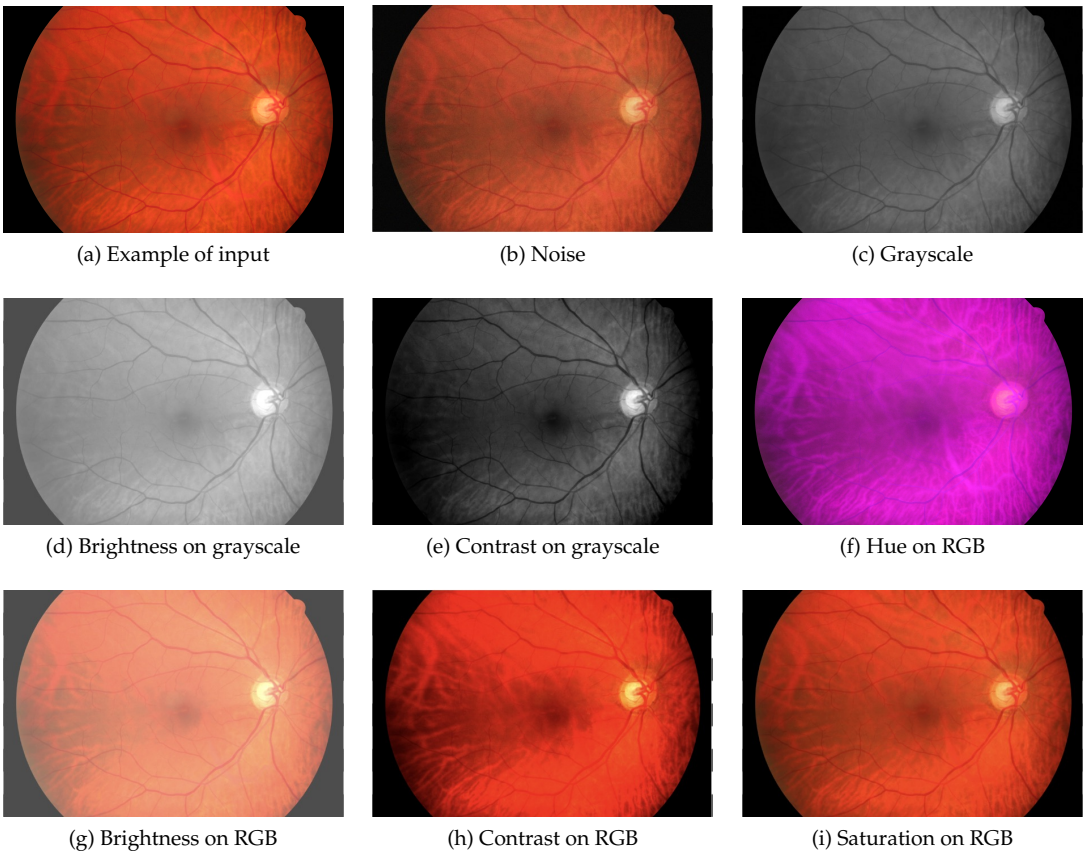
4.2.1 Dataset

The experiments are conducted on four well-labeled datasets: DRIVE, STARE, HRF, and IOSTAR. The original resolutions of these datasets are different, due to the capacity of our GPU, all images are padded and resized to 768×768 pixels. The train/test splits for DRIVE are provided by the authors (Staal et al., 2004), and for STARE the splits follow (Maninis et al., 2016). For HRF and IOSTAR, the splits are generated according to (Orlando et al., 2017b) and (Meyer et al., 2017), respectively. The details of the four datasets are shown in Table 4.1. (Galdran et al., 2020b) summarized the characteristics of the datasets. The dataset HRF has the highest resolution, and the same shape as STARE, which is not a complete image of the eyeball, while STARE is of poor quality because it is composed of scanned and digitized photographs. The dataset DRIVE and IOSTAR share the same shape of the whole eyeball, but the resolution of DRIVE is lower. All experiments are performed on a server with 8 GeForce RTX™ 2080 Ti Turbo 11G GPUs.

4.2.2 Data Augmentations Exploration

RQ1: What kind of data augmentation combination in the Mean Teacher model produces the best performance for retinal vessel segmentation?

In order to find the best performance data augmentation techniques applied in the Mean Teacher model for the retinal vessel segmentation task, Experiment 1 is designed and composed of three steps. Since the initialization network in (Perone and Cohen-Adad, 2018) is U-NET, the initialization network chosen first is U-NET. The data augmentation techniques that are applied in (Perone and Cohen-Adad, 2018) are rotation and Gaussian noise. In typical semantic segmentation tasks, all three categories of data augmentation techniques summarized in (Tajbakhsh et al., 2020) can be applied. However, the typical tasks usually have an area where the true-labeled pixels cluster. For retinal vessel segmentation, the true-labeled pixels are sparse and distributed all over the image, so the perturbation to spatial information will lead to wrong prediction. Therefore, the data augmentations that will change the layout of the image are not suitable for this task. Since rotation changes the layout of the image, it is not adopted in our experiments. In order to reproduce the result in (Perone and Cohen-Adad, 2018), Experiment 1.1 is conducted to apply Gaussian noise to the training samples. Experiment 1.1 shows the performance of the model on RGB images. The RBG images provide more information than grayscale images, which may affect the features our model learns from the data. Experiment 1.2 is designed to verify this assumption. In this experiment, grayscale and Gaussian noise are combined to perturb the input image. By comparing the results of Experiment 1.1 and Experiment 1.2, it should be clear whether the Mean Teacher model can perform better on RGB images or not. Speaking of data augmentations that change image appearance, the images in RGB have 4 attributions to change: brightness, contrast, hue, and saturation. But for grayscale images, only brightness and contrast can be changed, the values of hue and saturation are always 0. In Experiment 1.3, experiments are conducted on the image category that the Mean Teacher model performs better. If the results of Experiment 1.1 and Experiment 1.2 show the model has better performance in RGB images, the brightness, contrast, hue, and saturation adjustment will be applied to the input in turn, together with Gaussian noise. Only one attribution is changed in every experiment so that the effect of each attribution can be investigated. On the contrary, if the model performs better on grayscale images, the brightness and contrast adjustment will be applied to the input in turn, together with grayscale and Gaussian noise. The effects of the data augmentation techniques applied on one example image are shown in Figure 4.3.



(a) Example of input (b) Noise (c) Grayscale

(d) Brightness on grayscale (e) Contrast on grayscale (f) Hue on RGB

(g) Brightness on RGB (h) Contrast on RGB (i) Saturation on RGB

Figure 4.3: EFFECT OF DATA AUGMENTATION ON ONE EXAMPLE INPUT. This figure visualizes the effects of data augmentations adopted in the experiment. (a) is a sample from dataset HRF, (b) is the image got after adding Gaussian noise to the input, (c) is the image got after converting the input to grayscale, (d) is the result of adjusting the brightness of the grayscale image, (e) is the result of adjusting the contrast of the grayscale image, (f) is the result of adjusting the hue of the RGB image, (g) is the result of adjusting the brightness of the RGB image, (h) is the result of adjusting the contrast of the RGB image, (i) is the result of adjusting the saturation of the RGB image.

4.2.3 Initialization Network Comparison

RQ2: Will the choice of initialization network influence the performance of the Mean Teacher model?

In the data augmentation exploration, the performances of data augmentation combinations are compared. However, whether the choice of initialization network will affect the performance of the Mean Teacher model is not investigated. In Experiment 2, the best performance data augmentation combination in Experiment 1 is applied to the input, while the initialization network is changed to DRIU. By comparing the result of Experiment 2 with the result of Experiment 1, the effect of initialization network choosing can be explored.

4.2.4 Effect of Unlabeled Datasets Amount

RQ3: Will increasing the number of unlabeled datasets influence the performance of the Mean Teacher model on an unseen dataset?

The test sets in the above experiments are all split from the same dataset where the unlabeled train sets are from. Thus the performance of the Mean Teacher model on unseen datasets is still unknown. In experiment 3, the model is set according to the best performance model configuration in Experiment 2. The model is first trained with one dataset as labeled data, another one as unlabeled data, and tested on the test set of the other two datasets. The generalization ability of the model to the unseen datasets is tested in this step. After that, the number of unlabeled datasets used in semi-supervised training is increased to 2. When the training is finished, the performance of the model is evaluated on the test set of the one dataset left, which is unseen to the model. In this way, the effect of increasing unlabeled datasets on the model's generalization performance can be investigated.

4.2.5 Supervised Model Generalization Performance

In order to compare the generalization ability of the Mean Teacher model with the supervised model, a supervised cross-dataset test is conducted. The supervised models are the initialization networks used in the Mean Teacher model. All training samples are preprocessed by random augmentations: horizontal flipping, vertical flipping, rotation, and changes in brightness, contrast, saturation, and hue. The train /test splits of the four datasets stay consistent with the Mean Teacher model's configuration. Because the number of samples in the training set is not large enough to spilt out a validation set, a copy of the original training set is treated as the validation set. Both the U-NET and DRIU models are trained with one of these datasets, and tested on the other three datasets' test sets. Both models are trained with a learning rate of 0.001 and the number of epochs is 1000. The learning rate is decreased by a factor of 0.1 after 900 epochs. The batch size is 4 and 8, for U-NET and DRIU, respectively. The loss function and the metrics are the same as in the initialization network training. In this way, the generalization performance of the supervised model can be obtained.

Chapter 5

Experiment Results

In this chapter, the results of experiments described in Chapter 4 are shown and visualized. According to the experiment setup, the results are divided into four parts: data augmentations exploration, initialization network comparison, the effect of unlabeled datasets amount, and the supervised model generalization ability test. All performances are evaluated by the F1 score.

5.1 Data Augmentations Exploration Results

In this section, the results of the data augmentations exploration experiment are shown. In Experiment 1.1, only Gaussian noise is applied to the input samples, while in Experiment 1.2, grayscale and Gaussian noise are applied. Table 5.1 gives the details of Experiment 1.1 result.

Labeled Set \ Unlabeled Set	DRIVE	STARE	HRF	IOSTAR
DRIVE	**0.795**	0.507	0.502	0.681
STARE	0.734	**0.815**	0.652	0.656
HRF	0.635	0.404	**0.799**	0.671
IOSTAR	0.345	0.362	0.404	**0.824**

Table 5.1: EXPERIMENT 1.1 RESULT. The data augmentation adopted in Experiment 1.1 is Gaussian noise. The initialization network is U-NET. The bolded numbers in diagonals are the results of pre-training. The other numbers are the results of semi-supervised training with corresponding labeled and unlabeled datasets. The blue numbers indicate data pairs with performance in Experiment 1.1 better than in Experiment 1.2.

Figure 5.1 shows the comparison of F1 scores when Gaussian noise is applied to RGB images (Experiment 1.1) and grayscale images (Experiment 1.2) with the labeled dataset IOSTAR. The figure takes only part of the results to visualize the comparison. When looking into the details of the results, we find that on almost all dataset pairs, the model trained with grayscale images performs better than the model trained with RGB images. The only exception is the dataset pair of DRIVE as labeled data and IOSTAR as unlabeled data. The means of F1 scores of all dataset pairs in Experiment 1.1 and 1.2 are 0.547 and 0.700, respectively. The standard deviation of the F1 scores of all dataset pairs in Experiment 1.1 is 0.141, while the standard deviation of the F1 scores of all dataset pairs in Experiment 1.2 is 0.050. In other words, the performance of the model trained with grayscale images is better than the model trained with RGB images.

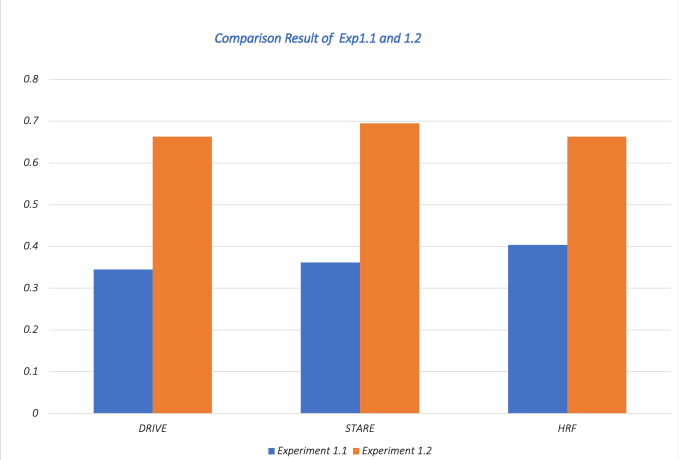


Figure 5.1: RESULT COMPARISON OF EXPERIMENT 1.1 AND 1.2. The initialization network is U-NET. The labeled dataset is IOSTAR. The dataset under the bar is the unlabeled dataset corresponding to the result. The blue bars show the results of Experiment 1.1, while the orange bars show the results of Experiment 1.2.

According to the experiment setup, in Experiment 1.3, the adjustments are applied to the grayscale images. There are two steps in Experiment 1.3: first, the brightness adjustment and Gaussian noise are applied to the grayscale images; then contrast adjustment and Gaussian noise are applied to the grayscale images. The two steps' results are compared with the result of Experiment 1.2. During experiments, we find the F1 scores of these three data augmentation com-

Experiment \ Unlabeled Set	DRIVE	STARE	HRF
Experiment 1.2	0.663	0.695	0.663
Experiment 1.3.1	0.679	0.724	0.661
Experiment 1.3.2	0.704	0.650	0.677

Table 5.2: COMPARISON OF PARTIAL RESULTS OF EXPERIMENT 1.2 AND EXPERIMENT 1.3. The labeled dataset is IOSTAR. The initialization network is U-NET. Experiment 1.2 applies grayscale and Gaussian noise, while Experiment 1.3.1 applies brightness adjustment, grayscale, and Gaussian noise, and Experiment 1.3.2 applies contrast adjustment, grayscale, and Gaussian noise. The number in the table is the F1 score of the model trained with the unlabeled dataset corresponding to the column in the Experiment corresponding to the row.

binations are very close, as partially shown in Table 5.2. Therefore, for every labeled/unlabeled dataset pair, 3 times of experiments are conducted to compare their performance statistically. The complete result is listed in Table 5.3.

For the entire result, a mean and standard deviation are computed to compare the performance of different data augmentations. The result is shown at the bottom part of Table 5.3. The higher the mean is, and the lower the standard deviation is, the better the performance is. From our experiments, when the initialization network is U-NET, the results show that the Mean Teacher model performs best with Gaussian noise when turning the input to grayscale. Eliminating the effect of saturation and hue information in the input image can improve the performance, while the brightness and contrast adjustments on the grayscale do not provide obvious improvement

	Experiment 1.2			Experiment 1.3.1			Experiment 1.3.2			Experiment 2		
	t_0	t_1	t_2	t_0	t_1	t_2	t_0	t_1	t_2	t_0	t_1	t_2
DRIVE												
STARE	0.764	0.760	0.744	0.768	0.778	0.769	0.756	0.755	0.754	0.759	0.753	0.755
HRF	0.622	0.643	0.620	0.644	0.629	0.656	0.639	0.639	0.640	0.616	0.616	0.621
IOSTAR	0.664	0.628	0.616	0.550	0.538	0.526	0.407	0.433	0.421	0.724	0.673	0.696
STARE												
DRIVE	0.742	0.745	0.737	0.736	0.749	0.735	0.727	0.740	0.733	0.704	0.702	0.706
HRF	0.655	0.672	0.662	0.685	0.684	0.691	0.667	0.663	0.677	0.593	0.615	0.624
IOSTAR	0.721	0.725	0.734	0.667	0.636	0.614	0.533	0.527	0.457	0.670	0.660	0.646
HRF												
DRIVE	0.687	0.686	0.677	0.655	0.677	0.686	0.665	0.665	0.671	0.692	0.682	0.682
STARE	0.682	0.707	0.704	0.698	0.701	0.670	0.664	0.654	0.646	0.677	0.698	0.699
IOSTAR	0.741	0.746	0.748	0.698	0.738	0.723	0.645	0.685	0.652	0.565	0.573	0.564
IOSTAR												
DRIVE	0.663	0.668	0.640	0.679	0.711	0.701	0.704	0.666	0.689	0.703	0.701	0.688
STARE	0.695	0.660	0.651	0.724	0.697	0.701	0.650	0.661	0.672	0.663	0.680	0.671
HRF	0.663	0.653	0.667	0.661	0.662	0.655	0.677	0.673	0.674	0.609	0.604	0.640
Mean	**0.689**			**0.680**			**0.644**			**0.665**		
Std	**0.044**			**0.059**			**0.091**			**0.051**		

Table 5.3: RESULTS OF EXPERIMENT 1.2, 1.3, AND 2. In the first column, the bolded datasets are used as the labeled dataset, and the italics datasets are used as the unlabeled dataset. t_0, t_1, and t_2 denote three times experiments with the same parameters. The initialization network in Experiment 1.2 and 1.3 is U-NET, and the initialization network in Experiment 2 is DRIU. Mean and Std are the mean and standard deviation of all three times results of the corresponding experiment. The highest mean is highlighted in blue, and the lowest standard deviation is highlighted in red.

Dataset	Mean	Std
DRIVE	0.673	0.064
STARE	0.710	0.037
HRF	0.709	0.029
IOSTAR	0.662	0.015

Table 5.4: RESULTS OF EXPERIMENT 1.2. The mean and standard deviation of the performance of the Mean Teacher model with different datasets used as the labeled data. The highest mean is highlighted in blue, and the lowest standard deviation is highlighted in red. The higher the mean is, and the lower the standard deviation is, the better the performance is.

in the performance of the model.

In order to further investigate which dataset provides the best generalization performance in the Mean Teacher model, the mean and standard deviation are computed for each dataset used as the labeled data. The result is shown in Table 5.4. It can be seen that HRF provides almost the same mean as the highest one, but a much lower standard deviation. This means the Mean Teacher model performs best when trained with HRF as the labeled dataset.

5.2 Initialization Network Comparison

In Experiment 1, the best performance data augmentation applies to the dataset when the initialization network is U-NET is explored. Another factor that may affect the model's performance is the initialization network. In this section, the initialization network is changed to DRIU and the data augmentation techniques are grayscale and Gaussian noise since they produce the best performance in Experiment 1. The result of Experiment 2 is close to Experiment 1, so Experiment 2 is conducted 3 times as well. The result is shown together with Experiment 1 in Table 5.3 to make it easier to compare. The mean and standard deviation of Experiment 2's results are 0.665 and 0.051, respectively. By comparing the mean and standard deviation with results in Experiment 1, it can be seen that the model's performance is affected by the choice of initialization network, and it performs better when the initialization network is U-NET than DRIU. The reason should be that in DRIU, the skip connection provides more information to the network, while in DRIUI, the extracted feature map contains coarser information.

5.3 Effect of Unlabeled Datasets Amount

In order to test the generalization ability of our model on an unseen dataset, we set one dataset as the unseen test set and one of the other three datasets as the labeled train set to train the model. The unlabeled train set comes from the remaining two datasets. The number of unlabeled datasets in the training increases from 1 to 2. The results are shown in Table 5.5.

Our results suggest that increasing the number of unlabeled datasets does not always improve or reduce the performance of the model. The hypothesis is that if the added unlabeled dataset provides the information labeled dataset lacks, increasing the number of unlabeled datasets will improve the performance of the model; if not, the performance would be worse. For example, when the model is tested on IOSTAR, the F1 score of the prediction decreases when the model is trained with DRIVE as the labeled dataset and the other two as the unlabeled datasets. However,

Labeled Data \ Unlabeled Data	STARE	HRF	IOSTAR	STARE HRF	STARE IOSTAR	HRF IOSTAR
STARE	-	0.645	0.712	-	-	0.754
HRF	0.670	-	0.664	-	0.669	-
IOSTAR	0.549	0.530	-	0.570	-	-

(a) Test Set: DRIVE

Labeled Data \ Unlabeled Data	DRIVE	HRF	IOSTAR	DRIVE HRF	DRIVE IOSTAR	HRF IOSTAR
DRIVE	-	0.693	0.764	-	-	0.634
HRF	0.693	-	0.664	-	0.510	-
IOSTAR	0.231	0.410	-	0.564	-	-

(b) Test Set: STARE

Labeled Data \ Unlabeled Data	DRIVE	STARE	IOSTAR	DRIVE STARE	DRIVE IOSTAR	STARE IOSTAR
DRIVE	-	0.646	0.634	-	-	0.580
STARE	0.585	-	0.595	-	0.689	-
IOSTAR	0.583	0.655	-	0.543	-	-

(c) Test Set: HRF

Labeled Data \ Unlabeled Data	DRIVE	STARE	HRF	DRIVE STARE	DRIVE HRF	STARE HRF
DRIVE	-	0.691	0.541	-	-	0.490
STARE	0.450	-	0.354	-	0.783	-
HRF	0.677	0.744	-	0.745	-	-

(d) Test Set: IOSTAR

Table 5.5: RESULTS OF EXPERIMENT 4. Each table corresponds to a dataset whose original test set is used as the model's test set. In every table, the datasets in the first column are used as labeled data with all their samples. The datasets in the first row are used as the unlabeled data with all their samples. - means there are no results here since the labeled dataset and unlabeled datasets overlap. The initialization network is U-NET, and the data augmentation is grayscale and Gaussian noise. The highest results of each test set are highlighted in color. Different colors mean the results are produced by different labeled datasets. Blue results are trained with the labeled dataset DTARE, and red results are trained with the labeled dataset DRIVE.

when the labeled dataset is HRF or STARE, the F1 score of the prediction increases along with increasing the number of unlabeled datasets. According to our hypothesis, dataset DRIVE is the most similar dataset to IOSTAR in terms of features learned by our model; adding any other dataset will not provide any new information to the model. Instead, it will only increase the noise learned by the model. Moreover, adding HRF as the unlabeled dataset generates a worse result than adding STARE, suggesting that HRF is more dissimilar to IOSTAR.

It can be seen that the Mean Teacher model performs best in predicting unseen datasets when STARE is used as the labeled dataset and trained with two unlabeled datasets. This is an interesting finding because the STARE dataset is the smallest dataset with poor quality. We hypothesize that the Mean Teacher model learns more information from more unlabeled datasets and eliminates the gap between the labeled dataset and the test dataset when the labeled dataset is not of good quality. The information learned from the unlabeled datasets helps the model generalize better to unseen datasets.

5.4 Supervised Model Generalization Performance

In Experiment 1, 2, and 3, the performance of the Mean Teacher model is tested. In order to compare with the supervised model, a generalization ability test on the two initialization networks is conducted. The result of the generalization ability of the supervised models on an unseen dataset is shown in Table 5.6. It can be seen that for some dataset pairs, the generalization ability is quite

Train Set \ Test Set	DRIVE	STARE	HRF	IOSTAR
DRIVE	0.814	0.795	0.701	0.791
STARE	0.741	0.825	0.750	0.671
HRF	0.636	0.735	0.803	**0.560**
IOSTAR	0.758	0.768	0.702	0.824

(a) U-NET Generalization Ability

Train Set \ Test Set	DRIVE	STARE	HRF	IOSTAR
DRIVE	0.811	0.797	0.705	0.766
STARE	0.756	0.828	0.742	0.597
HRF	0.576	0.641	0.797	**0.508**
IOSTAR	0.760	0.767	0.674	0.825

(b) DRIU Generalization Ability

Table 5.6: SUPERVISED MODEL GENERALIZATION PERFORMANCE. The numbers in the diagonals are the F1 scores when the corresponding model is trained and tested on the same dataset. The other numbers are the F1 scores when the corresponding model is trained with the train set and tested with the test set corresponding to the row and the column. The lowest cross-dataset result is highlighted in bold.

low. The mean and standard deviation of the F1 scores of the two supervised model cross-dataset test are 0.717, 0.069 for U-NET, and 0.691, 0.092 for DRIU, respectively. U-NET model performs better than DRIU model in generalization ability in supervised training, which is consistent with their performances in the Mean Teacher model. Although the mean of the F1 score of the U-NET model in supervised learning is higher than in the Mean Teacher model, the standard deviation is also higher, which means it is possible for the supervised model to produce relatively inaccurate predictions on some dataset pairs, for example, the HRF-IOSTAR pair. Its F1 score is the lowest number in Table 5.6 and is highlighted in bold, implying that HRF is more difficult to generalize to IOSTAR than other datasets. This is consistent with our result in Experiment 3.

Chapter 6

Discussion

Following the analysis of the experiment results in chapter 5, in this section, we discuss the findings of the experiments and the limitations of this work.

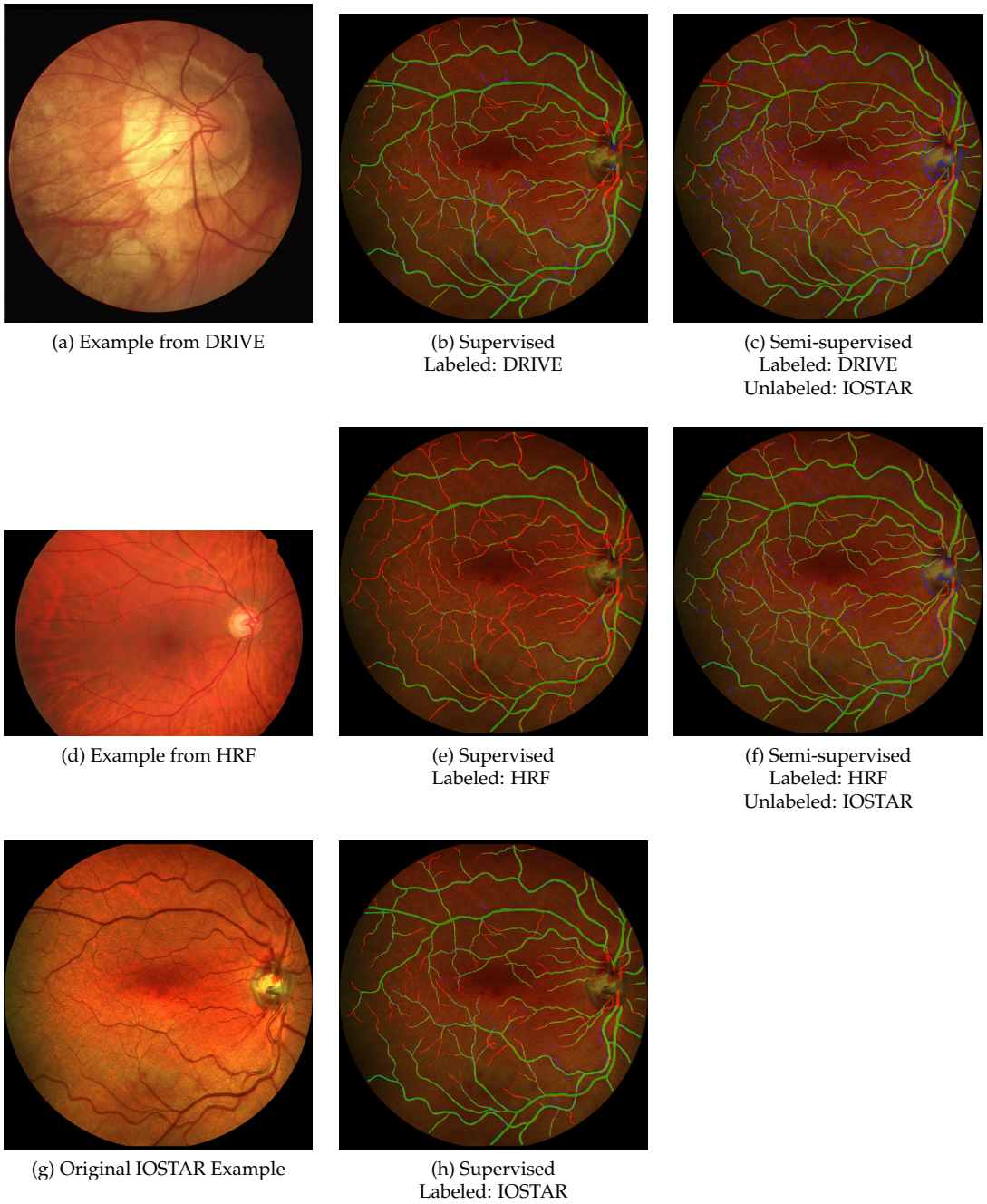
6.1 The Mean Teacher Model

By comparing the results of the experiments, we find that for the Mean Teacher model, the best performance occurs not always when the labeled data is of the best quality. When predicting on a known unlabeled dataset, the dataset with the highest resolutions achieves the best performance, even though the shape of the samples is not an entire eyeball. However, the performance of the model is not positively correlated with the resolution of the labeled dataset. It is hard to say what affects the performance most. The factors may be the number of samples, the quality of the samples, the position of the optic disc, or all of them. We can not draw a conclusion from this work due to the limited amount and the small size of the datasets.

When looking into the performance of the Mean Teacher model on different labeled/unlabeled dataset pairs, we can see that the worst generalized pair in the supervised learning model is better generalized. Figure 6.1 shows the comparison of predictions produced by the Mean Teacher model and the supervised model on the same dataset pairs. Among all labeled/unlabeled dataset pairs, the HRF/IOSTAR has the worst generalization performance in the supervised learning model, and the performance is significantly improved by the Mean Teacher model. However, the best performance pair (DRIVE/IOSTAR) gets slightly worse when turning the model from supervised to the Mean Teacher. The reason may be the samples in HRF are not entire eyeballs, the lost information is learned from the unlabeled dataset. In contrast, the DRIVE dataset is similar to IOSTAR, and the model learns more noise from the unlabeled dataset. However, other factors may also affect the performance such as the number of samples, the quality and the color of samples, and the position of the optic. Nevertheless, the Mean Teacher model can improve the performance of dataset pairs that can not be well generalized in the supervised learning models.

When predicting an unseen unlabeled dataset, the dataset with the poorest quality achieves the best performance by increasing the number of unlabeled datasets. The reason we hypothesize is that the unlabeled datasets with better quality complement the labeled dataset and improve the performance. It is worth exploring whether to keep increasing the number of unlabeled datasets can improve the performance of the model. Due to the lack of datasets, we can not conduct a comparative experiment to verify this finding.

However, increasing the number of unlabeled datasets does not always improve the performance of the model. If the labeled dataset and the test dataset are similar, the unlabeled dataset may provide information that can be seen as noise to the model and degrade the performance. For example, for our model, dataset DRIVE is the most similar dataset to IOSTAR; dataset HRF



(a) Example from DRIVE

(b) Supervised
Labeled: DRIVE

(c) Semi-supervised
Labeled: DRIVE
Unlabeled: IOSTAR

(d) Example from HRF

(e) Supervised
Labeled: HRF

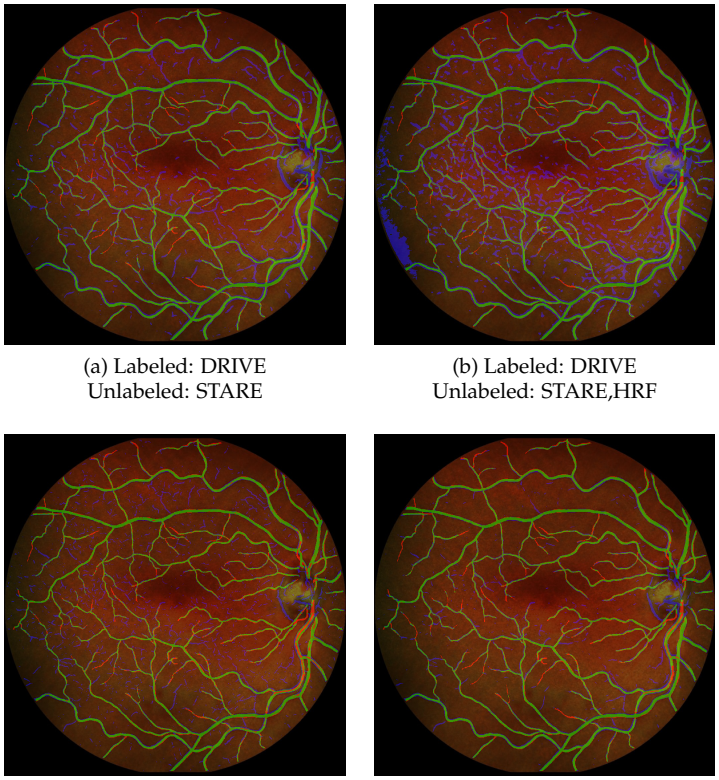
(f) Semi-supervised
Labeled: HRF
Unlabeled: IOSTAR

(g) Original IOSTAR Example

(h) Supervised
Labeled: IOSTAR

Figure 6.1: PREDICTIONS OF SUPERVISED MODEL AND SEMI-SUPERVISED MODEL. The supervised model is U-NET, the same as the initialization network used in the Mean Teacher model. Green lines indicate true positives, red lines indicate false negatives and blue lines indicate false positives. In supervised learning, DRIVE is better generalized to IOSTAR than HRF, while in semi-supervised learning, the model learns more noise which degrades the performance when the labeled dataset is DRIVE. In contrast, the Mean Teacher model trained with HRF and IOSTAR learns more useful information and achieves better performance than the supervised model. An example of the labeled dataset is shown in the first column.

is difficult to generalize to IOSTAR, which is suggested in the results of the supervised learning model. Figure 6.2 shows the comparison of predictions when increasing the number of unlabeled datasets in the Mean Teacher model. The figure clearly shows that for the DRIVE/IOSTAR pair, increasing the number of unlabeled datasets leads to more false positives, while for the HRF/ IOSTAR pair, increasing the number of unlabeled datasets reduces the number of false positives.



(a) Labeled: DRIVE
Unlabeled: STARE

(b) Labeled: DRIVE
Unlabeled: STARE,HRF

(c) Labeled: HRF
Unlabeled: DRIVE

(d) Labeled: HRF
Unlabeled: DRIVE,STARE

Figure 6.2: COMPARISON OF PREDICTIONS WHEN INCREASING THE NUMBER OF UNLABELED DATASETS. This figure shows different predictions on dataset IOSTAR when increasing the number of unlabeled datasets. Green lines indicate true positives, red lines indicate false negatives, and blue lines indicate false positives. The captions under each image indicate the labeled dataset and the unlabeled datasets used to train the model.

During semi-supervised training, it occurs that the model fails to converge in some experiments. The reason is that the optimizer used here is AdaBound. It is a new optimizer that combines the advantages of Adam and SGD. The loss function of this task is not so smooth that the optimizer may get stuck in a local minimum. In the future, we will try to use other optimizers to see if the problem can be solved.

In this work, the hyperparameters are not optimized. In the consistency weight function (3.2), the maximum value of the consistency weight w_{max}, the steps T the function takes to the maximum value and the weight function itself are possible to influence the performance of the model. It still needs to be explored what and how these hyperparameters affect the performance of the model.

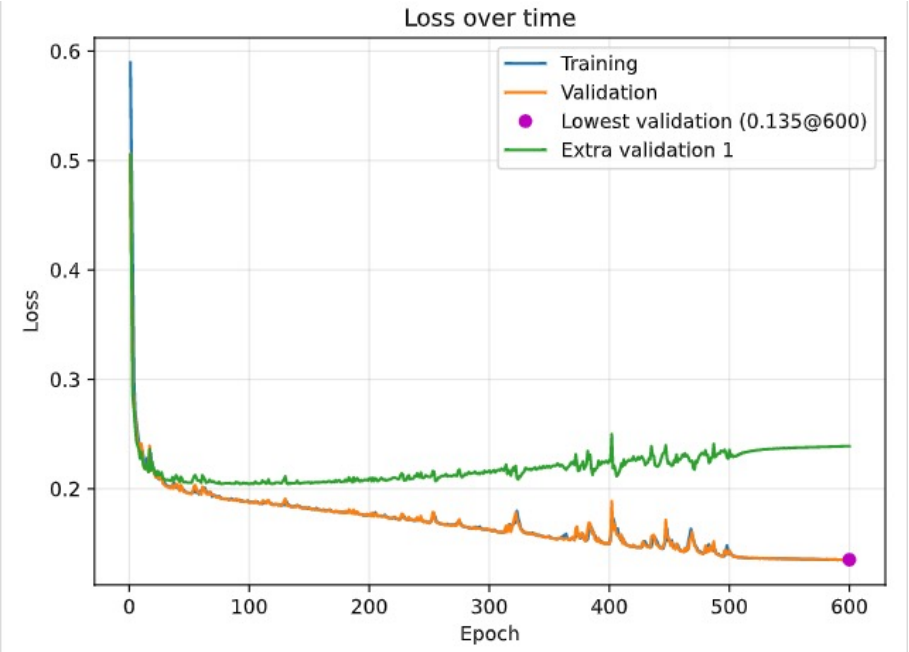


Figure 6.3: LOSS CURVE. This figure shows an example loss curve of U-NET pre-trained on dataset DRIVE as the labeled dataset after grayscale and adding Gaussian noise. The blue line indicates the training loss and the orange line indicates the validation loss. The extra validation set is the original testing set, and its loss is represented by the green line. The red dot indicates the lowest validation loss where the model parameters are saved and used to initialize the Mean Teacher model.

Due to the choice of the validation set and fixed training epochs, it can be seen from the loss curves of the pre-training that the model is overfitting to the train set. Figure 6.3 shows an example of the loss curve of U-NET pre-trained on dataset DRIVE as the labeled dataset after applying grayscale and Gaussian noise. The blue line indicates the training loss and the orange line indicates the validation loss. The red dot indicates the lowest validation loss where the model parameters are saved and used to initialize the Mean Teacher model. The extra validation set is the original test set, and its loss is represented by the green line. The green line is going up with the training loss decreasing, which indicates that the model is overfitting to the train set. In this case, the results of our experiments are lower than the result the model should achieve. However, our datasets are too small to split into a train set and validation set to prevent overfitting. It can be solved by using larger datasets when available.

Another limitation of this work is that the annotations of the ground truth in the datasets are not objective. Retinal vessels have complex structures and different experts may give different annotations on the same image. Even the same expert may give different annotations on the same image in different situations. This may lead to unstable performance. However, it is one of the dataset's characteristics and is difficult to change. It is worth discussing how to solve it properly.

6.2 Unsupervised learning

The performance of semi-supervised learning is affected by the size and quality of the labeled data. In order to overcome this limitation, unsupervised learning methods are proposed. Un-

supervised learning methods can be trained with unlabeled data only. The most common unsupervised learning method is self-supervised learning. The framework of self-supervised learning uses only unlabeled data to formulate a pretext learning task such as predicting the relative locations of the image patch or image rotation. These tasks should be useful for solving downstream practice problems. (Kolesnikov et al., 2019) summarized the self-supervised learning methods and their corresponding pretext tasks.

Recently, contrastive learning, which is a variant of self-supervised learning, becomes popular and draws a lot of attention. Contrastive learning regards the task of maximizing the similarity of different representations from one image as the pretext task directly (Chen et al., 2020). A simple illustration is shown in Figure 6.4. An input x_i is first transformed into two variants

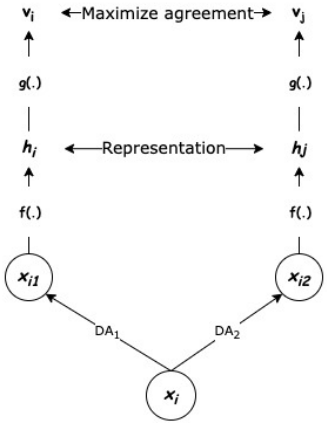


Figure 6.4: ILLUSTRATION OF CONTRASTIVE LEARNING. x_i is an input image, x_{i1} and x_{i2} are two variants after two different data augmentations DA1 and DA2. h_i and h_j are the representations of x_{i1} and x_{i2}, respectively, after a neural network $f(.)$. $g(.)$ is a projection function that maps the representations to the same space. v_i and v_j are the projections of h_i and h_j, respectively. The loss function is defined to maximize the similarity between v_i and v_j (Chen et al., 2020).

x_{i1} and x_{i2} by two different data augmentations. Then the representations of x_{i1} and x_{i2} are calculated by a neural network $f(.)$. The representations are then projected to the same space by a projection function $g(.)$. The projections of x_{i1} and x_{i2} are v_i and v_j, respectively. v_i and v_j are defined as positive pair. In (Chen et al., 2020), it is shown that the model performs better when the similarity is computed between v_i and v_j than between h_i and h_j. The choice of $g(.)$ also matters. A non-linear projection function is preferred. Let $sim(u, v)$ denote the similarity between two representations u and v. The loss function for a positive pair of examples(v_i, v_j) is defined as:

$$\ell_{i,j} = -log \frac{exp(sim(v_i, v_j)/\tau)}{\sum_{k=1}^{2B} \mathbb{1}_{[k \neq i]} exp(sim(v_i, v_k)/\tau)} \tag{6.1}$$

where B denotes the number of one batch examples and τ is a temperature parameter. $\mathbb{1}_{[k \neq i]}$ is an indicator function that is equal to 1 if $k \neq i$ and 0 otherwise. The final loss is computed across all positive pairs, both (i, j) and (j, i), in a batch and is defined as:

$$\mathcal{L} = \frac{1}{2B} \sum_{k=1}^{B} [\ell(2k-1, 2k) + \ell(2k, 2k-1)] \tag{6.2}$$

Equation (6.1) and Equation (6.2) indicate that the loss is computed at the image level. In

the segmentation task, loss at the pixel level is more required. To tackle this problem, (Wang et al., 2021) proposed a dense contrastive learning method. The authors calculated network loss which combined the global contrastive loss (Equation (6.2)) and a local dense contrastive loss. The feature vector of one view from one input was split into small grids. Two grids from the same location but different views of the same input were regarded as positive pair. The views from other inputs were pooled to the same size as the grid and formed the negative pairs. However, the grid positive pair is not easy to match as the method in (Chen et al., 2020). In order to match the positive pairs, all feature vectors were pooled to the same size as the grid. Then the similarity between the feature vectors was calculated and the most similar pairs were selected as the positive pairs. This method can partially solve the problem of pixel-level loss in typical semantic segmentation tasks, but still is not sufficient. Because true-labeled pixels in negative pairs may have the same amount as in the positive pair, which will confuse the model during matching positive pairs. The problem becomes even worse in the retinal vessel segmentation task because the true-labeled pixels are more dispersed. Therefore, it still needs further exploration to solve the problem of pixel-level loss in contrastive learning.

However, the idea of contrastive learning is promising. In some semi-supervised models, for example, (Hu et al., 2021) used contrastive learning to pre-train a network using the unlabeled data and then fine-tuned the model with labeled data. Some other works combined consistency learning and contrastive learning such as (You et al., 2021) and (Zhong et al., 2021), which is also a promising direction for further research in this task.

Chapter 7

Conclusion and Future Work

This work is aimed to alleviate the lack of labeled datasets in retinal vessel segmentation. The task is challenging because the true-labeled pixels belonging to the vessels are distributed all over the image so that limited data augmentation techniques can be applied to the samples. This work adapts a popular consistency learning model, the Mean Teacher model, to this task and tests the performance on four datasets. It provides some new insights into leveraging unlabeled datasets.

The Mean Teacher model is composed of two networks with the same structure. The two networks are initialized by any supervised learning model which is feasible for the task. Two variants of the same input are sent into the two networks and produce two predictions. The main idea of the model is to minimize the differences between the two predictions and maximize the agreement between the prediction and the ground truth (if available) by a combined loss. The two variants are generated by applying data augmentations to the input image. In our experiments, the model achieves the best performance when grayscale and Gaussian noise are adopted. The feasible data augmentations are limited by the distribution of the pixels belonging to the vessels. We expect a solution to apply stronger data augmentations to the inputs while not leading the model not able to reach convergency.

The results of our experiments show that the Mean Teacher model can exploit the unlabeled dataset to improve the generalization ability of the model when the supervised learning model is not able to generalize the labeled data to the unlabeled data. The choice of the labeled dataset used in training plays an important role in the model when trying to predict a known unlabeled dataset. It needs further investigation to find out the characteristics of the most suitable labeled dataset.

When predicting an unseen unlabeled dataset, the model provides a solution with one poor-quality labeled dataset and two good-quality unlabeled datasets and achieves improvement on some labeled/unlabeled dataset pairs. However, this solution needs more datasets and further experiments to verify. Moreover, the effect of increasing the number of unlabeled datasets also needs more datasets to investigate.

Our experiments imply that the performance can also be affected by the choice of the initialization network. Therefore, exploring the usage of other initialization networks is an interesting direction for future work. Due to the limited time, the effect of the hyperparameters is not investigated in this work, thus it can be further studied in the future.

List of Figures

List of Tables

Bibliography

Becker, C., Rigamonti, R., Lepetit, V., and Fua, P. (2013). Supervised feature learning for curvilinear structure segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 526–533. Springer.

Berns, G. S. and Berns, M. W. (1982). Computer-based tracking of living cells. *Experimental Cell Research*, 142(1):103–109.

Beucher, S., Bilodeau, M., and Yu, X. (1990). Road segmentation by watershed algorithms. In *PROMETHEUS Workshop, Sophia Antipolis, France*.

Caesar, H., Uijlings, J. R. R., and Ferrari, V. (2016). Region-based semantic segmentation with end-to-end training. In Leibe, B., Matas, J., Sebe, N., and Welling, M., editors, *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I*, volume 9905 of *Lecture Notes in Computer Science*, pages 381–397. Springer.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Christ, P. F., Elshaer, M. E. A., Ettlinger, F., Tatavarty, S., Bickel, M., Bilic, P., Rempfler, M., Armbruster, M., Hofmann, F., D'Anastasi, M., Sommer, W. H., Ahmadi, S., and Menze, B. H. (2016). Automatic liver and lesion segmentation in CT using cascaded fully convolutional neural networks and 3d conditional random fields. In Ourselin, S., Joskowicz, L., Sabuncu, M. R., Ünal, G. B., and III, W. M. W., editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2016 - 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II*, volume 9901 of *Lecture Notes in Computer Science*, pages 415–423.

Cui, W., Liu, Y., Li, Y., Guo, M., Li, Y., Li, X., Wang, T., Zeng, X., and Ye, C. (2019). Semi-supervised brain lesion segmentation with an adapted mean teacher model. In *International Conference on Information Processing in Medical Imaging*, pages 554–565. Springer.

Dong, H., Yang, G., Liu, F., Mo, Y., and Guo, Y. (2017). Automatic brain tumor detection and segmentation using u-net based fully convolutional networks. In del C. Valdés Hernández, M. and González-Castro, V., editors, *Medical Image Understanding and Analysis - 21st Annual Conference, MIUA 2017, Edinburgh, UK, July 11-13, 2017, Proceedings*, volume 723 of *Communications in Computer and Information Science*, pages 506–517. Springer.

Galdran, A., Anjos, A., Dolz, J., Chakor, H., Lombaert, H., and Ayed, I. B. (2020a). The little w-net that could: state-of-the-art retinal vessel segmentation with minimalistic models. *arXiv preprint arXiv:2009.01907*.

Galdran, A., Anjos, A., Dolz, J., Chakor, H., Lombaert, H., and Ayed, I. B. (2020b). The little w-net that could: State-of-the-art retinal vessel segmentation with minimalistic models. *CoRR*, abs/2009.01907.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11):139–144.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. B. (2017). Mask R-CNN. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2980–2988. IEEE Computer Society.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

Hu, X., Zeng, D., Xu, X., and Shi, Y. (2021). Semi-supervised contrastive learning for label-efficient medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 481–490. Springer.

Hung, W., Tsai, Y., Liou, Y., Lin, Y., and Yang, M. (2018). Adversarial learning for semi-supervised semantic segmentation. In *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*, page 65. BMVA Press.

Iglovikov, V., Seferbekov, S., Buslaev, A., and Shvets, A. (2018). Ternausnetv2: Fully convolutional network for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Jin, Q., Meng, Z., Pham, T. D., Chen, Q., Wei, L., and Su, R. (2019). Dunet: A deformable network for retinal vessel segmentation. *Knowledge-Based Systems*, 178:149–162.

Kolesnikov, A., Zhai, X., and Beyer, L. (2019). Revisiting self-supervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1920–1929.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Bartlett, P. L., Pereira, F. C. N., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 1106–1114.

Lahiri, A., Ayush, K., Biswas, P. K., and Mitra, P. (2017). Generative adversarial learning for reducing manual annotation in semantic segmentation on large scale miscroscopy images: Automated vessel segmentation in retinal fundus image as test case. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 794–800. IEEE Computer Society.

Laibacher, T. and Anjos, A. (2019). On the evaluation and real-world usage scenarios of deep vessel segmentation for funduscopy. *CoRR*, abs/1909.03856.

Laine, S. and Aila, T. (2017). Temporal ensembling for semi-supervised learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Lee, D.-H. et al. (2013). Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Luo, L., Xiong, Y., Liu, Y., and Sun, X. (2019). Adaptive gradient methods with dynamic bound of learning rate. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Maninis, K.-K., Pont-Tuset, J., Arbeláez, P., and Gool, L. V. (2016). Deep retinal image understanding. In *International conference on medical image computing and computer-assisted intervention*, pages 140–148. Springer.

Marín, D., Aquino, A., Gegúndez-Arias, M. E., and Bravo, J. M. (2011). A new supervised method for blood vessel segmentation in retinal images by using gray-level and moment invariants-based features. *IEEE Trans. Medical Imaging*, 30(1):146–158.

Meyer, M. I., Costa, P., Galdran, A., Mendonça, A. M., and Campilho, A. (2017). A deep neural network for vessel segmentation of scanning laser ophthalmoscopy images. In Karray, F., Campilho, A., and Cheriet, F., editors, *Image Analysis and Recognition - 14th International Conference, ICIAR 2017, Montreal, QC, Canada, July 5-7, 2017, Proceedings*, volume 10317 of *Lecture Notes in Computer Science*, pages 507–515. Springer.

Orlando, J. I., Prokofyeva, E., and Blaschko, M. B. (2017a). A discriminatively trained fully connected conditional random field model for blood vessel segmentation in fundus images. *IEEE Transactions on Biomedical Engineering*, 64(1):16–27.

Orlando, J. I., Prokofyeva, E., and Blaschko, M. B. (2017b). A discriminatively trained fully connected conditional random field model for blood vessel segmentation in fundus images. *IEEE Trans. Biomed. Eng.*, 64(1):16–27.

Perone, C. S. and Cohen-Adad, J. (2018). Deep semi-supervised segmentation with weight-averaged consistency targets. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 12–19. Springer.

Ren, S., He, K., Girshick, R. B., and Sun, J. (2015). Faster R-CNN: towards real-time object detection with region proposal networks. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 91–99.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Soares, J. V. B., Leandro, J. J. G., Cesar, R. M., Jelinek, H. F., and Cree, M. J. (2006). Retinal vessel segmentation using the 2-d gabor wavelet and supervised classification. *IEEE Trans. Medical Imaging*, 25(9):1214–1222.

Staal, J., Abràmoff, M. D., Niemeijer, M., Viergever, M. A., and van Ginneken, B. (2004). Ridge-based vessel segmentation in color images of the retina. *IEEE Trans. Medical Imaging*, 23(4):501–509.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 1–9. IEEE Computer Society.

Tajbakhsh, N., Jeyaseelan, L., Li, Q., Chiang, J. N., Wu, Z., and Ding, X. (2020). Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation. *Medical Image Analysis*, 63:101693.

Tarvainen, A. and Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30.

Wang, X., Zhang, R., Shen, C., Kong, T., and Li, L. (2021). Dense contrastive learning for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3024–3033.

Xie, S. and Tu, Z. (2015). Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, pages 1395–1403.

You, C., Zhao, R., Staib, L., and Duncan, J. S. (2021). Momentum contrastive voxel-wise representation learning for semi-supervised volumetric medical image segmentation. *arXiv preprint arXiv:2105.07059*.

Zana, F. and Klein, J.-C. (2001). Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation. *IEEE Transactions on Image Processing*, 10(7):1010–1019.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2017a). Understanding deep learning requires rethinking generalization. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Zhang, Y., Yang, L., Chen, J., Fredericksen, M., Hughes, D. P., and Chen, D. Z. (2017b). Deep adversarial networks for biomedical image segmentation utilizing unannotated images. In Descoteaux, M., Maier-Hein, L., Franz, A. M., Jannin, P., Collins, D. L., and Duchesne, S., editors, *Medical Image Computing and Computer Assisted Intervention - MICCAI 2017 - 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part III*, volume 10435 of *Lecture Notes in Computer Science*, pages 408–416. Springer.

Zhao, H., Li, H., Maurer-Stroh, S., Guo, Y., Deng, Q., and Cheng, L. (2018). Supervised segmentation of un-annotated retinal fundus images by synthesis. *IEEE transactions on medical imaging*, 38(1):46–56.

Zhong, Y., Yuan, B., Wu, H., Yuan, Z., Peng, J., and Wang, Y.-X. (2021). Pixel contrastive-consistent semi-supervised semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7273–7282.