

**CLASSIFICATION OF GLAUCOMA USING  
FUNDUS IMAGE WITH MACHINE LEARNING**

**A THESIS SUBMITTED TO THE  
INSTITUTE OF GRADUATE STUDIES  
OF  
NEAR EAST UNIVERSITY**

**By**

**HUSSEIN A.AZEEZ**

**In Partial Fulfillment of the Requirements for  
the Degree of Master of Science  
in  
Computer Engineering**

**NICOSIA, 2021**

**HUSSEIN A.AZEEZ**

**CLASSIFICATION OF GLAUCOMA USING FUNDUS  
IMAGE WITH MACHINE LEARNING**

**NEU  
2021**

**Classification of Glaucoma using Fundus Image with  
Machine Learning**

**A THESIS SUBMITTED TO THE  
INSTITUTE OF GRADUATE STUDIES  
OF  
NEAR EAST UNIVERSITY**

**By  
HUSSEIN A.AZEEZ**

**In Partial Fulfillment of the Requirements for  
the Degree of Master of Science  
in  
Computer Engineering**

**NICOSIA 2021**

Hussein Mohammed Nazar A.AZEEZ: Classification of Glaucoma using Fundus Image with Machine Learning

Approval of Director of Institute of graduate studies

Prof. Dr. Hüsnü CAN BAŞER

We certify this thesis is satisfactory for the award of the degree of Master of Science of  
Computer Engineering

Examining Committee in Charge:

Assoc. Prof. Dr Boran ŞEKEROĞLU

Committee Chairman, Computer Engineering  
Department, NEU

Asst. Prof. Dr Yöney KIRSAL EVER

Committee Member, Software Engineering  
Department, NEU

Assoc. Prof. Dr Kamil DIMILILER

Supervisor, Automotive Engineering Department,  
NEU

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required

by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work .

Name, last name: Hussein A.Azeez

Signature:

A handwritten signature in black ink, consisting of a horizontal line that curves upwards and then back down to the right.

Date: 08-08-2021

## **ACKNOWLEDGEMENTS**

I would like to gratefully and sincerely thank Assoc. Prof. Dr. Kamil Dimililer for his guidance. Understanding, patience, and most importantly, his supervising during my graduate studies at Near East University. His supervision was paramount in providing a well-rounded experience consistent my long-term career goals. He encouraged me to not only grow as an experimentalist, but also as an instructor and an independent thinker. For everything you've done for me Assoc. Prof. Dr. Kamil, I thank you. I would also like to thank Prof. Rahib Abiyev for giving me the opportunity to be a member in such university and such department. His help and supervision concerning taking courses was unlimited.

I would also like to thank NEU Grand library administration members, since it provided me with the appropriate environment for conducting my research and writing my thesis.

**To my family...**

## ABSTRACT

Glaucoma is a very serious eye condition that has gained infamy for being the most popular thief of sight. It is difficult to diagnose, irreversible and most often ends with the total blindness of the patient. This makes it a very scary condition. As a result, early detection is a very key aspect in helping to save the patient from a life of total blindness. The difficulty of the diagnosis of the condition and its often late manifestation has made it very important to find ways to augment and improve the conventional diagnosis process.

Computer aided diagnosis is a fast growing area that sees the involvement of computers to aid in diagnosis and sometimes treatment of ailments and health conditions. This is an area that offers a promise of more accurate and early diagnosis of diseases which would help patients get early treatments and save cost of having to treat more advanced and critical cases of their ailment or health conditions. For Glaucoma, machine learning offers an opportunity for early and very accurate diagnosis and detection that could help in saving the sight of many patients.

In this thesis, I discuss an approach to glaucoma detection using Convolutional neural networks. The model is trained on fundus images of patients with glaucoma and without glaucoma, accurately labeled by experts. The images are taken through a preprocessing procedure to reduce the dimensions of the feature domain, avoiding the curse of dimensionality. Also, in this application, the conventional fully connected layer the precedes the output layer is replaced with a global average pooling layer, which reduces the number of parameters in the model and avoids overfitting. This makes the CNN to be smaller yet more effective.

**Keywords:** Machine Learning; CNN; Glaucoma detection; Fundus Analysis; global average pooling

## ÖZET

Glokom, en popüler görme hırsızı olduğu için rezil olmuş çok ciddi bir göz rahatsızlığıdır. Teşhis etmek zordur, geri döndürülemez ve çoğu zaman hastanın toplam körlüğü ile sona erer. Bu, onu çok korkutucu bir durum yapar. Sonuç olarak, erken teşhis, hastayı tam bir körlük yaşamından kurtarmaya yardımcı olan çok önemli bir husustur. Durumun teşhisinin zorluğu ve genellikle geç ortaya çıkışı, konvansiyonel teşhis sürecini güçlendirmenin ve iyileştirmenin yollarını bulmayı çok önemli hale getirmiştir.

Bilgisayar destekli teşhis, hastalıkların ve sağlık durumlarının teşhisine ve bazen tedavisine yardımcı olmak için bilgisayarların dahil olduğu hızla büyüyen bir alandır. Bu, hastaların erken tedavi almalarına yardımcı olacak ve hastalıklarının veya sağlık durumlarının daha ileri ve kritik vakalarını tedavi etme maliyetinden tasarruf etmelerine yardımcı olacak daha doğru ve erken teşhis vaadi sunan bir alandır. Glokom için makine öğrenimi, birçok hastanın görüşünü kurtarmaya yardımcı olabilecek erken ve çok doğru teşhis ve tespit için bir fırsat sunuyor.

Bu tezde, Evrişimli sinir ağlarını kullanarak glokom tespiti için bir yaklaşımı tartışıyorum. Model, uzmanlar tarafından doğru bir şekilde etiketlenmiş, glokomlu ve glokomu olmayan hastaların fundus görüntüleri üzerine eğitilmiştir. Görüntüler, boyutluluk lanetinden kaçınarak özellik alanının boyutlarını azaltmak için bir ön işleme prosedürü ile alınır. Ayrıca, bu uygulamada, çıktı katmanından önceki geleneksel tam bağlantılı katman, modeldeki parametre sayısını azaltan ve aşırı uydurmayı önleyen bir global ortalama havuzlama katmanıyla değiştirilir. Bu, CNN'in daha küçük olmasına rağmen daha etkili olmasını sağlar.

**Anahtar Kelimeler:** Makine Öğrenimi; CNN; Glokom tespiti; Fundus Analizi; küresel ortalama havuzlama



## TABLE OF CONTENTS

<b>ACKNOWLEDGMENTS</b> .....	<b>Error! Bookmark not defined.</b>
<b>ABSTRACT</b> .....	iv
<b>ÖZET</b> .....	v
<b>TABLE OF CONTENTS</b> .....	vi
<b>LIST OF TABLES</b> .....	ix
<b>LIST OF FIGURES</b> .....	x
<b>LIST OF ABBREVIATIONS</b> .....	xi
<b>CHAPTER 1: INTRODUCTION</b> .....	1
1.1 Problem Definition .....	4
1.2 The Aim of the Thesis .....	4
1.3 The Importance of the Work .....	5
1.4 Limitations of the Study .....	6
1.5 Overview of the Thesis .....	7
<b>CHAPTER 2: LITERATURE REVIEW</b> .....	9
2.1 Popular Machine Vision Applications in Health Care .....	9
2.1.1 Detection of Tumors .....	10
2.1.2 Fracture Detection.....	10
2.1.3 Image Compression.....	10
2.2 Computer Aided Diagnosis (CAD) of Glaucoma .....	11
2.2.1 C5 Algorithm.....	11
2.2.2 Support Vector Machines (SVM) .....	12
2.2.3 Naive Bayes .....	13
2.2.4 Canny Edge Detection Filter .....	14
2.2.5 Attention Based Glaucoma Detection.....	15
<b>CHAPTER 3: CONVOLUTIONAL NEURAL NETWORKS (CNN)</b> .....	17

3.1 How they Work .....	18
3.1.2 Feature Extraction .....	19
3.1.3 Classification .....	20
3.2 Structure of a Convolutional Neural Network .....	20
3.2.1 Convolution layer .....	21
3.2.2 Pooling Layer.....	21
3.2.3 Activation Function.....	21
3.2.4 Output Layer .....	22
<b>CHAPTER 4: SOFTWARE TOOLS AND CNN ARCHITECTURE.....</b>	<b>23</b>
4.1 Software Development Tools.....	23
4.1.1 Keras.....	23
4.1.2 NumPy.....	24
4.1.3 Pandas.....	24
4.1.4 Matplotlib .....	25
4.1.5 OpenCV .....	25
4.1.6 Imgaug.....	25
4.2 Model Architecture.....	26
4.3 Dataset .....	29
<b>CHAPTER 5: METHODOLOGY .....</b>	<b>31</b>
5.1 Importing Required Modules and Libraries.....	31
5.2 Loading the Dataset.....	33
5.3 Defining Parameters .....	33
5.4 Image Preprocessing.....	34
5.5 Creating the Model.....	35
<b>CHAPTER 6: RESULTS AND DISCUSSIONS .....</b>	<b>36</b>
6.1 Running a Prediction .....	36
6.2 Comparison to SVM.....	37
6.3 Limitations to the Study.....	37

<b>CHAPTER 7 :CONCLUSION AND FUTURE WORK .....</b>	<b>38</b>
7.1 Conclusion .....	38
7.2 Future Works.....	39
<b>REFERENCES .....</b>	<b>40</b>
<b>APPENDICES .....</b>	<b>43</b>
Appendix 1: Structure of Model and Parameters .....	44
Appendix 2: Ethical Approval Letter .....	47
Appendix 3: Similarity.....	48

## LIST OF TABLES

<b>Table 4.1:</b> CNN Architecture .....	25
--	----

## LIST OF FIGURES

<b>Figure 1.1:</b>	An illustration of how intraocular pressure (IOP) affects the optic nerve.....	2
<b>Figure 1.2:</b>	Illustration of the clogging of drainage canals in Open-angle glaucoma .....	2
<b>Figure 1.3:</b>	Illustration of the clogging of drainage canals in Angle-closure glaucoma.....	3
<b>Figure 2.1:</b>	A graphical representation of possible separation planes through features.....	9
<b>Figure 2.2:</b>	Graph showing the best possible line to split the features.....	10
<b>Figure 2.3:</b>	Original fundus image (left), image with vessel segmented (center), “vessel-free” fundus image (right) .....	11
<b>Figure 2.4:</b>	A fundus image showing the optic cup and optic disk .....	12
<b>Figure 2.5:</b>	Attention based analysis of fundus images .....	13
<b>Figure 3.1:</b>	Biological Neural Unit.....	15
<b>Figure 3.2:</b>	Artificial Neural Unit.....	15
<b>Figure 3.3:</b>	Structure of a convolutional neural network CNN.....	16
<b>Figure 4.1:</b>	Tensorflow APIs and their levels .....	22
<b>Figure 4.2:</b>	Averaging each feature map to form a flattened vector to be sent to the the output layer.....	24
<b>Figure 6.1:</b>	Code to train the model with the previously defined parameters.....	32
<b>Figure 6.2:</b>	Code to calculate and display the accuracy of the SVM alternative .....	33

## LIST OF ABBREVIATIONS

<b>AI:</b>	Artificial Intelligent
<b>API:</b>	Application Programmer Interface
<b>CNN:</b>	Convolutional Neural Network
<b>NN:</b>	Neural Network
<b>TPU:</b>	Tensor Processing Unit
<b>RIGA:</b>	Retinal fundus images for glaucoma analysis
<b>OD:</b>	Optic Disk
<b>IOP:</b>	Intraocular Pressure

## CHAPTER 1

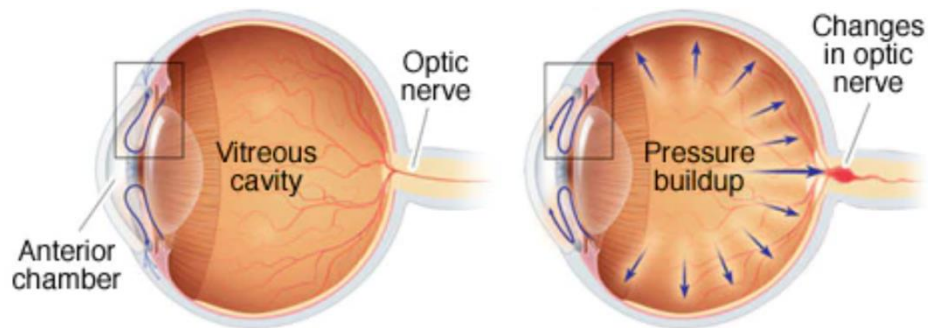
### INTRODUCTION

Glaucoma is an incurable condition that affects the human eye. It is a major cause of blindness and is quite difficult to detect. Glaucoma is generally caused by pressure on the optic nerve, which is very important for good eye sight. The damage caused by glaucoma progresses gradually, and in most cases, when the effects on the sight of a patient is noticed, the glaucoma is too advanced and cannot be managed. A key factor in the treatment of glaucoma is early detection which can help doctors or health practitioners prevent blindness or slow the progress of the condition. Machine learning has provided a powerful tool that can help in the early detection of glaucoma through the processing and analysis of retinal images.

There have been a lot of research involving the use of machine learning for the analysis of medical images which have helped in descriptive, diagnostic, and preventive measures for different conditions such as cardiovascular abnormalities, fractures and musculoskeletal injuries, neurological diseases and thoracic conditions and complications (HealthITAnalytics, 2018).

Glaucoma is an eye condition caused by abnormally high intraocular pressure (IOP) that causes damage to the optic nerve. Through history, the name glaucoma, derived from the Latin word *glauca* which means bluish-gray or green, was used loosely to describe conditions that involved clouding of vision such as cataracts and any similar conditions. In 1862, Franciscus Donders, a Dutch Ophthalmologist discovered that high intraocular pressure caused blindness and he called this condition *Glaucoma Simplex* although Richard Banister, the English ophthalmologist, was the first person

to define a connection between an increase in the tension of the eyeball and glaucoma. (Grewe, 1986). Figure 1.1 shows the effect of intraocular pressure on the optic nerve.

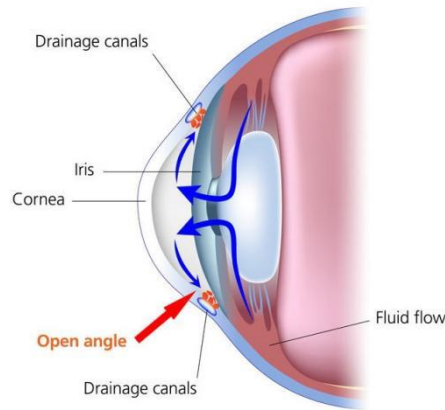


**Figure 1.1** An illustration of how intraocular pressure (IOP) affects the optic nerve

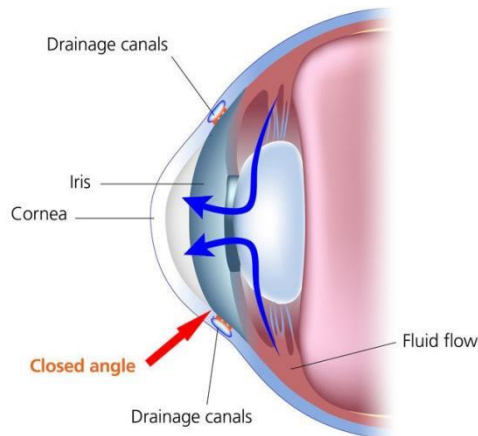
After this, glaucoma was approached and studied as an individual problem and its different types were distinguished.

Just like with some of the conditions mentioned above where machine learning has helped improve diagnosis, machine learning can also be applied to glaucoma to aid in its early detection. The two main types of glaucoma are *open-angle glaucoma* and *angle-closure glaucoma*, with open-angle glaucoma being the most common type of glaucoma with about 90% of glaucoma occurrences being open-angle. There are however a few other types of glaucoma which are essentially variants of the two main types such as *secondary glaucoma*, *pigmentary glaucoma*, *pseudoexfoliative glaucoma*, *traumatic glaucoma*, *neovascular glaucoma*, *irido corneal endothelial syndrome (ICE)* and *uveitic glaucoma*. The major difference between these two main types is the nature of the clogging of the drainage canals, which causes the IOP. In this thesis, I would be focusing on the open-angle type of glaucoma.





**Figure 1.2** Illustration of the clogging of drainage canals in Open-angle glaucoma



**Figure 1.3** Illustration of the clogging of drainage canals in Angle-closure glaucoma

Diagnosis of glaucoma is usually done by analysis of the Fundus, which is the back surface on the inside of the eye. It consists of the retina, optic disk, and blood vessels (Turbert, 2020). A fundus camera is used to take a picture of the back of the eye surface through the pupil in a process known as a Funduscopy. This image is then analysed by an ophthalmologist and used to make a diagnosis.

## 1.1 Problem Definition

One of the major challenges faced by ophthalmologists in diagnosing glaucoma is the fact that it is a slowly progressive condition that does not have much symptoms at its early stage and so can be very difficult to detect. In fact studies have estimate that half of the people with the condition are unaware that they have it. As a result, most of the time, when it is finally detected, it has already caused too much damage to the optic nerve making the blindness of the patient inevitable. It is currently the leading cause of blindness in the world. There is currently no cure for the condition, but early detection can help control its progress and either slow, or totally avoid blindness in the patient.

Another challenge to the diagnosis of glaucoma is the fact that glaucoma diagnosis has no specific pattern. Although optical pressure is the main causal factor of glaucoma, there are a few other risk factors, and other causes of pressure to the eye such as trauma can also cause glaucoma. Further more, its clinical representation could vary sometimes. This means that doctors often have to take a wild guess in their diagnosis if they intend to detect it at an early stage.

Glaucoma also has cost implications on the economy in the sense that it causes disabilities as well as reduced productivity (In some cases, peripheral vision may be lost before detection and consequently, treatment). As an example, through direct costs and the loss in productivity of people affected by the condition, glaucoma costs the US economy \$2.86 billion yearly (BrightFocus Foundation, 2020).

## **1.2 Aim of the Thesis**

The variation in the clinical representation of glaucoma makes it so that the best ophthalmologists to detect the condition at an early stage would be one with a lot of experience diagnosing and treating glaucoma. This is because, a longitudinal evaluation involving repeated observation of a lot of cases of the condition is

necessary to help the expert become more familiar with the condition. In most cases, the ophthalmologist might not have the opportunity to witness enough cases for an extensive longitudinal evaluation not to mention the fact that encountered cases prior to this level of expertise would be prone to error and also the fact that human memory might not serve too well.

The computation power of machine learning helps to reduce the time necessary for this longitudinal evaluation and increase the accuracy and efficiency of the detection afterwards.

In this thesis, I intend to take advantage of the opportunities provided by deep learning by developing a convolutional neural network (CNN) that would be trained over an array of retinal images of patients with glaucoma and patients without glaucoma. It is intended that the model would be trained enough to detect glaucoma at its onset and aid ophthalmologists to diagnose glaucoma faster and before more damage is done.

### **1.3 The Importance of the Work**

Although it is unfortunate that a cure has not yet been found for glaucoma, the fact that it can be controlled and blindness can be avoided in patients gives an great opportunity to the patient. The only key factor to this is that early detection is necessary.

Loss of sight is a terrible condition that leaves a person absolutely disabled. With blindness, even the most basic of tasks become very difficult, leaving the person not just unable to do anything, but also in constant need of assistance. In cases where blind people get trained enough to handle certain things themselves, a slight shift in routine or positioning of certain objects could bring their effectiveness to a halt. Furthermore, there are psychological implications to blindness such as sudden

outbursts of anger, feelings of neglect, depression, inferiority complex, anxiety, and others, as a result of the struggles brought about by the disability (Ishtiaq, 2016).

As the leading cause of blindness in the world, early detection of glaucoma, which would consequently aid in early treatment and prevention of blindness would go a long way in reducing cases of blindness and avoiding the previously mentioned situations. This would go along way in saving the sight of millions, avoiding the frustration of brought on by loss of sight, and keeping a lot of people and families happy.

#### **1.4 Limitations of the Study**

A key aspect of any machine learning application is the availability of good data. A major limitation to this study was the fact that in cases medical research, data is not always readily available to researchers because medical data is often considered confidential. In the cases where medical data is available, it could be very costly to obtain. In other cases, the data may have been truncated and lost their integrity, causing unreliable results.

Another limitation is the fact that available data for medical research might not be much because the process of collecting this data is quite extensive. Permissions must be sought, and sometimes, extensive paperwork maybe involved to ensure that the data is used within the ethical scope allowed by those making the data available. Also, there are not many medical institutions that gather real life data for medical research, and in cases where the institutions has an avenue for this, cultural perception could cause a lot of the patients not to be open to the idea.

## 1.5 Overview of the Thesis

The work in this thesis has been organized as follows.

In the second chapter, which would be the literature review, I would be discussing the different approaches of machine learning applications to medical diagnosis as an exposition to the idea of machine learning for medical applications. I would also be focusing on some approaches of machine learning already used in the detection of glaucoma.

In the third chapter, I would discuss convolutional neural networks with a brief explanation of deep learning, which is the class of machine learning that convolutional neural networks belong to. I would also discuss the key aspects of a convolutional neural network and the structure of its architecture.

In the fourth chapter, I would discuss my own convolutional neural network architecture and the software tools I would be using in my development process. I would explain the uses of each of the tools and how they affect the functionality and effectiveness of my model and system. I would also discuss the dataset used in training and testing the model in this chapter.

In the fifth chapter of this work, I would be discussing the development process of my model and system and explaining how the components work. I would also discuss any issues I might encounter in the development process and how these issues are solved.

In the sixth chapter, I would show my results and discuss the implications of these results. I would also discuss some tests that can be used to evaluate the effectiveness of the model.

In the seventh chapter, I would draw my conclusions on the work and discuss improvements that can be made as future work on the model to increase its effectiveness.

## CHAPTER 2

### LITERATURE REVIEW

The automation of the longitudinal evaluation in the glaucoma detection process is important because of the disparity, or in some cases, the obscurity of the early clinical representation of glaucoma. The disparity means that there are different approaches to the analysis of retinal images for the detection of glaucoma and these approaches can be correlated to different machine learning algorithms which can be used for the task of detection.

The most common methods for the diagnosis of glaucoma is through the observation of the optic disk (OD). The optic disk consists of two parts which are the neuroretinal rim, and the optic cup. The neuroretinal rim is the peripheral area of the fundus, while the optic cup is the bright yellowish area around the center of the fundus (Gundersen et al., 2009). These are some key features that ophthalmologists pay attention to in retinal images in the diagnosis of glaucoma. The automation of the analysis of these features is the aim of the application of machine learning to the process of glaucoma detection.

#### **2.1 Popular Machine Vision Applications in Health Care**

We have seen applications of machine learning in many fields, and health care is an area where machine learning shows a lot of promise. This is because there are a number of areas that could use the advantages and functionality provided by machine learning. A few of these areas are, and are not limited to the following.

### **2.1.1 Detection of Tumors**

Tumors are abnormal lumps of tissue that could either be benign or malignant. They could grow in most areas of the human body and usually have adverse effects on the host, sometimes even leading to death. For these reasons, detection of these tumors so they can be removed is very important. Machine vision has been applied in the detection of tumors in the body, like in the anterior segment of the eye (Dimililer et al., 2016), in the lungs (Dimililer et al., 2016) and brain (Sharma et al., 2014), among others. In these cases, machine learning is proposed to either help health care practitioners detect the tumors earlier or to help make detection easier in cases where detection of these tumors is difficult.

### **2.1.2 Fracture Detection**

Another area that machine vision has seen application in health care is in the detection of fractures, especially minor ones (Dimililer, 2017). Minor fractures are not easily detectable and could go unnoticed. This could cause pain in patients and if still untreated, could have more adverse effects on the patient. Applications like these could help detect and classify fractures, making sure nothing gets by the health care practitioner.

### **2.1.3 Image Compression**

Another application within the medical space for machine learning is in image compression. This is not a diagnostic application like in the previously mentioned examples, but is also very important. Management of the resource of storage space when dealing with Electronic Health Records (EHR) is very important. The formats in



which medical images tend to be stored is usually very bulky due to the attempt to ensure that all necessary aspects and features are preserved. Machine learning has been used to strike a balance of efficiency between managing storage space and maintaining quality of the images. Some applications have used back propagation to achieve lossless compression of medical images (Dimililer, 2013), and neural networks have also been used in the compression of X-ray images (Khashman & Dimililer, 2009).

## **2.2 Computer Aided Diagnosis (CAD) of Glaucoma**

Detection and diagnosis of glaucoma is a very difficult task that solely depends on high levels of expertise and experience from the ophthalmologists making the early detection of glaucoma a very important and sensitive issue. To aid in the process of diagnosis, research efforts have been directed towards Computer Aided Diagnosis (CAD) systems based on machine learning. This efforts aim to use the computation and automation opportunities provided by machine learning to aid the detection process. Some examples are as follows.

### **2.2.1 C5 Algorithm**

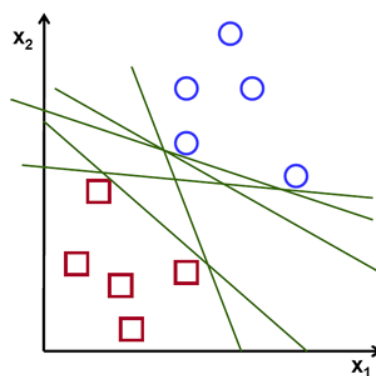
The C5 algorithm is an algorithm used to develop decision trees based on a dataset. It is the latest improvement to the ID3 (Iterative Dichotomiser 3) algorithm. ID3 works by iteratively splitting the features from the dataset into groups at each step. It employs a top-down greedy approach in developing decision trees, which means that the decision trees are developed from the top to the bottom and that at each step, the better or more relevant feature is preferred to be set as a decision node (Sakkaf, 2020).

In the C5 method, the algorithm decides on the split by using entropy to measure the purity(or homogeneity) of the feature and splitting it accordingly.

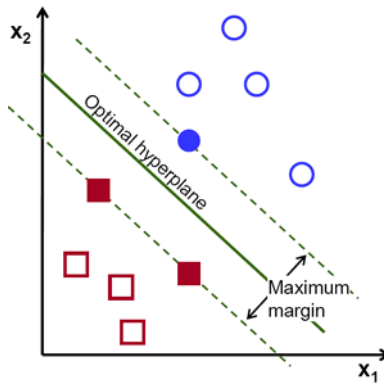
In glaucoma detection, The C5 algorithm has been used to develop decision trees based on features extracted using the Zernike moment method (McIntyre et al., 2004). Confocal Scanning Laser Tomography (CSLT) is a modern imaging technology that can be used to obtain high quality fundus images. In this example features are extracted from these CSLT images using the Zernike moment method, and the C5 algorithm is used to build a classification model.

### 2.2.2 Support Vector Machines (SVM)

Support vector machines (SVM) are a simple machine learning algorithm that are often used for linear regression and classification tasks. An SVM simply tries to find an optimal distinguishing line or plane (depending on 2D or 3D space) within the space or domain of the features with which it can separate data into separate classes.



**Figure 2.1** A graphical representation of possible separation planes through features



**Figure 2.2** Graph showing the best possible line to split the features

Figure 2.1 and 2.2 show an example of the working process of an SVM. If data points end up on different sides of the line or plane, then they are considered to be of different classes (Gandhi, 2018).

In this example ( Yu et al., 2005), Zernike moments method is used to extract features from CSLT images captured using the Heidelberg Retina Tomograph system and then the features are classified using an SVM.

### 2.2.3 Naive Bayes

In most cases of computer aided diagnosis of glaucoma, the diagnosis is based on shape or size analysis of sections of the fundus image after some form of image segmentation has been performed. In this approach (Bock et al., 2007), more focus is placed on a statistical evaluation of the data. As a result of this, the feature extraction, selection and preprocessing of the data is different from the other approaches and is very important if competent results are to be achieved. Features analyzed in this approach include pixel intensity values, textures, spectral features, and parameters of a histogram model

An example of this difference in preprocessing measures discussed above is the inpainting of the blood vessels in the fundus image to make for a “blood vessel free” fundus image. This is because the vessel sizes vary in size, shape and location from case to case and so could create an avenue for error considering the image-based approach of this method. This inpainting is done by doing a rough segmentation of the blood vessels and inpainting them spatially as seen in figure 2.3.



**Figure 2.3** Original fundus image (left), image with vessel segmented (center), “vessel-free” fundus image (right)

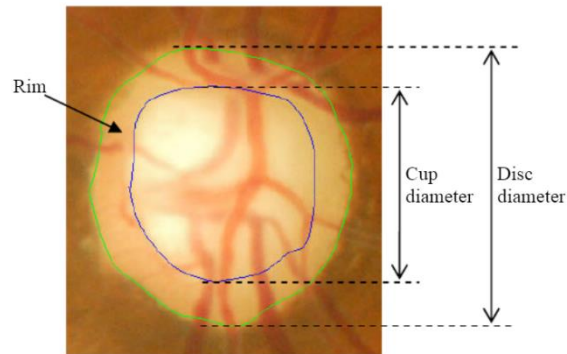
Also, In this approach the optic disk is centered in all images, and the images.

For the classification, the Bayes rule is used to determine the probability of an image belonging to a class with the assumption that the feature data is normally distributed, the predictive attributes are conditionally independent, and no other attributes influence the classification process.

#### **2.2.4 Canny Edge Detection Filter**

In this approach, glaucoma diagnosed by essentially analyzing the vertical cup-to-disk ratio in the fundus image (Hatanaka et al., 2010). To aid in detecting the optic disk in the neuroretinal rim, the blood vessels are first of all eliminated, and the optic disk is

detected using canny edge detection filter. When this is done, the size of the optic cup is measured in relation to the size of the optic disk as a way to diagnose glaucoma.

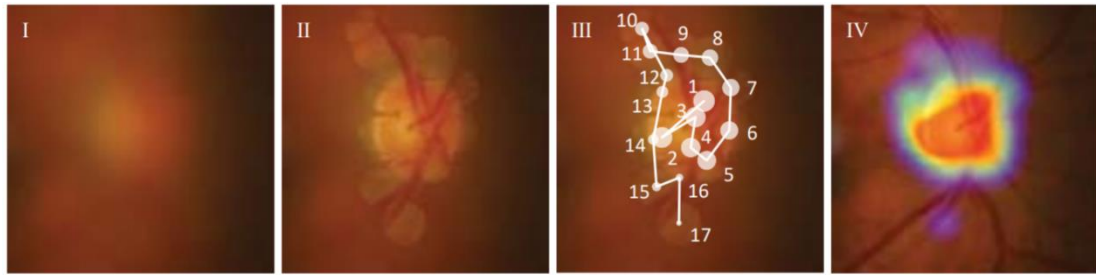


**Figure 2.4** A fundus image showing the optic cup and optic disk

This approach is based on the clinical presentation of glaucoma featuring an enlarged optic cup.

### 2.2.5 Attention Based Glaucoma Detection

This is an approach to Glaucoma detection with CNN and to image classification and object detection in general that involves analyzing the areas of an image that are most important and relevant to the classification or detection (Li et al., 2019). This ideology is similar to the way humans identify objects. The human eye does not parse the entire object to have an idea of what it is. Rather, we look at key features that are unique to the object and are relevant to the identification of the object.

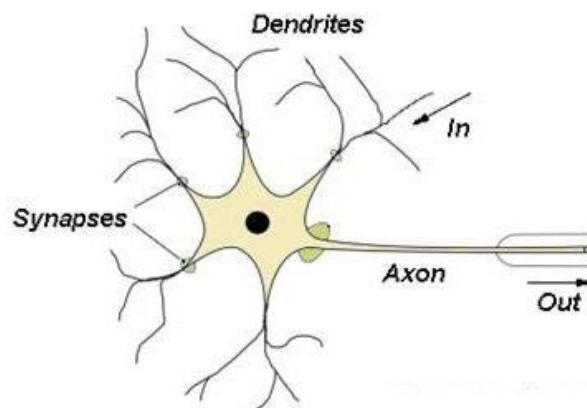


**Figure 2.5** Attention based analysis of fundus images; (I) Blurred image of a fundus; (II) the sharpened areas signify the areas of focus of professional ophthalmologists; (III) The order by which the areas of focus would be cleared; (IV) The generated image showing areas of focus

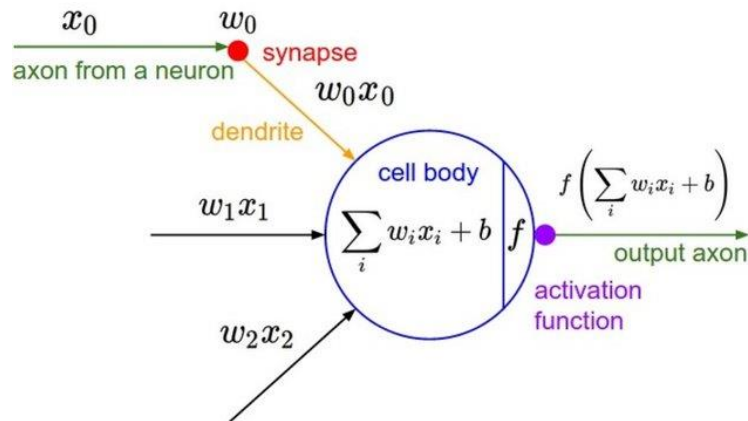
## CHAPTER 3

### CONVOLUTIONAL NEURAL NETWORKS (CNN)

Neural networks are an approach to machine learning that is based on the functionality of the human brain. Donald Hebb was the first to officially define some of the laws of neural networks, which he did in his book *The Organization of Behaviour* which was published in 1949, however work had already been done on the ideology of an artificial neural network by Warren McCulloch, a neurophysiologist, and Walter Pitts, a mathematician. In 1943, they published a paper describing how neurons work and they developed an electrical model of an artificial neural network.



**Figure 3.1** Biological Neural Unit



**Figure 3.2** Artificial Neural Unit

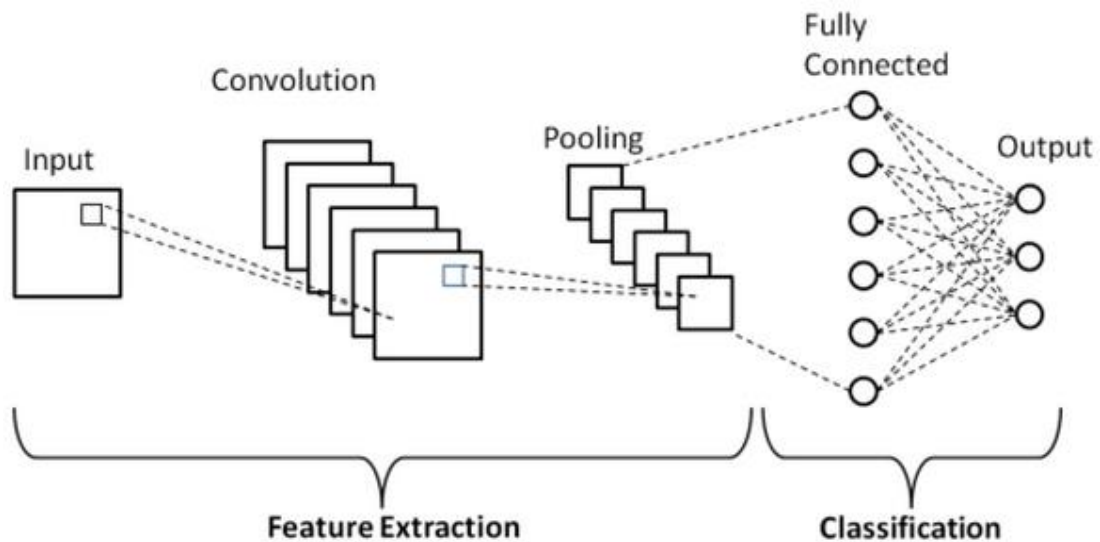
The human neural system consists of single neural units known as neurons, shown in figure 3.1. These units carry information in the form of stimuli caught at the dendrites through the synapses and axons and out onto the next neuron. The average human neural network is made up of about 86 billion of such neurons (Herculano-Houzel, 2012) passing information among each other constantly. Depending on the amount of stimulus coming from a preceding neuron, a neuron may be fired up or not.

Convolutional neural networks are a type of neural networks that are used mainly for computer vision applications. They are made up of layers of nodes (the artificial neural unit) all fully connected. The outputs of these nodes are adjusted based on the input value from the previous nodes, just like in the biological model (O'Shea & Nash, 2015).

### 3.1 How they Work

Convolutional Neural networks have a structure that can be split into two functional parts. The feature extraction part and the classification part. These two parts form the working system of the convolutional neural network.





**Figure 3.3** Structure of a convolutional neural network CNN

### 3.1.2 Feature Extraction

The functionality of a convolutional neural network is very similar to the functionality of the human eye. Two neurophysiologists, David H. Hubel and Torstel N. Wiesel in their research pointed out that the human brain processes visual data by extracting key features from the optic input captured by the eye, then combining these extracted features and making a decision on what the picture is believed to be. Essentially, the brain does not need to process every inch of the input image. This is the same ideology applied to feature extraction in convolutional neural networks. Specific parts of the image are selected and an inference is made based on the analysis of these selected parts.

The convolutional neural network model is able to decide on what parts of an image are important features for classification by supervised or unsupervised analysis of the training data, during which patterns or occurrences typical of a certain class are discovered and are watched for in the testing stage.

### 3.1.3 Classification

This is the part where the main work is done. Here, after the key features have been extracted from the input image, they are then analyzed and an inference is made. The feature extraction part actually plays a huge role in determining the effectiveness of the classification part. The features that would be extracted need to be very relevant to the classification process. Also, having too many features can increase the space, and having less data within this space could make the spatial difference of the data samples too high. This phenomenon is known as the *Curse of dimensionality*.

## 3.2 Structure of a Convolutional Neural Network

A typical Convolutional neural network is made of three major sections. The input layer, which is a single layer that takes in the input data into the neural network, the hidden layer, which could be made up of multiple layers interconnected and finally, the output layer, which gives the output of the neural network. The nodes in the hidden layer are interconnected in complex ways that aid in the training process. These layers are also set up with different tasks such as convolving and pooling.

The complexity of the network is also an important thing to pay attention to in building a convolutional neural network. Models with high complexities are more accurate, but require higher computation power and more training time. Also, the more the complexity of the network, the higher the risk of running into a phenomenon known as *overfitting*. This is a situation where the network is trained so much that its knowledge base becomes over specific to data that the model is trained on and might not work well with any new data. On the other extreme is a phenomenon known as *underfitting* which is where the model is so simple that it over generalizes the data causing a lot of false positives.

With the development of better CPUs and GPUs, complex machine learning models can now be used easily and higher accuracy levels can be attained.

### **3.2.1 Convolution Layer**

The convolution layers essentially select parts of the input image to be analyzed. This selection is in the form of matrices of pixels called convolution kernels. The kernels, which represent the key features of the image that are necessary for classification are used to form a feature map of the image. This feature map is made by convolving the inputs with the convolution kernel based on the activation function used in the nodes of the convolution layers. More convolution layers increase the accuracy of the CNN but increase the risk of overfitting (Brownlee, 2019).

### **3.2.2 Pooling Layer**

Pooling layers normally come after the convolution layers, and are used to downsize the dimensions of the feature map. The pooling operation is performed on each sample of the feature map (Tabian et al, 2019). The most common form of pooling used in CNN is max pooling.

Max pooling involves the selection of the maximum value in the convolution kernel. This results in a situation where the most present or major value in the kernel is highlighted as opposed to average pooling, where the average value of the kernel is highlighted (Brownlee, 2019). This method has been proven, through research to be more effective for image classification and object detection.

### **3.2.3 Activation function**

The activation function maps the input to each node to the output of the node. Just like in the biological neuron, it is used to determine if the node would be fired or not. There are a number of activation functions, which are selected based on the task at hand. Examples of activation functions are ReLU function, Sigmoid function, Tanh function, Leaky ReLU function and SoftMax function.

The most popular activation function is the ReLU (Rectified Linear Unit) activation function. It is also the preferred activation function for convolution layers in computer vision tasks like image classification and object detection. In the ReLU activation function, the output is set to zero for negative input, and the output is set to the same value as the input for positive values.

### **3.2.4 Output Layer**

The output layer is the last layer of a CNN. This is the layer that would give the result of a classification in the CNN. It also has a number of possible activations functions that are unique to output layers alone. Some examples are the Linear activation function, the logistic or sigmoid activation function, or the Softmax activation function, with the softmax being the most popular activation function in most CNNs. The Softmax activation function converts a numerical vector into a vector of probabilities (Brownlee, 2020). These probabilities correlate to and highlight the scale of the numerical vector. What this translates to in CNNs is that when the output or result is returned, it is returned as a vector of probabilities, showing the level or probability of each class being the chosen class of the input. This can be likened to the model telling us how certain it is of the classification.

## **CHAPTER 4**

### **SOFTWARE TOOLS AND SYSTEM DESIGN**

The development of my system was done with the python programming language, and some python libraries were also instrumental in the system design. There are some major software tools used in the development of the system, as well as some smaller tools used in smaller roles like in the preprocessing stages.

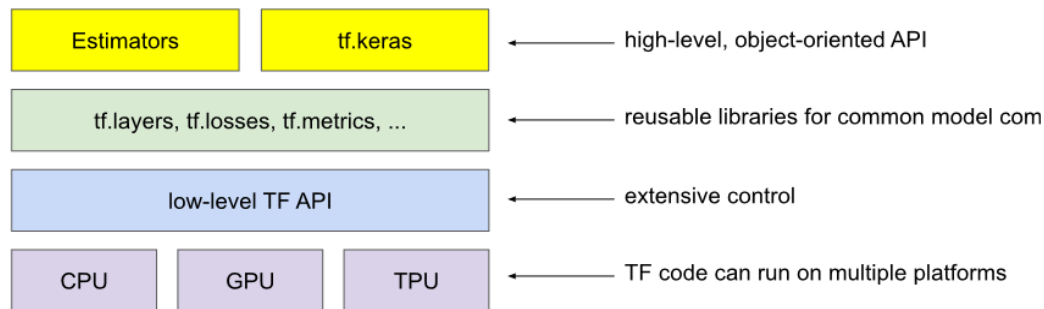
#### **4.1 Software Development Tools**

For the development, the python programming language was preferred because of its simplicity as well as the fact that a lot of machine learning research and development have been done with the python programming language meaning there are a lot of tools, tutorials and forums available to help in making the development process hassle free. Other libraries and tools used in the development process are given in this section. Details about how they are applied in this project would be given in chapter 5

##### **4.1.1 Keras**

Keras is a Tensorflow API designed with python that is set up with a modular structure that allows for fast prototyping. This makes it a favourite for production and research purposes. It is also user friendly and its modular structure makes it easily extendable.

Prior to the development of keras, another high level API that was used was the Estimator. The estimator is a high level API for the development of machine learning models that encapsulates four major actions - Training, evaluation, prediction and exporting.



**Figure 4.1** Tensorflow APIs and their levels

### 4.1.2 Numpy

Numpy is a python library for handling matrices and matrix operations. It consists of a powerful multidimensional array as well as tools and methods to be used in performing operations on them. It also makes it easy to develop data types with its array classes because of its easy and seamless array indexing. Because of these, it can be easily applied for scientific uses as well as applications involving generic data.

### 4.1.3 Pandas

Pandas is a python library that is mostly used for data manipulation and analysis. Pandas was created in 2008 by Wes McKinney with its name derived from *Python*

*Data Analysis*. It can also be used to clean up data to be used in data science and statistical applications.

#### **4.1.4 Matplotlib**

Matplotlib is another python library that I would be using in my work. It is used for representing mathematical data graphically or for making animated visuals. It can be used to make scatter plots, line plots, 3D plots, among other graphical representations for statistical data. It also comes with an object oriented API that makes it easy to integrate matplotlib into other applications.

#### **4.1.5 OpenCV**

OpenCV, which stands for Open source Computer Vision, is an image processing library created by Intel in 1999. In 2005, it was used in the development of Stanley, the vehicle that won the 2005 DARPA grand challenge. After that, it was improved and today is used on a large number of algorithms related to machine learning and computer vision. OpenCV array operations are usually carried out with the arrays in Numpy format, making it easy to use Numpy operations with OpenCV.

#### **4.1.6 Imgaug**

Imgaug is a library used in machine learning applications image augmentation. It is mostly used in the development of convolutional neural networks. It can also be used to set bounding boxes, segmentation maps, and to highlight keypoints in an image. It

can be used to execute these augmentation or highlighting operations on large amounts of images together or to perform a sequence of augmentation operations.

## **4.2 Model Architecture**

The CNN in this project is a supervised learning model. In my model, the hidden layer was split into four sections or blocks with 4 convolutional layers stacked on each other and the followed by a max pooling layer.

In the first block, I have four convolutional layers starting with 8 filters on the first layer and doubling up until 64 filters in the 4 layer. The ReLU activation function was used for all the four layers. They are then followed by a max pooling layer.

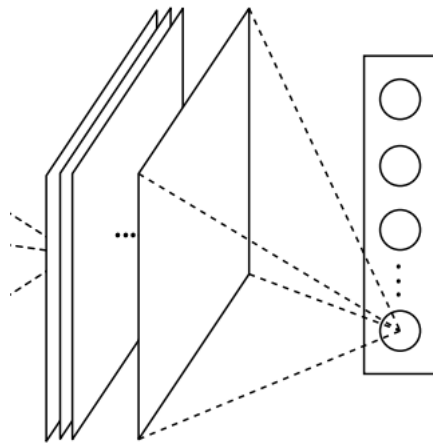
In the second section, I have four convolutional layers again, but starting with 32 filters on the first layer, 64 filters on both the second and third layers, and finally 128 filters on the fourth layer. The ReLU activation function was used for all four layers and they were followed by a max pooling layer.

In the third section, I have four convolutional layers, with 64 filters on the first and fourth layers, and 128 filters on the second and third layers. They all use the ReLU activation function and are followed by a max pooling layer also.

Finally in the fourth section, I have four convolutional layers , starting with 64 filters on the first layer, and reducing to 32 filters on the second layer, 16 filters on the third, and finally 8filters on the fourth layer. They use the ReLU activation function and are followed by a max pooling layer.



After these, in place of a fully connected layer, I have a global average pooling layer. Here, instead of finishing the architecture with fully connected layer as in conventional convolutional neural networks, the global average pooling layer takes the average of each feature map and sends the result as a vector to the output layer. These helps to generate a single feature map for each classification category.



**Figure 4.2** Averaging each feature map to form a flattened vector to be sent to the the output layer

Finally, in the output layer, I have a softmax activation function.

All layers are two dimensional, and the convolutional layers all have a kernel size of 3 by 3. The pooling layers are all 2 by 2.

**Table 4.1** CNN Architecture

Layers	Type	Filters	Kernel	Activation
--------	------	---------	--------	------------

			<b>Size</b>	<b>Function</b>
<b>Section 1</b>				
conv_11	Convolution	8	3 x 3	ReLU
conv_12	Convolution	16	3 x 3	ReLU
conv_13	Convolution	32	3 x 3	ReLU
conv_14	Convolution	64	3 x 3	ReLU
maxpooling	Pooling		2 x 2	Max Pooling
<b>Section 2</b>				
conv_21	Convolution	32	3 x 3	ReLU
conv_22	Convolution	64	3 x 3	ReLU
conv_23	Convolution	64	3 x 3	ReLU
conv_24	Convolution	128	3 x 3	ReLU
maxpooling	Pooling		2 x 2	Max Pooling
<b>Section 3</b>				
conv_31	Convolution	64	3 x 3	ReLU
conv_32	Convolution	128	3 x 3	ReLU
conv_33	Convolution	128	3 x 3	ReLU
conv_34	Convolution	64	3 x 3	ReLU
maxpooling	Pooling		2 x 2	Max Pooling
<b>Section 4</b>				
conv_41	Convolution	64	3 x 3	ReLU
conv_42	Convolution	32	3 x 3	ReLU
conv_43	Convolution	16	3 x 3	ReLU

conv_44	Convolution	8	3 x 3	ReLU
maxpooling	Pooling		2 x 2	Max pooling
<b>Section 5</b>				
GlobalAveragePooling	Pooling			Average Pooling
<b>Output Layer</b>				
Output				Softmax

### 4.3 Dataset

The dataset used in a machine learning application is key defining factor of the effectiveness of the machine learning model as bad data can have adverse effects on the effectiveness of a machine learning model (Ehrlinger et al., 2019). As such, the need to know quality of the data to be used in training a machine learning model can not be overemphasized. Even with this obvious fact, little research has been done regarding the evaluation of the quality of data. This is especially true in cases where data is collected in real time from sensors, as a one time assessment of the quality of the data would be largely insufficient. In such cases, data quality assessment would need to be incorporated into the development process as a key part of the system.

In the case of medical applications, data might not be as easily accessible as in other applications. There are a number of challenges with accessing medical data (Norwich University, 2020). As an example, in many countries, there are a number of laws that

highly regulate the accessibility, distribution and use of the medical information of patients.

Another challenge is the fragmentation of medical data. This means that the data is not uniform and can be hard to organize. In some cases even, data is recorded manually on paper. This consequently affects efficiency and can make some of the data to end up not being used. This also makes it necessary to have skilled experts to work on the data to organize it.

Another issue faced with medical data is the fact that the data repository is not centralized. This can make accessing the data to be a hassle. There are also varying procedures from source to source involved in the procurement of this data. This can extend the process of data collection, with no certainty of success in some cases.

For this project, I used the RIGA (Retinal Fundus Images for Glaucoma Analysis) dataset. This data set was compiled by reserachers in uniersities in Saudi Arabia and Canada and is available for free on the University of Michigan Deep Blue website.

## CHAPTER 5

### METHODOLOGY

In this section I would be discussing the implementation process of my project. The implementation is done on Google Colab. Google Colab allows users to run code on the Google cloud server. It is used mostly in machine learning applications so that the computational load is put on Googles GPUs and TPUs. With this even very complex model training can be done without fear of crashing my PC.

In the following sections, I would be highlighting the steps that I took in the development of my model

#### 5.1 Importing Required Modules and Libraries

First of all, I had to install all the required libraries and modules I needed in the development process. This includes all of the modules and libraries mentioned in chapter 4.

For the purpose of backward compatibility and to avoid the complex migration problems associated with the upgrade from TensorFlow 1.x to 2.x, I used the *disable\_v2\_behavior* function to force all global behaviors of TensorFlow to behave like it was designed for TensorFlow 1.x.

A number specific modules from the keras library were imported for specific purposes

The *ImageDataGenerator* module is used to carry out preprocessing or augmentation operations on the training and testing images. These are necessary to help extract more useful data from the images, hence increasing the effectiveness of the model.

The different types of layers necessary for the development of our CNN, such as convolution, max pooling and global average pooling are also imported from the *keras.layers* library. The modules used to invoke the activation functions on the layers are also imported from this library.

The keras backend is also imported. Keras has three backend implementations, namely TensorFlow, Theano, and CNTK. The backend implementation can be switched by changing the value of the “backend” variable in the *keras.json* configuration file.

The Keras *optimizer* module is used to update the model based on the results gotten from the loss function. This maps the loss function to the parameters of the model and hence increasing the accuracy of the model.

The *ModelCheckpoint* module is used to save the state of the model after each epoch. This is so that the training process is continued from the last saved state of the model. This also helps to know the current accuracy of the model at each stage so we can know if the last training iteration causes any change to the accuracy of the model. This also helped in the development process so I knew how many epochs were actually necessary for any tangible change to the accuracy of the model.

With the *imgaug* module *augmenters*, I set up a sequence of augmentations to be performed on the input images.

The Keras *plot\_model* is used to write the model to a file after it has been developed and trained so that it can be used later.

The *numpy* module is used help represent the images in a 2D array format and to help make it easy to perform mathematical operations on the 2D array created.

## **5.2 Loading the Dataset**

Here, the dataset is loaded from a google drive that requires authentication. The google modules for the google authentication are first installed. At the first time when the code is run, a unique access code is given for a signed in google account. This gives access to the zip file containing the dataset.

This is necessary, as the custodians of the dataset have the google drive set up as such. This ensures the integrity of the data set while also ensuring accessibility to the dataset. The authentication allows read-only access, so the dataset cannot be altered or edited by the users.

## **5.3 Defining Parameters**

Here, necessary parameters for the model are defined. The training samples, test samples, validation samples and epochs are set. The input shape is set to have 3 inputs; the image width, height and depth (3 channels for the RGB color scheme). The size of the input images is also set to 256 x 256.

## **5.4 Image Preprocessing**

I used `imgaug` to prepare a sequence of augmentations to be performed on the input images before they are run through the CNN. First of all, the Gaussian blur is used to smoothen the image, eliminating the blood vessels in the fundus, then the image is sharpened. The CoarseDropout is finally used to remove random rectangles in the input images to reduce overfitting.

## **5.5 Image Processing Tests**

To determine the best image augmentation techniques and to what extent these image augmentations should be used, I ran some tests. The first tests were to determine how much of the dataset I should perform the augmentation process on. I tested with augmentation on 10%, 50%, 60% and 100% of the dataset. For 10% I achieved an accuracy of 82%, for 50% I achieved an accuracy of 82% also, for 60%, I achieved an accuracy of 88%, and for 100% augmentation, I achieved an accuracy of 83%. Based on these results, I decided to perform the augmentation on 60% of the images.

After deciding on these parameters, I ran tests for the blurring section that handles the removal of the blood vessels. I tested with Gaussian blur, Average blur, median blur and bilateral blur. I achieved accuracies of 88%, 85%, 84% and 83% respectively. For this reason, I decided to use the Gaussian blur for the first level of processing.

## **5.6 Creating the Model**



Here, the model is created to the specifications discussed in section 4.2. The model is made up of 4 blocks in the hidden layer. The blocks all consist of convolution layers with the ReLU activation function and max pooling layers as shown in the following images.

In the first block, there are 4 convolution layers with 8, 16, 32, and 64 filters respectively, followed by a max pooling layer.

In the second block, there are 4 convolution layers with 32, 64, 64, and 128 filters respectively, followed by a max pooling layer.

In the third block, there are 4 convolution layers with 64, 128, 128, and 64 filters respectively, followed by a max pooling layer.

In the fourth block, there are 4 convolution layers with 64, 32, 16, and 8 filters respectively, followed by a max pooling layer.

This is then followed by a layer with global average pooling and finally the output layer with the softmax activation function.

## CHAPTER 6

### RESULTS AND DISCUSSIONS

After the model was developed, I used the `model.fit_generator` module to train the model with the predefined parameters shown in section 5.3.

The model was compiled and trained over 100 epochs and a training accuracy of 89.5% was achieved. The `modelCheckpoint` module was used to save the best accuracy attained during the training process.

```
Epoch 00099: acc did not improve from 0.89516
Epoch 100/100
8/8 [=====] - 3s 381ms/step - loss: 0.3471 - acc: 0.8891

Epoch 00100: acc did not improve from 0.89516
<keras.callbacks.History at 0x7f684b7b1400>
```

**Figure 6.1** Accuracy of the model

The test accuracy was 65%. In CNNs, the test accuracy must be less than the training accuracy to signify that there is no overfitting. If the test accuracy is far lower than training accuracy, this signifies underfitting.

#### 6.1 Running a Prediction

The prediction process is run over the test data, and an array of probabilities of the presence of glaucoma is returned as the result. These probabilities are then transformed with a threshold of 0.6. That is, probabilities above 0.6 are set to 1 and

probabilities below 0.6 are set to 0, with 1 representing glaucoma and 0 representing no glaucoma.

## 6.2 Comparison to SVM

For the sake of comparison, I found an implementation of glaucoma detection using SVM using extracted features from the same dataset I used in my CNN implementation so I could compare the accuracy of the implementation with that of mine. The SVM alternative had an accuracy of 75%.

```
print ("baseline accuracy " , ( (model.predict(X_test) == Y_test).astype("int8").sum() / Y_test.shape[0] ) )  
baseline accuracy 0.75
```

**Figure 6.2** Code to calculate and display the accuracy of the SVM alternative

## 6.3 Limitations to the Study

One major challenge that was faced in this work was the access to the dataset. A problem in most machine learning work and research is the use of bad data, which leads to the development of inefficient models which give wrong results. This fact coupled with the fact that medical data is not always readily available made this a delicate topic to work on. The availability of the RIGA data set helped solve this problem.

Another challenge was compatibility issues with Tensorflow versions 1 and 2. This forced me to disable all version 2 behaviors to avoid any conflict between the application of the versions in the code.

## CHAPTER 7

### CONCLUSION AND FUTURE WORK

#### 7.1 Conclusion

The implementation was a success and desirable results were achieved. Superior accuracy was gained as opposed to the alternative implementation with SVM. This was mostly unsurprising because CNNs have the best applications when it comes to classification or object detection applications, and this work also serves to prove that.

This application further proves that machine learning has a lot of potential to aid in the medical and healthcare field to help in the diagnosis of diseases and ailments that are normally difficult to diagnose even with the experience of an expert in some cases. This would help provide a more accurate opinion rather than getting multiple and sometimes, conflicting opinions on the same condition

Automated diagnosis is a growing and very necessary area to research. For a case like glaucoma, experts who performed a study at the Helsinki Private Eye Hospital on the accuracy of glaucoma diagnosis in everyday practice recommended using automated methods for the diagnosis of glaucoma (Lindbohm et al., 2017). This further highlights the importance of this area of work

## **7.2 Future Works**

In the future, improvements could be made on works such as this to make it possible for them to be used in the diagnosis of other eye conditions such as diabetic retinopathy. A more robust system could be developed that allows the ophthalmologist to simply input a fundus image and have any possible eye condition diagnosed. The results can then be assessed also by the expert to ascertain the diagnosis put forth by the system.

More generally, automated diagnosis for more diseases and conditions that are normally difficult to diagnose should be researched so as to take advantage of the computational power and efficiency of machine learning in healthcare applications.

## REFERENCES

- Bock, R., Meier, J., Michelson, G., Nyúl, L. G., & Hornegger, J. (2007). Classifying glaucoma with IMAGE-BASED features from Fundus photographs. *Lecture Notes in Computer Science*, 355–364. [https://doi.org/10.1007/978-3-540-74936-3\\_36](https://doi.org/10.1007/978-3-540-74936-3_36)
- BrightFocus Foundation. (2020, September 24). Glaucoma: Facts & figures. Retrieved March 02, 2021, from <https://www.brightfocus.org/glaucoma/article/glaucoma-facts-figures#:~:text=40%20and%20over%20more%20than%20three%20million%20Americans%20are%20living%20with%20glaucoma%2C%202.7,over%2011%20million%20by%202040>.
- Brownlee, J. (2019). *Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python*. Machine Learning Mastery.
- Brownlee, J. (2019, July 5). *A Gentle Introduction to Pooling Layers for Convolutional Neural Networks*. Machine Learning Mastery. <https://machinelearningmastery.com/pooling-layers-for-convolutional-neural-networks/>.
- Brownlee, J. (2020, June 23). Softmax activation function with Python. Retrieved April 18, 2021, from <https://machinelearningmastery.com/softmax-activation-function-with-python/>
- Dimililer, K. (2013). Backpropagation neural network implementation for medical image compression. *Journal of Applied Mathematics*, 2013, 1-8. doi:10.1155/2013/453098
- Dimililer, K. (2017). IBFDS: Intelligent bone fracture detection system. *Procedia Computer Science*, 120, 260–267. <https://doi.org/10.1016/j.procs.2017.11.237>
- Dimililer, K., Ever, Y. K., & Ratemi, H. (2016). Intelligent eye tumour detection system. *Procedia Computer Science*, 102, 325–332. <https://doi.org/10.1016/j.procs.2016.09.408>
- Dimililer, K., Ever, Y. K., & Ugur, B. (2016). ILTDS: Intelligent Lung Tumor Detection System on CT Images. *Advances in Intelligent Systems and Computing*, 225–235. [https://doi.org/10.1007/978-3-319-47952-1\\_17](https://doi.org/10.1007/978-3-319-47952-1_17)
- Ehrlinger, L., Haunschmid, V., Palazzini, D., & Lettner, C. (2019). A daql to monitor data quality in machine learning applications. *Lecture Notes in Computer Science*, 227-237. doi:10.1007/978-3-030-27615-7\_17

- Hatanaka, Y., Noudo, A., Muramatsu, C., Sawada, A., Hara, T., Yamamoto, T., & Fujita, H. (2010). Vertical cup-to-disc ratio measurement for diagnosis of glaucoma on fundus images. *Medical Imaging 2010: Computer-Aided Diagnosis*. <https://doi.org/10.1117/12.843775>
- Herculano-Houzel, S. (2012). The remarkable, yet not extraordinary, human brain as a scaled-up primate brain and its associated cost. *Proceedings of the National Academy of Sciences*, *109*(Supplement\_1), 10661–10668. <https://doi.org/10.1073/pnas.1201895109>
- HealthITAnalytics. (2018, November 05). Top 5 use cases for artificial intelligence in medical imaging. Retrieved March 01, 2021, from <https://healthitanalytics.com/news/top-5-use-cases-for-artificial-intelligence-in-medical-imaging>
- Ishtiaq, R. (2016). Psychosocial implications of blindness and low vision in students of Blind School Bahawalpur. *Pakistan Journal of Medical Sciences*, *32*(2). doi:10.12669/pjms.322.8737
- Gandhi, R. (2018, July 05). Support vector machine - introduction to machine learning algorithms. Retrieved March 08, 2021, from <https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47>
- Grewe, R. (1986). Zur Geschichte Des glaukoms. *Klinische Monatsblätter Für Augenheilkunde*, *188*(02), 167-169. doi:10.1055/s-2008-1050606
- Gundersen, K. G., Heijl, A., & Bengtsson, B. (2009). Sensitivity and specificity of structural optic disc parameters in chronic glaucoma. *Acta Ophthalmologica Scandinavica*, *74*(2), 120-125. doi:10.1111/j.1600-0420.1996.tb00054.x
- Khashman, A., & Dimililer, K. (2009). Medical radiographs compression using neural networks and Haar wavelet. *IEEE EUROCON 2009*. <https://doi.org/10.1109/eurcon.2009.5167831>
- Lindbohm, N. M., Vesti, E., Puska, P., Mäkitie, T., Friberg, K., & Harju, M. (2017). Diagnostic accuracy of glaucoma tests in everyday practice. *Ophthalmology @ Point of Care*, *1*(1). doi:10.5301/oapoc.0000020
- McIntyre, A. R., Heywood, M. I., Artes, P. H., & Abidi, S. S. R. (2004). Toward glaucoma classification with moment methods. *First Canadian Conference on Computer and Robot Vision, 2004. Proceedings.*, 265–272. <https://doi.org/10.1109/ccrv.2004.1301454>
- Norwich University. (2020, May 27). Current big data challenges in hospitals and healthcare. Retrieved April 15, 2021, from <https://online.norwich.edu/academic-programs/resources/data-challenges-in-healthcare>

- O'Shea, K., & Nash, R. (2015, December 2). *An Introduction to Convolutional Neural Networks*. arXiv.org. <https://arxiv.org/abs/1511.08458v2>.
- Sharma, K., Kaur, A., & Gujral, S. (2014). Brain Tumor Detection based on Machine Learning Algorithms. *International Journal of Computer Applications*, 103(1), 7–11. <https://doi.org/10.5120/18036-6883>
- Sakkaf, Y. (2020, September 12). *Decision Trees for Classification: ID3 Algorithm Explained*. Medium. <https://towardsdatascience.com/decision-trees-for-classification-id3-algorithm-explained-89df76e72df1>.
- Tabian, I., Fu, H., & Khodaei, Z. S. (2019). A Convolutional Neural Network for Impact Detection and Characterization of Complex Composite Structures. *Sensors*, 19(22), 4933. <https://doi.org/10.3390/s19224933>
- Turbert, D. (2020, May 20). Fundus. Retrieved March 02, 2021, from <https://www.aao.org/eye-health/anatomy/fundus>
- Varachiu, N., Karanicolas, C., & Ulieru, M. (2002). Computational intelligence for medical knowledge acquisition with application to glaucoma. *Proceedings First IEEE International Conference on Cognitive Informatics*, 233–238. <https://doi.org/10.1109/coginf.2002.1039303>
- Yu, J., Sibte, S., Abidi, R., Artes, P. H., McIntyre, A., & Heywood, M. (2005). Automated Optic Nerve Analysis for Diagnostic Support in Glaucoma. *18th IEEE Symposium on Computer-Based Medical Systems (CBMS'05)*, 97–102. <https://doi.org/10.1109/cbms.2005.36>



## **APPENDICES**

## Appendix 1

### Structure of Model and Parameters

Layer (type)	Output Shape	Param #
input_7 (InputLayer)	(None, 256, 256, 3)	0
norm_0 (BatchNormalization)	(None, 256, 256, 3)	12
conv_11 (Conv2D)	(None, 254, 254, 8)	224
conv_12 (Conv2D)	(None, 252, 252, 16)	1168
conv_13 (Conv2D)	(None, 250, 250, 32)	4640
conv_14 (Conv2D)	(None, 248, 248, 64)	18496
max_pooling2d_13 (MaxPooling)	(None, 124, 124, 64)	0
norm_1 (BatchNormalization)	(None, 124, 124, 64)	256
conv2d_40 (Conv2D)	(None, 124, 124, 16)	1040
conv_21 (Conv2D)	(None, 122, 122, 32)	4640
conv_22 (Conv2D)	(None, 120, 120, 64)	18496
conv_23 (Conv2D)	(None, 118, 118, 64)	36928
conv_24 (Conv2D)	(None, 116, 116, 128)	73856

max\_pooling2d\_14 (MaxPooling (None, 58, 58, 128)) 0

---

norm\_2 (BatchNormalization) (None, 58, 58, 128) 512

---

conv2d\_41 (Conv2D) (None, 58, 58, 64) 8256

---

conv\_31 (Conv2D) (None, 56, 56, 64) 36928

---

conv\_32 (Conv2D) (None, 54, 54, 128) 73856

---

conv\_33 (Conv2D) (None, 52, 52, 128) 147584

---

conv\_34 (Conv2D) (None, 50, 50, 64) 73792

---

max\_pooling2d\_15 (MaxPooling (None, 25, 25, 64)) 0

---

norm\_3 (BatchNormalization) (None, 25, 25, 64) 256

---

conv\_41 (Conv2D) (None, 23, 23, 64) 36928

---

conv\_42 (Conv2D) (None, 21, 21, 32) 18464

---

conv\_43 (Conv2D) (None, 19, 19, 16) 4624

---

conv\_44 (Conv2D) (None, 18, 18, 8) 520

---

max\_pooling2d\_16 (MaxPooling (None, 9, 9, 8)) 0

---

norm\_4 (BatchNormalization) (None, 9, 9, 8) 32

---

conv2d\_42 (Conv2D) (None, 9, 9, 2) 18

---

global\_average\_pooling2d\_2 ( (None, 2)) 0

---

activation\_2 (Activation) (None, 2) 0

=====

=====

Total params: 561,526

Trainable params: 560,992

Non-trainable params: 534

---

## APPENDIX 2: Ethical Approval Letter



### ETHICAL APPROVAL DOCUMENT

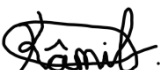
Date:08/08/2021

To the Institute of Graduate Studies

For the thesis project entitled as “Classification of Glaucoma using Fundus Image with Machine Learning”, the researchers declare that they did not collect any data from human/animal or any other subjects. Therefore, this project does not need to go through the ethics committee evaluation.

Title: Classification of Glaucoma using Fundus Image with  
Machine Learning

Name Surname: Assoc. Prof. Dr.Kamil Dimililer

Signature: 

Role in the Research Project: Supervisor