MASTERARBEIT | MASTER'S THESIS

Titel | Title

Anatomically Coherent Image Segmentation of Optic Disc and Cup

verfasst von | submitted by

Michael Oster BSc

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien | Vienna, 2025

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 066 645
Studienrichtung lt. Studienblatt | Degree programme as it appears on the student record sheet:	Masterstudium Data Science
Betreut von | Supervisor:	Dipl.-Ing. Mag. Hrvoje Bogunovic PhD

Kurzfassung

Zur Diagnose von Grünem Star werden häufig Fundoskopien herangezogen, welche den Sehnervenkopf und dessen Aushöhlung darstellen. Patienten, die an Grünem Star erkranken, leiden an einem erhöhten Augeninnendruck, was zu Schäden am Sehnerv führt und in besonders schlimmen Fällen mit der Erblindung des Patienten endet. Diese Schäden können in Fundoskopien durch eine wachsende Aushöhlung erkannt werden. Regelmäßige und flächendeckende Fundoskopien sind ein vielversprechender Kandidat zur Früherkennung von Grünem Star, doch die manuelle Auswertung durch medizinisches Fachpersonal ist zeitlich zu aufwendig für häufige Untersuchungen. Daher wurden in den letzten Jahre einige auf Deep Learning basierende Methoden entwickelt, um den Sehnervenkopf und dessen Aushöhlung in Fundoskopien zu segmentieren. Diese Masterarbeit beschäftigt sich mit U-Net, dem Model, auf dem viele der neueren Methoden aufbauen, und versucht anatomische Fakten in das Model zu integrieren. Das Ziel ist, ein Model zu bauen, dessen Segmentierungen anatomisch korrekt sind. Außerdem wird versucht, die benötigte Menge an annotierten Trainingsdaten zu reduzieren, da besonders im medizinischen Bereich Daten rar sind, während einiges an Vorwissen oft ungenutzt bleibt. Diese modifizierten Modelle werden dahingehend untersucht, wie dateneffizient sie lernen, also wie gut sie bereits mit wenigen Datenpunkten funktionieren.

Abstract

The diagnosis of glaucoma partly relies on retinal images, which show the optic cup within the optic disc of our eyes. A person with glaucoma suffers from high ocular pressure, which results in nerve damage and can ultimately lead to blindness. In retinal images, this can be observed as growth of the optic cup. Evaluating retinal images and especially the sizes of the optic disc and optic cup is a promising way to screen many people for glaucoma, however, medical experts would be overwhelmed by doing the evaluations manually. Therefore, the last years saw a rise of deep learning approaches, which allow for segmenting the optic disc and optic cup reasonably well. This Master's Thesis looks at U-Net, the architecture most of these approaches are built upon, and investigates the inclusion of anatomical prior knowledge. The goal is to make models which produce anatomically coherent outputs. Furthermore, this Master's Thesis aims to reduce the amount of labelled data needed to train the model, as data, especially in the medical domain, can be scarce, while anatomical knowledge is often present. Modifications of U-Net which include prior anatomical facts are tested and compared in terms of their data efficiency, i.e., how well they perform when trained with small amounts of data.

Acknowledgements

Writing this Thesis was both, much easier and more difficult than I expected. The writing and programming was manageable, my studies prepared me well in order to have the technical background for my tasks. However, the writing and programming was though, since you do it on your own and are the only person who has a strong incentive to continue. I was used to constant intellectual exchange with other people, this is what made my studies so joyful. Therefore, I thank my supervisor Hrvoje Bogunović and Botond Fazekas, who both supported me, even when the technical details got complicated. However, the person who spent most of her time supporting me was my manager, colleague, sparring partner and girlfriend Alina. Thank you for sharing some light in the darkest moments!

Contents

List of Tables

List of Figures

1 Introduction

This Master's Thesis falls within the domain of medical imaging. In particular, it deals with retinal images, which show the inside of the backside of an eye as it is taken through the pupil of a patient. These retinal images are used in the diagnosis of glaucoma and, among other things, show the optic cup within the optic disc. When a patient suffers from glaucoma, the optic cup grows within the boundaries of the optic disc, which results in irreversible damage to the optic nerve. Hence, a reliable and early detection of glaucoma is needed. Segmenting the optic disc and optic cup in retinal images is a promising candidate for this early detection, as a growing optic cup is an indicator for glaucoma. However, the human-based segmentation is a time-consuming process, which can be automated to make large scale glaucoma screenings feasible.

From a technical perspective, finding the optic disc and optic cup in a given image can be solved by image segmentation algorithms. These perform a pixelwise classification, assigning every pixel to one or multiple classes. The result is an output image which has the same size as the input image and contains a segmentation into, in our case, "background", "optic disc" and "optic cup". Please note that, since the optic cup is part of the optic disc, the segmentation algorithm is supposed to predict both, optic disc and optic cup, where the optic cup is present. Hence, it is not formulated as a multiclass classification per pixel in this Thesis, but rather three different binary classifications, one for each possible label.

In 2015, Ronneberger et al. presented a novel neural network architecture called U-Net, which was designed to perform image segmentation in the biomedical domain [1]. Since then, it has become a popular deep learning approach to image segmentation and is used in many variants and far beyond the field of biomedicine. The original architecture consists of convolutional layers, connected via max-pooling operations, to make the information present in the latent representation of the image more dense. This downsampling is then reversed using transposed convolutions. To make this architecture distinct from a conventional convolutional autoencoder, each iteration in the down-sampling phase gets fed into the according iteration in the up-sampling phase (the biggest image from the encoding phase is mapped to the biggest image in the decoding phase), as additional input. Hence, the name of the model, as it is typically depicted in a U-shape (see Figure 2.10 for the depiction in the original paper by Ronnerberger et al.[1]).

In 2018 and 2020 the REFUGE and REFUGE2 challenges took place, aiming (among other things) to find models that are good at segmenting the optic disc and optic cup [2] [3]. For this purpose, a dataset of 2000 labelled retinal images was provided by the hosts of the challenges, which is considered to be more than enough to train even advanced deep learning models for this particular problem. Most teams (14 out of 18) submitted architectures similar to U-Net, however, also Vision Transformers and DeepLabv3+ were used. Overall, state-of-the-art techniques were utilised to come up with good results, but they lacked guarantees of anatomical coherence.

It is the goal of this work to extend these models, by introducing prior knowledge to them and evaluating what effects this has on their performance and whether they become more anatomically coherent and resilient to a lack of vast amounts of available training data. Since the majority of the submitted models in the REFUGE challenges were based on the U-Net architecture, this Thesis explores the possibility to integrate prior knowledge into a U-Net model. The assumption is that provided knowledge decreases the amount of data needed to obtain the same results, as less information about the problem has to be learnt directly from the data. For this particular problem of segmenting the optic disc and optic cup, a lot of data is available, thanks to the REFUGE challenges. However, especially in medical imaging, data is very expensive. Therefore, finding ways to incorporate domain knowledge to reduce the need for data is one way to foster progress in this field.

There are two insights given by domain experts suitable for this investigation:

- The optic cup has to lie within the optic disc.
- Both, the optic disc and the optic cup, are usually round.

The first insight is incorporated by an additional loss term, which penalises models that predict the optic cup at places where no optic disc is predicted, thus violating basic anatomical facts. Furthermore, a model which is simply not capable of predicting the optic cup without the optic disc is employed. The second insight is used indirectly by transforming the images into polar coordinates, centred around a point within the optic cup. This transformation leads to images where models do not have to find circles within the image of interest, but rather rectangles. Also, this transformation disturbs the size of both the optic disc and optic cup to appear bigger.

This Thesis is structured in the following manner: Chapter 2 gives the reader an introduction to the current state-of-the-art for segmenting the optic disc and cup from retinal images. Furthermore, medical background revolving around glaucoma and an introduction to deep learning, especially in the field of segmentation, are provided. The Chapter closes with an analysis of papers which already tried to incorporate prior knowledge into deep learning architectures. Afterwards, Chapter 3 describes in detail what models are built and how prior anatomical knowledge is incorporated to foster their learning. Additionally, the data which is utilized for this Thesis is examined to provide a better understanding of the problem at hand. The models explained in Chapter 3 are then studied in Chapter 4 by comparing each new approach to a baseline U-Net. Finally, Chapter 5 concludes this Thesis by going through the main findings and possible future research.

2 Background & Literature Review

2.1 Medical Background - Glaucoma

Glaucoma describes an eye disease, which damages the optic nerve. One of the typical first symptoms of glaucoma is the loss, or impairment, of the periphery field of view. This impairment slowly progresses and, in severe cases, leads to the blindness of the patient. Due to the slow behaviour, patients often only realise the damage when the disease already progressed some time and daily activities such as reading become tough. This is the reason why glaucoma is called the “silent thief of sight”. According to estimates by Karen Allison et al., 118.8 million people will suffer from glaucoma by 2040, making it one of the leading causes of blindness [4]. Yet, around half of the patients who currently suffer from glaucoma are not diagnosed [2]. The mechanisms behind glaucoma are not fully understood, especially as the term describes a family of diseases. However, the eye pressure is one of the major risk factors [5]. Glaucoma is usually treated by the application of eye drops to lower the pressure, or even surgery to drain the extra fluid. As already mentioned, the disease progresses by damaging the optic nerve, which is irreversible. Therefore, diagnosing glaucoma as early as possible is vital to prevent most of the damage.

To make early detection on larger scales feasible, it is necessary to find a time-efficient and yet accurate diagnosing tool, which could be included in routine checks. Retinal images, combined with automated analyses of the taken images, are a good candidate for this task, as retinal images are relatively easy and quick to produce [2]. They typically allow a good view of the part of the retina where the optic nerve enters the eye, called the optic disc or optic nerve head[2]. On retinal images, it appears as a bright spot where blood vessels are converging. Within the optic disc, is the optic cup, which has no universal appearance, but sometimes looks like an even brighter spot inside the optic disc. Knowing the exact sizes of the optic disc and optic cup is of particular interest to medical experts, as calculating the vertical cup-to-disc-ratio (vCDR) is one of the standard diagnosis tools for detecting glaucoma. The disease results in damage to the optic nerve, perceived on retinal images as an increased size of the optic cup. This changes the vCDR, which is tracked over time to tell whether a patient has glaucoma.

Taking retinal images requires a so-called fundus camera (fundus meaning general backside of the eye, whereas in contrast retina is anatomically defined), which have seen development in the last years. Abràmoff et al. stated in 2010 that the portability of cameras was still an issue to be tackled, while today there are cameras approximately of the size of a nail gun, or even Smartphone-based devices which can take retinal images [6]. However, the quality of images should not be the price to pay for the enhanced portability.

Furthermore, for an image to be used for a reliable diagnosis, it should be taken by an expert. Once the retinal image is available for diagnosis, a medical professional locates the optic disc and optic cup to calculate the vCDR. Locating the optic cup can be challenging and errors can happen, especially when faced hundreds of times a day with such a tedious task. Automating this segmentation process would make the vCDR a cheap indicator for diagnosing glaucoma.

2.2 Image Segmentation

Image segmentation is the task of segmenting predefined classes in an image. For example, a medical doctor could be interested in the exact location of each bone in an X-ray. An image segmentation algorithm, specialised to decide for each pixel whether it shows a bone, would be able to tell the doctor where bones are present in the image. Although the doctor probably would be able to find all bones himself, there are more complicated segmentation problems, which are much harder to solve by hand. Furthermore, the automated segmentation allows for further computational analyses of the image, for instance, assessing whether all shown bones look healthy. Classical image segmentation methods are designed by defining rules that capture the characteristics of objects of interest, such as the colour, shape, or particular patterns in the texture. However, as deep learning advanced in the last years, innovations have come to the field of image segmentation. Before explaining a major milestone in Section 2.4, the basics of deep learning are needed as background.

2.3 Deep learning

Neural networks

The story of deep learning origins as an idea to solve classification problems, which means that we have some input data X and want to predict a label y. A classical example is spam detection for an email provider. Assume we have an email and we want to decide whether to show it to the user or place it in the spam folder. In order to do this, some information about the email is gathered in the form of numbers, e.g. amount of words, whether the sender is in the contacts of the user, how many hyperlinks are present in the text, etc. Finding these attributes can be tricky, but more on that later. These attributes are collected in a vector x, which now is the input for an algorithm which outputs $\widehat{y}$, where y is the true label (whether a given email is indeed considered spam) and $\widehat{y}$ its estimation. The algorithm that is fed with x mainly consists of a function $f : \mathbb{R}^n \to \mathbb{R}$, where n is the number of attributes, i.e., the length of the vector x. We estimate y by thresholding at 0:

$$\widehat{y} = \begin{cases} 1 \text{ if } f(x) > 0 \\ 0 \text{ else} \end{cases} \tag{2.1}$$

This is no sensible solution to any problem, as finding f, such that $\widehat{y}$ aligns well with

y, can be arbitrarily difficult. What functions should be considered when looking for a suitable f? To get a feeling for the problem at hand, imagine the following samples:

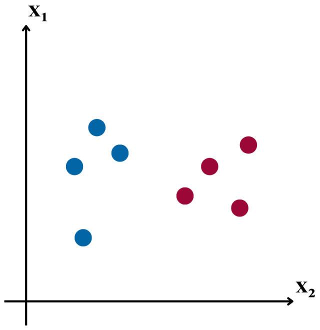

Figure 2.1: Labelled samples

Figure 2.1 shows an example of blue and red samples, which have two attributes x_1 and x_2. The goal of our function f is to separate the blue and red samples. In the case of Figure 2.1, a simple threshold for x_2 would already suffice, as Figure 2.2 shows:

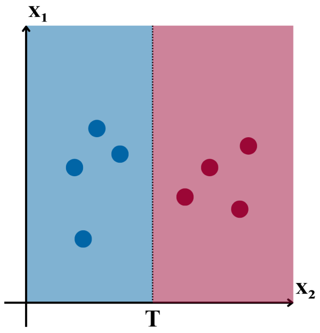

Figure 2.2: Labelled samples with classification threshold

The function associated to a threshold only in dimension x_2 (like in Figure 2.2) looks like:

$$f(x_1, x_2) = x_2 - T \tag{2.2}$$

When using this function f in Formula (2.1), all samples with $x_2 > T$ will get assigned to the class $\widehat{y} = 1$, which is assumed to be red. The dotted line in Figure 2.2 represents the decision boundary, i.e., $f(x) = 0$, where the model is most uncertain about its prediction.

In general, thresholding just a single dimension will not be enough to assign each sample to its correct space. The next logical step is to allow arbitrary straight lines across the plane:

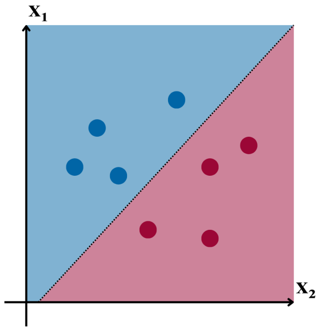

Figure 2.3: Labelled samples with linear f

Note that the position of some samples changed in Figure 2.3, which made the use of a threshold in just one dimension infeasible. The general form for functions f to obtain linear decision boundaries looks like:

$$f(x_1, x_2) = w_1 x_1 + w_2 x_2 + b \tag{2.3}$$

which is nothing more than a weighted sum of the attributes with a bias term. Without the bias term b, the decision boundary would have to pass through the origin, which can be easily verified by setting $x_1 = x_2 = 0$, hence including the bias term b adds flexibility. To generalise this function f to more than two input dimensions, the matrix notation is often used:

$$f(X) = W^T x + b = (\sum_{i=1}^{n} w_i x_i) + b \tag{2.4}$$

With this weighted sum in place, a big part of a simple artificial neuron is already done. To add another source of flexibility, a non-linear activation function is applied to the output of f. There are several famous activation functions, but for this introduction only the most basic ReLU function is of interest, which simply clips its input to be non-negative:

$$\sigma(x) = \max\{0, x\} \tag{2.5}$$

Now all parts are present to come to the actual neuron, as depicted in Figure 2.4. Applying the ReLU function to f before making a prediction using Definition (2.1) does not change the prediction. However, activation functions help to increase the potential of neurons when using more than a single one. Indeed, the universal approximation theorem tells us that already simple neuronal networks can approximate any continuous function, allowing for much more complex decision boundaries, compared to using only linear functions.

To understand why a neural net is more capable of producing complex functions f, consider the example in Figure 2.5.

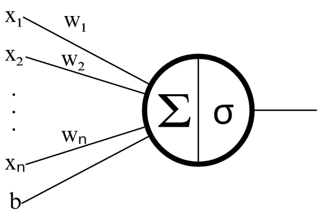

Figure 2.4: Neuron

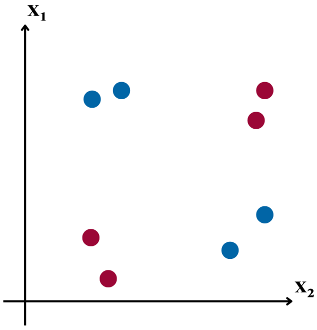

Figure 2.5: Samples with no linear separability

It shows a variation of the classical XOR problem, with red and blue samples lying on orthogonal diagonals. A single neuron is capable of producing an arbitrary straight line through the plane, however the XOR problem is not linearly separable, meaning that no straight line puts all samples into the correct class [7]. To solve this problem using neurons, a small neural net is necessary. Since each neuron is capable of producing a straight decision boundary, a first step could be to have two neurons which each separate one of the red groups from all other samples, as depicted in Figure 2.6.

Note that the neurons output positive values in their respective red areas and 0 in the blue areas, as negative values are clipped to 0 by the ReLU activation. In order to combine these two neurons, another neuron is used, which adds up the outputs of Neuron 1 and Neuron 2. The output of this final neuron will be positive, where either of its input neurons "fired", i.e. in both red areas of Figure 2.6. This leads to a function f which is positive where red samples are located, whilst being 0 at the blue samples, solving the XOR problem.

Figure 2.7 shows the full neural net to solve the XOR problem from Figure 2.5 [7]. The weights $w_{11}, w_{12}, w_{21}, w_{22}$ and biases b_1, b_2 are chosen to produce decision boundaries, as seen in Figure 2.6. The final neuron simply adds the outputs of the prior neurons. It

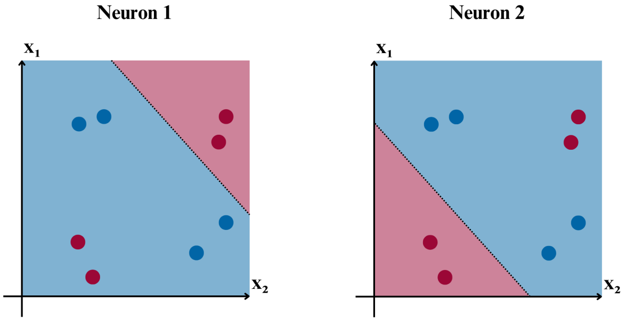

Figure 2.6: Two neurons producing two decision boundaries

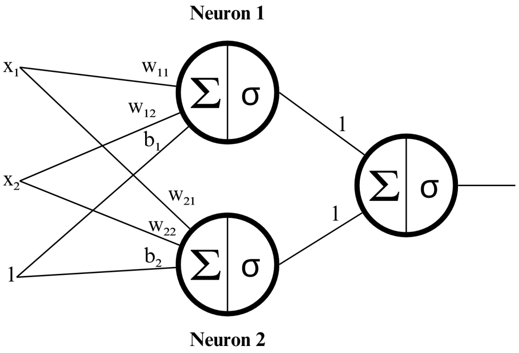

Figure 2.7: Neural net capable of solving the XOR problem

has no need for a bias term, it is implicitly set to 0. The Neurons 1 & 2 are located in the same "layer" of the neural network, as they share inputs. At the same time, the last output layer only consists of a single neuron.

This example leads back to the question: What are good attributes? As the XOR problem shows, a neural net is capable of creating its own attributes. Even though x_1 and x_2 are not perfectly suited to distinguish red from blue samples, the neural net is able to create two new attributes from them. Neuron 1 is active when samples are in the upper right corner, while Neuron 2 is active for samples close to the origin, and combining those already suffices for this problem. Neural nets are capable of producing near arbitrary decision boundaries, as long as equipped with enough neurons. This is to some extent due to their ability to create new, artificial attributes. However, this ability comes with the side effect that these new attributes are typically no longer interpretable by humans.

Convolutional neural nets

One of the main takeaways from the XOR problem is that neural nets are capable of looking for patterns which are important to solve the problem. In the XOR problem, we could observe that looking for samples which are in the top right corner, or close to origin, solves the problem. This idea of creating more abstract patterns (like where in the plane to look) plays a big role for a convolutional neural network (CNN). They are mainly used for image data, which usually has a very high amount of input attributes, or features. In this Thesis, for example, the input images have $512 \times 512 \times 3 = 786,432$ features, i.e., pixels. A naive approach, using neural nets, would involve a full neural network. This means that each neuron is connected to all neurons from the previous layer, or to all input features. Neurons located in the first layer therefore have $786,432$ weights, which all have to be tuned and optimized to enable this specific neuron to look for an interesting occurrence in the image. There is a more intuitive way to handle image data. Fully connected neural nets excel when the input features mostly share no behaviour, when the features are unstructured in some sense. Images, however, have in general very strong patterns and relations among different features. For example, two neighbouring pixels usually have very strong correlations, as natural images are mostly smooth. Hence, strong local changes (like edges) are of particular interest. The field of Image Processing provides quite a variety of methods to detect edges, but most of them rely on convolving a filter (or kernel) with the image of interest. These filters are typically much smaller than the images, e.g. 3×3 matrices for greyscale images. Convolving a filter with an image produces an image, which, assuming the image was padded accordingly, is as big as the input image. However, bright spots "mean" something different, as in the input image. In natural images, high values simply correspond to much light, but after convolving with a filter, this changes to a high resemblance with the pattern the filter searches.

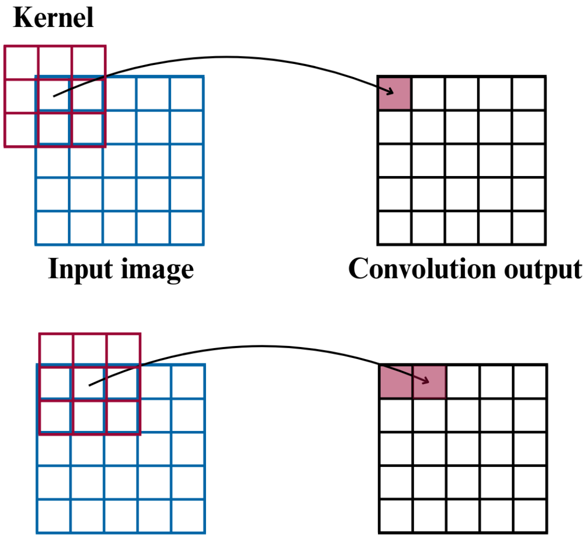

Figure 2.8: Convolution

Figure 2.8 visualises how convolution works. The filter is placed over the original image, and the overlaying pixels of the image and the filter are multiplied. These products are summed up and entered into the output image. The border of the image poses a problem, since some pixels of the kernel do not have a partner. There are several strategies to cope with this problem, like accepting smaller output images or artificially extending the input image. All convolutions in this Thesis use zero-padding, meaning that the input image gets a frame consisting of 0's. How thick this frame has to be, depends on the kernel size. The goal is to allow the centre of the kernel to always stay in the original part of the input image. A 3×3 kernel for example needs a 1-pixel wide padding around the input image.

Another important parameter of the convolution is the so-called stride, which controls how many pixels the kernel moves in each step. Figure 2.8 shows a convolution with a stride of 1, since the kernel moves 1 pixel in each step. If the stride was increased to 2, the kernel would skip half of the positions (since Figure 2.8 shows a 5×5 images, only 2 of the 5 positions are skipped), leading to an output image with size 3×3. Convolutions with stride greater than 1 therefore decrease the image size, making the information present more dense. These convolutions are called "strided convolutions" in this Thesis, even though all convolutions have a stride of at least 1.

Adding convolutions into neural networks marks a major milestone for deep learning and plays a big part in the success of neural networks. They are used by making the numbers in the kernel, in Figure 2.8 represented as a 3×3 matrix, learnable parameters. These numbers take the role of weights and biases of conventional neurons. While the neurons in Figure 2.7 could change their weights and biases to produce any linear decision boundary, the kernel parameters allow a convolutional unit to customize the pattern it searches for. By adding several units in parallel (making a layer wider), a CNN is able to search for many patterns, while only having a small amount of learnable parameters (the numbers the kernels are made of). The number of learnable parameters also does not change when a CNN is confronted with large images, in stark contrast to fully connected neural networks. Convolutional layers do not have to be located at the very beginning of a neural net, they can also be placed on top of each other, making the kind of pattern these later layers search for more abstract and hard to interpret.

Assuming a classification problem, like distinguishing pictures of cats from pictures of dogs, one could build a neural net by first stacking several convolutional layers and feeding the output of the last convolutional layer into a fully connected neural network. However, since fully connected neural networks are expensive in terms of parameters, it is desirable to make the output of the last convolutional layer smaller. Besides using strided convolutions, "pooling" can be applied, which typically aggregates 2×2 pixels into a single one, by taking the mean, max, or some other operation. This cuts the size of the image in each dimension in half, i.e. a pooling operation applied to a 2-dimensional image produces outputs with only a quarter of the pixels of the input images. These pooling operations can also be applied in between convolutional layers, to gradually decrease the size of the image.

Complementary to pooling, or strided convolutions, transposed convolutions can be used to increase the size of images. Again, this only applies to transposed convolutions with stride greater than 1. The increase is accomplished by adding rows and columns consisting of zeros into the input image. Figure 2.9 shows an 3×3 input image, which gains 2 rows and 2 columns in between. This bigger image is then used as input for a reversed convolution, where the kernel is weighted by the according centre pixel from the input image and added to the output. The kernel moves through the enlarged input images with a stride of 1 and where weighted kernels meet in the output image, they are added up at the overlapping pixels.

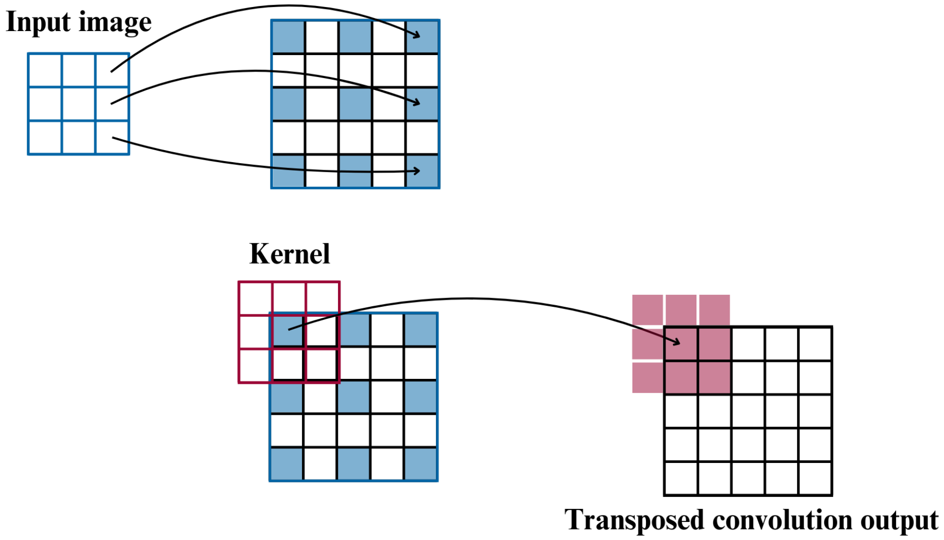

Figure 2.9: Transposed convolution

Note that increasing the stride for a transposed convolution does not mean the same as for a convolution. Instead of increasing the step size of the kernel, the stride increases the number of rows and columns added in the first step. A stride of 1 corresponds to no change, therefore a stride greater than 1 is necessary to increase the image size.

2.4 U-Net

In 2015 Ronneberger et al. presented a model called U-Net, which is widely used today [1]. It was developed for an image segmentation task in the biomedical domain, but its general structure allows for application in any domain. Using neural nets for image segmentation comes with the drawback that a lot of data is necessary to train a model. Furthermore, especially in the medical domain, an expert has to create this training data by segmenting example images. Ideally, these segmentations are correct on the pixel level of each image, which is tough to ensure when thousands of images are needed. The designers of U-Net claim that it works data efficient, which is desirable when working in the medical domain [1]. As visible in Figure 2.10, U-Net consists of a downsampling part, where convolutional layers alternate with pooling layers to make the information present in the latent spaces more dense. This process is then reversed by alternating between transposed convolutions to upsample the latent version of the image and normal convolutional layers. Crucially, U-Net feeds the corresponding latent image of the downsampling path as additional input

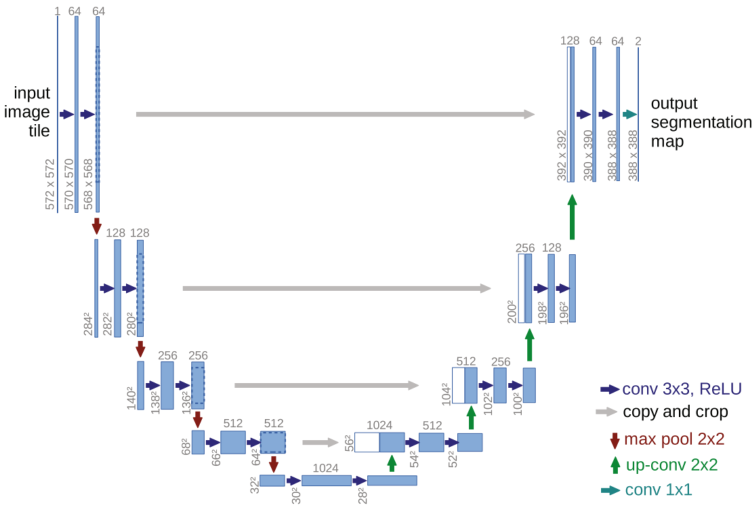

Figure 2.10: U-Net architecture from the paper by Ronneberger et al. [1]

to the upsampling path, indicated by the grey arrows in Figure 2.10. This allows for a crisp segmentation, as features of the original image can be used by the last convolutional layers. For this Master's Thesis, the python library MONAI was used to create a model similar to the original U-Net architecture.

In the original paper by Ronneberger et al., a cross-entropy loss is used to train the architecture. The cross-entropy loss is defined as:

$$L_{CE} = \sum_{p \in \Omega} \sum_{r=1}^{R} -y_p^r \cdot log(\widehat{y_p}^r) \tag{2.6}$$

where $p \in \Omega$ are all possible pixel indices for a given image, R is the number of classes, y_p^r is the ground truth label at a given pixel for a specific region r and $\widehat{y_p}^r$ is the outputted probability for the region r label at a given pixel, estimated by the model. By multiplying the log-likelihood of the prediction with y_p^r, only the correct region r at a given pixel p can have values greater than 0. Therefore, it is the goal to minimize $-log(\widehat{y_p}^r)$ for the correct class r. This can only be achieved by making $\widehat{y_p}^r$ as large as possible, however, since it resembles a probability, it is bound from above by 1. It is assumed that the prediction $\widehat{y_p}$ is put through a soft-max activation, which ensures that the probabilities for all regions r at a specific pixel p sum up to 1.

Nowadays, models which are based on U-Net are often trained using a Dice loss, which measures how much the ground truth image and the prediction overlap. The Dice loss is a natural extension of the Dice score (also called Sørensen-Dice index, Sørensen-Dice coefficient, Dice's coefficient, or Sørensen index), which is defined for two sets A and B:

$$DSC := \frac{2|A \cap B|}{|A| + |B|} \tag{2.7}$$

It can be observed that if A and B are equivalent, the Dice score is 1, since the intersection reduces to $|A|$ which is then multiplied by 2 and divided by $|A| + |A|$. Furthermore, if A and B share no elements, the cardinality of the intersection is 0, making the Dice score index 0 as well. Definition (2.7) can be extended to binary images, by viewing A and B as sets of active pixels in an image. In order to maximize this Dice score between a ground truth segmentation and a prediction produced by a neural network, the Dice loss is the most natural choice. For binarized images, it is usually defined as:

$$L_{Dice} = 1 - \sum_{r=1}^{R} \frac{1}{R} \sum_{p \in \Omega} \frac{2 y_p^r \cdot \widehat{y_p}^r}{y_p^r + \widehat{y_p}^r} \tag{2.8}$$

with the same notation as in Equation (2.6). The sum over all pixels $p \in \Omega$ essentially computes the Dice score, so it tells how similar the ground truth and the prediction are. This index is computed for each class r and then averaged over all R classes. A perfect overlap for all classes would make this average 1, hence, to turn this score into a loss, it is subtracted from 1 to get a value which an optimizer should decrease. Since the definition in Equation (2.8) is adopted from the Dice score, which is meant to be used for binarized images, the prediction $\widehat{y_p}^r$ should be converted from a probability to 0s and 1s. However, Milletari et al. argued in 2016 that Definition (2.8) is already differentiable with respect to $\widehat{y_p}^r$, making L_{Dice} suitable to be used as loss term [8].

2.5 Segmentation of Retinal Images

In the diagnosis of glaucoma, it is a common tool to look at the retinal images of a patient and calculate the vertical cup-to-disc-ratio (vCDR) [2]. This means that a medical professional locates the borders of the optic disc and optic cup to find out how much of the disc has already progressed to become part of the cup. Such a task can be easily formulated as a segmentation problem using retinal images as input. Before deep learning methods were widely used for machine learning problems, the researches concerned with segmentation tasks on retinal images used more classical approaches. In 2004, Lowell et al. proposed a method based on active contours to find the boundary of the optic disc [9]. Also, methods using morphological operations, like opening and closing, were proposed, in order to locate the optic disc, which is represented by a bright spot in gray-scale versions of retinal images. To additionally get the size of the optic disc, Sekhar et al. used a circular Hough transform, centred around their previously detected centre [10]. More advanced techniques, like active shape models, were also employed [11]. In 2013, Cheng et al. argued that using the purely circular Hough transform can harm the performances and especially introduce unnecessary errors to the vCDR, as the optic disc often appears as elliptic. They instead proposed a model based on superpixel classification [12].

The segmentation of the optic disc and optic cup via deep learning approaches gained some attention in recent years. There were two challenges, the REFUGE (Retinal Fundus Glaucoma Challenge) challenges, which were dedicated to apply state-of-the-art models to tackle today's problems in the usage of retinal images for diagnosis [2], [3]. In these challenges, the hosts provided the teams with high quality image segmentations, where the images were taken with different fundus cameras to mimic a real world setting.

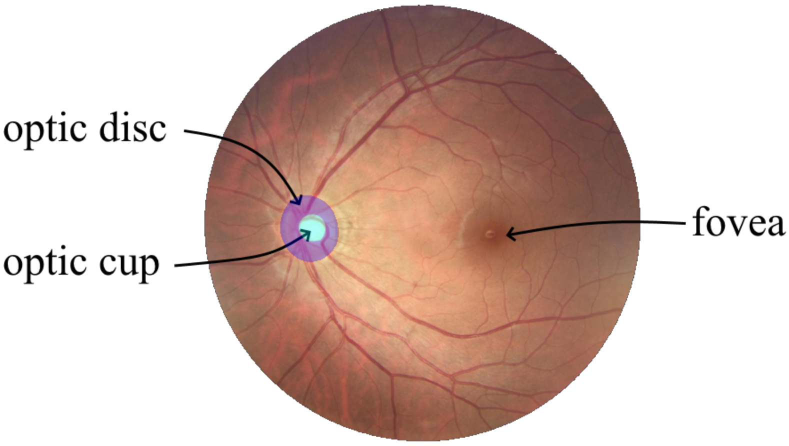

Figure 2.11: Example image, taken from the REFUGE dataset

The first challenge provided the twelve participating teams with 1200 images with segmentations, 400 each for training, validation and testing. See Figure 2.11 for an example, taken from this data. Equipped with this dataset, the teams were faced with two tasks: They should build a classifier to predict whether a person has glaucoma and a segmentation model to segment the optic disc and optic cup. In the scope of this Master's Thesis, only the segmentation models are of interest. The hosts came up with an evaluation metric to compare the quality of different segmentations of the same image. This metric consists of a weighted sum of the dice score on the optic disc and the optic cup and the mean absolute error of the vCDR, compared to the ground truth segmentation. All three best performing teams in the segmentation task used a two stage approach, where they first cropped the image to an area of interest (around the optic disc) and then applied some segmentation technique to get the final output. Most of the teams (14 out of 18) used architectures which build upon the original U-Net model (see Table 2.1).

For the second challenge, the hosts gave the teams the 1200 images from the first challenge as training data and 400 new images each for validation and testing, resulting in 2000 total images with segmentations. Additionally to the glaucoma classification and the segmentation of the optic disc and optic cup, the teams were faced with the problem of fovea localization. The fovea typically looks like a darker spot on retinal images (see Figure 2.11) and is home to cells, that are important for central vision [10]. For the segmentation task, the hosts used a similar metric to assess how good a segmentation is, again based on the dice score on the optic disc, the dice score on the optic cup and the mean absolute error of the vCDR. Even though the six teams who took part in the segmentation task were provided with three times more training data, the performance on their new test set is very comparable to the performance of the best teams of the first REFUGE challenge. This hints back at Ronneberger et al. who proposed U-Net and claim that U-Net is indeed working data efficient. Table 2.1 gives a broad overview over all models, which were used in both REFUGE challenges.

Team	**Idea**	**Models**	**Pre-processing**	**Post-processing**
AIML [2]	Two-step approach: ResNet-50 to identify the optic nerve head (ONH), cropping, ensemble for final prediction	ResNet-50, ResNet-101, ResNet-152, [13] ResNet-38 [14]	Augmentation using rescaling and rotation	Averaging across augmentations of the input and models
BUCT [2]	Two-step approach: U-Nets for optic disc (OD) and optic cup (OC) segmentation, OD segmentation used for cropping to help OC segmentation	U-Net [1]	Rescaling, cropping, converting to greyscale, augmentation using rotations and flippings	Taking largest connected component and ellipse fitting for OD segmentation
CUHKMED [2]	Two-step approach: U-Net to identify region of interest, ensemble of adversarially trained DeepLabv3+ architectures, including loss to enforce smooth border used for final prediction	U-Net [1], DeepLabv3+ [15] using backbone of MobileNetV2 [16]		Average across five models
Cvblab [2]	Two-step approach: U-Nets for OD and OC segmentation, OD segmentation used for cropping to help OC segmentation	U-Net [1]	Contrast Limited Adaptive Histogram Equalization, rescaling, using additional datasets for training	
Mammoth [2]	Ensemble of two-step approaches: both models first performed OD segmentation and then OC segmentation on cropped version	Mask-RCNN [17], Dense U-Net [1]	Division into training and validation set	Averaging probability outputs of both models

Masker [2]	Two-step approach: Mask-RCNN for optic nerve head (ONH) localization, cropping around optic nerve head (ONH), ensemble for final prediction	Mask-RCNN [17], U-Net [1], M-Net [18]	Cropped optic nerve head (ONH) images were divided using bagging principle and each subset got different pre-processing, including dehazing and edge-preserving multiscale decomposition	Voting
NightOwl [2]	Two-step approach: A dense U-Net localizes the optic nerve head (ONH) region, then another dense U-Net produces the final segmentation	Dense U-Net [19]	Histogram matching, exponential transformations, rescaling, augmentation to balance labels	Morphological operations, gaussian smoothing
NKSG [2]	Two-step approach: Extracting optic nerve head (ONH) area (method unspecified), DeepLabv3+ for segmentation	DeepLabv3+ [15]	Pixel-quantization	
SDSAIRC [2]	Two-step approach: M-Net for OD segmentation, cropping, then M-Net for OC segmentation	M-Net [18]	Polar transformation, histogram matching	Ellipse fitting
SMILEDeepDR [2]	Modified U-Net pretrained to predict label was fine-tuned to segment OD and OC	U-Net [1]		
VRT [2]	U-Nets for OD and OC segmentation, aided by an auxiliary CNN trained on vessel-segmentation masks	U-Net [1], auxiliary CNN [20]		Filling holes, convex hull
WinterFell [2]	Two-step approach: R-CNN to detect optic nerve head (ONH), cropping, ResU-Net for final segmentation	R-CNN [21], ResU-Net [22]	Normalizing to reference image, inversion of green channel	

cheeron [3]	Two-step approach: U-Net to segment OD, ResU-Net to segment OC	U-Net [1], ResU-Net [13]		
EyeStar [3]	Vision transformer with EfficientNet-B4 as backbone to find coarse feature maps which are put into the transformer	EfficientNet-B4 [23], Vision Transformer	Using additional datasets for training	
MAI [3]	Two-step approach: U-Net to segment OD, DeepLabv3+ using Test-Time Training strategy for final OD and OC prediction	U-Net [1], DeepLabv3+ [15]		
MIAG ULL [3]	Two-step approach: PSPNet to detect OD region, PSPNets for final OD and OC prediction	PSPNet [24] with ResNet50 [13] as encoder		
MIG [3]	Two-step approach: U-Net to segment OD, CE-Net for final OD and OC prediction	U-Net [1], CE-Net [25]	Using additional datasets for training	
VUNO EYE TEAM [3]	Two-branch network with whole image and vessel mask as input	U-Net [1] using EfficientNet-B0 [23] as encoder	Using additional datasets for training	

Table 2.1: Models used in both REFUGE challenges for the segmentation task, taken from their respective papers [2], [3].

Besides the REFUGE challenges, the task of segmenting the optic disc and optic cup via deep learning saw some interesting contributions. In 2018, Fu et al. presented M-Net, which can be seen as an extension of U-Net [18]. One of the key ideas behind U-Net and its success are the skip connections, which allows for high resolution inputs to be presented to the final layers. With M-Net, this idea is further explored, by scaling the original input down and feeding those smaller images to the corresponding encoding layers. Furthermore, the decoding path outputs are collected and considered in the final segmentation.

In a more recent paper by He et al., the anatomical relation between the fovea and the optic nerve head was utilized to obtain better results [26]. Like many of the teams on the REFUGE challenges, the authors used a two-step approach. First, the model produces a coarse localization of the optic disc and the fovea, which can foster each other since the distance of those two regions is similar for different retinal images. This coarse localization is used to crop the image, which is then fed into a U-Net-shaped architecture, with EfficientNet-B4 as encoder [23]. Additionally to the cropped retinal image, the output of the coarse segmentation is also used as an input for the fine segmentation [27]. The segmentation of the optic disc and especially the optic cup often comes with a high uncertainty. Small changes in the architecture, or a different choice for the initial weights, can alter the final segmentation. Wundram et al. proposed to actively use this uncertainty for downstream tasks. For the segmentation on fundus photographies, the most common downstream task is glaucoma classification, given the segmentation. They used non-deterministic models, which come up with different segmentations for different runs and used multiple samples to estimate the uncertainty in the final glaucoma classification [28].

In 2020, Bian et al. proposed the use of a generative adversarial network (GAN) approach to enhance the segmentation [29]. A typical GAN consists of two parts: a generator, which tries to learn a distribution and draw samples from it and a discriminator, that wants to distinguish real training examples from generated samples. The generator wants to deceive the discriminator by making realistic fakes, while the discriminator tries to be good at telling fakes from real examples. This leads to a game between the two players. Bian et al. implemented two cascaded U-Nets, one for the segmentation of the optic disc and one for the optic cup. The output of the optic disc is fed into the second U-Net, which outputs the segmentation for the optic cup. Both segmentation masks and the ground truth segmentations are then handed to a discriminator, which is trained to distinguish predictions from ground truth masks. This way, the cascaded U-Net model is additionally incentivized to produce results which look like correct segmentations. To mitigate the domain shift problem, Lei et al. proposed the use of a GAN-based approach [30]. Their goal was to make a model which is more robust to camera changes. This was done in an unsupervised setting, i.e., the model did not need the segmentations of a new, unseen camera, only the retinal images.

2.6 Incorporating Prior Knowledge

This section gives an overview over the methodology used in the literature to introduce prior knowledge, similar to the methods used in this Master's Thesis. The idea is that architectures like U-Net, equipped with one of the standard segmentation losses (cross-entropy, or dice loss), only draw their knowledge from the data provided. They aim to align their predictions with the ground truth and weight all mistakes equally. However, some mistakes are worse than others. Take for example the task of segmenting blood vessels in an image. The ideal model output looks like a bunch of quite narrow lines across the image. Assuming it is known that the image shows blood vessels which are all connected together, the model is supposed to output a segmentation which reflects this fact. But it does not "understand" that breaking a line is "worse" than making the blood vessels slightly too thick. For a model to be anatomically coherent, a medical professional should not be able to immediately tell whether a segmentation is nonsense. To compensate for this lack of awareness by the model, researchers try to incorporate prior knowledge into the models, for instance by introducing new loss terms. In the blood vessel example, a loss that increases when the output has many unconnected components could be introduced. El Jurdi et al. presented a survey in 2021, in which they investigate the current methods used to include prior knowledge in medical image segmentation [31].

2.6.1 Topological Loss

The term "topological" is inconsistently used throughout the literature on image segmentation. From a mathematical perspective, topology studies what objects share structural properties. When one asks a mathematician what he does when working in Topology, he will likely talk about the famous example of the mug and the doughnut, which are identical from a topological point of view. The reason for this is the fact, that it is possible to find a function which maps every point from the mug onto the doughnut, while preserving closeness within individual point-pairs. This is indeed possible, because they share a specific structure: They both have exactly one hole. This structural analysis can also be applied to images, for instance by looking at Betti numbers, which is an infinite sequence that encodes the topological structure of a binarized image (or much more complex objects). For example, the first Betti number describes the number of connected components, the second how many holes these components have, etc. It is possible to specify the desired Betti numbers and use a loss term to drive the models towards this specification [32]. But the Betti numbers only work on binarized images, and thus are non-trivial to use for training neural networks [33]. Hence, researchers looked at the theory of persistent homology, which suggest studying the evolution of shapes when varying the threshold applied to their prediction.

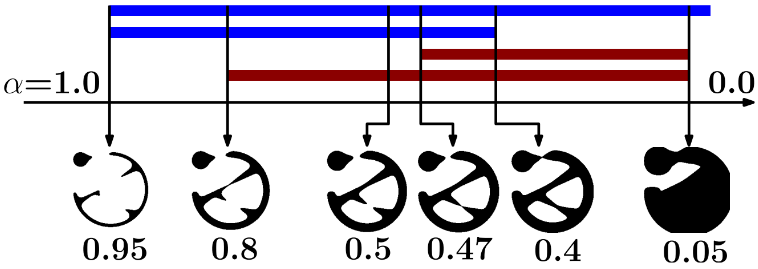

Figure 2.12: Barcode example [33]

These evolutions are usually drawn as barcodes (see Figure 2.12), where the x-axis refers to a threshold and each blue bar represents a connected component that emerges when applying the threshold on its left edge and merges with another on the right edge of the barcode. In Figure 2.12, each blue bar stands for the "lifespan" of a connected component. For example, at $\alpha = 0.4$ the second blue bar disappears, as the corresponding component is merged with the other component. The red bars tell how many holes are present in the thresholded image. For instance, at $\alpha = 0.8$ the lower component connects to itself, creating a ring, which therefore creates a hole. This hole "dies" at $\alpha = 0.05$, because it is filled in as the threshold decreases. There are several ideas on how to use these barcodes to force the model to produce topologically sound predictions [32]–[35]. Berger et al. expanded those ideas to multi-class segmentations in 2024 [36].

However, this Master's Thesis means something different when talking about topology. The structure, which is of interest to us, lives in the label relations, not the pixel relations. In 2014, Deng et al. proposed an image classification algorithm, where they utilize label structures [37]. In their toy example (see Figure 2.13), they introduce the reader to a classifier that can output "Husky", "Puppy", "Dog" and "Cat".

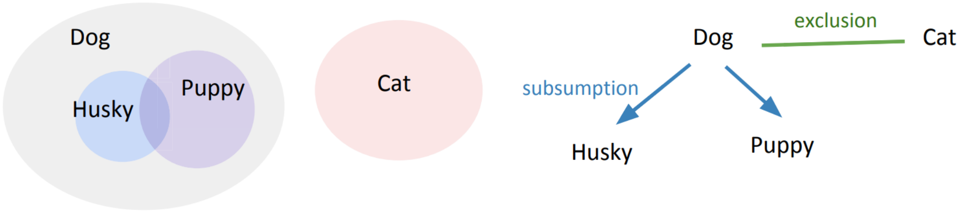

Figure 2.13: A toy example of a HEX graph [37]

A conventional classifier would output a score for each class, and the highest score wins. But due to the relations between labels, a more nuanced approach is needed: A "Husky" is always also a "Dog", the same holds for a "Puppy", while at the same time they all are mutually exclusive with "Cat". Deng et al. wanted a model that reflects the fact that a "Husky" is a "Dog", while still being able to only predict that it sees just a "Dog", without knowing its breed. Their solution employs HEX graphs (Hierarchy and Exclusion), as seen on the right side of Figure 2.13. To incorporate this knowledge represented by the HEX graph, they use a loss term that is told whether a combination of output labels is valid (for example "Puppy" and "Dog" can occur simultaneously, while "Puppy" and "Cat" is not a valid output).

This idea of defining a label hierarchy and including it in a model via a loss term, was later adopted for image segmentation. Examples are Bentaieb et al., or Reddy et al., who both had nested labels which were well suited for this kind of novel loss [38], [39]. The latter used a formulation in the form of

$$L_{Topo} = \sum_{p \in \Omega} \sum_{r=1}^{R} -y_p^r \cdot log(\widehat{y_p}^r) \cdot V(\widehat{y_p}) \tag{2.9}$$

where $p \in \Omega$ are all possible pixel indices for a given image, R is the number of classes, y_p^r is the ground truth label at a given pixel for a specific region r, $\widehat{y_p}^r$ is the outputted probability for the region r label at a given pixel, estimated by the model and V is the validity function. This validity function V is 0 for topological invalid predictions and 1 if the prediction follows the label hierarchy.

A different approach by He et al. includes a correction network [40]. They were faced with OCT images, which have a layer structure. Each layer is allowed to only touch neighbouring layers, building a label relation similar to retinal images (optic cup should not touch background). They trained a segmentation network and an additional correction network, which takes a segmentation that does not fulfil the topology and tries to correct it.

In 2022, Gupta et al. proposed to add a loss term, which focuses on topological critical pixels [41]. In their approach, they first identify all pixels which are currently violating a given topology and then add a pixelwise loss term which is masked to only include these critical pixels.

2.6.2 Polar Transformation

Besides using loss terms, the data can be transformed into a different domain to incorporate prior knowledge. For example, raw audio data is just a stream of numbers, which makes it tough to draw insights from this kind of data. It is often useful to change the perspective to get a better idea of the nature of given data. For audio data, researchers and engineers would look at the frequency domain of a signal to find out what frequency parts the signal consists of. This is such a powerful view on the data, that filters are even designed in the frequency domain, as it is much easier to describe the desired filter via its frequency behaviour. Mathematics provides us with a variety of different transformations, which each stands for a different view on the data. Finding a suitable transformation can make a given problem much easier to solve. For the task of segmenting the optic disc and optic cup in retinal images, the literature suggests to look at the polar domain, i.e. applying a polar transformation to the images. This changes the coordinate system the images live in from Cartesian to a radius-angle system.

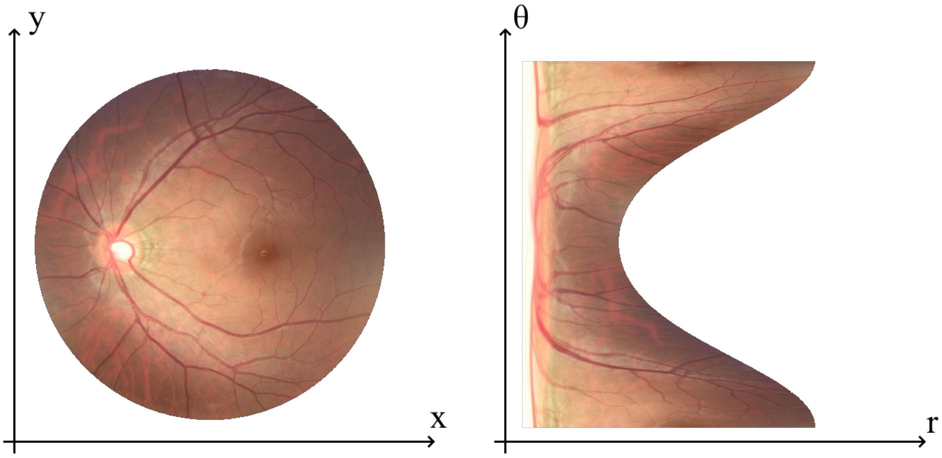

Figure 2.14: Cartesian and polar variants of the same image

This transformation converts the problem of finding an elliptic shaped object, or objects, into finding vertical boundaries. However, to perform the transformation the way Figure 2.14 shows, it is necessary to have the Cartesian coordinates of a point in the optic cup (ideally as close to the centre as possible). This centre point could be found by a neural network, which is trained to locate the centre pixel.

This Thesis assumes that such a neural network exists and uses the ground truth centre of mass of the optic cup as the centre point. By feeding the coordinates of this centre point within the optic cup to the polar transformation, the process becomes translation-invariant, as shifting the image in the Cartesian domain changes the coordinates of the point as well. Additionally, it becomes rotation-equivariant, since a rotation (around the chosen centre point) is reflected by a translation in the polar coordinate system along the angle-axis. The transformation also allows deep learning architectures to focus on a specific region, as all points of interest will have a low radius coordinate. The exact position of the optic disc varies in the Cartesian system. This fact would also allow some cropping, as a huge portion of the polar image only includes background pixels.

In 2017, Zahoor et al. used the polar transformation together with classical image processing and segmentation techniques for an algorithm to segment the optic disc, without the use of deep learning [42]. From that point on, the polar transformation was frequently used for the problem of segmenting the optic disc and optic cup, also together with deep learning approaches. This extends to other tasks, which include finding circular objects [18], [43], [44]. Usually, the segmentation is done in only one domain, however, Liu et al. decided to use both, the Cartesian and polar images, as input for their algorithm [44]. This theoretically allows the model to profit from the benefits of both domains.

2.6.3 Layer segmentation with topological guarantees

This subsection describes a paper by He et al. [45]. They introduce a novel method for layer segmentation of optical coherence tomography (OCT) images. These 3D images show the layers of the retina and, like other retinal images, help to diagnose eye-related diseases. Their goal is to segment the layers, while preserving the topological order, i.e., the order of these layers should be anatomically correct in the segmentation. They argue that conventional deep learning architectures, like U-Net, lack awareness for this topology in their prediction. To overcome this problem, they propose to add a second output path to an architecture like U-Net, which does not segment each layer, but rather predicts where the boundaries of each consecutive layer-pair lie. There are nine of these boundaries, hence, they use a nine channel output for this path. To convert the raw output to a proper probability, a softmax is applied to each column. This way, for each column, each row gets a certain probability for every boundary type. He et al. denote these probabilities by q_i in Figure 2.15, where i is an index to indicate the channel, or boundary type. Since for each column q_i represents a probability distribution (due to the column-wise softmax), they propose to calculate the expected value $\widehat{x_i}$ of these distributions. These expectational values are then gathered in $s_1, ..., s_L$, where L stands for the number of layers. Hence, s_i contains a vector, which includes the position of boundary type i in column j. To ensure the topological structure, i.e., $s_1(j) \leq s_2(j) \leq \ldots \leq s_L(j)$ must hold for all columns j,

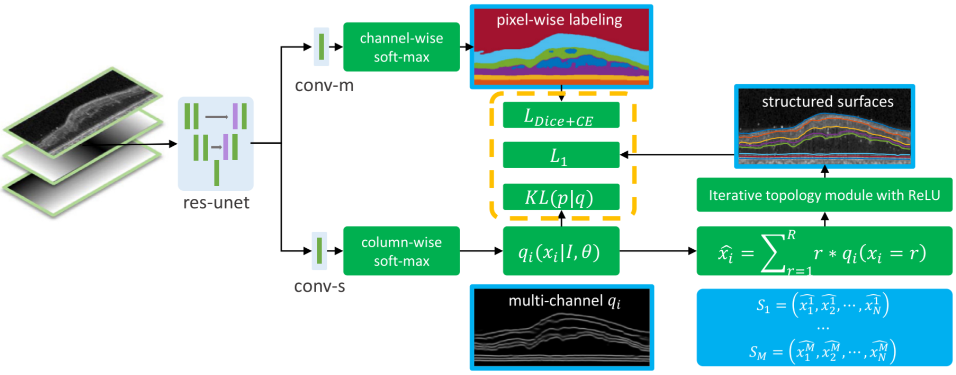
pixel-wise labeling
channel-wise soft-max
conv-m
res-unet
structured surfaces
$L_{Dice+CE}$
L_1
$KL(p|q)$
Iterative topology module with ReLU
column-wise soft-max
$q_i(x_i|I,\theta)$
$\hat{x_i} = \sum_{r=1}^{R} r * q_i(x_i = r)$
conv-s
multi-channel q_i
$S_1 = (\widehat{x_1^1}, \widehat{x_2^1}, \cdots, \widehat{x_N^1})$
...
$S_M = (\widehat{x_1^M}, \widehat{x_2^M}, \cdots, \widehat{x_N^M})$

Figure 2.15: Proposed method by He et al. for OCT layer segmentation. Image taken from original paper [45]

.

they apply an “iterative topology module with ReLU” (see Figure 2.15). This module applies the formula

$$s_i(j) = s_{i-1}(j) + ReLU(s_i(j) - s_{i-1}(j)) \tag{2.10}$$

which guarantees that each layer is predicted at least in the same row as the layer beneath. These updated expectations in $s_1, ..., s_L$ are then the final second output of the model. To train their model, they apply three losses:

- The sum of Dice loss and cross-entropy loss $L_{Dice} + L_{CE}$, applied to the segmentation output path
- Kullback Leiber Divergence applied to the probability distributions q_i to come as close to the real distributions (single point of mass in the row where the actual boundary lies, for each column) as possible
- A simple distance measure L_1 loss for the expected boundary row index $s_i(j)$ for each column j and boundary type i, after the topology module from Equation (2.10) is applied

He et al. suggest including index information to the input by concatenating two matrices, which together hold the coordinates of each pixel. They use the matrices X, Y from below,

$$X = \frac{1}{m}\begin{pmatrix} 1-\frac{m}{2} & 2-\frac{m}{2} & \dots & m-\frac{m}{2} \\ 1-\frac{m}{2} & 2-\frac{m}{2} & \dots & m-\frac{m}{2} \\ \vdots & \vdots & \ddots & \vdots \\ 1-\frac{m}{2} & 2-\frac{m}{2} & \dots & m-\frac{m}{2} \end{pmatrix} \quad Y = \frac{1}{n}\begin{pmatrix} 1 & 1 & \dots & 1 \\ 2 & 2 & \dots & 2 \\ \vdots & \vdots & \ddots & \vdots \\ n & n & \dots & n \end{pmatrix} \tag{2.11}$$

where m and n indicate the number of columns and rows respectively.

3 Methodology

This chapter explains in detail what models were implemented and evaluated for this Thesis. The baseline model is the starting point for all other models, hence it is described most extensively. Sections which are related to other models focus on the differences compared to the baseline model. These modifications each stand for an anatomical prior which is incorporated into the model. The goal of using these priors is to improve the segmentation quality, compared to the baseline model, when only few samples are used for training. Therefore, all models explained in this Chapter are evaluated in terms of several quality measures in Chapter 4 using plots, which allow the comparison of models trained with different amounts of data. Before diving into the implementation details of the models themselves, all data sources are briefly described to give a notion of the problem at hand. The code to replicate all mentioned models can be found here: `https://github.com/mo1333/anatomically_coherent_segmentation`

3.1 Data

A big part of this Thesis consists of measuring how well a model performs when only trained with small amounts of data. To simulate the availability of only few training samples, each experiment draws a random subsample of the training split of the REFUGE2 challenge. The validation and test splits are not decreased in size to keep a maximum of expressiveness. Additionally to the REFUGE2 dataset, the Chákṣu dataset is used for testing (not training). This allows the models to be comparable to many state-of-the-art architectures, as most of the papers regarding the segmentation of the optic disc and optic cup use at least one of the following datasets for evaluation.

3.1.1 REFUGE

The dataset used in the REFUGE2 challenge [3] is publicly available and was the main dataset used for this Master's Thesis. It can be downloaded on the main website of the REFUGE2 challenge here[1]. All 2000 fundus images in this dataset come from Chinese citizens. The dataset is already split into pre-specified training, validation and test sets. Since the REFUGE1 challenge was already finished when the REFUGE2 challenge started, the hosts decided to use all data from the REFUGE1 challenge as training data for the REFUGE2 challenge. The training set consists of 1200 images, taken with a Zeiss Visucam 500 or a Canon CR-2 camera. For the validation set, 400 images were taken by a KOWA device. The final test consists of additional 400 images, taken with a TOPCON

[1] `https://refuge.grand-challenge.org/`

TRC-NW400 camera. The training set includes 120 (that accounts for 10%) images from patients suffering from glaucoma. However, the validation and test set both have 80 (accounting for 20%) images of patients with glaucoma. As Figure 3.1 shows, different sets and therefore cameras look slighlty different. Having images taken from different cameras, and therefore distributions, mimics a real world setting, where a model should perform well with a big variety of possibly unknown cameras.

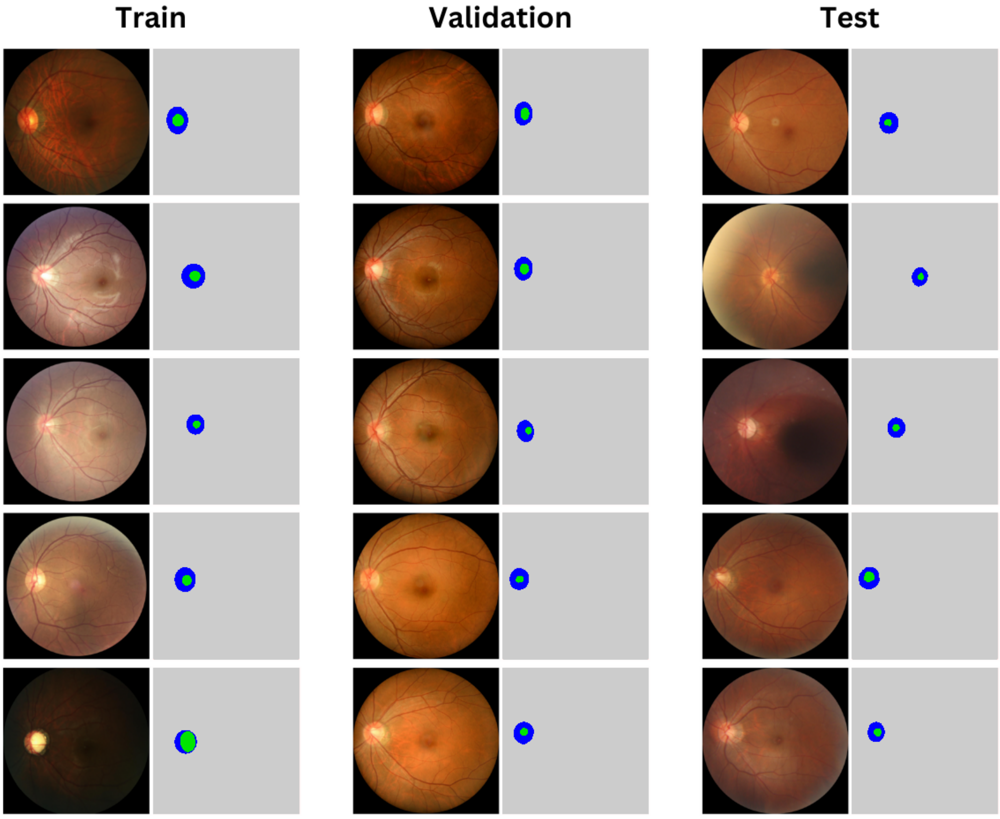

Figure 3.1: Random samples of train, validation and test split, taken from REFUGE2 challenge dataset, showing retinal image (left) and segmentation of optic disc and optic cup (right)

The segmentations are originally represented by 1-channel images, where the background is marked by 0's, the optic cup by 2's and the optic disc without the optic cup by 1's. Two pre-processing steps are taken:

1. The segmentations are converted into 3-channel images, where the first channel corresponds to the background, the second to the optic cup and the third to the optic disc (including the cup).
2. Both, the images and the segmentations, are rescaled to 512 × 512 pixels. This rescaling is done by interpolating the original pixels in the desired grid.

Figure 3.1 shows how segmentations after the pre-processing steps look like: They are 3-channel images, with each channel represented by a different color. The first channel (gray) depicts the background, the second the optic cup (yellow) and the third the optic disc (blue). Again it can be noticed that the inner circle, visualizing the segmentation of the optic cup, is not purely yellow, but also blue, since the optic cup is part of the optic disc. Hence, both channels (blue and yellow mix to green) are active where the optic cup is present, resulting in an overlay.

3.1.2 Chákṣu

As an addition to the REFUGE2 dataset, the Chákṣu dataset [46] is used to evaluate and compare different models. The dataset consists of 1345 fundus images of Indian citizens and, in contrast to the REFUGE2 dataset, includes multiple ground truth segmentations. It can be downloaded here[2]. Images in this dataset were taken by three different cameras: the Remidio non-mydriatic Fundus-on-phone camera (1074 instances), the Forus 3Nethra Classic fundus camera (126 instances) and a handheld Bosch fundus camera (145 instances). Each retinal image was segmented and classified by five different experts. Furthermore, Kumar et al. used four techniques to combine the expert segmentations: Mean, Median, Majority, and “Simultaneous Truth and Performance Level Estimation” (STAPLE). The STAPLE algorithm, as introduced in 2004 by Warfield et al.[47], is an iterative algorithm to weight the segmentations of the experts in order to come close to the true segmentation. The authors of the paper that introduces the Chákṣu dataset suggest, that STAPLE has the highest agreement with all experts simultaneously. Therefore, the STAPLE segmentations are used as the ground truth segmentations for the purpose of this Thesis [46].

The images that make up the Chákṣu dataset were not processed to look alike (see Figure 3.2), even though three distinct cameras were used. Note that the images differ in many subtle ways: They have different cropping, scaling, stretching and occasional text within the image. The cropping used for the Forus cameras hides part of the retina, even though the optic disc and optic cup are always visible. Similar to the REFUGE2 dataset, Chákṣu comes with a pre-defined split into training and test set, however this split is eliminated for the purpose of evaluation, since both splits are considered as an additional test set for this Thesis. The Chákṣu dataset is never used for training, only for testing models which are trained using REFUGE2 data. For the Chákṣu dataset, three pre-processing steps are necessary:

1. Central cropping by taking the biggest central square which still fits into the image, performed on images and segmentations.
2. Splitting the segmentation into a 3-channel image, like the REFUGE2 data.
3. Rescaling to 512 × 512 pixels, like the REFUGE2 data.

[2] `https://figshare.com/articles/dataset/Ch_k_u_A_glaucoma_specific_fundus_image_database/20123135`

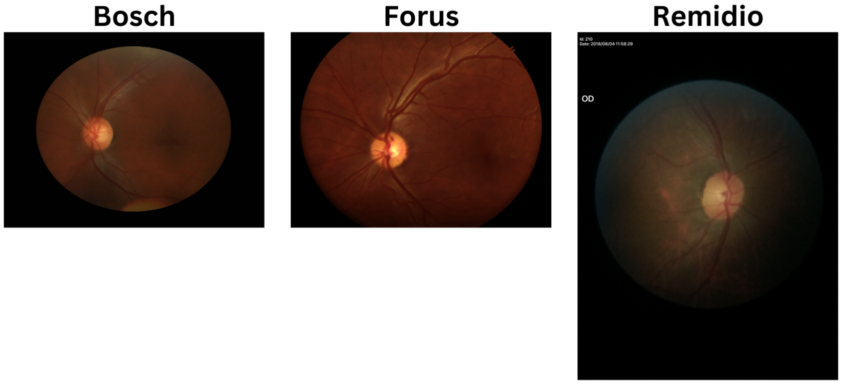

Figure 3.2: Different camera types used in the Chákṣu dataset

It was considered to change the order of rescaling and cropping for the Bosch images, as they have a horizontal stretch, however the results did not look more promising than using the mentioned order for the pre-processing steps. Even after pre-processing, some text parts are present in the Remidio images, but these fragments are not removed in order to investigate how different models treat these visual anomalies. Furthermore, the general appearance of the images does not match the training data (REFUGE2) in terms of colouring, cropping, stretching and position of the optic disc which is an interesting contrast, especially since the training data is very homogenous. These differences are seen as an additional challenge for the models and will show how resilient architectures are to drastic shifts in the distribution.

3.2 Models

The following sections describe the different models which are implemented and evaluated. It starts with a detailed analysis of the baseline, followed by four modifications. The goal of modifying the baseline model is to include prior, anatomical knowledge about the problem into the model, in order to make it perform better when only presented with small amounts of data.

3.2.1 Baseline

The baseline model is a simple U-Net, as introduced by Ronnerberger et al. [1] and explained in Chapter 2.4. A MONAI (Medical open network for AI) implementation of U-Net is used. MONAI is a Python library, which provides several state-of-the-art deep learning models which are specialised for the medical domain. Many inspirations on how to use MONAI's U-Net are drawn from this Tutorial[3]. The MONAI implementation differs in some ways from the original U-Net architecture, proposed by Ronneberger et al. [1]. For instance, in the downsampling path, the actual downsampling is not done via max pooling layers, but rather by striding in the convolution layer. For this Thesis, the striding value is set to 2 for all cases where downsampling is performed, which cuts the count of positions the convolution kernel attains in half. Hence, the size (in each dimension) of the latent image after the convolution layer is half the size of the input image. In the original U-Net paper, each encoding block consists of two convolutional layers (see Figure 2.10), both equipped with an activation function, followed by a max pooling operation. However, the MONAI implementation is more refined and uses residual units (see Figure 3.3). The details of these residual units are outlined in the following paragraphs.

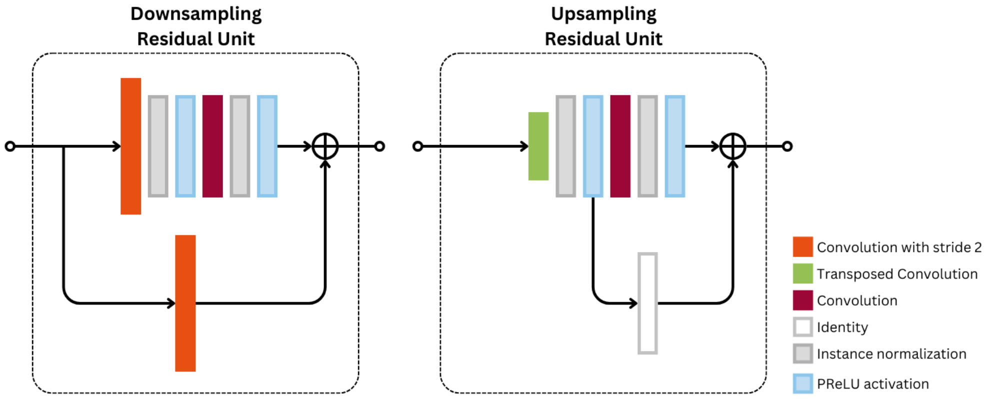

Figure 3.3: Residual Units

In the downsampling path (indicated by orange arrows in Figure 3.4) of MONAI's U-Net are the "Downsampling Residual Units". As Figure 3.3 shows, they consist of six blocks in their main path and a single block in their residual path. The unit first uses a strided convolution in the main path to decrease the size of the image, followed by a normalization layer and an activation layer. After this activation, another convolutional layer with a standard kernel size of (3×3) and zero-padding is in place to keep the same size. This convolution is again followed by normalization and activation layers, which concludes the main path. Additionally, there is a residual path within the Unit, which takes the input and applies whatever operation is necessary to get to the same size and number of channels as the output of the last activation in the main path. Since the unit downsamples the data using the strided convolution, it has to apply a strided convolution as well in the residual path, to decrease the size of the image, while possibly changing the number of channels. The output of the residual path is then simply added to the output of the last activation layer to produce the final output of the Unit.

[3] `https://github.com/Project-MONAI/tutorials/blob/main/3d_segmentation/unet_segmentation_3d_ignite.ipynb`

The upsampling path (indicated by green arrows in Figure 3.4) consists of “Upsampling Residual Units”, which work similar to their downsampling counterparts. They again have six blocks in their main path and use transposed convolutions (green block in Figure 3.3) to increase the size of the image, instead of the strided convolution in the “Downsampling Residual Units”. For the residual path, the output of the first activation layer in the main path is used, instead of the original input. This eliminates the need for alteration of the input into the residual path, compared to the “Downsampling Residual Unit”, since the size and number of channels is not changed after the first activation layer. Therefore, this residual path reduces to a skip connection, which skips the normal convolutional layer.

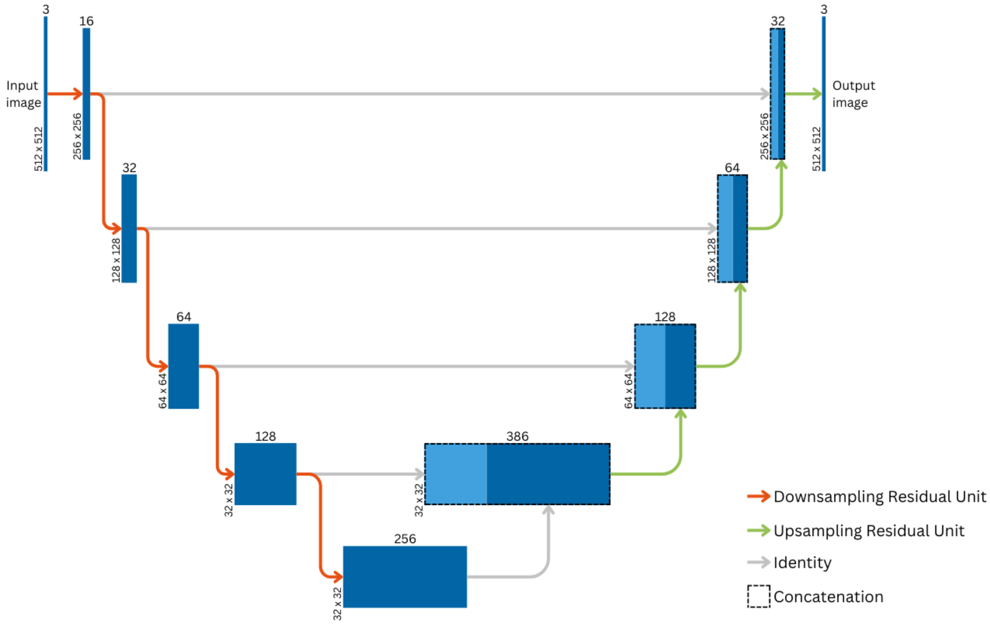

Figure 3.4: MONAI implementation of U-Net, marked with number of channels and image sizes used for this Thesis

To build the baseline architecture, a number of “Downsampling Residual Units” are chained, followed by “Upsampling Residual Units”. In the upsampling path, the output of the related downsampling step is used as additional input via concatanation (related in the sense that they have the correct output size). In Figure 3.4 this adopted U-Net architecture is depicted, using a similar illustration as in the original paper by Ronneberger et al. [1] (see Figure 2.10). Each blue block represents a latent space, with arrows connecting them to indicate what happens in each step.

Up to this point, only the architecture itself is discussed, however the parameters still need to be tuned. To train the baseline architecture for the task of segmenting the optic disc and optic cup, MONAI’s “DiceCELoss” and PyTorch’s “Adam” optimizer with a learning rate of 0.001 are used. As the name implies, the “DiceCELoss” breaks down into a weighted sum of cross-entropy and Dice loss (see definitions 2.6 and 2.8 respectively).

For the baseline model, these losses are equally weighted with 1. The output of U-Net is put through a sigmoid function, to ensure that each pixel has only values between 0 and 1. It is worth noting that a softmax function would probably be more common in the literature for this task. However, it makes a prediction of two classes at the same pixel tricky, since the channel-wise sum of the output at each pixel would be 1. For the task at hand, the model is supposed to predict the class "disc" and "cup" wherever the cup is present in the image. Hence, we would like the model to predict high values for "disc" and "cup", but a softmax function would rescale them. For example, if both classes are predicted to be 1, while the background is 0, the output probabilities would become 0.5 when applying the softmax, which is undesired. Instead, a sigmoid function is used, which operates pixelwise and does not look at different channels for its computation, making it suitable for the task.

In order to produce the final segmentation output of the model, which is necessary for evaluating the quality of the model performance, the probability output (after applying the sigmoid function) is thresholded. This is done by simply choosing a threshold value T and outputting 0 where the probabilities are smaller than T and 1 otherwise. The thresholds are searched as the last step of the training pipeline by stepping through 0 up to 1 in 100 steps and calculating the Dice score (see Definition 2.7) on the validation set for each threshold candidate. This process is done for the optic disc and optic cup in two separate steps, thus yielding two thresholds, one for each channel of interest. Thresholding gives an image which should be very similar to the ground truth segmentation. Details on how this output is used for evaluation can be found in Chapter 4.1.1.

3.2.2 Novel Topological Loss

The first modification introduces a new loss term to the baseline model, called topological loss. With a topological loss, label relations can be described, and the model is driven towards producing topologically sound outputs. This Thesis focuses on the topological relation of optic cup and optic disc, namely that the optic cup is inside the optic disc. Hence, the model should never predict an optic cup, where it does not predict the optic disc. Some losses which try to accomplish this goal can be found in the literature, like Reddy et al. [39] (see Definition 2.9), however their loss formulation does not lead to more sound predictions. Instead, a novel topological loss was developed to capture the inclusion of optic cup within the optic disc.

This new loss underwent a long evolution, starting from a naive idea: Punish pixels which predict a higher probability for the optic cup than for the optic disc. This loss can be easily implemented and mathematically described by the following formula:

$$L_{Topo} = \sum_{p \in \Omega} ReLU(\widehat{y_p}^{OC} - \widehat{y_p}^{OD}) \tag{3.1}$$

where $p \in \Omega$ are all pixel positions, $\widehat{y_p}^{OC}$ and $\widehat{y_p}^{OD}$ are the predicted probabilities of the optic cup and optic disc respectively and $ReLU$ is the ReLU function, which clips its input to be non-negative. The loss in Definition (3.1) is 0 when the probability of the optic cup is lower than the probability of the optic disc at all pixels. Therefore, it captures the inclusion property which is desired. However, it can impede the training process, as it can punish pixels which correctly predict an optic cup, whilst the prediction of the optic disc is not yet present. Especially in early training phases, this loss impedes the learning process.

The actual topological loss, which is used for this Thesis and the final experiments, looks a bit more refined:

$$L_{Topo} = \sum_{p \in \Omega} -\log\bigl(1 - ReLU(\widehat{y_p}^{OC} - \lfloor \widehat{y_p}^{OD} + 0.5 \rfloor - 0.5)\bigr) \tag{3.2}$$

This definition follows the same notation as Definition (3.1). The expression $ReLU(\widehat{y_p}^{OC} - \lfloor \widehat{y_p}^{OD} + 0.5 \rfloor - 0.5)$ at the heart of this loss is 0 for all pixels p where the probability of the optic disc is greater than 0.5, or the probability of the optic cup is less than 0.5, or both. For troublesome pixels, where the optic cup has a probability greater than 0.5, whilst the optic disc has a lower probability, the $ReLU$ expression will simplify to $\widehat{y_p}^{OC} - 0.5$, hence measuring how much the optic cup exceeds its threshold. As this grows larger, the input to the logarithm gets smaller, which therefore rapidly makes the logarithm go towards negative infinity. The loss weight λ_{Topo} is set to 10 for the final experiments. This hyperparameter is chosen empirically by doing a gird-like search and comparing results on the validation set. Bigger λ_{Topo} lead to a decrease of the mean Dice score on the validation set of the model trained on few training samples, while smaller λ_{Topo} lead to no apparent differences to the baseline.

3.2.3 Polar Transformation

Another way to incorporate prior knowledge, besides adding a loss term, is to transform the data into a different domain, where the segmentation becomes easier. The basics of the polar transformation are already discussed in Subsection 2.6.2. Since the optic disc and optic cup are quite circular, applying a polar transformation with the centre of the optic cup as transformation centre results in approximately rectangular shapes. The assumption is that this, together with the fact that the position of optic disc and optic cup get all mapped to the same locations in polar coordinates (see Figure 2.14), helps the model to learn. Additionally, areas close to the transformation centre are deformed to become larger, which makes the area of interest more present in the polar domain [18].

The transformation itself is done only once, to avoid unnecessary repetition of costly computations during the training. For the actual computation the library "polarTransform"[4] is used, which allows storing the settings of a transformation, making the back-transformation easier. Hence, each dataset has a file containing settings, which can be used for back-transformation. Both, images and segmentations, are transformed with the same settings. To compute where a pixel with Cartesian coordinate (x, y) ends up after the transformation, the following formula can be employed:

[4] `https://github.com/addisonElliott/polarTransform`

$$\begin{aligned} r &= \sqrt{(x - x_0)^2 + (y - y_0)^2} \\ \theta &= \operatorname{atan2}(y - y_0, x - x_0) \end{aligned} \tag{3.3}$$

where atan2 is the argument function, which corrects the arctan to be correct in all quadrants of the plane and (x_0, y_0) is the centre point of the transformation. For this Thesis, the centre point is chosen to be the centre of mass of the optic cup, i.e. the average of all indices at active optic cup pixels. This approach is not applicable in a real world setting, as the exact pixel positions of the optic cup are unknown for a new retinal image. Otherwise, there would be no need to segment it. It is assumed that a model exists, which is good at finding the centre of the optic cup. This model is simulated by taking the exact centre. Analogous to equation (3.3), the following formula tells how to transform polar coordinates into Cartesian coordinates:

$$\begin{aligned} x &= r \cdot \cos(\theta) + x_0 \\ y &= r \cdot \sin(\theta) + y_0 \end{aligned} \tag{3.4}$$

The final evaluation of a model trained on polar images is done in Cartesian coordinates to allow for a more interpretable comparison. It is worth noting that transforming images back and forth leads to small errors each time. Therefore, it is necessary to assume that these errors do not harm the meaningfulness of the results. These small errors are for instance inevitable when transforming the segmentation. Each channel consists only of 0's and 1's in the Cartesian domain, however, applying the transformation necessarily employs interpolation, which leads to ambiguity. The segmentations in the polar domain are thresholded at 0.5 to make them binarized again.

Models, which are trained on polar data, have an advantage over other models: the back-transformation includes the ground truth centre position of the optic cup (centre at (x_0, y_0)). Hence, the model prediction gets automatically mapped to the correct position.

3.2.4 Novel CDR Loss

The vertical cup-to-disc-ratio (vCDR) is an important attribute of retinal images. Medical experts use the vCDR as a tool to diagnose glaucoma. Therefore, it is necessary to get the vCDR of the segmentation as close to the correct vCDR as possible. In the REFUGE challenge [2], the mean absolute error of the vCDR of the prediction compared to the ground truth is one of the properties which are used to assess how good a model is. Therefore, the idea of using a loss which actively tries to minimize this error comes naturally. After investigating what problems the baseline has when faced with only a few training samples, it becomes clear that the model often first learns how to predict the optic disc correctly, whilst the optic cup is learnt later on (see Section 4.2.1). This is observable at the channel responsible for the optic cup: The model sometimes primarily focuses on the optic disc and only when equipped with adequate amount of epochs or training samples, the optic cup becomes more refined. This behaviour is actually intuitive, since the optic disc is much more obvious and homogenous in appearance. Furthermore, predicting 1's for the optic cup where the optic disc is present yields reasonably good results, as it is a valid first approximation.

Defining the CDR loss comes with some problems, as the vCDR is only properly defined for binarized images. However, the predictions consist of probabilities, which make it harder to tell the diameter of the optic disc in the prediction. Additionally, the computation of the diameter is non-trivial to implement differentiable, as changes of each pixel can affect the diameter either severely, or not at all. Take for example a single pixel at the centre of the optic disc. It suddenly changes from 1, like all its neighbours, to 0, making it an outlier. Has the diameter of the disc changed? On the other hand, a pixel which is far off the optic disc, which changes from 0 to 1 has the reverse problem. How should such changes be regarded?

To circumvent these differentiability problems, the loss is implemented to be intrinsically dynamic with its weight. This means that the model adjusts how much the loss should add to the total loss on the fly, depending on the vCDR. To compute the diameters in the prediction, the probabilities are thresholded at 0.5. In this binarized image, the vertical pixel diameter is computed by just subtracting the indices of the last and first row, which include a 1. The pixel diameter of the optic cup is then divided by the pixel diameter of the optic disc to get the vCDR. The squared difference of the ground truth vCDR and the predicted vCDR is then used as dynamic weight. It is worth noting that the computation of the vCDR is not differentiable, but the loss term has to have a part which can be used to update the parameters of the model. Next, the differentiable part of the vCDR loss is explained.

The loss either tries to decrease the size of the optic cup when the vCDR is too high (compared to the ground truth), or tries to increase the size of the optic cup when the vCDR is too small. This can be achieved by penalizing all optic cup predictions which are close to 1 when the vCDR is too big and vice versa penalize probabilities which are close to 0 when the vCDR is too small. For training samples, where the predicted vCDR is too big, the loss comes down to:

$$L_{CDR} = \delta_{vCDR}^2 \sum_{p \in \Omega} -\frac{1}{2}(\widehat{y_p}^{OC})^2 + \widehat{y_p}^{OC} + 0.05 \tag{3.5}$$

where δ_{vCDR}^2 is the squared difference of ground truth vCDR and predicted vCDR, $p \in \Omega$ are all pixel positions in the image and $\widehat{y_p}^{OC}$ is the predicted probability of the optic cup at pixel p. For training samples where the predicted vCDR is too small, the formula above is reversed, to encourage the optic cup to grow in size:

$$\begin{aligned} L_{CDR} &= \delta_{vCDR}^2 \sum_{p \in \Omega} -\frac{1}{2}(1 - \widehat{y_p}^{OC})^2 + (1 - \widehat{y_p}^{OC}) + 0.05 \\ &= \delta_{vCDR}^2 \sum_{p \in \Omega} -\frac{1}{2}(\widehat{y_p}^{OC})^2 + 0.55 \end{aligned} \tag{3.6}$$

with the same notation as in Definition (3.5). Note that the loss in Definition (3.5) can be minimized by making $\widehat{y_p}^{OC}$ smaller, whilst the loss in Definition (3.6) will be smallest when $\widehat{y_p}^{OC}$ approaches 1. This ensures that the optic cup either grows or shrinks, depending on whether the vCDR is too small or too big. In both cases, δ^2_{vCDR} should decrease, which corresponds to correct vertical diameters in the prediction. Additionally to the dynamic weight, a static hyperparameter λ_{CDR} is used and empirically chosen to be 10.

3.2.5 Enforcing Anatomical Coherence on Polar Data

The model presented in this subsection is based on a paper by He et al. [45] A technical introduction to their model is already given in Subsection 2.6.3, this subsection focuses on the application for segmenting the optic disc and optic cup. The model by He et al. was originally developed to find layers in an optical coherence tomography (OCT) image, which are important for detecting all kinds of eye-related diseases. The authors did not name their model, therefore it will be called ETUS (**E**nforcing **T**opology using **U**-Net **S**egmentation) in this Thesis. The task of segmenting the optic disc and optic cup in the polar domain can be interpreted as searching for layers and their boundaries, which is exactly what ETUS was designed to do. In contrast to all other models discussed so far, ETUS produces more than just a single output. It has a segmentation output, which is used to train the U-Net part, and an output that predicts where the boundaries of the different classes lie. The latter output has two channels, one for the boundary between optic cup and optic disc, and another for the boundary between optic disc and background. Having only two channels marks the biggest difference to the model presented by He et al., as they used nine boundary types instead of two. Figure 3.6 shows how an ideal output of the second "bounday" output looks like. Note that Figure 3.5 (which shows the ETUS architecture) mainly differs from 3.4 (which depicts the baseline U-Net model) by the additional output path.

The real power of ETUS lies in the way the boundary output is treated. Instead of a channel-wise softmax (which is applied to the segmentation output), a row-wise softmax is applied to the boundary output. This converts the output to a bunch of probability distributions, one for each row, which estimate the ground truth boundary (see Figure 3.6 "Layer Boundaries" and Figure 3.7 for reference). The architecture desires a high probability where the boundary actually occurs and a probability of zero everywhere else. In order to tell how close the predictions of these distributions are to the actual distributions, the Kullback–Leibler divergence is used and turned into a loss. This loss is defined as

$$L_{KL} = \sum_{p \in \Omega} q_p \cdot \log \left(\frac{q_p}{\widehat{q_p}} \right) \tag{3.7}$$

where $p \in \Omega$ are all pixel positions in the image, q_p is the ground truth boundary distribution at pixel p and $\widehat{q_p}$ is the predicted boundary distribution at pixel p, i.e., the "Output boundaries" in Figure 3.5. Figure 3.7 shows an example of two ground truth

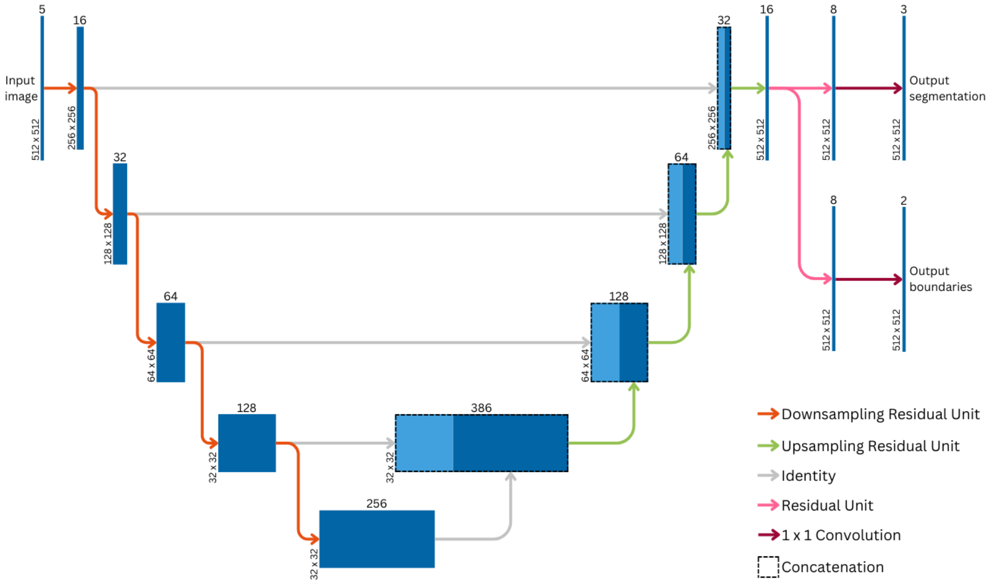

Figure 3.5: ETUS architecture

distributions q_p (one for each channel and thereby boundary), which are approximated by $\widehat{q_p}$. Since they show the distributions for a single row, they can be interpreted as a horizontal slice of the "Layer Boundaries" image from Figure 3.6. The approximations $\widehat{q_p}$ are pushed toward q_p by the Kullback–Leibler divergence. Since a row-wise softmax is applied to obtain $\widehat{q_p}$, the Kullback–Leibler divergence from Definition (3.7) will try to increase $\widehat{q_p}$ where $q_p = 1$ in each row separately. Due to the softmax, this automatically decreases $\widehat{q_p}$ where $q_p = 0$, thus the output of the model gets closer to q_p, which is desired. Furthermore, the expected values (s_{OC} and s_{OD} for the optic cup and optic disc respectively) are computed from these row-wise distributions, which tell for each row in what column the model "expects" the boundary to lie. These expectations are then fed into the topological module, which iterates over the channels and makes sure that in each row, the boundary between optic cup and optic disc occurs left of the boundary between of optic disc and background. The topology is assured by the following formula:

$$\begin{aligned} s^*_{OD}(i) &= s_{OD}(i) \\ s^*_{OC}(i) &= s^*_{OD}(i) + ReLU(s_{OC}(i) - s^*_{OD}(i)), \quad \text{for } i = 1, ..., n \end{aligned} \tag{3.8}$$

where $s_{OC}(i)$ and $s_{OD}(i)$ are the expected boundaries for the optic cup and optic disc in row i respectively, s^*_{OC} and s^*_{OD} contain the updated expectations for the optic cup and optic disc, n stands for the number of rows and $ReLU$ stands for the ReLU function which clips its input to be non-negative. Using this correcting mechanisms for the expected boundary of the optic cup, ensures that $s^*_{OC}(i) \leq s^*_{OD}(i)$ will hold, which is known to be true for retinal images. By applying this topology module, the topology is enforced on the output, meaning that the model is not capable of predicting an optic cup where

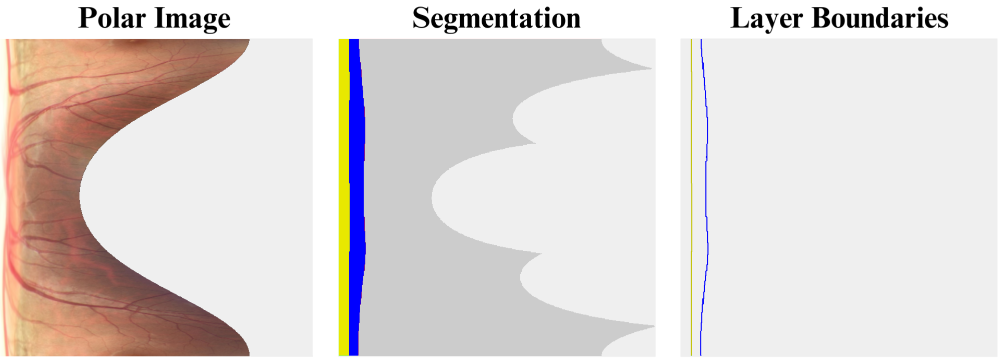

Figure 3.6: Polar image with segmentation and boundaries

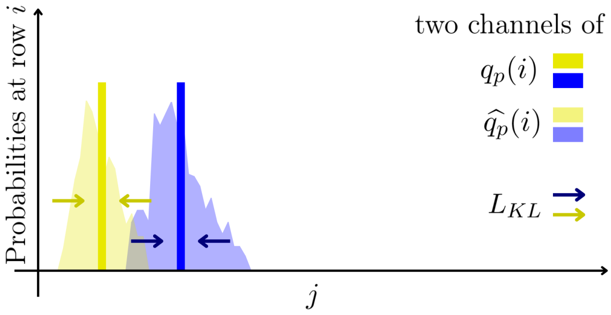

Figure 3.7: Example distribution of q_p and $\widehat{q}_p$ at a specific row i over all columns j

it does not predict an optic disc. Since the output of the topology module is the only output with guaranteed topological correctness, it is used to make the final output for evaluations. This output lives in the polar domain, therefore it is back-transformed like in Subsection 3.2.3.

Similar to He et al., matrices which contain information about the row and column indices are concatenated to the input image (see Subsection 2.6.3). This Thesis uses X and Y as defined below,

$$X = \frac{1}{m}\begin{pmatrix} 1 & 2 & \dots & m \\ 1 & 2 & \dots & m \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 2 & \dots & m \end{pmatrix} \qquad Y = \frac{1}{n}\begin{pmatrix} 1 & 1 & \dots & 1 \\ 2 & 2 & \dots & 2 \\ \vdots & \vdots & \ddots & \vdots \\ n & n & \dots & n \end{pmatrix} \tag{3.9}$$

where m is the number of columns and n is the number of rows. These matrices allow ETUS to obtain additional spatial information [45].

4 Results

4.1 Experiments

This Thesis studies the effects of incorporating prior knowledge into the model. The main hypothesis of this Thesis suggests that including prior knowledge makes a model more data-efficient, i.e., a model with incorporated prior knowledge needs less training data than a regular model to come to a similar performance. To test this hypothesis, the experiments conducted for this Thesis aim to answer the questions 1.-4. listed below. Each modification explained in Section 3.2 is tested independently of the others, even though some combinations would be feasible. Before analysing modifications and their effects, a detailed study of the behaviour of the baseline U-Net when trained with fewer training samples is necessary. This leads to the question:

1. *How well does the baseline U-Net perform when reducing the amount of training data?*

To address the hypothesis regarding improved data-efficiency, the following question is discussed:

2. *Do modifications explained in Section 3.2 increase the performance compared to the baseline for reduced training samples?*

Furthermore, changes such as switching cameras or medical staff are common in the medical field. Deep learning models need to be able to handle such changes, which is referred to as the model's ability to generalise. To validate the hypothesis also in this regard, the following question is addressed.

3. *Is the baseline U-Net still able to generalise to unseen datasets when trained with fewer training samples?*

Lastly, this Thesis studies whether the modifications are able to improve the generalisability.

4. *Do the modifications explained in Section 3.2 enhance the generalisability to unseen datasets when trained with fewer training samples compared to the baseline?*

In order to answer these questions, it is necessary to establish quality measures, which are able to compare performances. Therefore, Subsection 4.1.1 introduces the employed evaluation metrics. Section 4.2 endeavours to provide responses to the questions above. Questions related to the baseline model alone are explored in Subsection 4.2.1. The results are summarised and further discussed in Section 4.3, where Tables 4.1, 4.2 and 4.3, as well as Figures 4.25, 4.26 and 4.27 give a condensed overview of all results. These results are put into perspective by comparing them with the state-of-the-art models from the REFUGE2 challenge.

4.1.1 Evaluation Metrics

In order to ensure comparability with other research in this field, the evaluation is heavily based on the first REFUGE challenge [2]. Therefore, the Dice score, the Hausdorff distance and the vertical cup-to-disc-ratio (vCDR) are computed for all predictions on the REFUGE2 test set and the Chákṣu dataset as a whole (see Section 3.1 for details on these data sets). Definition and explanation of the Dice score were already given in Definition (2.7) in Section 2.2. This score measures how much two areas overlap, divided by the sum of their sizes. For the evaluation, the predicted areas are compared to their ground truth counterparts for optic disc and optic cup individually.

As an additional metric, the Hausdorff distance is used, since it behaves quite differently to the Dice score, while still being intuitive. The following explanation is inspired by [48]. Given sets A and B, the one-sided Hausdorff distance from A to B is defined as

$$d_H^1(A, B) := \max_{a \in A} \min_{b \in B} d(a, b) \tag{4.1}$$

where $d(\cdot, \cdot)$ can be any distance function. For the purpose of this Thesis, d is the Euclidean distance $d(x, y) = ||y - x||_2$. This one-sided Hausdorff distance can be interpreted as a game, where the first player (max) tries to find a point $a \in A$, which is as far from any point in B as possible, while the second (min) picks the point $b \in B$ which is closest to a. One can easily see that this distance function is not symmetric, take $A \subset B$, where B is much larger than A. Then it obviously holds that $d_H^1(A, B) = 0$, while $d_H^1(B, A) > 0$. To overcome this, the Hausdorff distance is defined by taking the maximum of these two one-sided Hausdorff distances

$$d_H(A, B) := \max\{d_H^1(A, B), d_H^1(B, A)\} \tag{4.2}$$

which is symmetric. This Hausdorff distance is computed between the optic disc ground truth and prediction, as well as between the optic cup ground truth and prediction. Hence, each image has two Hausdorff distances associated with it.

The vertical cup-to-disc-ratio (vCDR) is an important characteristic of retinal images, as it is used by medical experts as an indicator to diagnose glaucoma. Therefore, the difference of the ground truth and predicted vCDR should be as small as possible. The vCDR is computed by taking the vertical diameter (in pixels) of the optic disc and the optic cup. The diameter of the optic cup is then divided by the optic disc to obtain the ratio:

$$
\begin{aligned}
vCDR &= \frac{\mathrm{vdiam}(I^{OC})}{\mathrm{vdiam}(I^{OD})} \\
&= \frac{\max\{i \mid \exists j : I^{OC}(i,j) = 1\} - \min\{i \mid \exists j : I^{OC}(i,j) = 1\}}{\max\{i \mid \exists j : I^{OD}(i,j) = 1\} - \min\{i \mid \exists j : I^{OD}(i,j) = 1\}}
\end{aligned}
\tag{4.3}
$$

where vdiam is the function that computes the vertical diameter, I^{OC} and I^{OD} are the optic cup and optic disc channel of a binarized segmentation respectively and i, j are the row and channel indices respectively. These indices must satisfy $1 \leq i \leq n$ and $1 \leq j \leq m$, where n is the number of rows and m is the number of columns. To compare the prediction to the ground truth, this Master's Thesis looks at the absolute error between the vCDR of the prediction and the ground truth.

All three metrics are evaluated in terms of their mean score and worst score when reducing the amount of training data. Analysing the worst performances is of importance in the medical domain, as rare cases of complete failures can have severe consequences. Hence, the mean score alone does not suffice to assess the quality of a model. Since the variance of each performance measure increases when using less training data, the results of at least five runs are used simultaneously to increase expressiveness. This increased variance is due to the fact that sampling of potentially only 12 samples from a pool of 1200 introduces a significant random factor. Furthermore, each of these runs differs in their parameter initialisation.

4.2 Models

This section aims to answer the questions raised in Section 4.1 by iterating over each model and analysing its performance when trained with fewer training samples on the REFUGE2 test set and the Chákṣu dataset. The latter has significant differences to the REFUGE2 set and therefore represents a domain shift, since no Chákṣu data was used for training. Observations of lesser importance are not represented with a dedicated illustration. Instead, the summarized results from Section 4.3 are referenced. First, the performance and generalisability of the baseline model are explored in detail.

4.2.1 Baseline

Question 1: Performance using less training data

It is expected that the performance decreases, as less training data presents the distribution the model tries to learn more sparsely. Furthermore, reducing the number of training samples also reduces the number of training steps linearly, since the number of epochs is set to 100 for all experiments. As a first approach, the baseline model is trained with $10\%, 20\%, ..., 100\%$ of the available REFUGE2 training data. In total, the training data consists of 1200 images, meaning that 10% correspond to 120 training images.

Figure 4.1 shows the mean Dice score of the baseline U-Net on the y-axis, over the fraction of the REFUGE2 training that was used for training. At 100% of training data,

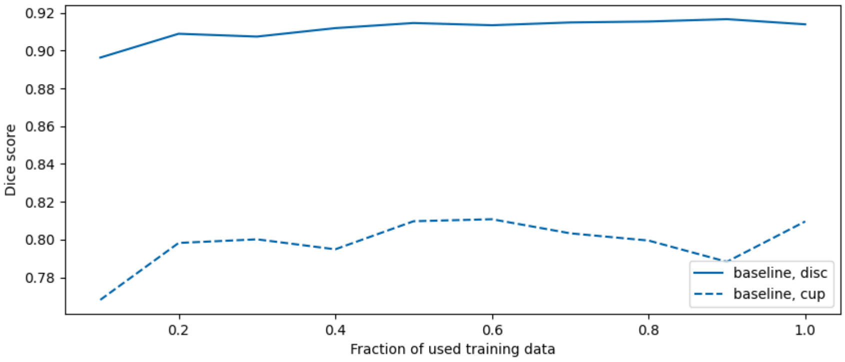

Figure 4.1: Baseline, mean Dice score on REFUGE2 test set

the baseline achieves a mean Dice score of around 0.91 for the optic disc and around 0.81 for the optic cup. The performance is only slightly impeded when reducing the amount of training data. At 10% of training data, the baseline still achieves a mean Dice score of around 0.9 for the optic disc and around 0.77 for the optic cup. To put the results in Figure 4.1 into perspective: The best teams in the REFUGE2 challenge achieved mean Dice scores up to 0.961 and 0.865 for the optic disc and optic cup respectively on the test set. These performances correspond to the lines in Figure 4.1 at 100%, as the models in the REFUGE2 challenge were trained using all training data. Figure 4.1 leads to the impression, that the model behaves quite similar when decreasing the amount of available training data, which raises two questions:

- Since Figure 4.1 shows just the mean, is this graph actually expressive enough to draw a conclusion, i.e. do the models actually behave similarly, or is the mean alone too simple of a statistic?

- When does the model start to collapse performance-wise, as models trained with 10% of the training data still perform quite well?

To answer the first question, boxplots are employed. They provide more information about the distribution, which makes them more expressive than the mean alone. Figure 4.2 shows the boxplot of the Dice score on the optic cup. Indeed, the boxes look similar, with only slight changes visible when decreasing the amount of training data. The performance distribution often looks like the one depicted in Figure 4.2: The majority of points centred around the mean, or median, with a long tale towards bad performances. In the case of the Dice score, this tale stretches towards 0, as low Dice scores correspond to poor segmentations. If not stated otherwise, it can be assumed that the boxplots show similar results to the mean statistics. The mean statistics are preferred over boxplots when the same message is conveyed due to their readability and flexibility when including more models or statistics.

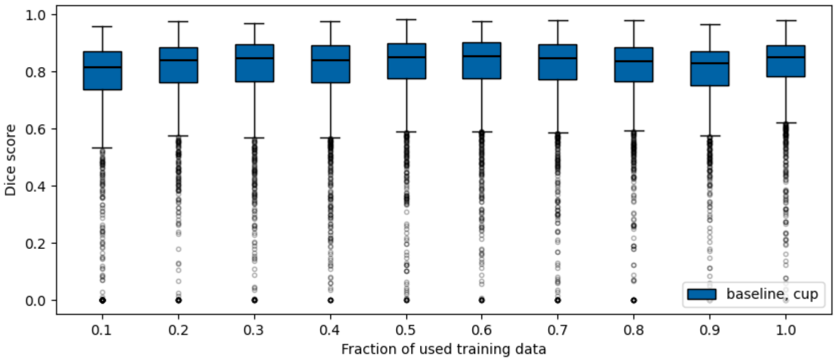

Figure 4.2: Baseline, boxplot of Dice score on optic cup on REFUGE2 test set

The second question, which tries to find the minimum of training data, without drastically decreasing the performance of the models, leads to a major design decision of the Master's Thesis: Instead of $10\%, 20\%, ..., 100\%$ of the training data, the performance for reduced sizes of the training set is assessed on $1\%, 2\%, ..., 10\%$ of the original REFUGE2 training set. Since this data set consists of 1200 samples, 1% amounts to 12 training samples.

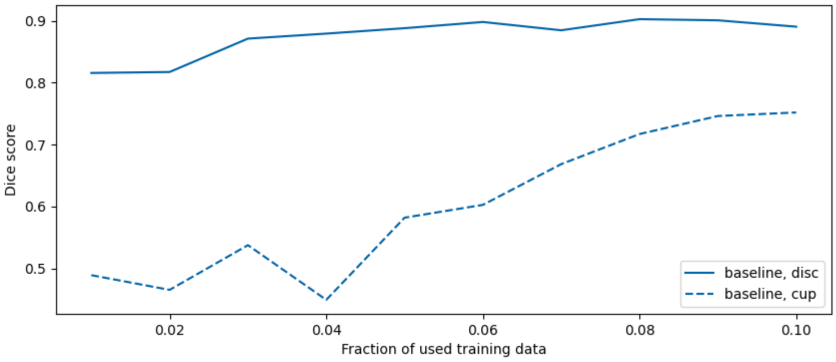

Figure 4.3: Baseline, mean Dice score on REFUGE2 test set

Figure 4.3 shows the mean Dice score of the baseline U-Net, trained with only $1\%, 2\%, ..., 10\%$ of the original REFUGE2 training set. Indeed, contrary to Figure 4.1, Figure 4.3 shows a significant performance decrease. The mean Dice score for the optic disc drops from around 0.9 at 10% of the training data to around 0.82 at 1%, while the mean Dice score for the optic cup even drops from around 0.75 at 10% of the training data to around 0.5 at 1%. Please note that all graphs show estimations of the true statistics, which is especially important to bear in mind for smaller percentages of training data. Drawing 12 images out of 1200 allows for a much worse representation of the true distribution than drawing 120. Therefore, unexpected lows, or highs, like in Figure 4.3

for the optic cup at 4%, should be interpreted as the result of many random processes, which have a high chance of deviating from the mean, since the variance increases with less training samples.

To provide a more detailed analysis of the performance decrease for reduced training sets, the Hausdorff distance is used as an additional metric. As a reminder: The Hausdorff distance measures how far two objects are apart, by choosing a point a in, e.g., the prediction, which is as far away from the ground truth as possible, and calculating its distance to the closest point b to a which lies in the ground truth (see Definition 4.2) for details). Therefore, small outliers can drastically increase the Hausdorff distance, while other metrics (like the Dice score) barely change. Another more obvious difference to the Dice score lies in the desired value range. While the Dice score should be as close to 1 as possible, the Hausdorff distance becomes 0 for perfect predictions. The distance is given in Euclidean pixel distance, i.e., a value of 100 means that the positions of the pixels a and b from the min-max problem have a Euclidean distance of 100. The images on which the Hausdorff distance is computed have a size of 512×512, meaning that the theoretical maximal Hausdorff distance is $512 \cdot \sqrt{2} \approx 724$. However, the optic disc and optical cup are usually not located in the corners of the image, which is why the actual maximal distance is smaller and varies from image to image. Like with the Dice score, the Hausdorff distance is computed for the optic disc and optic cup separately.

It is expected that the mean Hausdorff distance will get worse for fewer training samples, similar to the Dice score. Figure 4.4 shows the mean Hausdorff distance of the baseline U-Net. It illustrates that the model makes big breakthroughs for small increases of used training data, as the performance gain in the range between 1% and 3% of used training data shows. The mean Hausdorff distance for the optic disc drops from around 160 at 1% of used training data to around 40 at 3%. However, it is worth noting that the distribution of the Hausdorff distance for the optic disc on the REFUGE2 test samples is not gaussian. To illustrate the distribution of the Hausdorff distance for the optic disc of REFUGE2 test samples, Figure 4.5 shows a violin for each fraction of used training data. The thickness of each violin represents the density of the distribution at a specific Hausdorff distance.

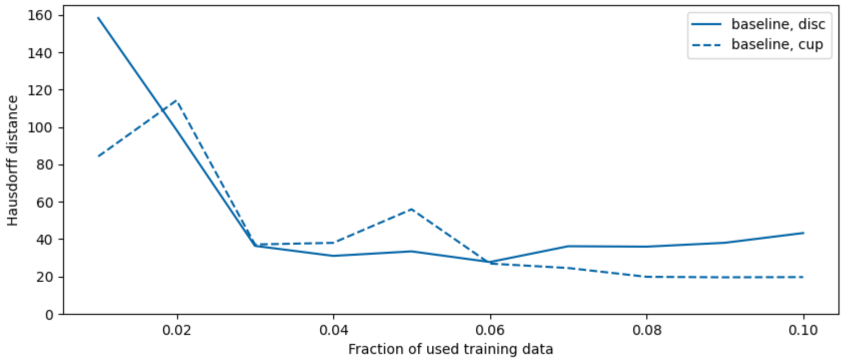

Figure 4.4: Baseline, mean Hausdorff distance on REFUGE2 test set

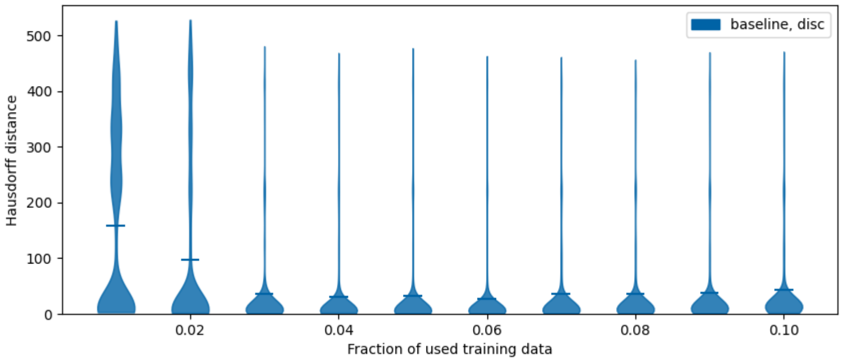

Figure 4.5: Baseline, Hausdorff distance of optic disc on REFUGE2 test set

Especially for 1% and 2% of used training data, it can be observed that the baseline model struggles with many REFUGE2 test samples. This behaviour is worth investigating, as the form of the tail at 1% of training data in Figure 4.5 is unexpected. For this purpose,

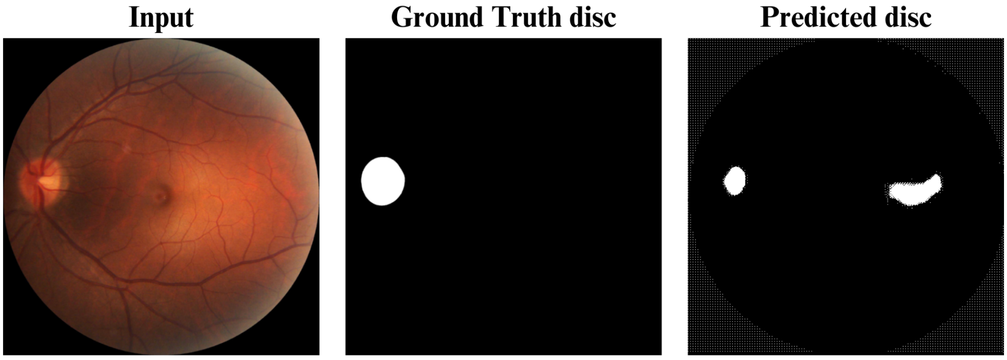

Figure 4.6: Baseline model trained with 1% of training data; input, ground truth and prediction of REFUGE2 test sample Nr. 253

Figure 4.6 shows a particularly bad prediction of the baseline model trained with 1%, where the Hausdorff distance exceeds 500. Juxtaposed to the predicted disc is the ground truth optic disc, as well as the retinal image, which is used as the model input. The model not only predicts the disc too small, but also includes another part of the image in its prediction. Furthermore, the rims of the prediction show white dots where the input is black. This is a strong indicator that the training of this particular model is not finished, which is plausible, since it only had $\#training\ samples \cdot \#epochs = 12 \cdot 100 = 1200$ training steps in the entirety of its training process. As expected by the looks of Figure 4.6, the baseline also achieves with under 0.25 an unusually low Dice score on the optic disc. However, Figure 4.5 shows that even models trained with 10% of the training data sometimes fail to achieve low Hausdorff distances. How do these predictions look like?

Figure 4.7 shows that a low mean Hausdorff distance does not imply better performance on all samples. The input image in Figure 4.7 is unusual dark with an unpronounced optic disc (see Figure 3.1 for comparison), which is in contrast to Figure 4.6, where the model struggles due to an unfinished training. This darkness seems to mislead the model, which instead predicts the optic disc in a bright area of the input. As expected, the Dice score for the predicted optic disc from Figure 4.7 is 0. This extreme example demonstrates the importance of an in-depth analysis of each model, which is intended to be used in the medical domain. Even models with promising results can fail miserably when presented with a challenging sample. Therefore, only comparing mean Dice scores, or other oversimplifying statistics does not suffice to deem a model fit for application. It is possible that a model with slightly worse mean Dice score has less distressing examples, which could confuse a medical expert.

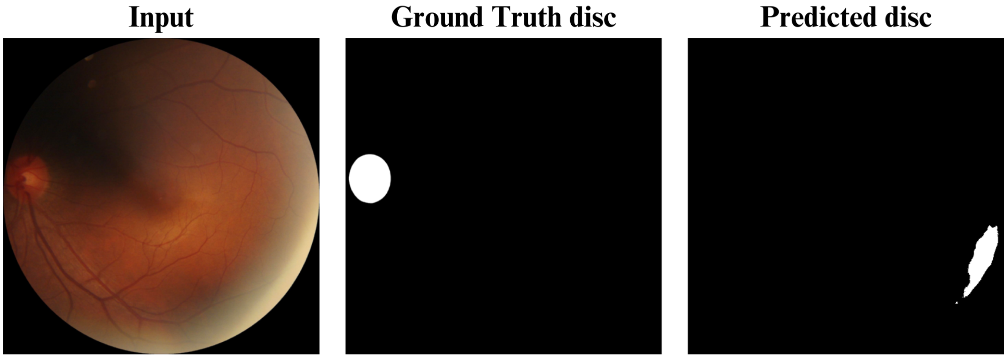

Figure 4.7: Baseline model trained with 10% of training data; input, ground truth and prediction of REFUGE2 test sample Nr. 110

Since the vCDR is a common diagnosis tool for glaucoma, models which assist medical experts with the diagnosis are expected to capture this characteristic of retinal images particularly well. Therefore, studying how the mean absolute vCDR error changes when reducing the amount of used training data is necessary to assess the changes in performance. The mean absolute vCDR error is expected to decrease with more training data. In order to achieve a low error for the vCDR, the models need to get the ratio of the predicted optic disc and optic cup as close to the ground truth ratio as possible. In theory, this does not necessarily correspond to a good segmentation, as a spatial translation (no rotation) of the prediction would lead to the same vCDR. Figure 4.8 shows the mean absolute vCDR error of the baseline on the REFUGE2 test set. It does not follow a monotonic decline when increasing the fraction of used training data, but rather fluctuates. At 1% of used training data, the mean absolute vCDR error is around 0.36. The error jumps to around 0.8 at 2% of training data, just to decline to around 0.3 at 3% training data. This spike is unexpected and unwanted, as it suggests that the performance of the baseline model is heavily dependent on the drawn training samples or the parameter initialisation. The absolute vCDR error itself is sensitive to small outliers, as Section 3.2.4 already

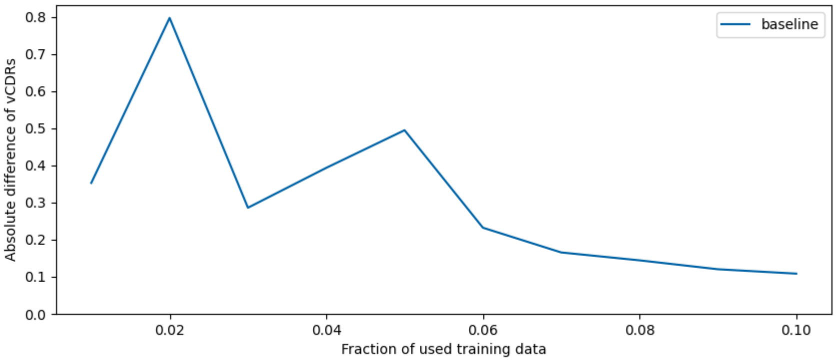

Figure 4.8: Baseline, mean absolute vCDR error on REFUGE2 test set

explored. Just a single pixel outlier can increase the vertical diameter drastically and thus the ratio. Figure 4.8 shows how this can affect the error, as especially models trained with fewer training data tend to produce outliers. This leads to errors even beyond 1, for rare cases where the optic cup is predicted to be bigger than the optic disc. These outliers happen much less frequently for more training data, and thus the mean absolute vCDR error steadily declines from 60% of the training data onward.

Question 3: Generalisability to unseen dataset

Similar to the performance on the REFUGE2 test set, the baseline model is expected to increase its capability to generalise to a new dataset when presented with more training samples. This is tested by evaluating the baseline model on the Chákṣu dataset. Figure 4.9 shows the mean Dice score of the baseline model on the Chákṣu dataset for

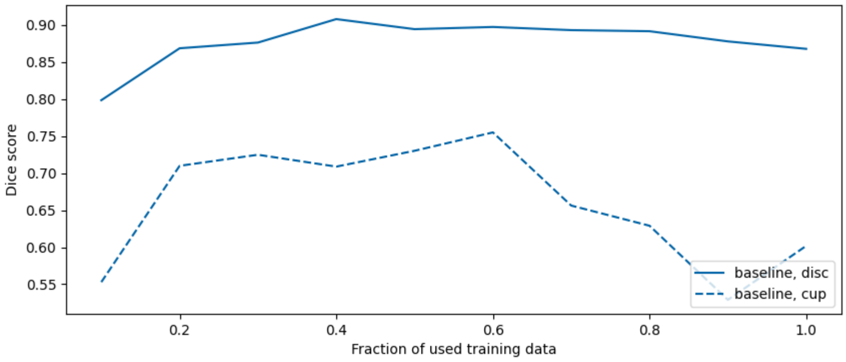

Figure 4.9: Baseline, mean Dice score on Chákṣu

$10\%, 20\%, ..., 100\%$ of the available REFUGE2 training data. The mean Dice score for the optic disc starts around 0.8 at 10% of the training data and steadily increases to around 0.9 at 40% of the training data. However, it starts to slowly drop at this point for increased training data. The optic cup shows a similar behaviour in Figure 4.9, with a performance decline from 60% of used training data onward. The baseline model seems to overfit to the REFUGE2 data, which was used for training. Hence, the model looses its capability to generalise with more training data. At the same time, the mean Dice score rapidly increases from 10% of used training data to 20%. The mean Dice score of the optic disc increases from around 0.8 to 0.87, while the mean Dice score of the optic cup has an even more drastic increases from around 0.55 to 0.72. This indicates that the baseline needs at least 20% of the REFUGE2 training to properly generalise to the unseen Chákṣu dataset.

Figure 4.9 is extended to $1\%, 2\%, ..., 10\%$ of the available training data in Figure 4.10. Figure 4.10 shows that decreasing the amount of used training data steadily decreases the mean Dice score as well. It starts around 0.85 for the optic disc and 0.55 for the optic cup at 10% training data, and reduces to around 0.72 for the optic disc and 0.5 for the optic cup at 1%. Please note that Figures 4.9 and 4.10 show two independent estimates for the mean Dice score at 10% training data. They slightly differ, as Figure 4.10 shows a mean Dice score of around 0.85 for the optic disc and 0.58 for the optic cup at 10% training data (in contrast to 0.8 and 0.55 respectively in Figure 4.9).

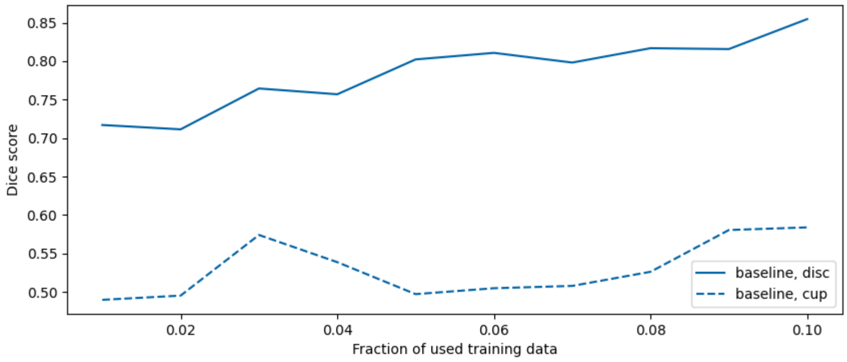

Figure 4.10: Baseline, mean Dice score on Chákṣu

As Figure 4.10 shows, the baseline model has problems to generalise in terms of the mean Dice score. The Hausdorff distance is assumed to reveal a similar pattern. This assumption is confirmed by Table 4.2b, because the baseline model has a mean Hausdorff score of 104.5 for the optic disc and 56.0 for the optic cup even at 10% of training data. These results are significantly worse than for the REFUGE2 test set, as the juxtaposed Table 4.2a indicates. A more detailed analysis of the Hausdorff distance on the Chákṣu dataset is presented in Figure 4.11. It shows the distribution of the Hausdorff distance of the Baseline regarding the optic disc on Chákṣu. Instead of a single big near gaussian distribution, the Chákṣu samples are arranged in three separate clusters. This pattern is most prominent for lower percentages of training data used, but is still visible at 10%. This behaviour is partly due to the fact that the Chákṣu dataset is inhomogeneous. Figure 3.2

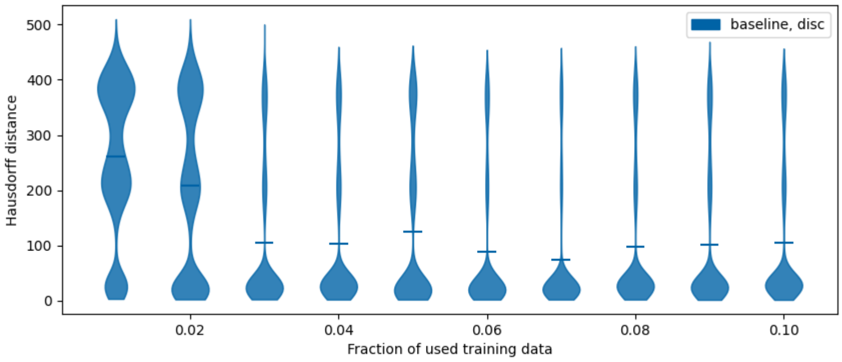

Figure 4.11: Baseline, Hausdorff distance of optic disc on Cháksu

already shows that the three different camera types, used to build Cháksu, have different characteristics. However, the clusters do not align perfectly with the camera types. When looking at the representatives of the samples which achieved a Hausdorff distance less than 100 when trained with 1% of the training data, only few Cháksu samples with text in it are present. Another reason for this clustering could lie in the different techniques of the people who took the retinal images: Some retinal images place the optic disc in the centre (making the maximal Hausdorff distance small), while others have their optic disc closer to the edge of the image. The pattern in Figure 4.11 indicates that parts of the Cháksu dataset are easier for the baseline model to generalise to.

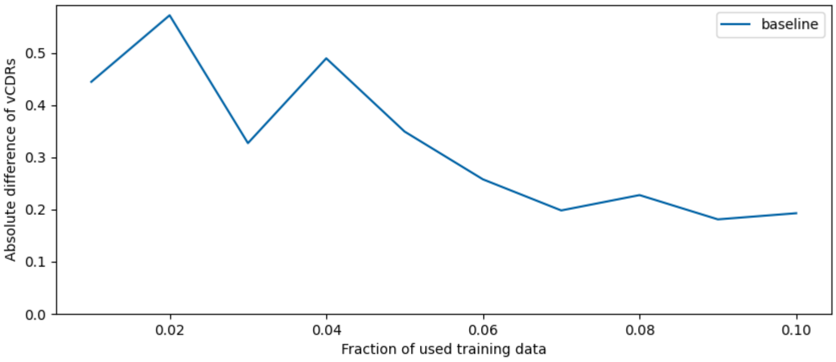

Figure 4.12: Baseline, mean absolute vCDR error on Cháksu

Lastly, the generalisability in terms of the mean absolute vCDR error is discussed. Figure 4.12 shows the mean absolute vCDR error on the Cháksu dataset. It shows a similar pattern to Figure 4.8 with strong fluctuations for small fractions of used training data. Nonetheless, the baseline model is able to steadily reduce the mean absolute vCDR error from 4% of used training data onward, ending around 0.19 at 10% training data. This mean error is still significantly higher than on the REFUGE2 test set, as can be observed in table 4.3, which suggests a poor generalisation from the REFUGE2 training data to the Cháksu dataset.

4.2.2 Novel Topological Loss

Question 2: Performance using less training data - Comparison with baseline

The topological loss, as introduced in Section 3.2.2, aims to produce anatomical correct predictions by penalising models that predict the optic cup, where no optic disc is predicted. According to the hypothesis of this Thesis, this would make the model more data-efficient, i.e., the performance will be better with few training samples, compared to the baseline. To test this hypothesis, the mean scores and the worst scores are analysed. Figure 4.13 shows the mean Dice of the baseline U-Net, compared to the same model trained with the additional topological loss, for the REFUGE2 test set. The mean Dice score for the optic disc is comparable, except for 1% of the training data, where the model trained with topological loss performs around 0.7, while the baseline U-Net achieves a mean Dice score above 0.8. However, the model trained with topological loss performs significantly worse for the optic cup, as the dotted lines in Figure 4.13 indicate. This suggests that the mean performance is not increased using the topological loss. But what about the worst performances? Figure 4.14 shows boxplots of the absolute vCDR error of the baseline and the model trained with topological loss on the REFUGE2 test set. For smaller percentages, 1% - 5%, the model trained with topological loss has less severe worst cases. The baseline model has some outliers with errors greater than 1, which means that the optic cup is predicted to be vertically larger than the optic disc, which violates the anatomy. This phenomenon cannot be observed with the topological loss, which is designed to increase the probability of the optic disc or decrease the probability of the optic cup. Both cases only decrease the vCDR. Since the severe outliers of the baseline in Figure 4.14 are cases where vCDR is (much) higher than 1, the topological loss has an advantage, as these cases become less likely.

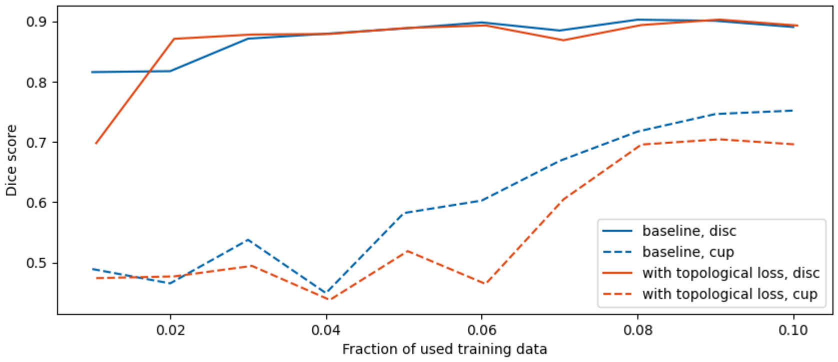

Figure 4.13: Topological loss, mean Dice score on REFUGE2 test set

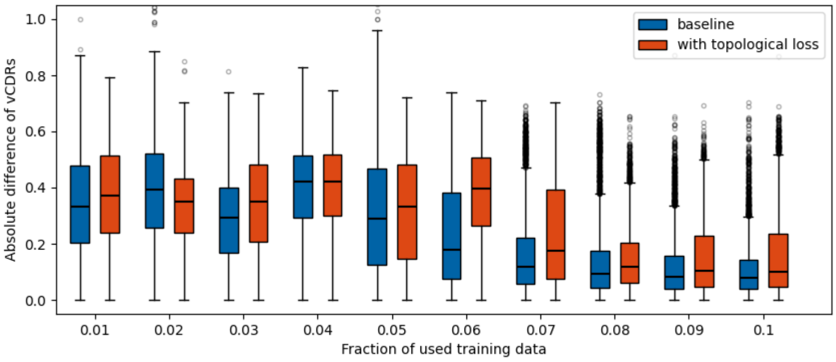

Figure 4.14: Topological loss, boxplot of absolute vCDR error on REFUGE2 test set

Summary To summarise: The mean performance is slightly reduced using the topological loss, but the highest baseline errors in terms of the absolute vCDR error get less severe by using the topological loss. This can be seen in Figure 4.14 at percentages of training data between 1% and 5%, where the upper whiskers of the boxplots (or the highest outliers) have smaller absolute vCDR errors with the topological loss. Depending on the priorities of a potential user, this could be a considerable trade.

Question 4: Generalisability to unseen dataset - Comparison with baseline

Using the topological loss nudges the model to outputs where the optic cup is only predicted where the optic disc is predicted as well. Since this reflects an anatomical fact, it is expected that the model trained with the topological is able to generalise better, as this anatomical fact is present in all datasets, not only the training distribution. Figure 4.15 shows the mean Dice score of the baseline, compared to the same model additionally trained with the topological loss, evaluated on the Cháksu dataset. For both, the optic

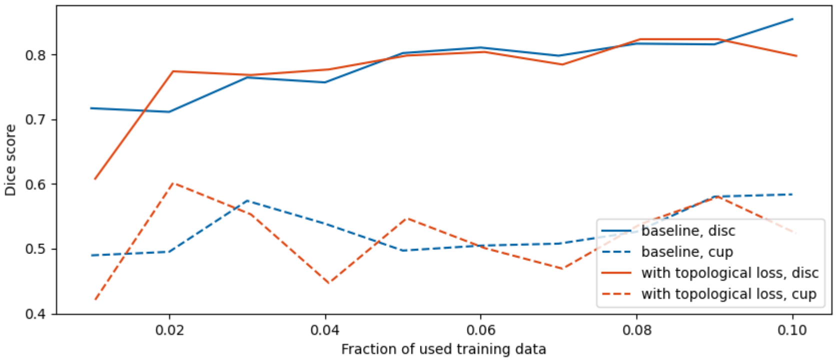

Figure 4.15: Topological loss, mean Dice score on Cháksu

disc and the optic cup, the models in Figure 4.15 behave similarly. This is also true for the mean Hausdorff distance, as can be seen in Table 4.2b.

Looking at Table 4.3b reveals an advantage of the model trained with topological loss that was already discussed in Figure 4.14: Especially for small percentages of training data, the topological loss helps to prevent severe outliers in terms of the absolute vCDR error. Table 4.3b shows that this is also true for the mean on the Chákṣu dataset.

Summary Overall, the topological loss has no significant effect on the generalisability, except for its tendency to prevent severe malfunctions in terms of absolute vCDR error.

4.2.3 Polar Transformation

Question 2: Performance using less training data - Comparison with baseline

Models associated with this subsection are trained on REFUGE2 training samples, which are transformed into the polar domain beforehand (see Section 3.2.3). For the evaluation, the predictions of the model, which also live in the polar domain, are transformed back to Cartesian coordinates to make comparisons fair. Figure 4.16 shows the mean Dice score of the baseline, compared to the same model trained on polar data. Their performance for the optic disc is similar, but the model trained on polar data starts with a higher mean Dice score for the optic cup at 1% of used training data. However, unlike the baseline, the model trained on polar data does not improve its mean Dice score on the optic cup when increasing the amount of used training data. Therefore, the baseline surpasses the model trained on polar data at 5% training data. The mean Hausdorff distance reveals a similar pattern in Table 4.2a and Figure 4.26: Even though the model trained on polar data starts at a smaller average Hausdorff distance, it loses its advantage eventually when increasing the amount of training data. Table 4.3a and Figure 4.27 furthermore show that the model trained on polar data also has difficulties to achieve small mean absolute vCDR errors.

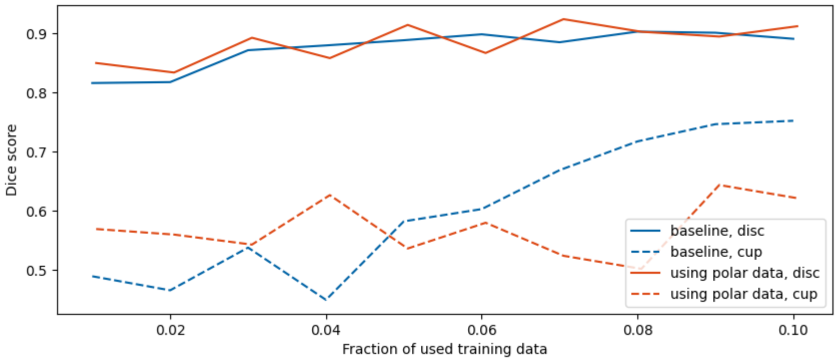

Figure 4.16: Polar transformation, mean Dice score on REFUGE2 test set

Summary To summarise the differences of the baseline and the model trained on polar data in terms of performance on small amounts of data: Using polar data is an effective strategy for very small amounts of data, like 1% (only 12 training samples). However, using the Cartesian data makes it easier for the model to improve when increasing the number of training samples. The model trained on polar data did not gain as much performance as the baseline with more training data.

Question 4: Generalisability to unseen dataset - Comparison with baseline

Applying a polar transformation the way this Master's Thesis does, produces images which look more alike. For example, the optic disc and optic cup get relocated to proximally the same position, the left edge of the image. It is assumed that this mechanism helps the model to generalise to new data, as this new data is transformed the same way. Figure 4.17 shows that, indeed, using polar data helps to increase the mean Dice score on an unseen dataset, compared to the baseline. This can be observed with both, the optic disc and the optic cup. However, the model trained on polar data does not improve its mean Dice score for the optic cup with more training data. It stagnates around 0.6, which is better than the baseline, especially for smaller percentages of used training data, but the baseline is able to narrow the performance gap for the optic cup with increased amounts of training data. Surprisingly, the advantage in terms of average Dice score is not reflected in Tables 4.2b and 4.3b, which show the mean Hausdorff distance and mean absolute vCDR error on the Cháksu dataset respectively. That the model trained on polar data has problems in terms of the Hausdorff distance is particularly unexpected, as the back-transformation into the Cartesian coordinates already eliminates severe misplacements, due to how the back transformation is implemented (see Subsection 3.2.3 for details).

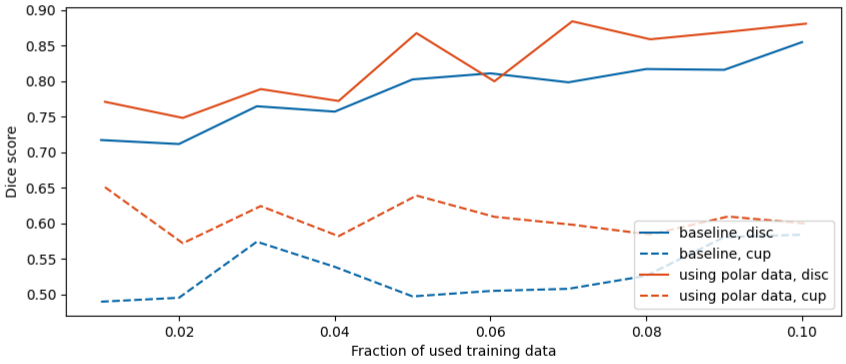

Figure 4.17: Polar transformation, mean Dice score on Cháksu

Summary To summarise: Using data which was transformed into the polar domain helps the model to generalise to the Cháksu dataset in terms of mean Dice score, as

Figure 4.17 proves. However, the mean Hausdorff distance and mean absolute vCDR error are not improved by using polar data.

4.2.4 Novel CDR Loss

Question 2: Performance using less training data - Comparison with baseline

The CDR loss is designed to nudge the vCDR of a prediction as close to the ground truth vCDR as possible. This is done by trying to increase, or decrease the size of the predicted optic cup, depending on whether the vCDR is too small, or too big. It can be observed that the baseline model performs poorly for the optic cup, by often overestimating the size of it. Hence, the CDR loss is expected to reduce this source of error. Figure 4.18 shows the mean Dice score of the baseline, compared to the same model trained with the CDR loss as additional loss term. They behave very similarly for the optic disc, but introducing the CDR loss slightly improves the mean Dice score on the optic cup. This improvement is only present for fractions of used training data between 4% and 7%. Analysing the mean Hausdorff distance and mean absolute vCDR error in Tables 4.2a and 4.3a confirms that introducing the CDR loss helps the model, but only in a certain percentage range of training data. The columns 2% and 5% in Tables 4.2a and 4.3a show an improvement compared to the baseline, but columns 1% and 10% indicate that the advantage is not universal.

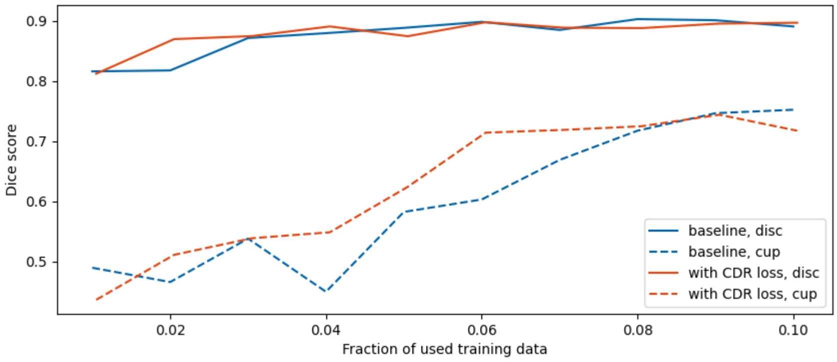

Figure 4.18: CDR loss, mean Dice score on REFUGE2 test set

As the name already suggests, the CDR loss ultimately aims to reduce the absolute vCDR error, by penalizing predictions where the vCDR is far off the ground truth. Figure 4.19 shows a boxplot of the mean absolute vCDR errors. The most significant difference of the baseline and the model trained with CDR loss is at 1% training data, where at least every forth sample output of the model trained with CDR loss has an absolute vCDR error that exceeds 1. This is a significant setback of performance in terms of worst absolute vCDR error. Including the CDR loss reduces the median error in the aforementioned range of 4% up to 7% of training data.

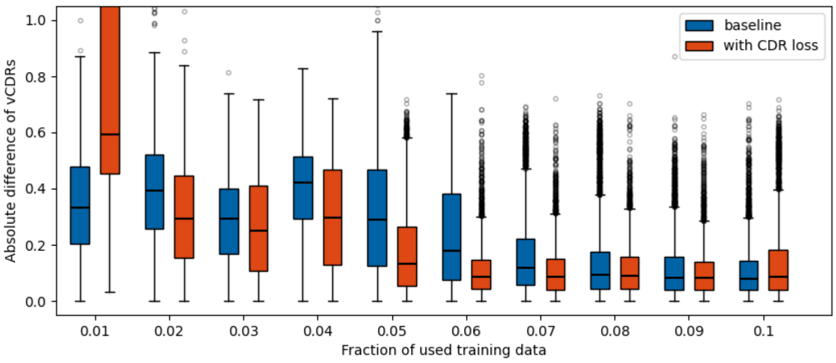

Figure 4.19: CDR loss, boxplot of absolute vCDR error on REFUGE2 test set

Summary To summarize: Unless a potential user has between 4% and 7% (48 and 84 training images, respectively), adding the CDR loss does not increase the performance. Especially at 1% training data, the performance is even diminished in terms of all performance measures. A possible explanation for the specific range of used training data, where the CDR loss leads to an improvement, could lie in the fact that the additional loss penalizes models which ignore relations in terms of size of the optic disc and optic cup. For higher amounts of training data, the baseline model eventually learns to get the sizes correct, but including the CDR loss puts more focus on the size relations, which are therefore corrected early on.

Question 4: Generalisability to unseen dataset - Comparison with baseline

The CDR loss aims to assist the model to produce segmentations with a similar vCDR as the ground truth segmentation. Introducing this loss should therefore increase the generalisability, especially in terms of the average absolute vCDR error.

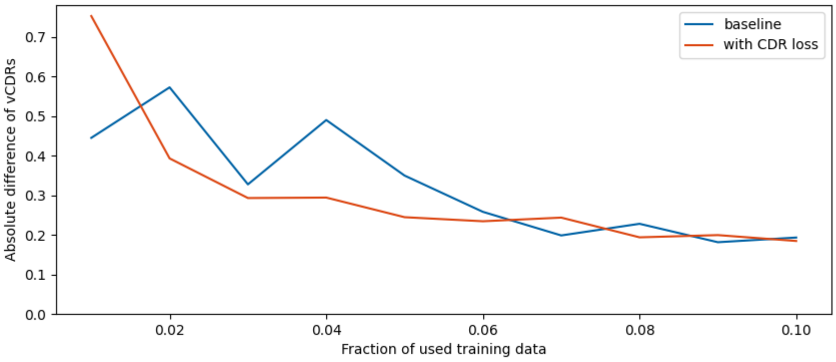

Figure 4.20: CDR loss, mean absolute vCDR error on Cháksụ

Figure 4.20 shows the mean absolute vCDR error of the baseline, compared to the model additionally trained with the CDR loss on the Chákṣu dataset. As already discussed for Figure 4.19, introducing the CDR loss has a negative impact on the vCDR error for the model trained with 1% of the available training data. This setback can also be observed for the Chákṣu dataset in Figure 4.20. However, increasing the training data shows that including the CDR loss helps the model to achieve lower mean absolute vCDR error earlier on. This can be seen in Figure 4.20 between 2% and 6% of training data. Tables 4.1b and 4.2b reveal that the CDR loss does not significantly improve the generalisability of the model in terms of mean Dice score or mean Hausdorff distance.

Summary To reiterate: Applying the CDR loss helps the generalisability in terms of mean absolute vCDR error for specific percentage ranges of training data, but this slight advantage is not reflected by other metrics.

4.2.5 Enforcing Anatomical Coherence on Polar Data

Question 2: Performance using less training data - Comparison with baseline

The model discussed in this Subsection 3.2.5 encapsulates the most changes compared to the baseline. For instance, ETUS is not capable to produce predictions, which do not follow the topological label relation of optic disc and optic cup. This is also in contrast to the topological loss, which only tries to nudge the model toward topological sound predictions, rather than enforcing them. Furthermore, the data used to train ETUS lives in the polar domain. Since the model showed promising results for layer segmentation in the paper by He et al. [45], it is expected that these successes can be replicated for the segmentation of the optic disc and optic cup in the polar domain, which is translated into a layer segmentation problem in the polar domain.

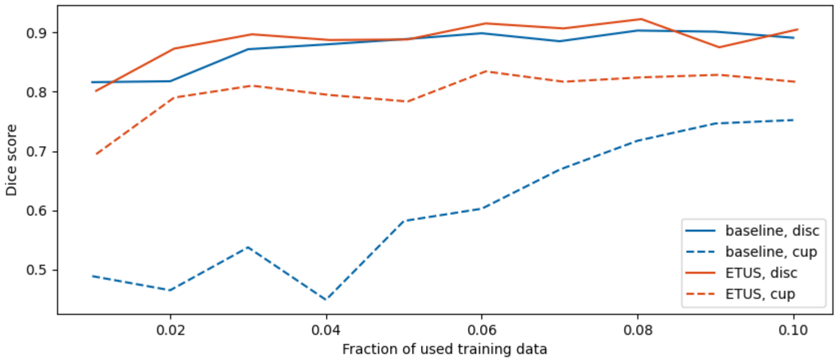

Figure 4.21: ETUS, mean Dice score on REFUGE2 test set

Figure 4.21 shows that already when trained with only 1% (only 12 images) of the training samples, ETUS achieves a mean Dice score on the optic cup of around 0.7, compared to the 0.5 of the baseline. Increasing the training data to 2% lifts the mean

Dice score to around 0.79, which is only slightly worse than the baseline model trained with 100% of the data (see Figure 4.1). This means a considerable improvement in terms of mean Dice score on the optic cup. The optic disc, however, shows a comparable performance to the baseline model. Figure 4.21 indicates that ETUS is capable to produce significantly better segmentations of the optic cup, but is this advantage reflected by the absolute vCDR error? Figure 4.22 shows boxplots of the absolute vCDR error of the baseline, compared to ETUS. Not only the median is significantly reduced, especially for smaller amounts of training data, but also the worst performance outliers. The superiority of ETUS is further confirmed in Table 4.2a, which shows the mean Hausdorff distances.

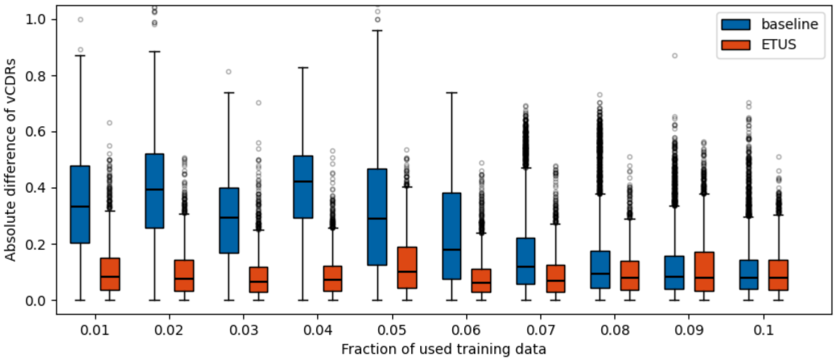

Figure 4.22: ETUS, boxplot of absolute vCDR error on REFUGE2 test set

Why is ETUS so successful on the REFUGE2 test set? Especially its segmentation performance on the optic cup is considerably better compared to the baseline. This is most likely the reason for the low median absolute vCDR errors in Figure 4.22 as well. The improvement is probably due to the biggest change to the baseline: ETUS does not produce segmentations, but predicts where the boundaries lie in the polar domain. To do this correctly, a Kullback-Leiber loss and a L_1 on the radius at each angle are employed. These seem to be very effective and drive the model towards good predictions of the boundaries. Furthermore, it could be the case that the boundaries have a more homogenous appearance compared to the optic disc and optic cup themselves. This would make the identification of the boundaries easier compared to segmenting the objects, as all other models do.

Summary To summarize: ETUS is capable to produce sound segmentations with drastically reduced training sets. The superiority of ETUS in optic cup segmentation lead to the decision to expand the analysis of ETUS, to also include models trained with $10\%, 20\%, ..., 100\%$. This allows for a fair comparison with state-of-the-art models, which is included in Section 4.3.

Question 4: Generalisability to unseen dataset - Comparison with baseline

As already discussed in Section 4.2.3, transforming retinal images into the polar domain distorts images to look more similar, by mapping the region of interest to the left edge in each image. Since ETUS is used in the polar domain, it is assumed that ETUS increases the generalisability to unseen data sets compared to the baseline. Furthermore, the topological label relation (optic cup is predicted within the optic disc) is enforced, which eliminates a potential error. Figure 4.23 shows the mean Dice score of the baseline, compared to ETUS on the Chákṣu dataset. ETUS is able to improve the mean Dice score compared to the baseline for both, the optic disc and optic cup, at every fraction of used training data, except for 5%, where it performs slightly worse on the optic disc. This improvement is more pronounced for the optic cup, where the performance gap is significant. Tables 4.2b and 4.3b show that ETUS also generalises considerably better in terms of mean Hausdorff distance and mean absolute vCDR error, respectively.

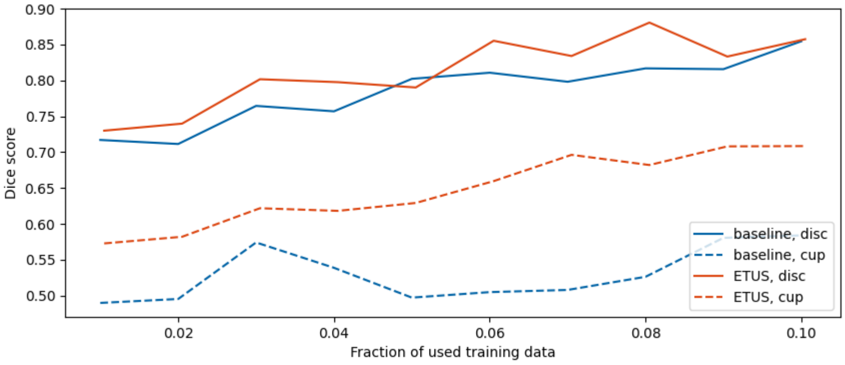

Figure 4.23: ETUS, mean Dice score on Chákṣu

As discussed in Figure 4.9, the baseline model overfits to the REFUGE2 set, as its performance on the Chákṣu dataset diminished with increased size of the training set. To put ETUS to a final test, the corresponding plot is analysed. Figure 4.24 shows the mean Dice score of the baseline, compared to ETUS on the Chákṣu dataset in the percentage range $10\%, 20\%, ..., 100\%$ of REFUGE2 training data. In contrast to the baseline, ETUS is able to steadily increase the mean Dice score with more training samples, even though no Chákṣu samples were used for training. Hence, ETUS seems to be more robust against overfitting than the baseline model.

Summary To summarize: ETUS is able to extend its superiority over the baseline to unseen datasets, even though the performance gap on the optic cup is less severe for small data points. Thus, it is able to generalise better than the baseline. Furthermore, ETUS shows no signs of overfitting in Figure 4.24, making it easier to train properly.

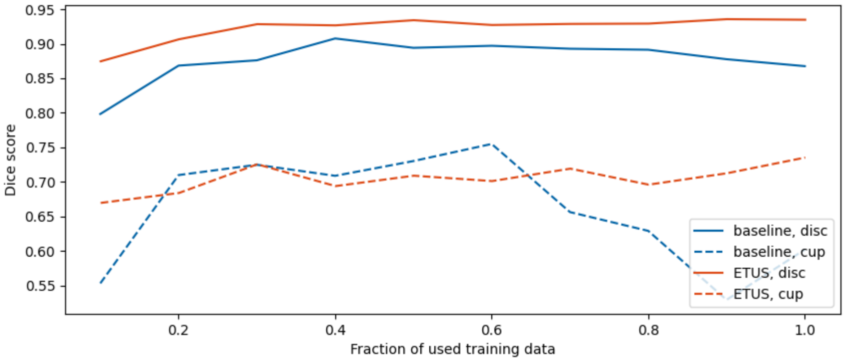

Figure 4.24: ETUS, mean Dice score on Chákṣu, up to 100% training data

4.3 Discussion

To assist the comparison and to provide the reader with a richer analysis, some summary tables are included, which show the results of all models on both evaluation sets (REFUGE2 test set and Chákṣu), represented by models trained with $1\%, 2\%, 5\%$ and 10%. Furthermore, plots showing the mean performances of all models on the REFUGE2 test set are included.

Table 4.1: Mean Dice score on both evaluation sets of optic disc (top) and optic cup (bottom), with the highest score for optic disc and optic cup separately highlighted per column

	(a) REFUGE2 test set				(b) Chákṣu			
Model	1%	2%	5%	10%	1%	2%	5%	10%
Baseline	0.816	0.817	0.888	0.891	0.717	0.711	0.802	0.855
	0.489	0.465	0.582	0.752	0.49	0.495	0.497	0.584
Topological loss	0.698	0.871	0.889	0.893	0.608	**0.774**	0.798	0.798
	0.474	0.477	0.519	0.696	0.421	**0.602**	0.547	0.524
Polar transform	**0.85**	0.834	**0.914**	**0.912**	**0.771**	0.748	**0.867**	**0.881**
	0.569	0.56	0.536	0.621	**0.651**	0.572	**0.639**	0.6
CDR loss	0.812	0.87	0.874	0.897	0.7	0.762	0.759	0.776
	0.436	0.511	0.624	0.717	0.412	0.543	0.438	0.54
ETUS	0.801	**0.872**	0.888	0.905	0.73	0.74	0.79	0.857
	0.695	**0.79**	**0.783**	**0.816**	0.573	0.582	0.629	**0.708**

This Thesis studies the behaviour of deep learning segmentation algorithms faced with small amounts of training data and how to improve them. U-Net is chosen as a baseline model, as it is often used and easy to implement by using pre-existing libraries. Figure 4.1,

Table 4.2: Mean Hausdorff distance on both evaluation sets of optic disc (top) and optic cup (bottom), with the lowest score for optic disc and optic cup separately highlighted per column

	(a) REFUGE2 test set				(b) Chákṣu			
Architecture	1%	2%	5%	10%	1%	2%	5%	10%
Baseline	158.4	98.38	33.46	43.28	261.7	208.0	124.1	104.5
	84.3	114.3	56.0	19.69	237.2	227.6	99.8	56.0
Topological loss	169.6	**38.03**	30.42	37.18	240.2	202.4	97.85	110.9
	158.6	44.3	32.63	23.59	211.3	172.6	62.0	51.4
Polar transform	**87.84**	95.98	42.52	54.94	220.3	244.6	88.96	110.3
	51.8	74.56	42.82	46.71	207.8	230.6	74.62	95.18
CDR loss	211.7	38.98	36.6	39.92	251.8	162.1	112.3	66.47
	206.4	37.85	25.58	22.73	260.9	144.5	56.93	34.15
ETUS	95.13	39.71	**16.53**	**11.63**	**83.11**	**53.23**	**39.68**	**35.6**
	67.79	**17.41**	**9.964**	**9.12**	**75.9**	**32.41**	**27.82**	**32.17**

Table 4.3: Mean absolute vCDR error on both evaluation sets, with the lowest score per column highlighted

	(a) REFUGE2 test set				(b) Chákṣu			
Architecture	1%	2%	5%	10%	1%	2%	5%	10%
Baseline	0.353	0.798	0.495	0.109	0.445	0.572	0.349	0.193
Topological loss	0.366	0.337	0.317	0.17	0.341	0.371	0.288	0.247
Polar transform	0.497	0.461	0.512	0.244	0.492	0.545	0.41	0.286
CDR loss	1.054	0.307	0.185	0.15	0.752	0.393	0.244	0.184
ETUS	**0.107**	**0.1**	**0.129**	**0.102**	**0.161**	**0.145**	**0.142**	**0.15**

and by extension all results on a percentage range of 10% up to 100%, are surprisingly stable in performance. Models trained with only 10% (120 images) of the training data are expected to perform substantially worse than models trained with all the training data. There are two main theories for this good performance:

- U-Net is indeed very data efficient
- The problem of segmenting the optic disc and optic cup is easy to solve by deep learning models

Ronneberger et al. claim that U-net is data efficient, however the version used for this Thesis is considerable bigger in terms of tunable parameters, which should harm the data efficiency [1]. Nonetheless, even the results of models trained with only 1% of the training data and insufficient training steps for convergence, show that the models are able to perform reasonable segmentations. Table 4.1b furthermore indicates that especially models trained on polar data (Polar transform and ETUS) are in parts able to generalise their performance onto unseen datasets. Hence, the claim of data efficiency is supported by the results of this Thesis. Section 2.5 already tells that segmenting the optic disc is a task which was tackled before the rise of deep learning. The researchers came up with segmentation algorithms, which were able to perform reasonably well for this task. Due to its outstanding characteristics, the researchers could find patterns that could be used to locate and segment the optic disc. The fact that the problem is solvable by rule based algorithms already hints that deep learning approaches could work very well, even with little data.

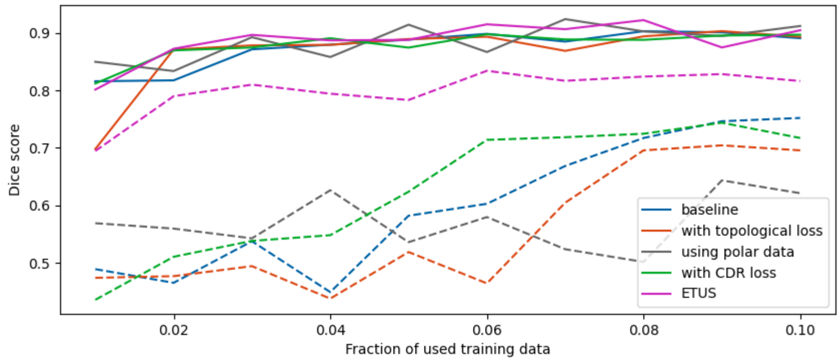

Figure 4.25: All models, mean Dice score on REFUGE2 test set, for optic disc (solid lines) and optic cup (dashed lines)

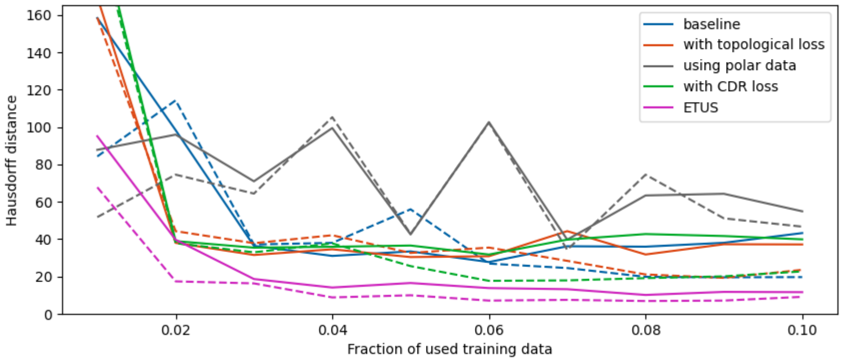

Figure 4.26: All models, mean Hausdorff distance on REFUGE2 test set, for optic disc (solid lines) and optic cup (dashed lines)

Besides the overall surprising good performance of U-Net trained with small amounts of data, each modification changes the behaviour in an interesting way, as Section 4.2 explores in detail. The topological loss is able to reduce the likelihood of severe malfunctions in

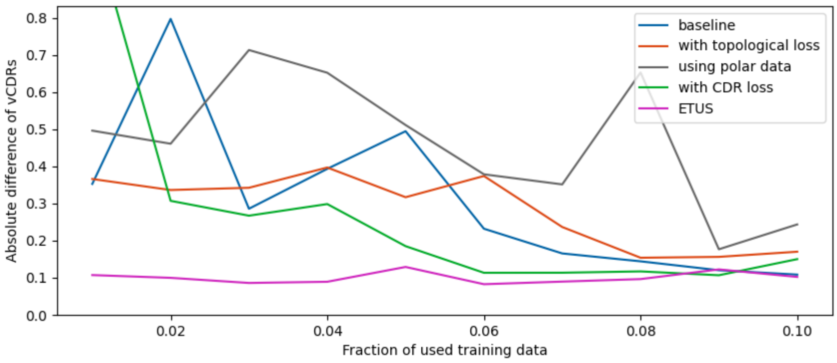

Figure 4.27: All models, mean absolute vCDR error on REFUGE2 test set

terms of absolute vCDR error. Transferring the problem into the polar domain helps the model to generalise to new, unseen data, which can again be seen in Table 4.1. As discussed in Subsection 4.2.4, introducing the CDR loss to the model slightly improves the performance and generalisability, but only in specific ranges of used training data. However, ETUS definitely brings the most changes compared to the baseline. ETUS generally performs better than the baseline and is arguably the best adaptation according to Tables 4.1, 4.2 and 4.3. It combines the improved generalisability onto unseen data from the polar transformation with the topological coherence, especially leading to low mean absolute vCDR errors, as Table 4.3 and Figure 4.27 show. On top of good performance for small amounts of training data, ETUS does, unlike the baseline, not overfit to the REFUGE2 dataset when trained with more than 60% of the training data (see Figure 4.24).

The REFUGE2 dataset that was used for training and evaluation stems from the REFUGE2 challenge. To put the results of this Thesis into the context of state-of-the-art models, Table 4.4 shows the performance of the teams that participated in the segmentation task of the REFUGE2 challenge. This Table is copied from the corresponding paper [3] and extended by the performances of the baseline and ETUS. All models in Table 4.4 had access to the same data, the REFUGE2 train and validation sets, and were evaluated on the same set, the REFUGE2 test set. Please note that ETUS is able to outperform all other models in terms of mean Dice score on the optic cup. Besides that, ETUS performs comparable for the optic disc and rather poorly for the mean absolute vCDR error. It is important to bear in mind that ETUS had an unfair advantage compared to the other models, as its data is transformed to the polar domain beforehand. To perform this transformation, the ground truth centre of mass of the optic cup is used as the centre point of transformation. In order to make up for this advantage, another preprocessing model would have to be trained, which specializes in finding the centre point. It is assumed for this Thesis that such a model exists and that it performs very accurately. In contrast to the other models, the baseline performs poorly, especially in terms of the mean absolute vCDR error. This is unsurprising, as the teams employed

Table 4.4: Performance of Baseline and ETUS at 100% training data, compared to the results of the REFUGE2 challenge [3], with the best result per column highlighted

Architecture	Dice OC	Dice OD	MAE of vCDR
cheeron [3]	0.865	**0.961**	0.055
MAI [3]	0.854	0.960	0.060
VUNO EYE TEAM [3]	0.845	0.960	0.058
MIG [3]	0.846	0.949	0.055
EyeStar [3]	0.831	0.939	**0.054**
MIAG ULL [3]	0.851	0.918	0.064
Baseline	0.810	0.914	0.102
ETUS	**0.889**	0.943	0.063

specialised architectures, whereas the baseline is a pre-implemented standard model.

As ETUS shows, it is possible to utilize prior knowledge to improve the performance of a model, even trained with few training samples. Compared to the other modifications, ETUS is rather far from the original architecture. It utilizes many ideas, some of which are not as intuitive as those of other modifications. Therefore, it is hard to tell whether the incorporation of a particular prior fact is helpful. It could be the case that the different approach ETUS uses (finding boundaries, instead of classical segmentation) is well suited for the task of segmenting the optic disc and optic cup. Nonetheless, the research question can be answered positively: Including priors is capable of reducing the amount of data needed to get to a reasonable performance, in terms of all chosen quality metrics. How can this insight be used outside this Thesis?

Even though the training data is reduced to simulate a setting where data is scarce, the evaluation is done on extensive test sets. Models, which are trained with extremely few samples, have a big variance in their performances. This could make the evaluation and selection of modifications tough in a real world setting. Furthermore, finding prior knowledge, which is suitable to be incorporated, can be non-trivial. It is not always the case that relations, such as the inclusion of the optic cup within the optic disc, are present. And even if prior knowledge is provided, it can be difficult to implement it as a loss, or use it in other ways.

5 Conclusion

The goal of this Master's Thesis is to investigate how prior knowledge can benefit image segmentation models, especially when only few training samples are available. Chapter 3 explains what prior facts are used for the task of segmenting the optic disc and optic cup in retinal images.

Indeed, incorporating knowledge via additional losses, or changes to the input data, can have an effect on the behaviour of the model. Most of the modifications have minor or even insignificant effects. However, the ETUS model, which furthermore has structural differences to the baseline (see Subsection 3.2.5), achieved very promising results, even for few training samples. It uses an alternative segmentation strategy compared to all other models in this Thesis, by solving a different version of the segmentation problem, which is shown to be successful. Generally, the performance of the models, in particular of the baseline, is surprisingly stable when reducing the fraction of training data, which is used for training. This is why the range of percentages of used training data was chosen between 1% and 10%.

Data can be a scarce resource, especially in medical research, where samples can be expensive and too sensitive to be used carelessly. The results of this Thesis show that a good model choice is a considerable alternative to mass collection of training data. However, the luxury of vast validation and test sets was still utilized, which increases the expressiveness of the results. It is still desirable to have many test samples, even when few training samples are needed. A clever model choice can hardly supplement the certainty of an extensive evaluation, which includes many test samples.

To use the results of this Thesis in the real world, the biggest challenge is to find a sensible way to incorporate prior knowledge. The segmentation of the optic disc and optic cup is a well studied field of research, where many ideas have already been investigated, such as the polar transformation. To extend the finding that incorporating prior knowledge can foster learning with only few data points, it is inevitable to try different approaches to come up with a good model. Even then, the lack of many test samples could pose a problem, as comparisons of models are less expressive.

Further research could employ even more methods which allow the model to use their data efficiently, like data augmentation, or applying an adversarial discriminator. The latter would push the model towards segmentations, which are difficult to tell apart from ground truth segmentations, thus forcing the model to learn details of how correct segmentations look like. Furthermore, it could be worth investigating how much additional training data is needed to reduce the effect of the domain shift to other datasets. For example, how many samples of the Chákṣu dataset would be needed for training to close the performance gap between the REFUGE2 test set and Chákṣu? Expanding research in the field of data-efficient models could yield relatively cheap indicators for a wide variety of conditions, leading to earlier detections, earlier treatments, and therefore a higher effectiveness of our healthcare system.

Bibliography

[1] O. Ronneberger, P. Fischer and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells and A. F. Frangi, Eds., Cham: Springer International Publishing, 2015, pp. 234–241.

[2] J. I. Orlando, H. Fu, J. Barbosa Breda *et al.*, "Refuge challenge: A unified framework for evaluating automated methods for glaucoma assessment from fundus photographs," *Medical Image Analysis*, vol. 59, p. 101 570, 2020, ISSN: 1361-8415. DOI: `https://doi.org/10.1016/j.media.2019.101570`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S1361841519301100`.

[3] H. Fang, F. Li, J. Wu *et al.*, *Refuge2 challenge: A treasure trove for multi-dimension analysis and evaluation in glaucoma screening*, 2022. arXiv: `2202.08994 [eess.IV]`.

[4] K. Allison, D. Patel and O. Alabi, "Epidemiology of glaucoma: The past, present, and predictions for the future," en, *Cureus*, vol. 12, no. 11, e11686, Nov. 2020.

[5] A. K. Schuster, C. Erb, E. M. Hoffmann, T. Dietlein and N. Pfeiffer, "The diagnosis and treatment of glaucoma," en, *Deutsches Ärzteblatt Int*, vol. 117, no. 13, pp. 225–234, Mar. 2020.

[6] M. D. Abràmoff, M. K. Garvin and M. Sonka, "Retinal imaging and image analysis," *IEEE Reviews in Biomedical Engineering*, vol. 3, pp. 169–208, 2010. DOI: `10.1109/RBME.2010.2084567`.

[7] I. Goodfellow, *Deep Learning.* MIT Press, 2016.

[8] F. Milletari, N. Navab and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*, Ieee, 2016, pp. 565–571.

[9] J. Lowell, A. Hunter, D. Steel *et al.*, "Optic nerve head segmentation," *IEEE Transactions on medical Imaging*, vol. 23, no. 2, pp. 256–264, 2004.

[10] S. Sekhar, W. Al-Nuaimy and A. K. Nandi, "Automated localisation of optic disk and fovea in retinal fundus images," in *2008 16th European Signal Processing Conference*, 2008, pp. 1–5.

[11] F. Yin, J. Liu, D. W. K. Wong *et al.*, "Automated segmentation of optic disc and optic cup in fundus images for glaucoma diagnosis," in *2012 25th IEEE international symposium on computer-based medical systems (CBMS)*, IEEE, 2012, pp. 1–6.

[12] J. Cheng, J. Liu, Y. Xu *et al.*, "Superpixel classification based optic disc and optic cup segmentation for glaucoma screening," *IEEE transactions on medical imaging*, vol. 32, no. 6, pp. 1019–1032, 2013.

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

[14] Z. Wu, C. Shen and A. Van Den Hengel, “Wider or deeper: Revisiting the resnet model for visual recognition,” *Pattern recognition*, vol. 90, pp. 119–133, 2019.

[15] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff and H. Adam, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[17] K. He, G. Gkioxari, P. Dollár and R. Girshick, “Mask r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[18] H. Fu, J. Cheng, Y. Xu, D. W. K. Wong, J. Liu and X. Cao, “Joint optic disc and cup segmentation based on multi-label deep network and polar transformation,” *IEEE Transactions on Medical Imaging*, vol. 37, no. 7, pp. 1597–1605, 2018. DOI: `10.1109/TMI.2018.2791488`.

[19] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[20] J. Son, W. Bae, S. Kim, S. J. Park and K.-H. Jung, “Classification of findings with localized lesions in fundoscopic images using a regionally guided cnn,” in *International Workshop on Ophthalmic Medical Image Analysis*, Springer, 2018, pp. 176–184.

[21] R. Girshick, “Fast r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[22] S. M. Shankaranarayana, K. Ram, K. Mitra and M. Sivaprakasam, “Joint optic disc and cup segmentation using fully convolutional and adversarial networks,” in *Fetal, Infant and Ophthalmic Medical Image Analysis: International Workshop, FIFI 2017, and 4th International Workshop, OMIA 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 4*, Springer, 2017, pp. 168–176.

[23] M. Tan and E. Le Q V, *Rethinking model scaling for convolutional neural networks. 2019*, 1905.

[24] H. Zhao, J. Shi, X. Qi, X. Wang and J. Jia, “Pyramid scene parsing network,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[25] Z. Gu, J. Cheng, H. Fu *et al.*, “Ce-net: Context encoder network for 2d medical image segmentation,” *IEEE transactions on medical imaging*, vol. 38, no. 10, pp. 2281–2292, 2019.

[26] H. He, L. Lin, Z. Cai and X. Tang, "Joined: Prior guided multi-task learning for joint optic disc/cup segmentation and fovea detection," in *International Conference on Medical Imaging with Deep Learning*, PMLR, 2022, pp. 477–492.

[27] R. Kamble, P. Samanta and N. Singhal, "Optic disc, cup and fovea detection from retinal images using u-net++ with efficientnet encoder," in *Ophthalmic Medical Image Analysis: 7th International Workshop, OMIA 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 8, 2020, Proceedings 7*, Springer, 2020, pp. 93–103.

[28] A. M. Wundram, P. Fischer, S. Wunderlich *et al.*, "Leveraging probabilistic segmentation models for improved glaucoma diagnosis: A clinical pipeline approach," in *Medical Imaging with Deep Learning*, 2024.

[29] X. Bian, X. Luo, C. Wang, W. Liu and X. Lin, "Optic disc and optic cup segmentation based on anatomy guided cascade network," *Computer Methods and Programs in Biomedicine*, vol. 197, p. 105 717, 2020.

[30] H. Lei, W. Liu, H. Xie, B. Zhao, G. Yue and B. Lei, "Unsupervised domain adaptation based image synthesis and feature alignment for joint optic disc and cup segmentation," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 1, pp. 90–102, 2021.

[31] R. El Jurdi, C. Petitjean, P. Honeine, V. Cheplygina and F. Abdallah, "High-level prior-based loss functions for medical image segmentation: A survey," *Computer Vision and Image Understanding*, vol. 210, p. 103 248, 2021.

[32] J. R. Clough, N. Byrne, I. Oksuz, V. A. Zimmer, J. A. Schnabel and A. P. King, "A topological loss function for deep-learning based image segmentation using persistent homology," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 12, pp. 8766–8778, 2020.

[33] X. Hu, F. Li, D. Samaras and C. Chen, "Topology-preserving deep image segmentation," *Advances in neural information processing systems*, vol. 32, 2019.

[34] N. Byrne, J. R. Clough, I. Valverde, G. Montana and A. P. King, "A persistent homology-based topological loss for cnn-based multiclass segmentation of cmr," *IEEE transactions on medical imaging*, vol. 42, no. 1, pp. 3–14, 2022.

[35] N. Stucki, J. C. Paetzold, S. Shit, B. Menze and U. Bauer, "Topologically faithful image segmentation via induced matching of persistence barcodes," in *International Conference on Machine Learning*, PMLR, 2023, pp. 32 698–32 727.

[36] A. H. Berger, N. Stucki, L. Lux *et al.*, "Topologically faithful multi-class segmentation in medical images," *arXiv preprint arXiv:2403.11001*, 2024.

[37] J. Deng, N. Ding, Y. Jia *et al.*, "Large-scale object classification using label relation graphs," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, Springer, 2014, pp. 48–64.

[38] A. BenTaieb and G. Hamarneh, “Topology aware fully convolutional networks for histology gland segmentation,” in *International conference on medical image computing and computer-assisted intervention*, Springer, 2016, pp. 460–468.

[39] C. Reddy, K. Gopinath and H. Lombaert, “Brain tumor segmentation using topological loss in convolutional networks,” 2019.

[40] Y. He, A. Carass, Y. Yun *et al.*, “Towards topological correct segmentation of macular oct from cascaded fcns,” in *Fetal, Infant and Ophthalmic Medical Image Analysis: International Workshop, FIFI 2017, and 4th International Workshop, OMIA 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 4*, Springer, 2017, pp. 202–209.

[41] S. Gupta, X. Hu, J. Kaan *et al.*, “Learning topological interactions for multi-class medical image segmentation,” in *European Conference on Computer Vision*, Springer, 2022, pp. 701–718.

[42] M. N. Zahoor and M. M. Fraz, “Fast optic disc segmentation in retina using polar transform,” *IEEE Access*, vol. 5, pp. 12 293–12 300, 2017.

[43] M. Benčević, I. Galić, M. Habijan and D. Babin, “Training on polar image transformations improves biomedical image segmentation,” *IEEE access*, vol. 9, pp. 133 365–133 375, 2021.

[44] Q. Liu, X. Hong, W. Ke, Z. Chen and B. Zou, “Ddnet: Cartesian-polar dual-domain network for the joint optic disc and cup segmentation,” *arXiv preprint arXiv:1904.08773*, 2019.

[45] Y. He, A. Carass, Y. Liu *et al.*, “Structured layer surface segmentation for retina oct using fully convolutional regression networks,” *Medical image analysis*, vol. 68, p. 101 856, 2021.

[46] J. H. Kumar, C. S. Seelamantula, J. Gagan *et al.*, “Chákṣu: A glaucoma specific fundus image database,” *Scientific data*, vol. 10, no. 1, p. 70, 2023.

[47] S. K. Warfield, K. H. Zou and W. M. Wells, “Simultaneous truth and performance level estimation (staple): An algorithm for the validation of image segmentation,” *IEEE transactions on medical imaging*, vol. 23, no. 7, pp. 903–921, 2004.

[48] D. P. Huttenlocher, G. A. Klanderman and W. J. Rucklidge, “Comparing images using the hausdorff distance,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 15, no. 9, pp. 850–863, 1993.

Acronyms

CDR cup-to-disc-ratio. vii, viii, xii, 33, 34, 54–56, 59, 60, 62

CNN convolutional neural network. 9, 10, 16

ETUS **E**nforcing **T**opology using **U**-Net **S**egmentation. ix, xi, xii, 35–37, 56–63, 65

GAN generative adversarial network. 18

MAE mean absolute error. 63

MONAI Medical open network for AI. 28

OC optic cup. 15–17, 63

OCT optical coherence tomography. xi, 22, 23, 35

OD optic disc. 15–17, 63

ONH optic nerve head. 15, 16

STAPLE Simultaneous Truth and Performance Level Estimation. 27

vCDR vertical cup-to-disc-ratio. ix, xii, 3, 4, 13, 14, 33–35, 40, 41, 46, 47, 49–58, 60, 62, 63