Optical Coherence Tomography: Prediction of Progression of Diabetic Retinopathy Severity Through Machine Learning

by

Cyrus Eberth WaChong

Thesis Proposal Submitted in Partial Fulfillment of the
Requirements for the Degree of
Bachelor of Applied Science

in the
School of Engineering Science

Attached: Thesis

cwachong@sfu.ca

301306459

Abstract

With advancements in technology and computation power, machine learning is becoming increasingly popular in clinical environments due to its ability to generalize data and provide timely and accurate results, removing some of the workload from medical personnel. In this thesis, two machine learning methods were tested and compared for the prediction of diabetic retinopathy (DR) progression. Vascular and avascular biomarkers were extracted from optical coherence tomography angiography (OCTA) en face images to use as input for one of the machine learning methods, as well as a verification method of clinical personnel. The first method involved training a single-hidden layer feedforward neural network (SLFN) that was trained using the extracted biomarkers; the second method used a deep neural network (DNN) trained directly on pre-processed OCTA en face images. The final ResNet50 image-based DNN predicted the progression of DR with an accuracy of 85.4%, specificity of 97.62%, and sensitivity of 16.67%.

Acknowledgements

I would like to sincerely thank all of those that supported me throughout the process and completion of this undergraduate thesis project. The completion of this project was only possible with the support of many individuals.

First, I would like to sincerely thank my supervisor, Dr. Marinko V. Sarunic, for kindly accepting me as an undergraduate thesis student. His patience, technical expertise, constant support, and guidance all helped me stay on track and gave me the confidence needed to persevere through the problems that appeared during the course of this thesis.

I would also like to express my gratitude for my committee members, Dr. Mirza Faisal Beg and Dr. Myeong Jin Ju for their valuable encouragement, support, and time. They have been crucial to the completion of this thesis project.

I would like to thank the Biomedical Optics Research Group (BORG) as a whole for their support throughout this project. I am especially thankful to BORG member Mr. Timothy Yu, who has been incredibly supportive throughout this entire project, always available to help me through technical issues, conceptual issues, and anything else that I needed help with. Timothy also provided parameters extracted using pipelines he designed, such as the retinal biomarkers extracted from OCTA en face images. I would like to also thank BORG member Mr. Julian Lo for his support during the weekly meetings and for providing the segmented en face OCTA images that were crucial for the completion of this project. Additionally, I would like to thank Dr. Da Ma for his constant support during this process, always helping clarify key concepts and ideas.

Finally, I would like to express my gratitude for my family for their support, patience, and love for me. Their support during the hard and frustrating moments truly pushed me to complete this project.

Table of Contents

List of Tables:

List of Figures:

Glossary:

AI: Artificial intelligence

CNN: Convolutional neural network

DA: Diagnostic accuracy

DCP: Deep capillary plexus

DNN: Deep neural network

DR: Diabetic retinopathy

ETDRS: Early treatment diabetic retinopathy study

FAZ: Foveal avascular zone

LR: Learning rate

ML: Machine learning

NPDR: Non-proliferative DR

NRDR: Non-referable DR

OCT: Optical coherence tomography

OCTA: Optical coherence tomography angiography

PDR: Proliferative DR

RDR: Referable DR

ReLU: Rectified linear unit

SCP: Superficial capillary plexus

SE: Structural element

SLFN: Single-hidden layer feedforward neural network

TN: True negative

TP: True positive

Chapter 1: Introduction

1.1 Overview

Diabetic Retinopathy (DR) is a complication of diabetes that causes damage to the blood vessels of the retina [1]. DR can lead to vision loss and, if left untreated, can lead to blindness. It is estimated that by 2050, the number of people suffering from DR will double from 7.7 million to 14.6 million [2], with nearly 415 million diabetic patients at risk worldwide [2][3]. DR is the leading cause of visual impairment and blindness in working-age adults in the USA [2] and can lead to other serious eye conditions, such as diabetic macular edema (DME), neovascular glaucoma, and could result in retinal detachment [4]. Caused by high blood sugar content in the blood, over time DR will damage the blood vessels in the retina and optic nerve. These vessels can swell, leak, and lead to ischemia (restriction of blood flow to tissues that may cause permanent damage), all of which can cause loss of vision due to limited or non-existent blood supply. These are the early stages of DR, known as non-proliferative diabetic retinopathy (NPDR), the stage in which vision becomes blurry [5]. If DR is not treated early on, it may progress to proliferative diabetic retinopathy (PDR), where new blood vessels are created to compensate for the lack of blood flow. These new blood vessels often leak into the vitreous, and if left untreated, can cause blind spots in the eye, or even complete blindness.

DR severity is clinically classified using five severity levels (5-ary classification): normal, mild, moderate, severe, and proliferative DR (PDR). Alternatively, DR can be classified using three severity levels (3-ary classification): normal, non-proliferative DR (NPDR), and PDR. These two classifications are related; NPDR encompasses the mild, moderate, and severe classifications of the 5-ary severity model since these severities cause minimal to no symptoms in the patients, whereas PDR patients experience symptoms such as vision loss due to proliferating vessels [1][6].

Currently, methods for diagnosing and determining severity of DR involve ophthalmologists manually reviewing a patient's retinal images. The objective of this project is to design a deep neural network method that will automatically provide quantitative parameters describing its prediction of the progression of a patient's DR using retinal images. These results can expedite the time-consuming process of manually reviewing images by providing evidence-based results that can be used to predict the progression of DR accurately and efficiently in a patient.

1.2 Publications on Classification in Ophthalmology

Machine learning has been applied in many medical fields since it is capable of leveraging data that has already been acquired to provide evidence-based results much more efficiently than traditional methods. To illustrate, machine learning algorithms have been applied to successfully classify severity of DR using fundus images. A study conducted by Google performed a binary classification between referable DR (RDR) and non-diseased fundus images. The network had an F-score (A combined sensitivity and specificity metric with a max value of 1) of 0.95, while the ophthalmologists received an F-score of 0.91 [3]. Another publication tested multiple convolutional neural network (CNN) architectures with pretrained weights; the network with the best performance achieved a validation sensitivity of 95% and a specificity of 96% [7]. This study also performed DR classification with 3-ary and 4-ary classifiers of disease severity, (classification of three and four levels of DR severity, respectively) achieving sensivities of 85% and 75% for no DR and severe DR respectively [7] but only 29% mild class sensitivity. The study could not achieve

comparable results with the 4-ary classifier due to the lack of training data needed to effectively train a CNN. It is important to realize that the most cutting-edge studies were effective due to utilizing large dataset. Googles network used a dataset of 128,000 fundus images graded for DR [3], while the second study mentioned used an open source dataset composed of 35,000 retinal images graded for DR [7]. Although these studies, along with many of the other publications, have demonstrated high accuracies in their results, they simply make use of cross-sectional datasets instead of longitudinal datasets, therefore these networks cannot predict the progression of DR in patients. An example of a healthy and diseased fundus image is presented in Figure 1.

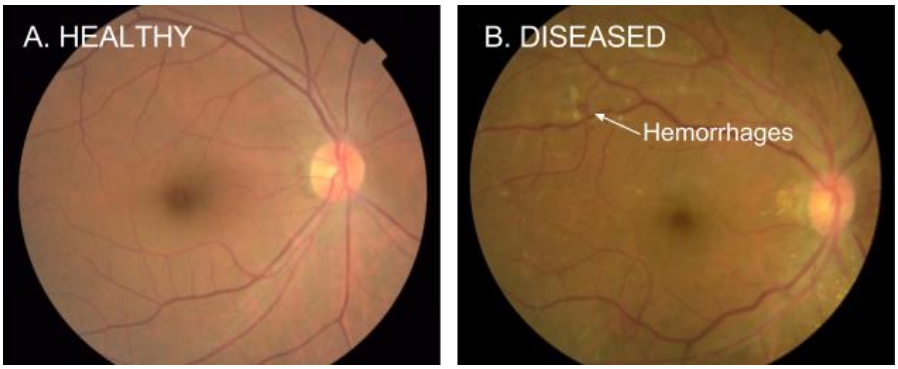

Figure 1: Healthy retina compared to retina of patient with DR [3].

Fundus images are normally obtained alongside Optical Coherence Tomography (OCT) and OCT Angiography (OCTA). They are acquired using a fundus camera which records colour images of the interior surface of the eye [8]. OCT and OCTA images, representative examples of which are shown in Figure 2, provide further detail on the retina, including cross-sectional information, and more detailed microvasculature of the retina.

The classification examples described above do not utilize optical coherence tomography (OCT) and OCT angiography (OCTA), which can provide information on the individual layers of the retina, as well as fluid flow through the retinal blood vessels. Fundus images do not reveal microvascular detail, and since DR is a disease of the microvasculature, OCT and OCTA can be used together to provide additional information of the microvasculature for diagnosis and prognostication.

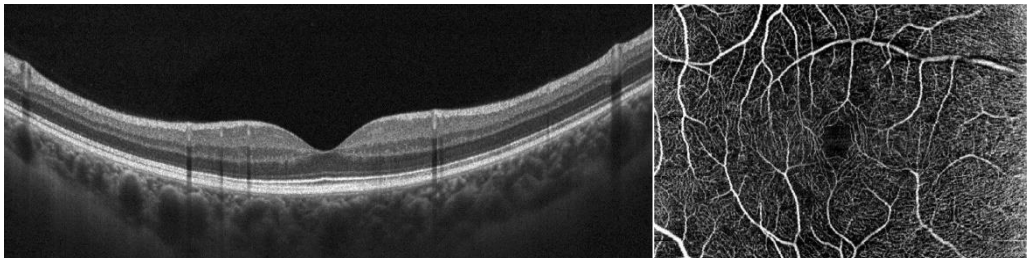

Figure 2: Left to right: OCT B-scan, en face OCTA [9] [10].

1.3 OCT and OCTA Image Acquisition

OCT is a non-invasive imaging technique that produces cross-sectional images, allowing an ophthalmologist to view the patient's distinctive retinal layers [11]. OCT is akin to a time-of-flight delay from the reflections on the retinal boundary layers to create volumetric images of the retina. These

volumetric scans are composed of multiple cross-sectional 2D images, known as B-scans. B-scans are composed of many A-scans, the vertical columns of the image that are produced by a focused light source which map the reflective profile of the sample in depth [12][13][14]. This focused light is moved laterally across the surface of the retina, acquiring a series of A-scans. There are two main OCT methods, namely time domain (TD)-OCT, and Fourier domain (FD)-OCT [12]. TD-OCT is normally performed by mechanical means, whereas the FD-OCT can be spectrometer based (so called Spectral Domain, SD) or implemented using a turntable laser or a swept source (SS). While these methods vary with how they are conducted and their advantages and disadvantages, the theory behind OCT remains the same. OCT uses low-coherence light (a broad range of wavelengths) typically in the near infrared (NIR) spectrum, allowing increased penetration of the retinal tissue. The light from the source is split into two by an optical splitter, the output is then fed into the arms of an interferometer [12][15]. One arm directs light to the tissue, while the other arm directs light to a reference mirror. The backscattered light is then combined with the light from the reference arm to generate an interference pattern [12][15]. This interference signal is made up of light that has travelled the same distance in both arms of the interferometer. By adjusting the reference mirror, the distance the light travels in the reference arm changes. Therefore, different depths of the retina can be chosen for imaging. Figure 3 demonstrates and labels the thirteen layers that are typically seen in OCT images.

Retinal Layers	
Abbr.	Name
ILM	Internal Limiting Membrane
RNFL	Retinal Nerve Fibre Layer
GCL	Ganglion Cell Layer
IPL	Inner Plexiform Layer
INL	Inner Nuclear Layer
OPL	Outer Plexiform Layer
ONL	Outer Nuclear Layer
ELM	External Limiting Membrane
PR	Photoreceptor Layers
RPE	Retinal Pigment Epithelium
BM	Bruch's Membrane
CC	Choriocapillaris
CS	Choroidal Stroma

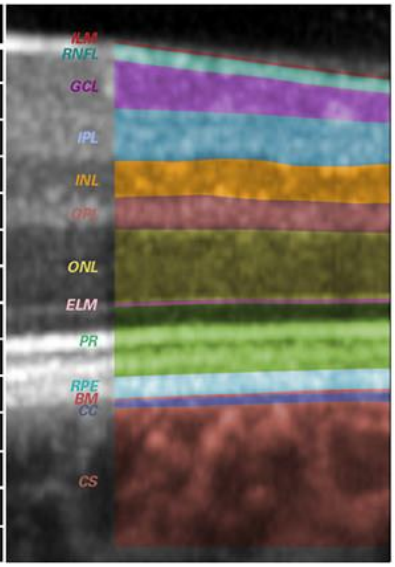

Figure 3: All layers of the retina overlaid on an OCT B-scan.

OCTA is an extension of OCT, providing the same structural B-scans, while also providing information on volumetric blood flow in the microvasculature of the retina and choroid without the use of injectable dyes. OCTA images are created by comparing sequentially acquired B-scans at the same location of the retina to visualize the retinal blood flow[16][17]. Locations that do not contain vasculature have minimal or no variation, resulting in low intensity pixels in the OCTA image; however, locations that correspond to blood vessels have larger changes in reflection due to motion of blood cells, resulting in a high intensity pixel in the OCTA image[18].

Many methods of imaging and analyzing the volumetric blood flow have been developed since the beginning of OCT imaging. Split-spectrum amplitude-decorrelation angiography (SSADA), a method of

imaging developed at the Oregon Health & Science University (OHSU) Casey Eye Institute, is a method of imaging which involves splitting the full OCT spectrum into narrower bands and calculating decorrelation of the spectral bands. SSADA significantly improved both the SNR for flow detection and connectivity of the microvasculature compared to other algorithms[19]. A second method, pioneered by Dr. Ruikang Wang, involved imaging the backscattered signals created by moving blood cells in the blood vessels [20]; this imaging technique is called Optical microangiography (OMAG).

Unlike OCT images that are normally viewed as cross-sections, OCTA images are commonly viewed in the *en face* plane. These *en face* images are generated by segmenting the retinal layers using of each B-scan in a volume (as presented in Figure 3), then these coordinates are then transferred to the corresponding OCTA volume. The *en face* image is then generated by projecting the OCTA volumetric data onto a 2D plane.

The OCTA data is used to analyze the vascular changes at different layers of the retina. These vascular changes are strong indicators of DR [18]. To fully analyze these results, multiple OCTA *en face* plexuses are examined, comparing the correlation between vascular parameters, and clinically diagnosed severity of DR. There are two commonly used *en face* plexuses for analysis, namely the superficial capillary plexus (SCP) which is the capillary network held within the ganglion cell layer and/or the nerve fiber layer, and the deep capillary plexus (DCP), which is located in the inner nuclear layer [18].

1.4 Classification of Severities of DR

Studies highlight the importance of OCTA *en face* images as a method of early diagnosis for DR, since abnormalities which cannot be seen in fundus nor OCT images can be used for diagnosis before structural abnormalities occur [21]. These abnormalities/biomarkers that are only found in OCTA images include microaneurysms, foveal avascular zone (FAZ) size, and capillary features. A series of OCTA *en face* images of eyes with varying severities of DR are demonstrated in Figure 4.

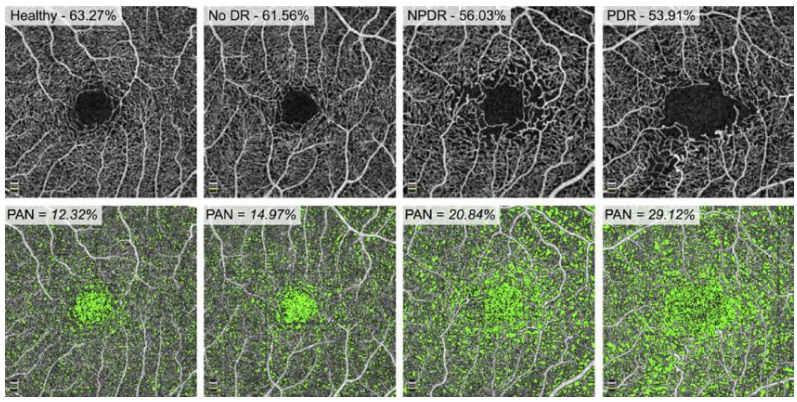

Figure 4: OCTA en face images of patients with varying severities of DR (top), areas of nonperfusion shown in green (bottom) [22].

As explained in Section 1.1, DR severity is normally classified using 3 or 5 classes; due to a limitation on OCTA *en face* images available for this thesis project, a binary (2-ary) classification approach will be taken. Since the goal of this project is to predict the progression of DR, and not to classify, the two classes of input

data will be eyes that experienced DR progression, and those that did not experience DR progression. The definition of DR progression was explained in Chapter 1.5.

1.5 High-Level Overview of this Thesis Project

The overall purpose of this thesis is to design a machine learning algorithm to predict the progression of DR using OCTA images. The project targets a binary classification between progression of DR, and no progression of DR. The data supplied for this project followed a 15-step DR severity scale, and DR progression is defined as an increase in 2 or more steps of severity level compared with the baseline image [23].

Two machine learning approaches were taken to solve this problem. The first approach involved extracting parameters from the OCTA images that related to vessel features, which were used to predict the progression. The second approach used OCTA images as direct input into the model for classification. An important different between the methods is that the first requires hand-crafted features, whereas the second method relies on the machine learning algorithm to identify the relevant features for classification.

Thesis Goal

Predicting the progression of DR is difficult, as diagnoses vary between clinicians. To provide a useful assistive tool for prediction of DR progression, I aim to design a neural network to predict DR progression of a patient as defined above with an accuracy of 75%

1.6 Contributions

My work for this project began as an undergraduate thesis student with the BORG. The initial code for extracting relevant parameters, such as FAZ area, perimeter, and minimum/maximum diameter was provided by Timothy Yu. I performed manual segmentation to assist with Julian Lo's vessel segmentation DNN which is used to segment the vasculature in OCTA *en face* images. Timothy Yu provided the extracted biomarkers from the data supplied for this thesis project.

Work done for my ENSC undergraduate thesis includes the pre-processing script to create the input specified in Chapter 3.3, the scripts used to find biomarkers correlated to DR progression, and all the implementations specified in Chapters 4 and 5 of the parameter-based and image-based neural networks.

1.7 Thesis Organization

This thesis is organized as follows: Chapter 2 will present background information related to artificial intelligence and machine learning methods, including convolutional neural networks, and the elements that compose them. Chapter 3 will present the dataset, along with how the data was acquired, and what the criteria was for the subjects. Chapter 3 will also cover the extracted OCTA retinal biomarkers that are correlated to the progression of DR (and thus will be used as input). Chapter 4 details the methods and results of a parameter-based neural network to predict the progression of DR in a patient. Chapter 5 covers the image-based neural network, including the architecture and the methods used to tune the network to maximize performance. Chapter 6 presents the final results, the challenges faced during this project, the potential future work, and a conclusion.

Chapter 2: Introduction to Artificial Intelligence and Machine Learning

Diagnosis and prognosis of retinal images is a challenging and human-error prone task. To aid clinicians with these complicated tasks, machine learning (ML) will be used in this project. This chapter will introduce the applications of machine learning and deep neural networks (DNNs) in this project, as well as their limitations, how performance is analyzed, and ultimately how the top performing networks were identified.

An important subcategory of DNNs called convolutional neural networks (CNNs) will also be presented, and their importance to the project will be highlighted. The purpose of each layer of the CNN will be explained, along with a graphical representation and the methods that were applied for maximum performance. Other important design features such as activation functions and hyperparameters will be explained.

2.1 General Background and Biomimicry

Artificial intelligence (AI) is the endeavor to replicate or simulate human intelligence in machines. AI is an incredibly broad term, describing any technology that its environment and takes action to maximize its chance of success of achieving a certain defined goal. Machine learning (ML), a field that is currently quite popular in the medical field, is a subset of AI. ML is a field of computer science revolving around computer algorithms that improve automatically through experience without being explicitly programmed.

Many of the current approaches in the field of ML are attempting to replicate the way in which humans think. Biomimicry is the imitation of nature and is used extensively in the progress of ML. Neural networks, the backbone of ML, are a form of biomimicry of the human brain; In the human brain, input signals are received at the dendrites of neurons, where they are summed with other inputs. If the summation of these signals surpasses a certain threshold, the cell will depolarize and fire an action potential. Mimicking this process, neural networks used in ML consist of layers of nodes acting as neurons which connect to other nodes. These nodes fire using weighted summation once activation thresholds are exceeded. These neural networks are improved through an iterative training process which optimizes the weighted connections between nodes to best represent the provided training dataset.

2.2 Single Hidden Layer Feedforward Neural Networks and Deep Neural Networks

Single hidden layer feedforward neural networks (SLFNs) make use of nodal architectures with one hidden layer (layer in between the input and output layers). The input layer will receive the input values, which are then fed forwards into the hidden layer. The hidden layer is tasked with detecting patterns and optimizing/tuning the weighted connections to the output layer's nodes to best represent the training data. The output layer then receives and summates all the outputs from the hidden layer, it is then tasked with classification of the data to best match the ground truth paired with the input data. The number of output nodes is determined by the number of classes specified by the classification problem. Figure 5 (left) demonstrates a typical SLFN used to classify handwritten digits in the Modified National Institute of Standards and Technology (MNIST) dataset. The input layer has 784 nodes, each corresponding to a pixel in the 28x28 input image of a handwritten number "9". The hidden layer will detect segmented sections of the number "9", and upon weighted summation of the input nodes that exceed a threshold, the hidden node that represents a certain feature will activate. The final layer corresponds to the 10 possible numbers that can be classified, which will be determined by the combination of features that were detected in the hidden layer.

Deep neural networks (DNNs) are a subset of artificial neural networks which contain more than one hidden layer; the input and output layers remain the same as SLFNs. DNNs naturally contain more parameters, such as the weighted connections between neurons, due to the increased number of neurons and layers. This allows DNNs to solve more complex problems than SLFNs can with increased efficiency since they can detect higher level and more complex features in the images. Each hidden layer detects sub-features from those that were detected in previous layers, which can be useful in more complicated applications or problems. A DNN with an additional hidden layer is presented in Figure 5 (right), where the second layer receives the results from the previous layer and uses them to recognize more complex features, such as a circle and a line. The recognition of these more complex features results in less overall input into to the output nodes, limiting variability and error, ultimately increasing accuracy. The one major issue with deeper neural networks is that they require more data to generalize the dataset effectively.

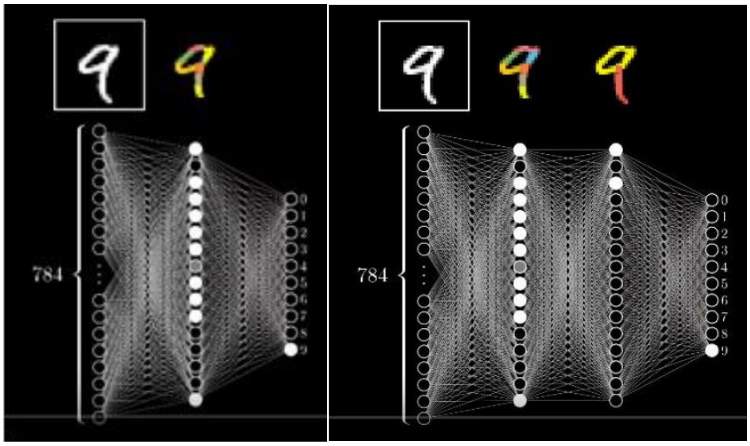

Figure 5: Simplistic example of a single hidden layer feedforward neural network (left) and DNN (right) used to classify the Modified National Institute of Standards and Technology (MNIST) dataset [24].

2.2.1 Types of Machine Learning

There are three types of machine learning: supervised learning, semi-supervised/reinforcement learning, and unsupervised learning. Supervised learning requires training data paired with corresponding ground truth data for verification and tuning of the network. Semi-supervised learning is normally used when there is an abundance of training data, but minimal labelled ground truth to accompany it. By providing both groups of data for training, the network will use both groups of data and produce improved results compared to when just labelled data is used. Unsupervised learning analyzes large batches of data that has no corresponding ground truth; the network is intended to detect patterns in the data.

2.2.2 Types of Machine Learning Problems

As with common types of machine learning, there are common problems that these networks seek to solve, namely: classification, regression, and structured output. Classification problems are solved by neural networks called classifiers which train on provided input data, and ground truth data indicating which class the training input belongs to. These classifier networks can then be used to predict the class of future inputs. Regression neural networks predict a continuous output variable, making these networks commonly used to predict things such as the stock market prices, or housing prices. Regression predictive modelling is the task of approximating a mapping function that relates input variables to a continuous output variable. Structured output neural networks are mainly used for segmentation and identification, examples being

OCT layer segmentation, OCTA vessel segmentation, both of which are discussed and utilized in this project.

2.2.3 Types of Classification

There are three types of classifiers: binary, categorical, and multiclass. Binary classifiers have two potential output values, categorical classifiers have more than two output classes, and multiclass classifiers can identify multiple different classes within a single image.

2.2.4 Overfitting and Underfitting

Overfitting refers to a model that has too closely tuned its parameters and weighted connections towards the input data, failing to generalize to the potential data relevant to the current problem. This normally occurs with limited datasets, where excessive training can cause the model to "memorize" the dataset or tune the model to the noise and artifacts of the training data, all of which will ultimately reduce the ability of the network to generalize and model potential future inputs. Common approaches to limit overfitting include reducing the complexity of the network, applying dropout layers, using callback functions to optimize the epoch count, and if possible, training with more data. Underfitting on the contrary describes the network not properly tuning parameters due to lack of exposure to the training data, which results in poor outcomes with future data. This is normally corrected by analyzing the validation loss function and increasing the overall number of epochs of which the network trains.

2.2.5 Data Augmentation

Data augmentation is used to provide variability to the input data by means of flipping, translating, cropping, and rotating, among other operations prior to training. These techniques are pivotal to increasing the training dataset size, which aids in reducing overfitting by providing more unique training data to the network to help further generalize its mapping of potential input data.

2.3 Convolutional Neural Networks

Machine vision is a field of study experiencing exponential growth as technology advances and computation power increases. Machine vision applications can currently be found in Tesla's self driving cars, automated assembly lines, surveillance systems, and many other places. The input data for these networks may have extremely high resolutions, each pixel being crucial to the overall output of the network. CNNs, which are subsets of DNNs, are algorithms that reduces the overall resolution of an input image into a form that is simpler to process, while maintaining crucial information and characteristics of the image [25]. CNNs accomplish this by organizing neurons in three spatial dimensions: width, height, and depth.

A regular neural network trained on a volume consisting of multiple images appended is computationally demanding, as each pixel of the volume corresponds to an input node. Not only is this approach computationally demanding, but it is also much more prone to overfitting. CNNs replicate the connectivity patterns of neurons in the visual cortex, where individual neurons respond to stimuli only in a restricted region of the visual field; a collection of these fields overlap to cover the entire visual area. This approach minimizes the connection between neurons when compared to fully connected layers, making them less computationally demanding while lowering the chances of overfitting.

2.4 Transfer Learning

Transfer learning is another form of biomimicry; the human brain uses previously trained neural connections to transfer knowledge to other similar tasks [26], similar to what we attempt to replicate with transfer learning. Neural networks apply this concept with transfer learning techniques to increase the performance and speed of training for problems with smaller datasets. The application of transfer learning for artificial intelligence has been thought of since the creation of the earliest neural networks; Companies such as Google, Tesla, and Facebook leverage their large datasets to create generalizable base models and weights for transfer learning. The success of big data has provided many researchers with the opportunity to explore the potential of machine learning. ImageNet is a prime example of research conducted with a large dataset to provide a great base for most machine vision projects. The ImageNet network is a classifier that was trained with 14 million images pertaining to thousands of classes, the difficulty here was to create a network large enough to generalize all the classes, allowing the network to identify each class independently. This project made use of VGG-16, a 16-layer model developed by Oxford University, to attempt to generalize this dataset.

When training a neural network, especially in the medical field, many bottlenecks can present themselves, such as time needed to acquire data, number of patients willing to be imaged, patient data sensitivity, etc. For this project, there was a strict limit on the size of the dataset due to the number of patients that were imaged as a baseline, and also imaged after 6 months. Cross-sectional data is already difficult to acquire, but requiring the second imaging session after a specified period introduces more uncontrollable factors, such as patient deaths, patients who do not return, etc. This small amount of data has the potential to cause overfitting of the network but can be combatted by transferring already trained weights from a network trained on a larger, related dataset. A less complex DNN is trained for classification of extracted OCTA image features. One major issue that affects transferability is the complexity of deeper neurons. Naturally, the deeper hidden layers focus on more complex features of the input, reducing the probability that the features that are transferred are directly related to the new problem. This can be alleviated by “freezing”, or locking the weights of earlier, shallower layers of a model trained on a related task, therefore only tuning the finer, more complex features, using the features acquired from shallower layers as inputs to the deeper, trainable layers. The number of layers to retrain is dependent on the dataset size, and the similarity between the data of both the pretrained network and the new application [27].

2.5 Classification Performance Metrics

There are common classification performance metrics used to represent the overall performance of an examination, such as diagnostic accuracy (DA), specificity, and sensitivity. The classification is considered to be correct when the test result is positive, and the disease is present (true positive; TP), or the test result is negative, and the disease is not present (true negative; TN). There are also two cases in which the classification is considered to be incorrect, this is when the disease is not present, but the test result is positive (false positive; FP), and when the disease is present, but the test result is negative (false negative; FN). The contingency matrix in Figure 6 illustrates the performance metrics.

		Disease	
		+	−
Test	+	TP	FP
	−	FN	TN

Figure 6: Contingency matrix visualizing diagnostic accuracy, specificity, and sensitivity [28].

As mentioned, one important metric used to analyze the classification performance of a neural network is diagnostic accuracy (DA). DA (Eq. 2.1) is the ratio between the number of subjects that the test classifies accurately, and the total number of subjects tested [28],

$$DA = \frac{TP + TN}{TP + FP + FN + TN} \tag{2.1}$$

Sensitivity and specificity are two evaluation metrics to track the performance of a set of diagnoses. Sensitivity (Eq. 2.2) describes the tendency of a test to classify a subject with the disease accurately,

$$Sensitivity = \frac{TP}{TP + FN} \tag{2.2}$$

Conversely, specificity (Eq. 2.3) relates to the tendency of a test to classify a non-diseased subject correctly,

$$Specificity = \frac{TN}{TN + FP} \tag{2.3}$$

2.6 Convolutional Layer

Convolutional layers are used in many image-based neural network applications due to their capabilities to retain relevant information while being computationally efficient. Convolutional layers use kernels, a sliding, weighted window, to perform convolutions which ultimately detect the presence and location of features in the input. These convolutions are linear time-invariant (LTI) systems; the system combines its input with a matrix using matrix multiplication. Since the convolution is considered an LTI system, in signal processing terms, its impulse response is referred to as a filter or kernel, as it is commonly referred to in the field of machine learning. The number of kernels used for feature detection is determined by the number of input channels. Figure 7 demonstrates the process of reducing the resolution of the input through convolution, while retaining relevant information and features.

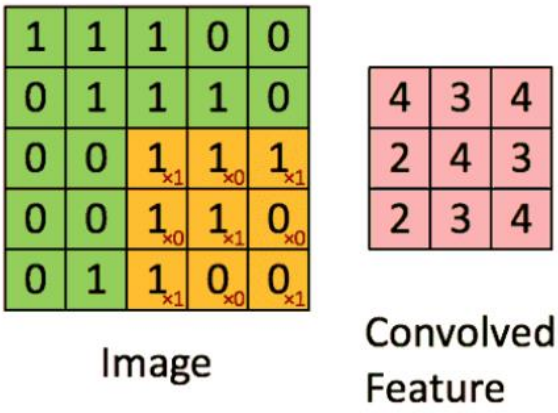

Figure 7: Convolution of a 5x5x1 image with a 3x3x1 kernel (weights in red characters in yellow boxes) with a stride of 1 to output a 3x3x1 convolved feature [25].

The kernel travels across the image, performing matrix multiplication with the kernel and overlapping portion of the input. The output of this process is a matrix representing the convolution between the kernel and the input, also considered the convolved feature, or feature map. This process is illustrated in Figure 8. [29].

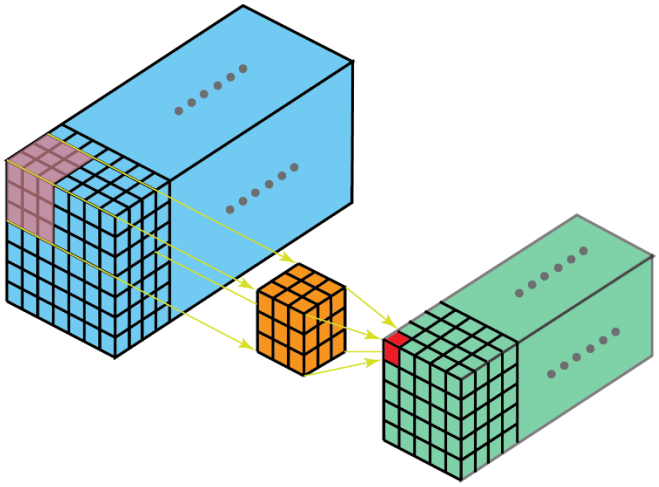

Figure 8: A kernel moving along its path, following the defined stride parameters, to produce a feature map [29].

This process of convolving the kernel with the input is what differentiates the CNN from other DNNs. The convolutional layer contains four main parameters:

- Kernel width, height, and depth: W_{kernel}, H_{kernel}, and D_{kernel}
- Stride length: S
- Input padding: P
- Number of kernels: N

2.6.1 Kernel Size

Just as weighted connections between neurons are tuned for regular DNNs, convolutional layers have trainable kernels that are finely tuned for feature detection and mapping. The size of these kernels determines the receptive field of each node to say how much of the image it will analyze for features with each stride. This ultimately reduces the overall number of connections between the neural layers, minimizing computation requirements and loss of information.

2.6.2 Stride Length

Stride describes the movement of the kernel as it convolves with the input, defining the number of pixels the filter shifts with each convolution. Larger stride lengths reduce the number of overlapping pixels in the receptive field, demonstrated in Figure 9. Increasing the stride length down-samples the image, reducing the overall resolution of the output, and minimizing computational load [30].

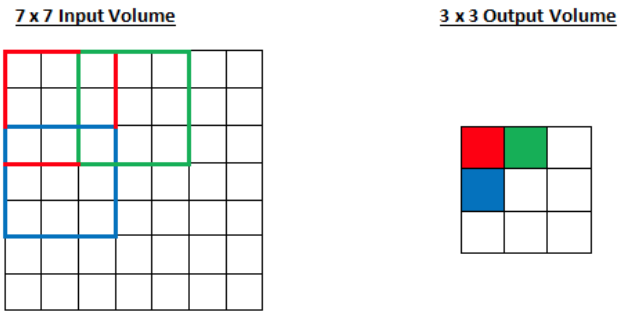

Figure 9: Visualization of convolutional filter with a stride of 2 convolving with a 7x7 input image [30].

2.7 Pooling Layers

Pooling layers target the variation between different input data. Artifacts may be present in input data, normally caused by errors introduced by technology, hardware operators, or problems during image acquisition. Pooling layers work by down-sampling the feature maps that are provided as input, preventing the overall neural network from overfitting to the artifacts present in the image. Two types of pooling layers that are commonly used are the max pooling layers and the average pooling layers, which detects the maximum value or averages the values within the pooling range, respectively. The effects of max pooling are demonstrated in Figure 10.

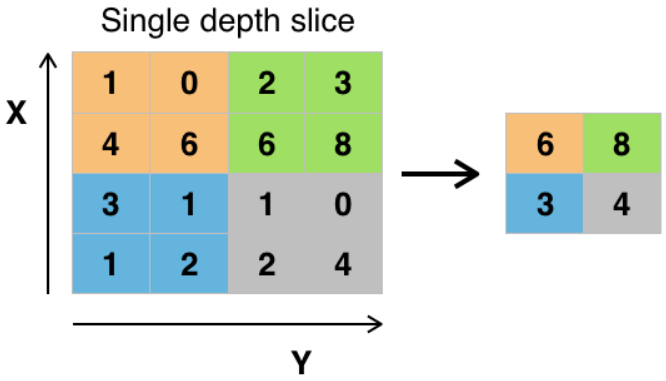

Figure 10: Example of max pooling with filter size and stride of 2 [30].

2.8 Dropout Layers

Dropout layers, as many other layers in CNNs such as pooling layers, are used to reduce overfitting and limit computational load. Unlike pooling layers, dropout layers will randomly ignore certain nodes, simulating a nosier dataset, and forcing the nodes to focus on more important features of the input without memorizing them, therefore minimizing the networks tendency to overfit to the dataset [31]. Figure 11 compares a regular fully connected neural network to one with dropout layers implemented.

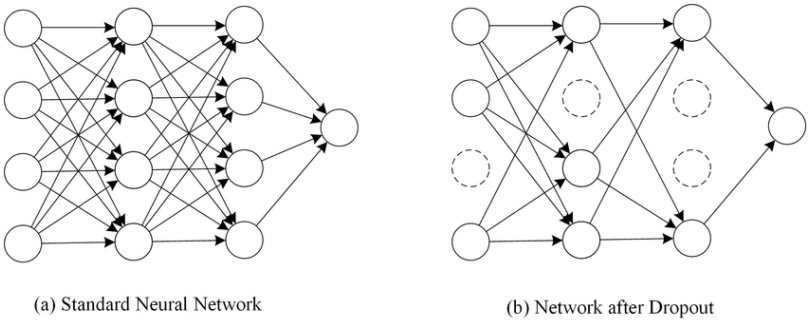

Figure 11: Standard fully-connected neural network (left), feedforward neural network with dropout layers (right) [32].

2.9 Fully Connected/ Dense Layer

Fully connect layers, also known as dense layers, are the hidden layers of DNNs where each neuron from the previous layer connects to every neuron of the next layer. An example of a fully connected layer is presented in Figure 12, where each one of the weighted connections is tuned during training.

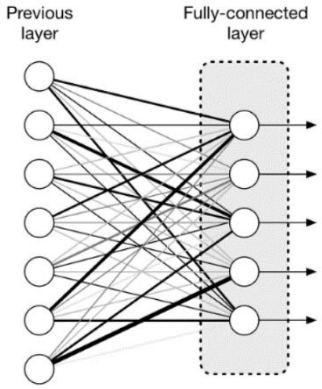

Figure 12: Example of a fully connected layer with their corresponding weighted connections [33].

Fully connected layers are commonly used at the final parts of a neural network architecture to concatenate all features from the preceding layers to form an N dimensional vector representing the number of classes in the problem. The fully connected layer combines features detected in the input image detected in the previous layers, attempting to generalize the results to calculate the confidence level for each class. Each node in the output layer will correspond to the networks confidence of the input image pertaining to a certain class, therefore the class with the largest confidence level will be considered the class the input image belongs to; often a confidence threshold will be implemented to increase the overall accuracy of the network. A fully connected layer summates the multiplication of each neuron value of the previous layer

$h^{(n-1,k)}$ with corresponding weighted connection $w^{(n,k)}$ with an added bias $b^{(n,k)}$. This layer requires an activation function to decide whether the neuron outputs a value ("fires") or not.

$$h^n = \sum_{k=1}^{i} h^{(n-1,k)} w^{(n,k)} + b^{(n,k)} \tag{2.4}$$

Where n is the current layer, and k is the current node.

2.10 Activation functions

Considering Equation 2.4, which describes the outputs of neurons of a previous layer inputting to a neuron of the next layer, this neuron requires a threshold to determine whether a specific feature has been detected. Activation functions serve this purpose, they act as transfer functions that are implemented to act as thresholds and introduce non-linearity to a neural network. Output values can range from 0 to 1 or -1 to 1, depending on the activation function chosen. These functions non-linearly scale the values of each node, which can be incredibly useful for inputs that cannot be modelled to linear output values. The following activation functions are commonly used to introduce non-linearity to neural networks.

2.10.1 Sigmoid Activation Function

The sigmoidal activation function (Eq 2.5) maps values between 0 and 1 and is commonly used to model probability. It was adapted to determine the probability of each nodal activation.

$$\emptyset(z) = \frac{1}{1 + e^{-z}} \tag{2.5}$$

2.10.2 Tanh Activation Function

Similar to the sigmoid function, the tanh activation function is sigmoidal. The tanh (Eq. 2.6), also known as the hyperbolic tangent function, is similar to the sigmoid function, except it extends the range of values from -1 to 1. This activation function is centered a zero but increases the rate of change near the origin compared to the sigmoid function.

$$tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \tag{2.6}$$

2.10.3 ReLU Activation Function

The issue with the sigmoid and tanh activation functions is the vanishing gradient problem, where the gradient used to adjust the weighted connection between neurons become infinitesimally small, ultimately leading to a halt in network training. Since a "vanishing" gradient is not desirable, a popular solution with a constant gradient is the Rectified Linear Unit (ReLU) activation function (Eq. 2.7),

$$R(z) = max\ (0, z) \tag{2.7}$$

The ReLU activation function is used to identify extreme differences in the input while also being computationally efficient. Along with these advantages, the function itself is simple, converting all negative values to 0, and leaving positive values untouched, meaning that multiple ReLU functions in series do not diminish results, since positive values are unaltered.

2.10.4 Softmax Activation Function

For many networks, the most important activation function is that at the end of a neural network, as that is the activation function that calculates the confidence of each output node. The three most common output layer activation functions are: the linear function, commonly used for regression problems; the sigmoid activation function, used most for binary and multilabel classification problems; Softmax, commonly used for multiclass classification.

The softmax activation function (Eq. 2.8) emphasizes the maximum element and acts on the entire layer unlike many other activation functions. Softmax is a differentiable generalization and selects the maximum value:

$$[f^{(L)}(x)]_i = \frac{e^{xi}}{\sum_{j=1}^{n_L} e^{x_j}} \tag{2.8}$$

2.11 The ResNet Architecture

The architecture that will be used for the image-based approach in this project will be the ResNet [34]. ResNet starts with a convolutional layer, proceeded by batch normalization and max pooling layer. Any number of ResNet layers can be added to the architecture. The depth of the model depends on the number of ResNet layers. These residual blocks consist of repeated sequences of two convolutional layers whose output is connected to the input of the block, making the residual connection. Figure 13 shows a simple residual block.

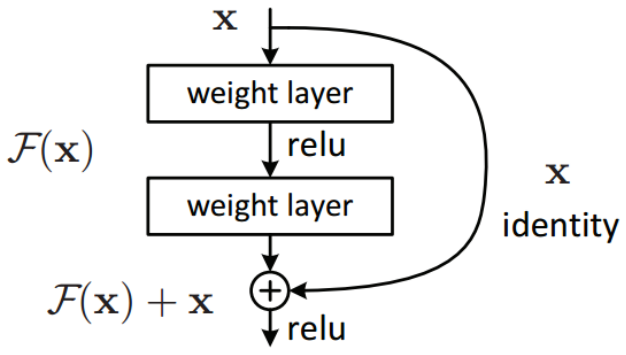

Figure 13: Visual representation of a simple residual block [34].

2.12 Summary

This chapter presented an overview on AI, ML, deep learning, and diagnostic metrics. Important characteristics of deep neural networks were explained, including their roles in processing input to produce meaningful outputs. The next chapter will cover the dataset used for this project.

Chapter 3: Parameter and Dataset Overview

This chapter describes the retinal OCTA image dataset, along with the image parameters that were extracted. These biomarkers were correlated to the clinically diagnosed severity and progression of DR conducted by the collaborating research group who acquired the OCTA image dataset; the trends of correlation were compared with previously published results as well. The biomarkers were examined for statistically significant differences between normal and DR, the results of which are discussed in Section 3.1.2.

3.1 OCTA Dataset

3.1.1 Retinal Imaging Instrumentation and Protocol

The following information is sourced from a study conducted by the Chinese University of Hong Kong [23]. The OCTA en face images used in this project were obtained by the Chinese University of Hong Kong's (CUHK) Eye Center. The initial en face images were acquired from July 2015 to November 2016, and the subjects had been consecutively followed up with for at least 2 years, where each subject attended a second visit at month 6 after baseline examinations. Subjects without DR or mild NPDR were followed annually afterwards, while subjects with moderate or greater NPDR were followed up every 6 months.

Criteria for inclusion into the study included that: (1) the patient must be at least 18 years of age; (2) diagnosis of type 1 or type 2 diabetes; (3) at least 24 months of follow-up. Exclusion criteria were: (1) proliferative diabetic retinopathy (PDR) was present at the baseline; (2) eyes had a history of pan-retinal photocoagulation or focal laser treatment within 6 months before recruitment; (3) eyes had a history of cataract surgery or other intraocular surgery within 6 months before recruitment; (4) eyes with ungradable OCTA images, structural OCT images, or color fundus photographs; (5) eyes with ocular condition other than DR at baseline or during follow-ups; (6) eyes that underwent cataract surgery 6 months before development of Diabetic Macular Edema (DME) during follow-ups, with the intention to exclude potential cases of Irvine-Gass syndrome.

Patients in this study underwent OCTA using a Swept-source OCT. The study used 3x3mm volumetric scans centered at the fovea, each consisting of 320 A-scans per B-scan for a total of 320 B-scans, which were used to obtain OCTA images.

This dataset was used to analyze the progression of DR in a patient by following a 15-step DR severity scale. The CUHK defined DR progression as an increase in 2 or more steps of severity level compared with the baseline image.

3.1.2 Parameter Definitions

The extracted vascular parameters for this project included fractal dimension, skeletonized density, perfusion density, and average vessel diameters of the different regions specified by the Early Treatment Diabetic Retinopathy Study (ETDRS) chart, an example of which is presented in Figure 14. Since the images used in this project are 3x3mm OCTA en face images, the five regions of interest are the Central subfield (Cen), Superior inner ring (Sin), Nasal inner ring (Nin), Inferior inner ring (Iin), and the Temporal inner ring (Tin); These are the terms used to describe the regions where the biomarkers were extracted. Along with the vascular parameters, certain avascular parameters were also extracted from the OCTA images, all of which are related to the Foveal Avascular Zone (FAZ). Once these parameters were extracted, t-tests were performed to see which of the parameters were clearly correlated to DR progression in the

CUHK dataset. After performing these tests, seven parameters were found to be correlated to the progression of DR in the eyes present in the dataset, therefore these were the parameters used as inputs for the parameter-based network. The extracted parameters that were correlated to DR progression are listed in Table 1 below.

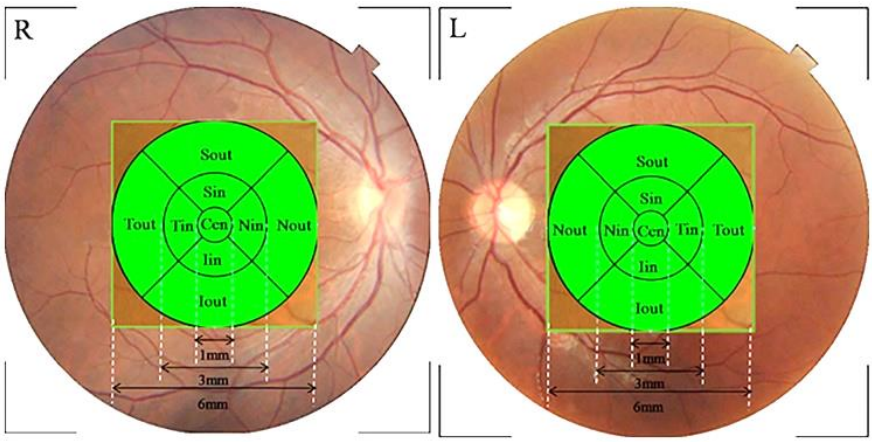

Figure 14: Example of an Early Treatment Diabetic Retinopathy Study (ETDRS) chart with different regions superimposed on a fundus image [35].

Table 1: Extracted parameters correlated to DR progression.

Fractal Dimension (Cen)
Fractal Dimension (Tin)
Fractal Dimension (Sin)
Fractal Dimension (Full Image)
Max FAZ Diameter
FAZ Area
FAZ Perimeter

3.1.3 Literature and Trend Analysis

The correlation between extracted biomarkers and DR severity were examined and compared with recently published results on early predictive biomarkers of DR. The statistical significance between each biomarker for eyes pertaining to the normal or DR class was calculated using t-tests and compared to published results.

The CUHK study using the same data as this thesis project did not split up the OCTA images into the regions specified in Figure 14 but found three parameters that correlated closely with DR progression: FAZ Area, vessel density, and fractal dimension. The results from the t-tests showed that FAZ area and fractal dimension among other parameters were correlated to DR progression but found no clear correlation between vessel density and DR progression.

3.2 Diabetic Retinopathy Progression Dataset

The dataset supplied by the CUHK consisted of 205 healthy eyes and 32 diseased eyes, along with risk factors such as patient age, BMI, diastolic and systolic blood pressure, and glycosylated hemoglobin level (HbA_{1c}). For this undergraduate thesis project, the OCTA en face images were used along with whether DR progressed or not within a 6-month period. In the future, the risk factors may be incorporated into the training process to provide the network more information, potentially leading to better performance. Both

parameter-based and image-based network approaches were taken for this project to see which provided the best results. For each eye, OCTA en face images for both the Superficial Capillary Plexus (SCP) and the Deep Capillary Plexus (DCP) were provided [18].

3.3 Combination of Image Data for Training

For the image-based classification approach, the SCP and DCP were used, along with a Maximum Intensity Projection (MIP) of both segmented versions of both the SCP and DCP. Segmentations were initially manually drawn, but due to the size of the dataset and the detail present in the OCTA images, BORG Master's Graduate Julian Lo used the vessel segmentation DNN that he developed. These segmentations were then manually reviewed to assure their quality. To combine all the information from the different modalities into a single input for training, the MIP of both the SCP and DCP segmentations was calculated. An example of the two segmented images, along with the MIP of the two images, is presented in Figure 15.

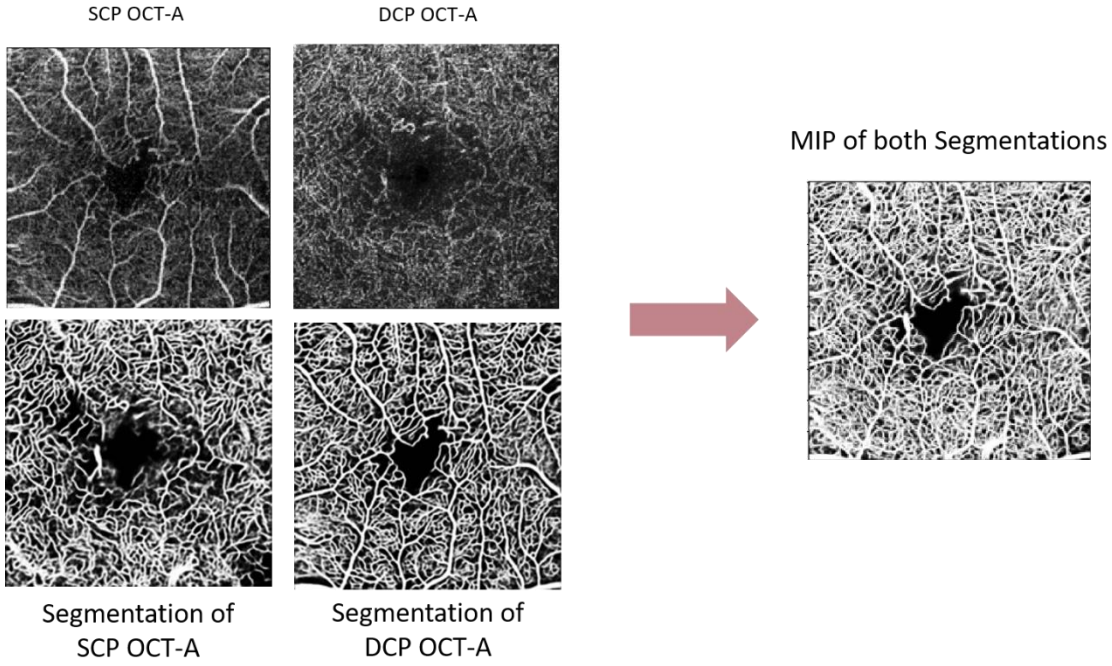

Figure 15: Segmentations of SCP, DCP, and MIP of the two.

The final input for the network was a 3-channel image, where the first two channels are the SCP and DCP and the third channel is the MIP of these layers, as explained in Figure 15. In the future, the third channel would be replaced with the OCT intensity images; the MIP of the OCTA images is being used in place of structural OCT images until this data is supplied by the collaborators. An example of the network input is illustrated in Figure 16.

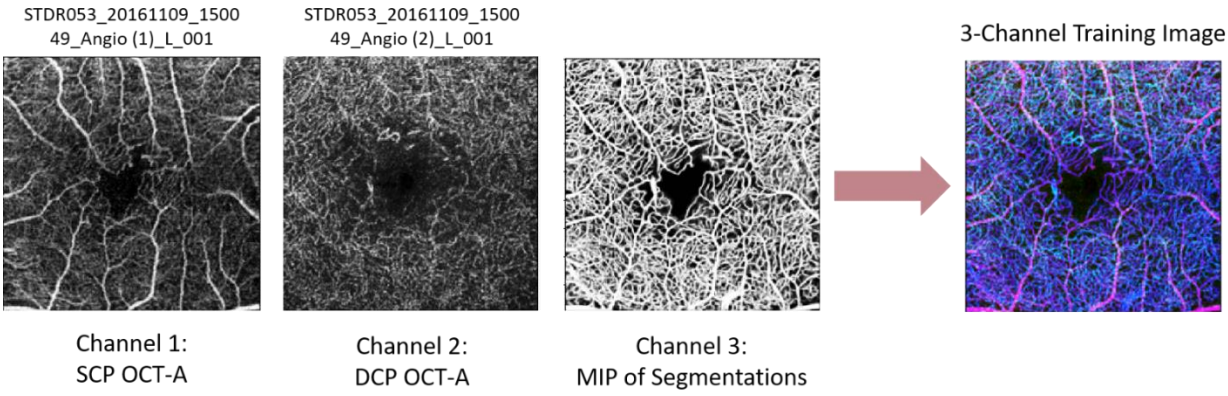

Figure 16: The 3 channels used as input for the neural network. Left to right: SCP, DCP, MIP.

3.4 Summary

In this chapter, the data used for both the parameter-based network and image-based network were presented, along with examples of these inputs. This chapter also covered the CUHK dataset, the condition of the eyes in the dataset, and how progression was defined and used at the ground truth for the training of the network. The pre-processing and organization of the data to create a single 3-channel input image for the neural network was presented and explained. Chapter 4 will discuss how the extracted parameters were used to train the parameter-based neural network, while Chapter 5 will discuss how this 3-channel input was used to train the neural network.

Chapter 4: Design and Results of Parameter-Based Neural Network Progression Classifier

This chapter will discuss the design and implementation of the Single-hidden Layer Feedforward Network (SLFN) used to predict the progression of DR in an eye. The SLFN architecture was developed using the Keras API through Python, and the neural network was trained on CEDAR using an NVIDIA Tesla V100 Volta 32GB GPU. As mentioned in Chapter 3.1.2, there are seven parameters of those that were extracted that were found to correlate to the progression of DR in the patients imaged by the collaborating group at CUHK.

4.1 Methods for single hidden layer feedforward network using parameter-based data

The extracted parameters are retinal biomarkers that are not only used as input into the network, but as information that can be provided to clinicians alongside the network's prediction for verification. Since these parameters have been found to be correlated to DR progression, they can be used by clinicians to interpret the results produced by the network. The extracted biomarkers are therefore fed through an SLFN to predict the progression of DR, while also providing a useful benchmark for the accuracy of the networks results. The parameter-based SLFN used in this chapter has 7 hidden-layer nodes; an example of the SLFN architecture used is presented in Figure 17.

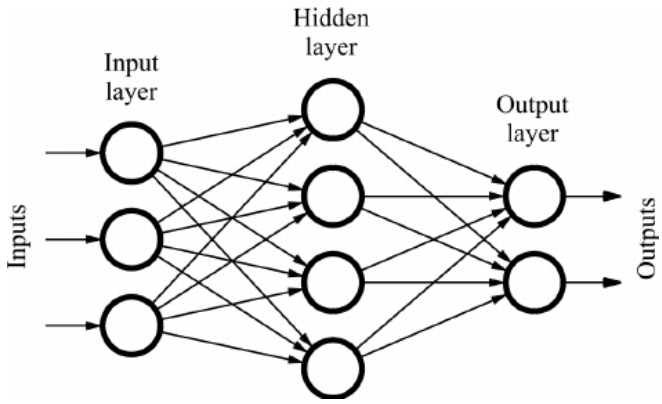

Figure 17: Example of SLFN used for Parameter-based Approach [36].

One of the issues of using a smaller dataset, such as the one used in this project, is the overall limited ability to generalize the dataset. As the problem being solved becomes more complex, such as increasing the number of biomarkers at the input, the larger the dataset needs to become, or the network will experience a decrease in overall performance [37][38]. For this reason, the optimal number of extracted parameters for training from the CUHK dataset was found using trial and error. To find the optimal number of parameters for training on this dataset, a fixed learning rate value was chosen, then a varying number of parameters were tested. A validation set composed of 20% of the total dataset was withheld to represent the true performance of the network, since it has not been trained on the validation data. Once the optimal number of input parameters is found, the optimal hyperparameters are chosen using trial and error.

The network continuously overfit to the larger class due to an imbalanced dataset, therefore for these tests the validation loss value was used to choose the optimal hyperparameters. Figure 18 presents the results of the tests done for a range of inputs and concludes that the best parameter-based neural network for the initial

cohort of 237 subjects is trained using the 3 parameters that are most correlated to DR progression. These 3 parameters were the Fractal Dimension (Tin), FAZ Area, and Fractal Dimension (Cen).

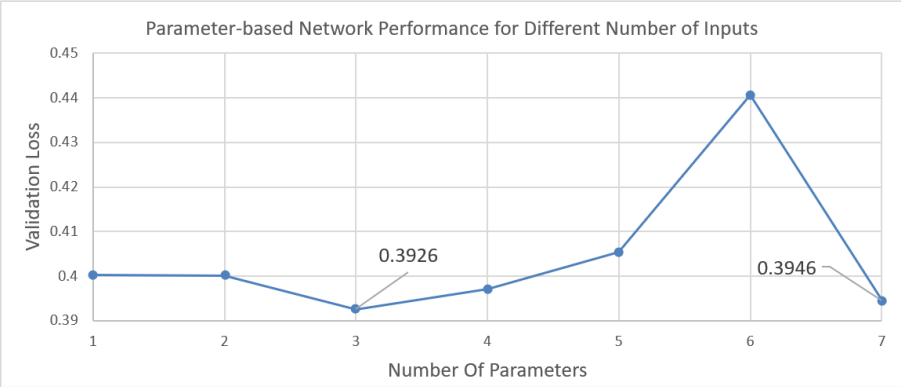

Figure 18: Performance of SLFN with different number of input parameters.

The results are difficult to analyze and potentially less dependable than desired because the dataset is much smaller than typically used for these applications, but the validation loss value is at a minimum when three input parameters are used. To combat the issue of overfitting while also potentially decreasing validation loss values, Chapter 4.2 will cover the implementation and testing of weighted classes.

4.2 Using Weighted Classes to Combat Performance Issues due to Imbalanced Dataset

As mentioned in Chapter 4.1, the DR progression dataset provided by the CUHK is imbalanced; 86.5% of eyes do not experience progression of DR as defined by the study conducted by the CUHK, meaning only 13.5% of eyes in the dataset experienced DR progression. These imbalances can force the network to favour one class, and when paired with a relatively small dataset, can cause a network to predict only values from the larger class, due to overfitting. The results from Chapter 4.1 are an example of this case, the network predictions resulted in a specificity of 1 and sensitivity of 0, meaning it predicted all inputs to experience no DR progression, even with the order of input parameters being randomized every epoch. The combination of a small and imbalanced dataset caused the network to lose the ability to generalize and accurately classify the retinal images. To try to combat this issue, weighted classes were tested.

Weighted classes are a method to combat imbalanced datasets. Both the majority and minority classes are assigned different weights to compensate for the skewed distributions. These weights will influence the classification of the classes during the training phase, making each input from the minority class more influential in the tuning process of the network in comparison to inputs from the majority class. The purpose of weighted classes is to heavily penalize the misclassification of inputs pertaining to the minority class by assigning each input a higher weight.

These weighted classes were incorporated using the Keras API, and the weights associated to each class corresponded the representation of that class in the overall dataset. Therefore, each retinal image that experienced no progression had much less influence in the overall tuning of the network when compared to a retinal image that experienced DR progression, because they represent a much larger percentage of the overall CUHK dataset. The results for the network using weighted classes are presented in Figure 19.

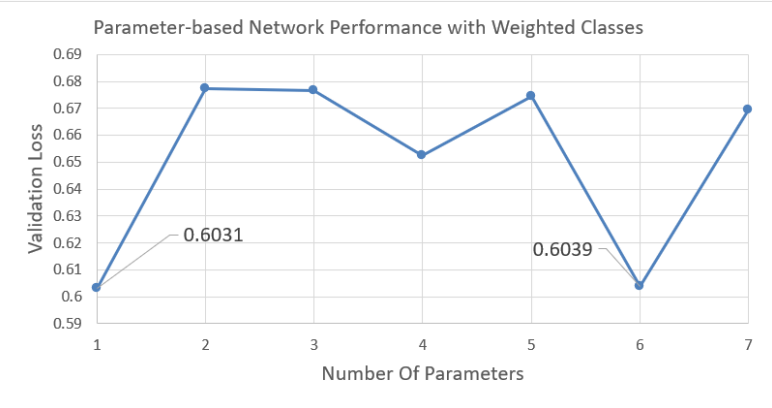

Figure 19: Performance of SLFN with weighted classes.

Though weighted classes normally help with imbalanced datasets, after multiple training iterations, the networks still overfit to the larger class of the CUHK dataset, producing a specificity value of 1, while producing worse validation loss values than the networks that did not incorporate weighted classes. A limitation of this early-stage development is that the number of images available in the CUHK dataset is still smaller than what is ideally needed for training a neural network. The neural network performed poorly in both cases due to dimensionality and limited data; in general, dimensionality proved to be a major problem for this project, but this is an issue for any small dataset.

Currently, the solution to the poorly performing network would be to acquire more data and test these methods with the additional data. In the situation where more data were acquired, and if these approaches still produced low accuracy, the larger class could be scaled down to have the same number of data points as the smaller class (to balance the datasets).

4.3 Results of Parameter-based Network

After analysis of the results of the previous two tests, a conclusion is drawn: There is simply not enough data at this preliminary stage to train a robust network capable of reliable results. The parameters most correlated to DR progression in the CUHK dataset were found and used as inputs, but even the optimal number of inputs produced a network that was unreliable. The network will greatly benefit when new data is acquired, allowing more effective generalization of the dataset.

4.4 Summary

In this chapter, the design and implementation of the parameter based SLFN was introduced. The parameters most correlated to DR progression were found, and combinations of input data and input size were tested; none of the resulting networks were satisfactory at this preliminary stage. However, the computational methods are in place and may be readily used when more data is available. The next chapter will describe the image-based approach of the project, where the network receives OCTA data as opposed to the numerical retinal parameters used in this chapter.

Chapter 5: Design and Results of Image-Based Neural Network Progression Classifier

This chapter examines the design and implementation of the image-based neural network used to predict the progression of DR in an eye. This approach will use OCTA images as input as opposed to the parametric values used in Chapter 4. The DNN architecture used for this chapter is a typical Convolutional Neural Network (CNN) known as ResNet50 [34]. Multiple methods of tuning and editing the network and its weights are covered in this chapter, all of which are done with the goal of maximizing the performance of the network.

5.1 Methods for Initial Image-based Deep Neural Network Classifier

Chapter 3.3 explored the pre-processing pipeline used to produce the inputs for training, which included combining the SCP, DCP, and MIP of the two images into a single 3-channel input. This chapter will explain the network architecture used, and the initial test conducted for the image-based prediction of DR progression.

5.1.1 The Implementation of the ResNet50 Architecture

Smaller CNN architectures are useful for simple problems such as recognizing handwritten digits or differentiating between dogs and cats, but when inputs are complex and what differentiates the classes is not as pronounced, a deeper network is required to detect those features.

There are caveats when implementing these deeper network architectures, one of which is the potential of experiencing the vanishing gradient problem. The vanishing gradient problem is an issue that presents itself as the model depth increases [39]. This occurs when the gradient, which is used to update the weights of the model, becomes infinitesimally small. This causes the updates to the weights to become negligible, resulting in lack of improvement in the model being trained. To overcome this issue that arises with deep models, ResNet makes use of residual mapping in the layers of the model. See Section 2.11 for a more detailed description.

There are many different ResNet architectures, all of which differ in how many residual blocks they contain. For this project, the ResNet50 model, which has 50 residual blocks, is used due to it being deep enough to detect the microvascular features in the OCTA images without being excessively deep to require an unreasonable amount of data. Using a ResNet architecture will mean that as more data is acquired, the network can be trained without worry of the vanishing gradient problem. For the work in this thesis, the ResNet50 architecture was created using the Keras API through Python, and the neural network was trained on CEDAR using an NVIDIA Tesla V100 Volta 32GB GPU. Since the ResNet50 model provided by Keras has an output value of 7x7x2048, two additional dense layers were added at the tail end to produce a binary output. The architecture is presented in Table 2.

Table 2: ResNet50 architecture used for this project.

Layer	Output Shape
Regular ResNet50	7x7x2048
Dropout	7x7x2048
Dense	7x7x16
Batch Normalization	7x7x16
Dropout	7x7x16
Flatten	1x784
Dense	1

Different strategies were applied to maximize the performance of the network throughout this chapter, including transfer learning, data augmentation, and cross-validation.

5.2 Results for Initial Image-based Deep Neural Network Classifiers

To test the ability of the network to predict the progression of DR, the ResNet50 model presented in *Table 2* was used with randomly initialized weights as a benchmark. The input used is described in Chapter 3.3, with the SCP being the first channel of the input, the DCP being the second, and the MIP of the segmentations being the third. Metrics such as sensitivity and specificity of the best training epoch are provided for each case, to show not only the accuracy of the overall predictions, but also how well the network is performing for each class. Since there were only 32 inputs pertaining to the "DR Progression" class, it was decided that the validation dataset was composed of only 20% of the entire dataset to maximize the training data. The network's performance was analyzed through comparison of the validation accuracy and loss plots, as well as the best performing epoch's sensitivity and specificity values. The initial neural network performed poorly and produced non-ideal results, independent of the learning rate chosen. Every test showed the network quickly learning to predict a value of 0 (corresponding to no DR progression) for every eye in the dataset, independent of the actual ground truth value. Figure 20 contains the loss and accuracy plots of the initial test. To attempt to remedy this poor performance, some additional approaches were tested in the following chapters.

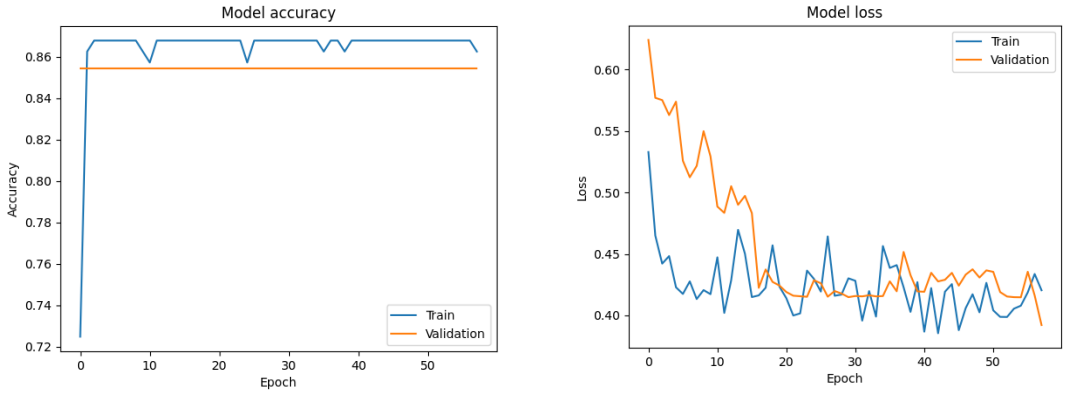

Figure 20: Accuracy and loss plots for initial network (Sensitivity: 0, Specificity:1).

5.3 Using Transfer Learning for Better Initial Network Weights

The test done in Chapter 5.2 used randomly initialized weights, which can produce varying performance for the final network. One way to increase the overall performance of an un-trained network, especially

with limited data, is to initialize with weights from a previously trained network that has been trained on relevant data.

5.3.1 Transfer Learning with ImageNet Weights

One of the most common transfer learning techniques is to initialize with weights of a network trained on the ImageNet database. The ImageNet database has been created by researchers worldwide to provide a vast and easily accessible labeled dataset for image recognition. Initializing with ImageNet weights gives the network a breadth of knowledge on many different sorts of objects, spanning anywhere from fungi to construction signs [40]. The network in this section was initialized with ImageNet weights.

Due to the size of the dataset, the shallower layers (those that are closer to the input layer) that are initialized with ImageNet weights were frozen, meaning that they will not be trainable. This should theoretically allow the network to finely tune the layers responsible for detecting the smaller details in the OCTA images, which is important for predicting the progression of DR. The results from this test are presented in Figure 21. The sensitivity and specificity metrics remained at zero and one respectively, meaning the network predicted all inputs will experience no DR progression; this means the network struggled to generalize the dataset, which is likely due to the current limitation on data. The following section will investigate a different set of weights used for transfer learning.

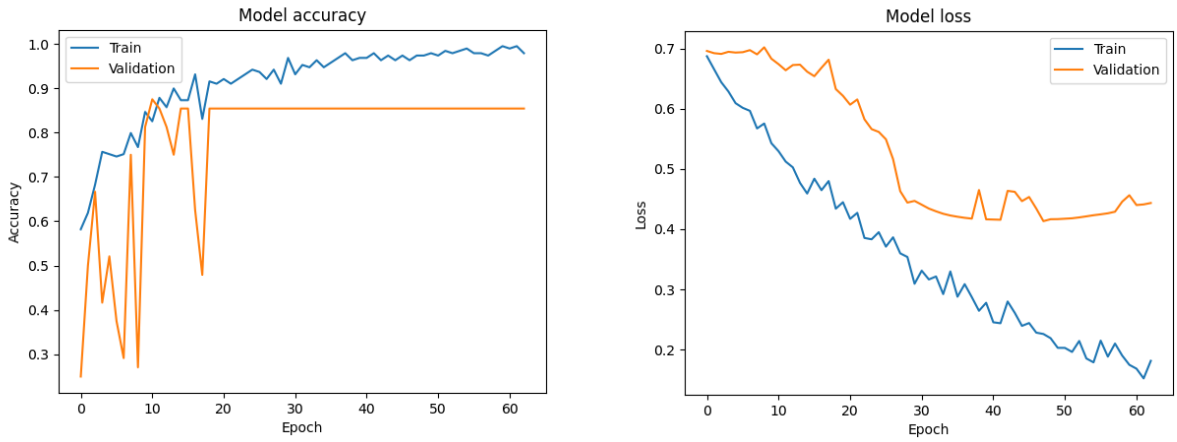

Figure 21: Accuracy and loss plots using ImageNet weights (Sensitivity: 0, Specificity:1).

5.3.2 Transfer Learning with DR Classification Weights

ImageNet weights work well for simpler classifiers, or for 'natural' images similar to those present in the dataset such as trees and fruits, however for images as detailed and complex as OCTA images, a more refined set of initial weights are needed, especially with a small training dataset. A ResNet50 network was trained on the OCTA data Mr. Timothy Yu used for DR classification; those weights were then used for initialization of the DR progression detection network. The results of this test are presented in Figure 22. The DR classification weights result in a much more stable training process, with final validation loss values lower than the network that used ImageNet weights. The network still overfits to the large class, as can be deduced from the sensitivity and specificity values. To counter the overfitting, data augmentation was tested, the results of which are explained in the following section.

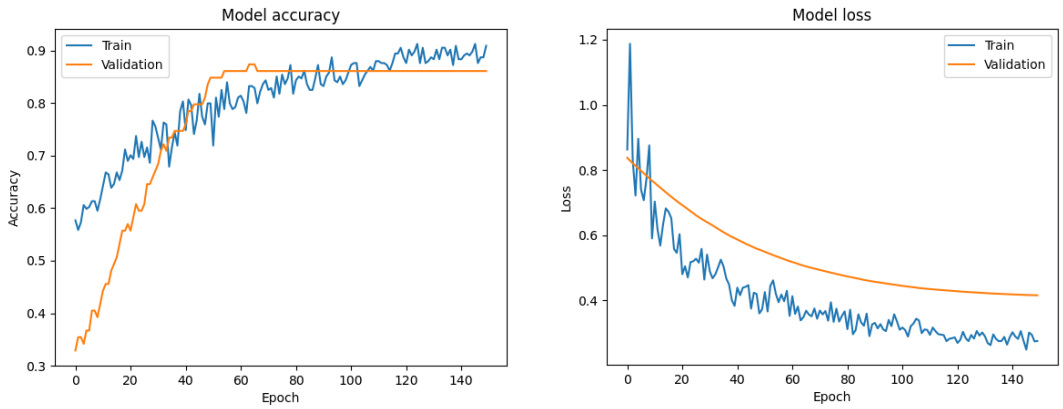

Figure 22: Accuracy and loss plots using DR classification weights (Sensitivity: 0.0, Specificity: 1.0).

5.4 Down-sampling the Dataset to Eliminate Imbalance

One of the potential limitations in the early-stage development of binary classification DNNs trained on medical data is a naturally occurring imbalance between the normal/healthy class, and the diseased class. The CUHK dataset provided for this project was composed of 32 "Progression" eyes and 205 "No progression" eyes; this imbalance of class sizes is one of the factors leading to the network overfitting to a single class, but results should ameliorate once more data is acquired. There are multiple ways to eliminate the imbalance, and for the first attempt, the most simplistic approach was taken: down-sampling. This means that the larger class (in this case the "No progression" class) was sampled so that both classes used for training had an equal number of input images.

This test produced a network with lower accuracy and higher loss values, especially for the validation dataset since the network is now only being trained on 25 images from each class. What is very different is that the network was no longer overfitting to the "No progression" dataset, but actually predicting more of the "Progression" dataset correctly. All of the images that were not used for training were used for validation and testing; of the 205 "No progression" images, 180 were used for validation, while 7 of the "Progression" images were used for validation. The final network predicted all 7 "Progression" validation images correctly, while predicting only 29 of the 180 "No Progression" images correctly, resulting in a final validation accuracy of 19.3%.

This result is not ideal but proves that down-sampling the larger class has the potential to solve the issue of over-fitting. The limitation of down-sampling is that data must be down-sampled to the size of the smaller class, which for this test was 25 images; to ameliorate the final validation accuracy, new data must be acquired. The loss and accuracy plots of this test are presented in Figure 23.

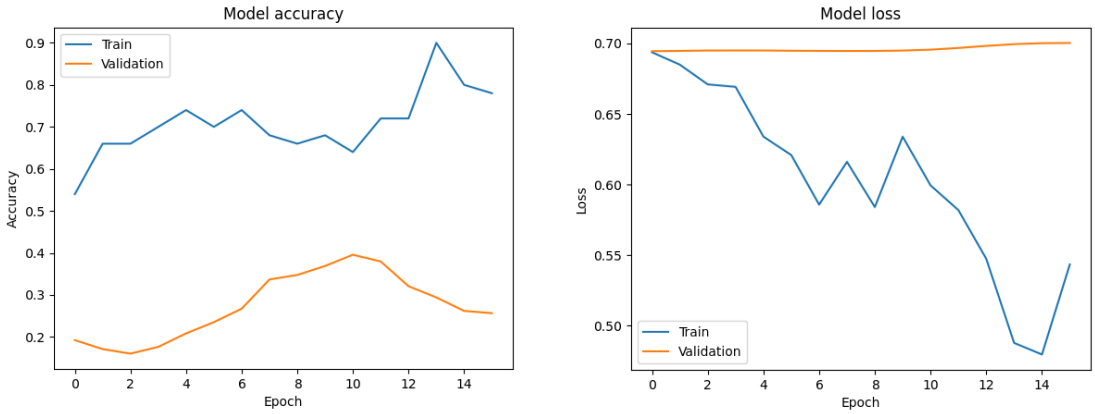

Figure 23: Accuracy and loss plots after down-sampling (Sensitivity: 1, Specificity:0.161).

5.5 Using Weighted Classes to Eliminate Effects of an Imbalanced Dataset

Weighted classes, which were also tested in Chapter 4, were used as the second method to prevent overfitting. This method of combatting the effects of an imbalanced dataset involves weighing the influence an input has on the overall tuning of the network based on its class representation in the training dataset. If the input belongs to a class that makes up a large portion of the overall training dataset, it has less individual influence on the overall tuning of the network weights. As in the previous implementation described in Section 4.2, this was done by including the *class_weight* argument in the *model.fit* function. The results were an improvement over the two previous tests: the model still overfitted to the larger class but the validation loss decreases smoothly to a value slightly above the final validation loss value of the initial/benchmark network presented in Section 5.2. These plots are shown in Figure 24. Weighted classes have an advantage over down-sampling the larger class, since they do not decrease the size of the training dataset, but as demonstrated, the network trained with weighted classes still experiences overfitting to the larger class, even though the inputs were weighted. The final method of balancing the dataset, a process named data augmentation, is tested in the following section.

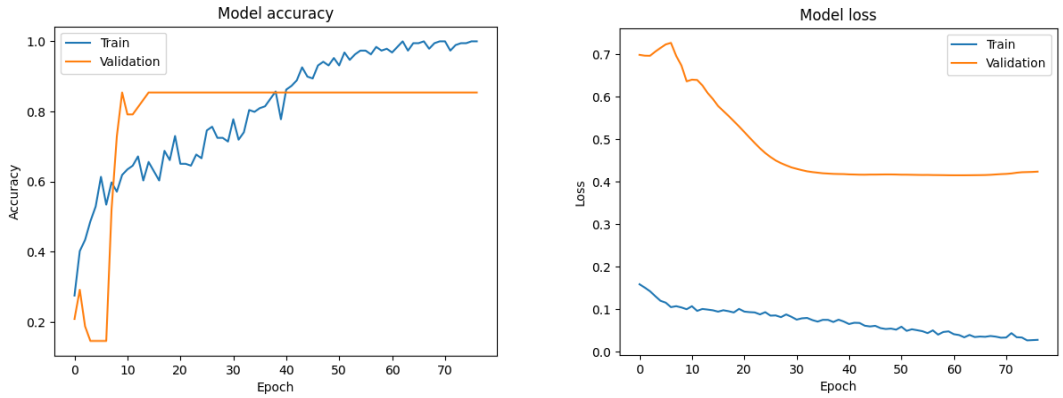

Figure 24: Accuracy and loss plots for weighted classes (Sensitivity: 0.0, Specificity: 1.0).

5.6 Using Data Augmentation to Balance the Dataset

Data augmentation is a popular method of up-sampling a smaller class to minimize the imbalance in a dataset, as was presented in Chapter 2.2.5. The different forms of data augmentation that were tested in this chapter were vertical and horizontal flips, intensity shifts, and rotations (maximum of 15 degrees). These transformations were applied to the smaller class in an attempt to simulate additional data so that the network can generalize the dataset more effectively, without overfitting. The plots from this test are presented in Figure 25. The implementation of data augmentation produced the same final validation accuracy (85.4%) as previous tests that experienced overfitting, but the difference with this test is that this network predicted 1 of the 7 "Progression" validation images correctly, while correctly predicting 40 of the 41 "No progression" images. The use of down-sampling or data augmentation to balance the dataset produces networks that do not predict the same value for all inputs, which is a promising result when trying to combat overfitting to one class. According to the results, data augmentation is the most effective way to eliminate the imbalance of the dataset, because it does not overfit entirely to a single class, while maintaining a rather high accuracy value, therefore it is used in the final network.

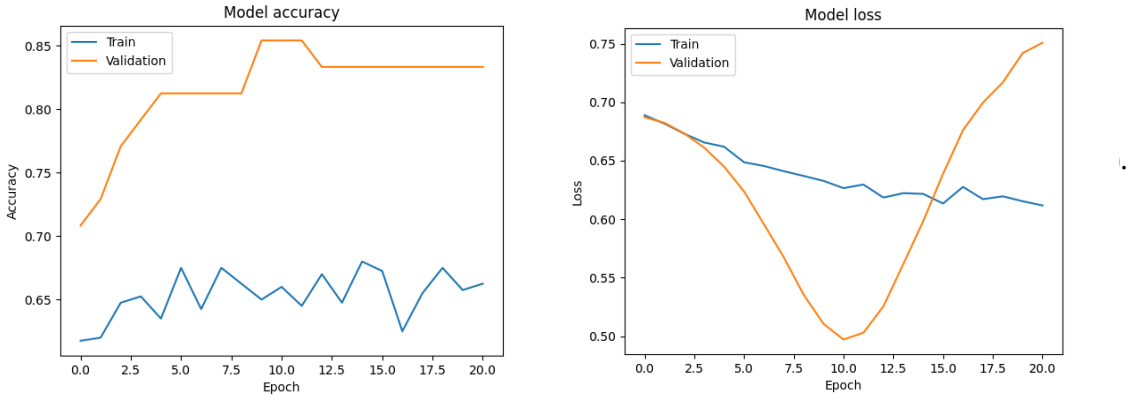

Figure 26: Accuracy and loss plots for data augmentation (Sensitivity: 0.143, Specificity: 0.976).

5.7 Summary

This chapter presented the design and implementation of the image-based DNN. Various methods of training were outlined, along with the architecture ultimately chosen for this project. The following chapter will present the challenges and future opportunities of this project.

Chapter 6: Challenges, Future Work, and Conclusion

This thesis project examined the use of machine learning for OCTA image analysis, specifically for prediction of DR progression. The first approach taken was a feature-based network that was trained on retinal biomarkers extracted from the OCTA en face images, which can also assist medical personnel to verify results. The second approach taken was an image-based network, which used OCTA data directly as input. Multiple methods of training were tested for the image-based network including using pre-trained weights and down-sampling of the data, all of which were done to test the ability of the network to predict the progression of DR.

As mentioned previously, there have not been any machine learning studies revolving around prediction of progression of DR, but studies have covered DR classification using fundus images or OCT images. OCT and OCTA are relatively new imaging techniques, resulting in data being difficult to obtain unless one acquires it themselves. One group utilized OCT images where the retina was segmented into 12 layers, and they attempted to distinguish normal subjects from NPDR subjects. Using a Computer-Assisted Diagnostic (CADx) system, they were able to distinguish between normal and NPDR subjects with an accuracy of 92.5%, and distinguish between subclinical and mild/moderate NPDR with 95% accuracy [41]. Another group reported that their detection program, named the Iowa Detection Program (IDP), distinguished NRDR from RDR eyes with a sensitivity of 96.8% and specificity of 59.4% in comparison to the sensitivity of 80.0% and 98.0% by retinal specialists [42]. A third study conducted by the BORG and Dr. Morgan Heisler evaluated the role of deep learning in the classification of DR using OCTA images, achieving DR classification accuracies of 92% and 90% for the two methods tested [43]. These three groups were able to successfully classify DR due to the amount of data available; the group using the CADx system had a "ground truth" dataset of 200 images, along with validation and testing datasets composed of 160 and 80 eyes, respectively. This group also extracted 36 features per OCT scan, which were then used to train a classifier. The group that developed the IDP simply used fundus images but had a dataset of 1748 eyes. Dr. Morgan Heisler's study used 463 volumes from 308 eyes, using both OCT and OCTA en face images to train the network. This thesis attempted to solve a more complex problem, which is prognosis of DR instead of diagnosis; this more complex problem, along with a smaller dataset, is likely what caused the difference in final performance values between the one found in this thesis, and the three studies discussed above.

The ResNet50 network that was initialized using ImageNet weights and implemented data augmentation provided the highest level of accuracy (85.4%) for prediction of DR progression, while not entirely overfitting to one class (specificity/sensitivity of 0.976/0.143). The specificity value of the final network used in this thesis is comparable to the group who used the IDP, but the sensitivity value is much lower. This is likely due to the size of the dataset used for this project, lack of exposure to the "Progression" class, and the added complexity of predicting the progression of DR in an eye as opposed to classifying the current DR severity of the eye.

Many of the methods that were examined during this thesis resulted in networks that overfitted to the larger class (specificity of 1, and sensitivity of 0), but two methods did not experience overfitting: the first included down-sampling of the dataset, and the second implemented data augmentation to up-sample the dataset. The down-sampling test trained the network with 25 OCT images of each class ("DR progression" and "No DR progression"), while validating on the remaining dataset (180 "Progression" image, and 7 "No progression" images). This resulted in a network that correctly identified all 7 "Progression" images, while only correctly identifying 29 of the 180 "No progression" images. This was the only test that had a

sensitivity value (1) above 0.2 but resulted in a very low specificity value (0.161); down-sampling the dataset provided a network with a nearly opposite classification behaviour compared to all other tests conducted. Since the validation set was composed mostly of images that experienced no progression, and the network itself performed poorly on these sorts of images, the final validation accuracy value was substantially lower than the other tests. The network with data augmentation also did not experience complete overfitting; the network predicted 40 of the 41 "No progression" images in the validation set correctly, while predicting 1 of the 7 "Progression" images correctly. The results of all tests conducted in this thesis are presented in Table 3.

Table 3: Results acquired throughout this project.

Model (ResNet50 Architecture)		Val. Accuracy (%)	Specificity	Sensitivity
Benchmark Network	(Section 5.2)	85.4	1	0
Transfer Learning with ImageNet Weights	(Section 5.3.1)	85.4	1	0
Transfer Learning with DR Class. Weights	(Section 5.3.2)	85.4	1	0
Down-sampling Dataset	(Section 5.4)	19.3	0.161	1
Weighted Classes	(Section 5.5)	85.4	1	0
Data Augmentation	(Section 5.6)	85.4	0.976	0.143

6.1 Challenges

One of the major challenges encountered throughout this process was size of the CUHK dataset available for this project. Most of the methods tested throughout this project were attempts to maximize performance of the network using this small dataset; this, combined with the complex nature of the problem at hand made the entire process difficult to test. The purpose of this thesis was to explore engineering design for improving the classification results for a challenging problem for the case of small datasets. Ultimately, the results from this thesis have been promising, and could be improved substantially once new data is acquired, helping push this research towards clinical utility.

The results from the image-based network are superior to those of the parameter-based network. Both networks were trained on the same number of inputs, but the final image-based method was more successful in accurate predictions without entirely overfitting to the larger class. These challenges proved to be major roadblocks in this project.

The data used in this project paired OCTA data of each eye with a ground truth value describing whether the DR in the eye progressed (progression for this dataset is defined in Section 1.5). The study by the CUHK found that metrics extracted at the baseline, which are used to determine the severity of DR, are strong predictors for risk of progression. To capitalize on the correlation between DR severity at the baseline and DR progression, DR severity may be used along with the input images in the future to provide the network with crucial information that can increase performance drastically.

6.2 Future Work

By increasing the overall size of the dataset, many of the problems faced in this project can be remedied, such as the network's ability to generalize. Once this larger dataset is acquired, various neural network architectures can be tested, including deeper and more complex network architectures. Collecting data from different sources can supply the network with a more general dataset, thus increasing performance for all

methods of acquisition and demographics of patients. More data may improve the prediction capabilities of the network, ultimately providing more accurate results.

Using information such as patient age, sex, Body Mass Index (BMI) along with the extracted parameters can also be used to potentially produce finer results as they provide more information for the network to find patterns. This network can then be compared fairly to the image-based method, and a fair conclusion on which method is superior can be made once more data is acquired.

Along with an increase in dataset size, structural OCT images can be used as one of the channels of the image-based input. This provides additional streams of information, potentially providing superior results for the prediction of DR progression. All methods presented have the potential to increase the overall performance of the network, potentially creating a network that is valuable in real-life clinical situations.

6.3 Conclusion

In this thesis, the ResNet50 model with ImageNet pre-trained weights proved to be the machine learning approach with the best potential for the prediction of DR progression. Utilizing pre-processed OCTA images as explained in Chapter 3.3, along with data augmentation, the network was able to predict the progression of DR with an accuracy of 85.4%, specificity of 0.976, and sensitivity of 0.143. These classification metrics show the predictive properties of the network using OCTA image data for DR progression.

For the parameter based approach, extracted parameters are used for both prediction of DR progression, as well as to assist ophthalmologists with verification of the results. The parameters that demonstrated correlation to DR progression included Fractal Dimension, FAZ Diameter, FAZ area, and FAZ perimeter, which was similar to the results found from the CUHK study [23].

Machine learning methods have the potential to assist methods currently employed by ophthalmologists. Current methods include manually reviewing images to predict the progression of DR, but insufficient data hindered the performance of the networks tested in this project. There is significant potential for the approaches designed through this thesis project as more data gets acquired.

Chapter 7: References

[1] Mayo Clinic, “Diabetic retinopathy - Symptoms and causes - Mayo Clinic,” *Mayo Foundation for Medical Education and Research*, 2018. https://www.mayoclinic.org/diseases-conditions/diabetic-retinopathy/symptoms-causes/syc-20371611 (accessed Jan. 03, 2021).

[2] “Diabetic Retinopathy Data and Statistics | National Eye Institute,” 2010. Accessed: Jan. 03, 2021. [Online]. Available: https://www.nei.nih.gov/learn-about-eye-health/resources-for-health-educators/eye-health-data-and-statistics/diabetic-retinopathy-data-and-statistics.

[3] L. Peng and V. Gulshan, “Google AI Blog: Deep Learning for Detection of Diabetic Eye Disease,” *Google AI Blog*, no. Figure 1, pp. 1–4, 2016, Accessed: Jul. 23, 2020. [Online]. Available: https://ai.googleblog.com/2016/11/deep-learning-for-detection-of-diabetic.html.

[4] “Diabetic Retinopathy Data and Statistics | National Eye Institute,” 2010. Accessed: Jul. 23, 2020. [Online]. Available: https://www.nei.nih.gov/learn-about-eye-health/eye-conditions-and-diseases/diabetic-retinopathy.

[5] K. Boyd and R. Janigian, “What Is Diabetic Retinopathy? - American Academy of Ophthalmology,” *American Academy of Opthalmology*. 2017, Accessed: Jul. 23, 2020. [Online]. Available: https://www.aao.org/eye-health/diseases/what-is-diabetic-retinopathy.

[6] L. Wu, P. Fernandez-Loaiza, J. Sauma, E. Hernandez-Bogantes, and M. Masis, “Classification of diabetic retinopathy and diabetic macular edema,” *World J. Diabetes*, vol. 4, no. 6, p. 290, 2013, doi: 10.4239/wjd.v4.i6.290.

[7] C. Lam, D. Yi, M. Guo, and T. Lindsey, “Automated Detection of Diabetic Retinopathy using Deep Learning.,” *AMIA Jt. Summits Transl. Sci. proceedings. AMIA Jt. Summits Transl. Sci.*, vol. 2017, pp. 147–155, 2018, Accessed: Jul. 23, 2020. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/29888061.

[8] “Color Fundus Photography | Department of Ophthalmology.” https://ophthalmology.med.ubc.ca/patient-care/ophthalmic-photography/color-fundus-photography/ (accessed Nov. 04, 2020).

[9] “OCT (tomografia de coerência óptica) - Saiba tudo na Clínica Belfort.” https://www.clinicabelfort.com.br/exames/oct/ (accessed Jul. 24, 2020).

[10] Heidelberg Engineering, “OCT Angiography.” .

[11] D. Turbert, “What Is Optical Coherence Tomography? - American Academy of Ophthalmology,” *American Academy of Ophthalmology*, 2015. https://www.aao.org/eye-health/treatments/what-is-optical-coherence-tomography (accessed Jul. 23, 2020).

[12] A. G. Podoleanu, “Optical coherence tomography,” *J. Microsc.*, vol. 247, no. 3, pp. 209–219, Sep. 2012, doi: 10.1111/j.1365-2818.2012.03619.x.

[13] D. Huang *et al.*, “Optical Coherence Tomography,” *Science*, 1991. https://www.researchgate.net/publication/232175518_Optical_Coherence_Tomography (accessed Feb. 10, 2021).

[14] J. G. Fujimoto, C. Pitris, S. A. Boppart, and M. E. Brezinski, “Optical coherence tomography: An emerging technology for biomedical imaging and optical biopsy,” *Neoplasia*, vol. 2, no. 1–2. Nature Publishing Group, pp. 9–25, 2000, doi: 10.1038/sj.neo.7900071.

[15] “Optical Coherence Tomography | Biophotonics Imaging Laboratory.” https://biophotonics.illinois.edu/?q=imaging-technology/optical-coherence-tomography (accessed Jan. 05, 2021).

[16] L. Wang, Y. Li, Y. Li, and K. Li, “Improved speckle contrast optical coherence tomography angiography - PubMed.” https://pubmed.ncbi.nlm.nih.gov/30416648/ (accessed Jan. 05, 2021).

[17] P. Le and B. Patel, “Optical Coherence Tomography Angiography,” .

[18] J. Khadamy, K. Abri Aghdam, and K. Falavarjani, “An update on optical coherence tomography angiography in diabetic retinopathy,” *Journal of Ophthalmic and Vision Research*, vol. 13, no. 4. Wolters Kluwer Medknow Publications, pp. 487–497, Oct. 01, 2018, doi: 10.4103/jovr.jovr_57_18.

[19] Y. Jia *et al.*, “Split-spectrum amplitude-decorrelation angiography with optical coherence tomography,” *Opt. Express*, vol. 20, no. 4, p. 4710, Feb. 2012, doi: 10.1364/oe.20.004710.

[20] R. K. Wang, “Optical microangiography: A label-free 3-D imaging technology to visualize and quantify blood circulations within tissue beds in vivo,” *IEEE J. Sel. Top. Quantum Electron.*, vol. 16, no. 3, pp. 545–554, May 2010, doi: 10.1109/JSTQE.2009.2033609.

[21] W. J. Choi *et al.*, “Choriocapillaris and choroidal microvasculature imaging with ultrahigh speed OCT angiography,” *PLoS One*, vol. 8, no. 12, p. 81499, Dec. 2013, doi: 10.1371/journal.pone.0081499.

[22] P. L. Nesper *et al.*, “Quantifying Microvascular Abnormalities With Increasing Severity of Diabetic Retinopathy Using Optical Coherence Tomography Angiography,” *Invest. Ophthalmol. Vis. Sci.*, vol. 58, no. 6, pp. BIO307–BIO315, May 2017, doi: 10.1167/iovs.17-21787.

[23] Z. Sun *et al.*, “OCT Angiography Metrics Predict Progression of Diabetic Retinopathy and Development of Diabetic Macular Edema: A Prospective Study,” *Ophthalmology*, vol. 126, no. 12, pp. 1675–1684, Dec. 2019, doi: 10.1016/j.ophtha.2019.06.016.

[24] R. Ghaida, “Neural Networks | Deep Learning : Teaching a Computer to Read | by Ryan Ghaida | Medium.” https://medium.com/@ryaboug/neural-networks-deep-learning-teaching-a-computer-to-read-3e856e2d4a50 (accessed Jan. 05, 2021).

[25] S. Saha, “A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way | by Sumit Saha | Towards Data Science,” *Towards data Science*, 2018. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53 (accessed Jan. 06, 2021).

[26] L. Torrey and J. Shavlik, “Transfer Learning.”

[27] “CS231n Convolutional Neural Networks for Visual Recognition.” https://cs231n.github.io/transfer-learning/ (accessed Jan. 06, 2021).

[28] J. L. Prince and J. Links, *Medical Imaging Signals and Systems, 2nd Edition*. Boston, MA, 2015.

[29] K. Bai, “Towards Data Science - A Comprehensive Introduction to Different Types of Convolutions in Deep Learning,” vol. 12, pp. 1–43, 2019, Accessed: Jan. 06, 2021. [Online]. Available: https://towardsdatascience.com/a-comprehensive-introduction-to-different-types-of-convolutions-in-deep-learning-669281e58215.

[30] A. Deshpande, “A Beginner’s Guide To Understanding Convolutional Neural Networks – Adit Deshpande – Engineering at Forward | UCLA CS ’19.” 2016, Accessed: Jan. 06, 2021. [Online]. Available: https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks-Part-2/.

[31] J. Brownlee, “A Gentle Introduction to Dropout for Regularizing Deep Neural Networks,” pp. 1–9, 2018, Accessed: Jan. 06, 2021. [Online]. Available: https://machinelearningmastery.com/dropout-for-regularizing-deep-neural-networks/.

[32] A. Ben Khalifa and H. Frigui, “Multiple Instance Fuzzy Inference Neural Networks,” Oct. 2016, Accessed: Jan. 06, 2021. [Online]. Available: http://arxiv.org/abs/1610.04973.

[33] “Fully-Connected Layer with dynamic input shape | by Teco KIDS | Medium.” https://medium.com/@tecokids.monastir/fully-connected-layer-with-dynamic-input-shape-70c869ae71af (accessed Jan. 06, 2021).

[34] S. Sahoo, “Detailed Guide to Understand and Implement ResNets – CV-Tricks.com.” https://cv-tricks.com/keras/understand-implement-resnets/ (accessed Jan. 26, 2021).

[35] B. Chen *et al.*, “Profile and determinants of retinal optical intensity in normal eyes with spectral domain optical coherence tomography,” *PLoS One*, vol. 11, no. 2, p. 148183, Feb. 2016, doi: 10.1371/journal.pone.0148183.

[36] “Simple Feed Forward Neural Network code for digital Handwritten digit recognition | by Chetan Warke | Random TechPark | Medium.” https://medium.com/random-techpark/simple-feed-forward-neural-network-code-for-digital-handwritten-digit-recognition-a234955103d4 (accessed Jan. 25, 2021).

[37] J. Brownlee, “Introduction to Dimensionality Reduction for Machine Learning,” pp. 1–5, 2020, Accessed: Jan. 23, 2021. [Online]. Available: https://machinelearningmastery.com/dimensionality-reduction-for-machine-learning/.

[38] J. Raj, “A beginner’s guide to dimensionality reduction in Machine Learning,” *Towar. Data Sci.*, pp. 1–7, 2019, Accessed: Jan. 23, 2021. [Online]. Available: https://towardsdatascience.com/dimensionality-reduction-for-machine-learning-80a46c2ebb7e.

[39] “Residual blocks — Building blocks of ResNet | by Sabyasachi Sahoo | Towards Data Science.” https://towardsdatascience.com/residual-blocks-building-blocks-of-resnet-fd90ca15d6ec (accessed Jan. 26, 2021).

[40] “ImageNet.” http://image-net.org/about-overview (accessed Jan. 27, 2021).

[41] H. S. Sandhu, A. Eltanboly, A. Shalaby, R. S. Keynton, S. Schaal, and A. El-Baz, “Automated diagnosis and grading of diabetic retinopathy using optical coherence tomography,” *Investig.*

Ophthalmol. Vis. Sci., vol. 59, no. 7, pp. 3155–3160, Jun. 2018, doi: 10.1167/iovs.17-23677.

[42] M. D. Abràmoff *et al.*, "Automated analysis of retinal images for detection of referable diabetic retinopathy," *JAMA Ophthalmol.*, vol. 131, no. 3, pp. 351–357, Mar. 2013, doi: 10.1001/jamaophthalmol.2013.1743.

[43] M. Heisler *et al.*, "Ensemble deep learning for diabetic retinopathy detection using optical coherence tomography angiography," *Transl. Vis. Sci. Technol.*, vol. 9, no. 2, pp. 1–11, Jan. 2020, doi: 10.1167/tvst.9.2.20.