M.Sc. Engg. Thesis

DESIGN AND DEVELOPMENT OF A DEEP LEARNING BASED APPLICATION FOR DETECTING DIABETIC RETINOPATHY

by
Md. Tarikul Islam Papon

Submitted to

Department of Computer Science and Engineering

in partial fulfillment of the requirements for the degree of
Master of Science in Computer Science and Engineering



Department of Computer Science and Engineering

Bangladesh University of Engineering and Technology (BUET)

Dhaka 1000

May 2019

Dedicated to my loving parents and to my beloved wife

Author's Contact

Md. Tarikul Islam Papon

Lecturer

Department of Computer Science and Engineering

Bangladesh University of Engineering and Technology (BUET).

Email: tarikulpapon@cse.buet.ac.bd, tarikulpapon@gmail.com

The thesis titled "DESIGN AND DEVELOPMENT OF A DEEP LEARNING BASED APPLICATION FOR DETECTING DIABETIC RETINOPATHY", submitted by Md. Tarikul Islam Papon, Roll No. **1015052028 P**, Session October 2015, to the Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, has been accepted as satisfactory in partial fulfillment of the requirements for the degree of Master of Science in Computer Science and Engineering and approved as to its style and contents. Examination held on May 14, 2019.

Board of Examiners

1. _____

Dr. A.K.M. Ashikur Rahman — Chairman (Supervisor)
Professor
Department of CSE, BUET, Dhaka.

2. _____

Dr. Md. Mostofa Akbar — Member (Ex-Officio)
Head and Professor
Department of CSE, BUET, Dhaka.

3. _____

Dr. Rifat Shahriyar — Member
Associate Professor
Department of CSE, BUET, Dhaka.

4. _____

Abu Wasif — Member
Assistant Professor
Department of CSE, BUET, Dhaka.

5. _____

Dr. Mohammad Rashedur Rahman — Member (External)
Professor
Department of Electrical and Computer Engineering,
North South University, Dhaka-1229.

Candidate's Declaration

This is hereby declared that the work titled "DESIGN AND DEVELOPMENT OF A DEEP LEARNING BASED APPLICATION FOR DETECTING DIABETIC RETINOPATHY" is the outcome of research carried out by me under the supervision of Dr. A.K.M. Ashikur Rahman, in the Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka 1000. It is also declared that this thesis or any part of it has not been submitted elsewhere for the award of any degree or diploma.

Md. Tarikul Islam Papon

Candidate

Acknowledgment

I express my heartiest gratitude, profound indebtedness and deep respect to my supervisor, Dr. A.K.M. Ashikur Rahman for his constant supervision of this work. He helped me a lot in every aspect of this work and guided me with proper directions whenever I sought one. His patient hearing of my ideas, critical analysis of my observations and detecting flaws (and amending thereby) in my thinking and writing have made this thesis a success.

I would also want to thank the members of my thesis committee for their valuable suggestions. I thank Dr. Md. Mostofa Akbar, Dr. Rifat Shahriyar, Abu Wasif, and specially the external member Dr. Mohammad Rashedur Rahman.

In this regard, I remain ever grateful to my loving parents and to my beloved wife Sadia who always exist as sources of inspiration behind every success of mine I have ever made.

Abstract

Diabetic retinopathy (DR), a complication of diabetes, is one of the leading causes of blindness globally. Since early detection of DR can reduce the chance of vision loss significantly, regular retinal screening of diabetic patients is an essential prerequisite. However, due to inefficient manual detection as well as lack of resources and ophthalmologists, early detection of DR is severely hindered. Moreover, subtle differences among different severity levels and the presence of small anatomical components make the task of identification very challenging. The objective of this study is to develop a robust diagnostic system through integration of state-of-the-art deep learning techniques for automated DR severity detection. We used the concept of deep Convolutional Neural Networks (CNNs), which have revolutionized different branches of computer vision including medical imaging. Our deep network is trained on the largest publicly available Kaggle data set using our very own novel loss function. After several preprocessing and augmentation, 159,464 images were used for the training of the model. 10,000 images of Kaggle data was kept separate for testing purpose. Unlike most retrospective studies which perform binary classification (DR vs no DR), our model is trained to output five classes of DR severity as per international standard. An accuracy of 79.57% with a sensitivity of 79.58%, specificity of 82.81%, precision of 79.57% and F1 score of 0.778 was achieved on the test data. The model is also validated using two independent databases: Messidor and E-Ophtha to demonstrate its efficacy and generalization ability. In addition, a general comparison with some existing studies has been carried out to show that our model's performance is comparable to the recent state-of-the-art models. The implementation of such a model to identify DR severity level accurately can reduce the risk of vision loss drastically by referring the affected to an ophthalmologist for further screening and treatment.

Contents

Board of Examiners **ii**

Candidate's Declaration **iii**

Acknowledgment **iv**

Abstract **v**

1 Introduction **1**

1.1 Motivation 4

1.2 Contribution 5

1.3 Thesis Organization 6

2 Background **7**

2.1 Fundus Image and Features of Diabetic Retinopathy 8

2.2 Classification of Diabetic Retinopathy 10

2.3 Literature Review 13

2.3.1 DR classification with feature extraction 13

2.3.2 DR classification without feature extraction 16

3 Methodology **20**

3.1 Data Analysis and Cleaning 22

3.1.1 Ungradable Image Detection 23

3.1.2 Training and Test Data Selection 25

3.2 Image Preprocessing . . . 26
3.2.1 Cropping . . . 27
3.2.2 Downsampling . . . 27
3.2.3 Sharpening . . . 29
3.2.4 Clipping . . . 30
3.2.5 Summary . . . 30
3.3 Image Augmentation . . . 32
3.4 Model Development . . . 35
3.4.1 VGGNet . . . 35
3.4.2 Proposed Network Architecture . . . 35
3.5 Model Training . . . 38
3.5.1 Custom Loss and Metric Function . . . 38
3.5.2 Hyper-parameter tuning . . . 39
3.5.3 Training our model . . . 39
3.6 Web Application Development . . . 41
3.7 Software and Hardware . . . 41

4 Results **44**
4.1 Training and Validation Performance . . . 44
4.2 Performance on Test Data . . . 50
4.3 A Closer Look at Performance . . . 52
4.4 Comparison with Retrospective Models . . . 54
4.4.1 Two-class Performance . . . 54
4.4.2 Generalization Capability and Comparison . . . 56

5 Conclusion **59**

List of Figures

2.1 Fundus image of a healthy retina 8
2.2 Features for DR Detection [1] 9
2.3 Four different fundus images, representing 4 different stages of NPDR and PDR with their respective features labeled. 12
2.4 Steps involved in DR classification with feature extraction 13
2.5 Limitation of segmentation method to detect optic disc 14

3.1 Flow Diagram of the Steps Involved 21
3.2 Variation among the images belonging to the same class. Here all images belong to class-0 or healthy eye. 24
3.3 Example of some ungradable images. 24
3.4 Example of some low-contrast but gradable images 25
3.5 Steps involved in preprocessing 28
3.6 A sample Gaussian kernel 30
3.7 Preprocessing helps identify lesions of DR 31
3.8 Effect of applying SUACE and CLAHE on Figure 3.7a 32
3.9 Skewed Distribution of Data 33
3.10 Sample augmented images of a class-4 image 34
3.11 VGG-16 Architecture 36
3.12 Web Application for DR Detection 42

4.1 Accuracy and Loss curve of the model for MAEC loss function 46
4.2 Accuracy and Loss curve of the model for CCE loss function 47

4.3 Accuracy and Loss curve of the model for MSE loss function 48
4.4 Accuracy and Loss curve of the model trained with 256×256 images 49
4.5 Very little anatomical difference between a class-0 and class-1 image 53
4.6 Class-0 and class-1 images misclassified as class-4 images 53
4.7 Examples of some severe misclassifications 55

List of Tables

2.1 Summary of the anatomical components present at different stages of DR 11

3.1 Summary of the severity level and distribution of data in Kaggle dataset 22
3.2 Training and test data selection 26
3.3 Summary of augmentation procedure 34
3.4 Proposed Network Architecture. 37
3.5 Hyper-parameters of our network 40

4.1 Confusion matrix for DR classification on 10000 test images 50
4.2 Performance metrics of the model on test data 51
4.3 Global performance metrics of the model on test data (multi-class) 51
4.4 Confusion matrix for affected (class 1,2,3,4) vs healthy (class 0) classification . . 56
4.5 Confusion matrix for vtDR (class 2,3,4) vs rDR (class 0, 1) classification 56
4.6 Performance metrics for two binary classification schemes 56
4.7 Comparison with retrospective studies 58

Chapter 1

Introduction

Diabetes is a chronic disease that affects the production of insulin in human body or impairs the body's ability to process insulin. With the course of time diabetes affects body's circulatory system, integumentary system, reproductive system, and central nervous system- leading to the damage of retina. Diabetes Retinopathy (DR) is a medical condition where the retina is damaged because of fluid leaks from blood vessels into the retina. The presence of DR is quite frequent among diabetes patients [2] and the percentage of diabetes patients worldwide is expected to increase from 2.8% in 2000 to 4.4% in 2030 [3]. DR is the most common microvascular complication and if left untreated, it may lead to visual impairment or blindness [4]. In 2010, out of the 126.6 million people who were diagnosed with DR, 0.8 million were blind and 3.7 million was found to be visually impaired due to this deleterious disease [5, 6]. To make things worse, the number of people affected by DR may reach up to 191 million by 2030. Furthermore, since there is no prominent visual or perceptible symptom of DR in the earlier stages, most patients remain unaware of this disease. As such, DR has become a leading cause of vision loss and a major public health problem.

Many studies have demonstrated that early detection and proper clinical treatment can reduce the risk of vision loss, mitigating the inimical impacts of the disease [7, 8]. As a result, early detection of Diabetic Retinopathy (DR) using retinal photography, known as retinal screening, is crucial because of its potential for reducing the number of cases of blindness. As mentioned earlier, DR occurs when diabetes damages the blood vessels inside the retina.

Due to the fluid leak into the retina, features such as microaneurysms, haemorrhages, hard exudates, cotton wool spots or venous loops, neovascularisation are formed [4]. DR is generally diagnosed by examining retinal images, known as fundus images, for the above-mentioned features or abnormalities (also known as lesions) by an experienced ophthalmologist. Ophthalmologists analyze these images for these lesions and based on the properties of these anatomical components present in the images, the patient is graded for DR. From a broad perspective, Diabetic Retinopathy can be classified as Nonproliferative Diabetic Retinopathy (NPDR) and Proliferative Diabetic Retinopathy (PDR). Among the other different grading conventions of DR present in the medical community, the severity scale proposed by Wilkinson et. al. [9] is the most popular and widely used. They proposed five stages of DR in their work:

i No Diabetic Retinopathy

ii Mild Nonproliferative Diabetic Retinopathy (NPDR)

iii Moderate Nonproliferative Diabetic Retinopathy (NPDR)

iv Severe Nonproliferative Diabetic Retinopathy (NPDR)

v Proliferative Diabetic Retinopathy (PDR)

Currently, detecting DR is a time-consuming and manual process which is prone to human error. An automatic retinal image analysis (ARIA) model can reduce the workload of clinicians and can provide a cost-effective and easily implementable method. These ARIA models can improve the efficiency of the grading process by a great deal, consequently increasing the throughput. The goal of the automatic screening algorithms is to refer the patients to an eye care provider when the image has a higher DR risk. In this way three purposes can be achieved: 1) Timely referral for treatment in the DR population, 2) Minimization of the vision loss risk, and 3) Effective use of the healthcare resources for the patients.

Traditionally, these automatic retinal image analysis models involve explicit feature extraction from the fundus images using image processing and machine learning techniques to identify the lesions. The extracted features from the images are then used for DR classification

using machine learning models and/or image processing techniques. However, the majority of these models focus on only feature engineering rather than classification [10]. In addition, most of the retrospective works identify only a subset of the features whereas DR classification needs the identification of all the features. Furthermore, this often introduces a generalization problem because of the difference in datasets. As a result, the performance of the models that attempted DR classification, in general, has been moderate. Although sensitivities are found to be high in most work, specificities remain moderate which is not sufficient to deploy in clinical environments. Moreover, most of the works focus on binary classification (DR vs no DR) whereas ophthalmologists prefer the grading scale proposed by Wilkinson et. al. [9]

Deep learning (DL) is a class of state-of-the-art machine learning techniques that have gained exceptional popularity in the last few years [11]. Deep learning models find intricate patterns between different types of data by deriving relevant necessary representations from the data without the requirement of manual feature engineering. DL models are made up of multiple layers with different functions and the algorithm adjusts the functional parameters based on the ground truth levels. Compared with conventional techniques, DL has achieved significantly higher performance in many domains, including natural language processing, computer vision [12, 13] and voice recognition [14]. Deep learning has also been successfully applied to many medical imaging analysis to detect various medical conditions [15, 16, 17]. Convolutional neural network (CNN, or ConvNet), a class of deep neural network, is considered state of the art for general image classification task because of their outstanding performance [18]. In mathematics, convolution is a mathematical operation on two functions to produce a third function that expresses how the shape of one is modified by the other [19]. CNN uses many layers with convolutions that use filters to extract complex features from an image in order to classify the image [11]. Recently CNNs have been used on fundus images to detect different ophthalmic diseases including Diabetic Retinopathy [20, 21, 22, 23], Glaucoma [24], Age-related Macular Degeneration (AMD) [25] and Retinopathy of Prematurity (ROP) [26]. The use of deep convolutional neural networks, coupled with telemedicine, may be a long-term solution to screen and monitor patients for primary eye care environments.

In this work, we have developed a novel deep learning based DR severity detection model.

We used convolutional neural network (CNN) for the detection task. The network is trained using our own loss function with a large heterogeneous dataset. To our best knowledge, no existing work has been able to identify the five stages of DR with same level of accuracy as ours. In addition, the model is also validated on two popular external datasets (Messidor [27] and E-Ophtha [28]) to demonstrate its robustness.

1.1 Motivation

Liew G. et. al. [29] showed that during the decade when eye screening programs were introduced in the United Kingdom, blindness due to Diabetic Retinopathy was reduced despite the fact that diabetes population increased during that period. Ophthalmologists also recommend that the risk of blindness due to DR can be significantly mitigated through screening programs. However, a vast portion of diabetes patients are not screened annually due to various factors [30]:

i **Inadequate diabetic eye screening programs**

In low and middle-income countries, there are very few eye screening programs compared to the number of possible patients due to long-term financial sustainability. Moreover, most of these programs are located in big cities or major urban areas. As a result, people living in rural areas get very little or no opportunities for eye screening.

ii **Inadequate resources**

Fundus images are generally captured using a specialized camera consisting of an intricate microscope. These cameras are too costly. As a result, in the low and middle-income countries, there is a scarcity of this technology. In addition, electronic patient records are almost non-existent in these countries.

iii **Inadequate number of specialists**

Currently, there are too few ophthalmologists compared to the number of patients in most countries. As a result, a specialist has to screen a huge number of images by himself. There are over 70 million people with diabetes in India whereas the number of

ophthalmologists is roughly around 12,000 [31]. Early screening and proper treatment of the patients is very difficult due to this extreme overburdened patient-care system.

iv **Lack of awareness among patients**

A vast majority of the potential patients are unaware of DR and its fatal consequences. As mentioned earlier, there is little or no visual symptom of DR at the earlier stages. Hence, people do not feel the necessity to visit the eye screening programs for regular checkup.

v **Economic status**

The prevalence of Diabetic Retinopathy is higher among the people of Africa, South Asia and Latin America [32]. The vast majority of this population is underprivileged and remain below the poverty line. As such, they are often unwilling to visit ophthalmologists on a regular basis.

Generally, DR is diagnosed by a careful investigation of the fundus images by an experienced ophthalmologist. The images are examined for the existence of the anatomical components like microaneurysms, hemorrhages, and exudates. In addition, optic disk and blood vessels are also examined for anomaly. The whole process is highly subjective and laborious. Furthermore, since a specialist has to examine a huge number of fundus images, the screening process is error-prone. Therefore, there is an urgent need to develop a cost-effective, efficient and easy-to-use automated retinal screening system that can improve the overall ophthalmic status of the diabetes patients.

1.2 Contribution

The major contributions of this work can be enumerated as:

i. Design and development of a deep learning based model that can identify the five stages of DR severity from fundus images. The model is trained using a large dataset where there is high variation among the images like any real-world dataset. Image preprocessing

and augmentation is performed to improve performance and amend the class imbalance problem.

ii. A novel custom loss function for training the above-mentioned model. The loss function is designed in such a way that can capture the essence of the ordered output (severity levels) as well as penalize the parameters accrodingly.

iii. A web application for identification of the severity of DR from fundus image. One can upload a fundus image and the application will show the stage of DR for that image using our pre-trained model.

iv. Validation of the model using two other public databases. By evaluating the performance of the model on external datasets, the model's generalization ability is demonstrated. In addition, comparison with some existing models is also performed through this validation.

1.3 Thesis Organization

This thesis is organized as follows. Chapter 1 gives a brief introduction to the problem. It also discusses the motivation to investigate the problem as well as the contributions of this work. Chapter 2 presents the necessary background of Diabetic Retinopathy (DR) and discusses the relevant researches to detect DR severity. Chapter 3 elaborately discusses all the steps involved to develop, train and test the model. The results obtained along with comparison with some existing studies are presented in details in Chapter 4. Finally, Chapter 5 provides some concluding remarks and possible future researches along this direction.

Chapter 2

Background

Diabetic Retinopathy (DR) affects mainly patients with diabetes type 1 and some with diabetes type 2. As mentioned in the previous chapter, this disease is mainly caused by damage to the small blood vessels that oxygenate the retina. Patients with DR develop blurred vision, spotty vision, night vision problems and in some cases, total blindness [33]. Early detection of sight-threatening DR allows laser therapy to be performed to prevent or delay visual loss.

This chapter provides the background for the different components involved in the research. The chapter starts with the details of fundus image and the anatomical components of DR in Section 2.1, followed by Section 2.2 which describes the popular classification scheme used for identifying DR severity. The section also discusses different symptoms for different stages of DR. Section 2.3 presents a brief study of the existing works in the literature. These works are categorized based on the use of feature engineering. Finally, a brief discussion on the effectiveness of using deep learning based approaches for DR classification is summarized at the end of Section 2.3.2.

2.1 Fundus Image and Features of Diabetic Retinopathy

Fundus imaging is the most commonly used imaging technique to capture retinal images. Typically, 'fundus' refers to the back of the pupil. A specialized fundus camera consisting of an intricate microscope attached to a flash enabled camera are generally used to capture a magnified and upright view of the fundus. Fundus image of a healthy retina (no DR) can be seen in Figure 2.1.

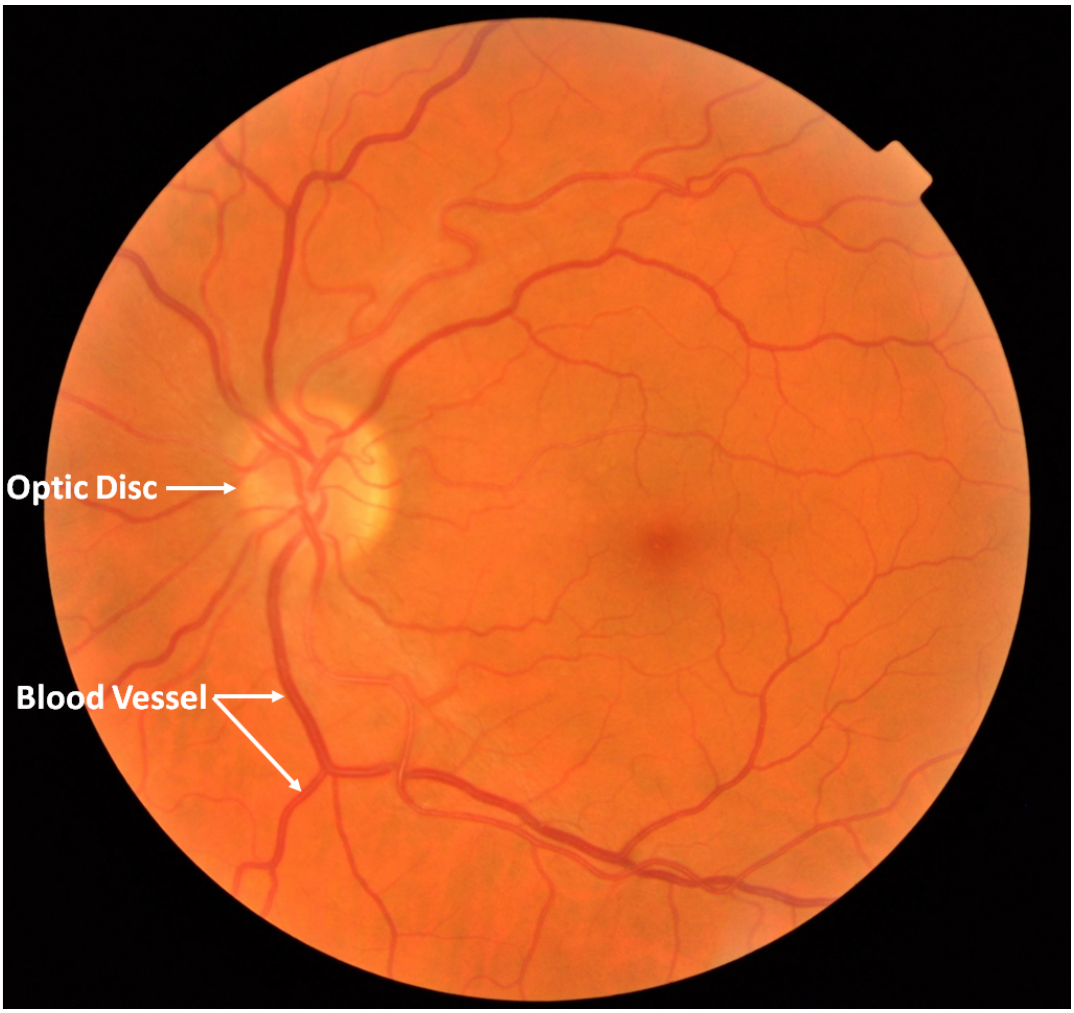


Figure 2.1: Fundus image of a healthy retina

Generally, fundus images are analyzed for some specific features, also called lesions, for the screening of DR. Making a proper assessment of the severity or stage of retinopathy requires the identification of the following lesions [34, 9] which can be seen in Figure 2.2:

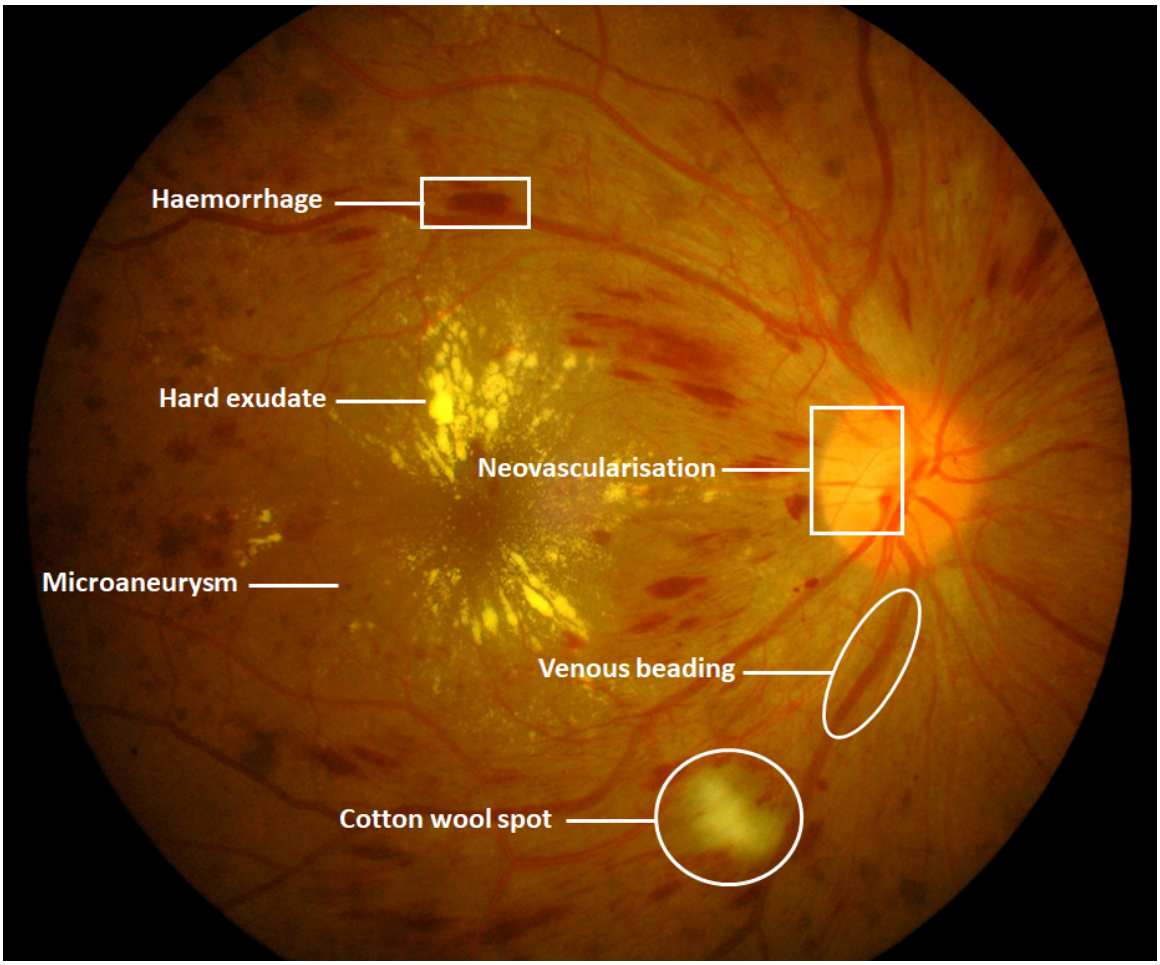


Figure 2.2: Features for DR Detection [1]

i **Microaneurysms** represent the earliest visible change of DR. They appear as small round, red dots, mainly in the posterior part of the eye as shown in Figure 2.2, and usually increase with the progression of DR [9].

ii **Haemorrhages** also appear at the earlier stage of DR following microaneurysms. However, their shapes can be more irregular than microaneurysms. They appear due to the leakage of blood in the inner nuclear layer [34].

iii **Cotton wool spots** are grayish or white patches of discoloration in the nerve fiber layer. The lack of oxygen in the retina, called *ischemia*, causes this damage in the nerve fiber layer of the retina [34].

iv **Hard exudates** are formed by leaked cellular lipids from abnormal intra-retinal capillaries. They usually have a bright yellow color with irregular boundaries and they vary from small spots to larger patches [9].

v **Venous beading** is a type of vascular abnormality which occurs in segments in the veins. The degree of venous beading can be a useful sign of proliferative diabetic retinopathy.

vi **Neovascularisation** refers to the process of abnormally growing new vessels. These new vessels grow when there is not enough oxygen provided to the retina. These newly formed vessels are fragile and bleed easily causing various complications [9].

2.2 Classification of Diabetic Retinopathy

The severity of DR symptom vary significantly between individuals according to the presence of multiple factors. As mentioned earlier, Diabetic Retinopathy (DR) can be broadly categorized as Nonproliferative Diabetic Retinopathy (NPDR) and Proliferative Diabetic Retinopathy (PDR). However, a vast majority of the existing works have combined these two classes into one as Referable Diabetic Retinopathy (rDR) [35, 20, 22]. As a result, they have classified DR into two binary categories based on just the presence of DR without considering the severity level. We will discuss more about these works in Section 2.3. On the other hand, ophthalmologists recommend DR to be classified for screening purposes according to the proposed international clinical classification system developed by Wilkinson et. al. [9]. This consists of five stages of disease severity starting from no DR to PDR.

i **No DR,** where the retina is healthy and the fundus image is normal. Features like microaneurysm, haemorrhage, exudate, cotton wool, etc. are not present in the image.

ii **Mild Nonproliferative Diabetic Retinopathy (NPDR),** where a few microaneurysms appear as small red spots on the superficial layers of the retina [36].

iii **Moderate Nonproliferative Diabetic Retinopathy (NPDR),** where more lesions appear as more capillaries become damaged, and the retina become more ischemic due to

Screening Classification	Anatomical Components
No DR	No abnormalities
Mild Nonproliferative DR	Microaneurysms
Moderate Nonproliferative DR	Microaneurysms, haemorrhages, exudates or cotton wool spots
Severe Nonproliferative DR	More prominent microaneurysms, haemorrhages, exudates or cotton wool spots. In general, the 4-2-1 rule is followed to classify.
Proliferative DR	Neovascular growth with the above symtomps.

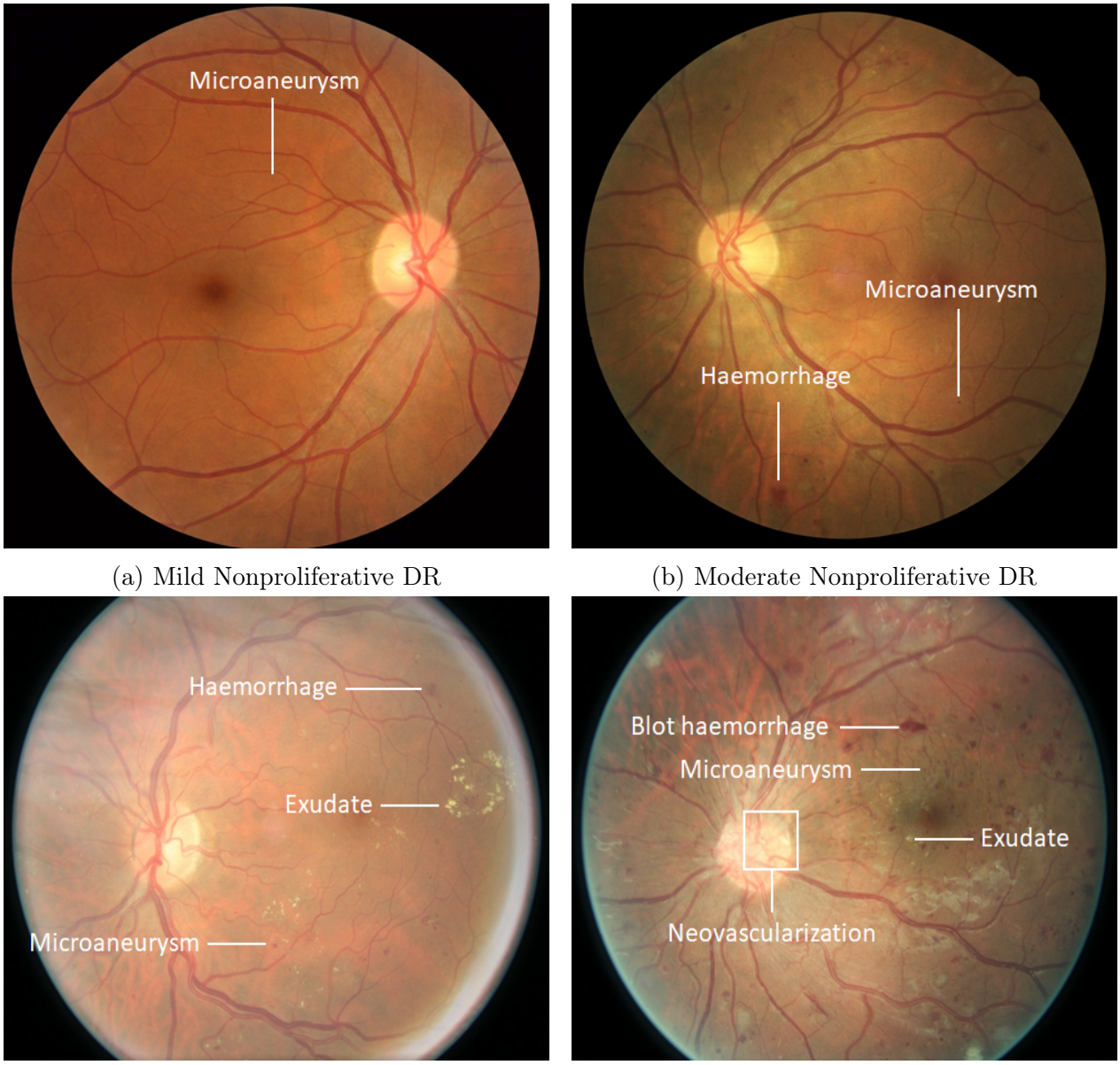
Table 2.1: Summary of the anatomical components present at different stages of DR

lack of blood flow, and therefore lack of oxygen. Haemorrhages, soft exudates and hard exudates start to appear in the fundus image [36].

iv **Severe Nonproliferative Diabetic Retinopathy (NPDR),** where more blood vessels are affected. Features like haemorrhages, soft exudates, and hard exudates become extremely frequent. However, new blood vessel growth is not yet found in this phase. According to the proposed international clinical classification of DR, a *4-2-1 rule* are indicative of Severe nonproliferative DR [9]. The *4-2-1 rule* is not described in this thesis because it falls out of the scope of this work. Severe nonproliferative DR can rapidly advance to PDR or remain static.

v **Proliferative Diabetic Retinopathy (PDR),** where new vessels (neovascularization) begin to grow along the inner surface of the retina in response to the need for oxygen. These new vessels are compromised and fragile, which causes severe bleeding and consequent vision loss.

A summary of this severity grading is presented in Table 2.1. An example of the four stages NPDR and PDR can be seen in Figure 2.3.



(a) Mild Nonproliferative DR

(b) Moderate Nonproliferative DR

(c) Sever Nonproliferative DR

(d) Proliferative DR

Figure 2.3: Four different fundus images, representing 4 different stages of NPDR and PDR with their respective features labeled.

2.3 Literature Review

The systems developed for automatic screening of Diabetic Retinopathy can be broadly categorized into two classes:

1. DR classification with feature extraction
2. DR classification without feature extraction

Our work uses the second approach for DR classification. A brief study of the existing works is presented in the following subsections.

2.3.1 DR classification with feature extraction

Since image processing can help the specialists to identify different features from fundus image, there has been an increase in the application of digital image processing techniques for automatic detection of DR [37]. This approach usually employs these image processing techniques to preprocess the image, followed by machine learning approaches to identify features and classify DR severity. In some cases, the machine learning step is also replaced by hard-coded image processing techniques. Some of these works have explicitly focused on feature extraction only. These features are generally the anatomical components of a retinal image like optic disc, blood vessels, microaneurysms, haemorrhages, exudates, etc. Figure 2.4 shows a general block diagram of the steps involved in these systems.

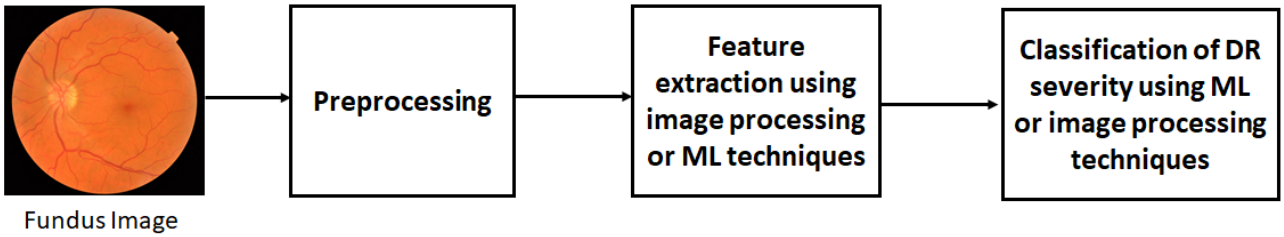
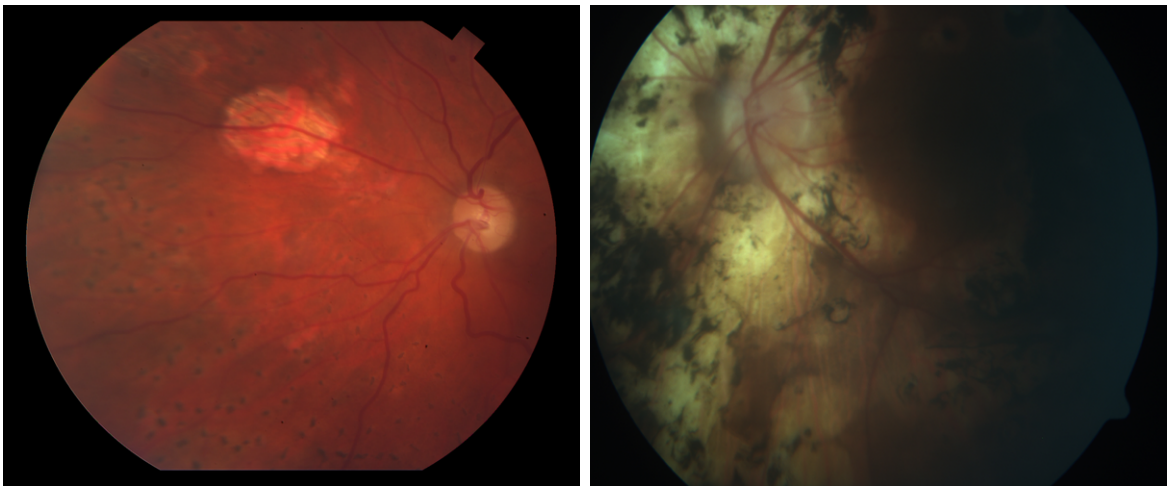


Figure 2.4: Steps involved in DR classification with feature extraction

Identification of optic disc and blood vessel is central for the detection of DR. Techniques like Adaptive thresholding, green channel extraction, morphological operations, contrast

enhancement (such as histogram equalization), active contour models, principal components analysis (PCA), and the watershed transform [38] are most commonly used in this regard. Aravind et al. [39] used green channel extraction, histogram equalization, contrast enhancement, and morphological operations as their preprocessing steps. Later, they used an SVM for the classification purpose which provided a 90% accuracy, 92% sensitivity, and 80% specificity. Sinthanayothin et. al. [40] localized the optic disc by identifying the area with the highest variation in the intensity of adjacent pixels. Although they achieved sensitivity and specificity of 99.1%, their work was reported to fail for practical datasets with a large number of white lesions and light artifacts [41]. Ravishankar et. al. [42] used the major vessels to identify the location of the optic disc with an accuracy of 97.1%. [43] used thresholding and morphological operations to identify optic disc based on the assumption that optic disc is the brightest part of a fundus image. However, their assumption is not always true for a practical dataset as shown in Figure 2.5. Alipour S. H. M et al. [44] used a curvelet-based algorithm in combination with contrast limited adaptive histogram equalization, illumination equalization, and morphological operations.



(a) Another region brighter than optic disc (b) Difficult to detect

Figure 2.5: Limitation of segmentation method to detect optic disc

A number of works [45, 46] have used two-dimensional matching filters for identifying blood vessels. [47] used two-dimensional matching filters along with region-based attributes by segmenting blood vessels. Welikela et. al. [48] used a genetic algorithm based approach for

detecting blood vessels. Hayashi et. al. [49] developed a system that can detect blood vessel intersections and it can identify abnormal widths in blood vessels. Orlando et. al. [50] used a vessel segmentation method based on fully connected conditional random fields. [51] identified blood vessels by detecting largely connected components in a binary image. Then they used an SVM classifier to identify the remaining thin vessels. Lupascu et. al. [52] constructed a 41-D feature vector and trained a feature-based AdaBoost classifier for vessel detection. They reported an accuracy of 95.97% for a test set of 20 images. Nagaveena et. al. [53] segmented the blood vessels using adaptive median thresholding. They reported average accuracy, specificity, and sensitivity of 91%, 96%, and 70% respectively for 40 images. Recently convolutional neural networks (CNN) have also been used for vessel detection [54, 55, 56].

An important step of exudate detection is the removal of prominent structures of the retina, such as optic disc and blood vessels. Wang et. al. [57] proposed a novel approach which combines brightness adjustment procedure with statistical classification method and local window-based verification strategy. Hunter et. al. [58] used neural network based exudates detection where they introduced a hierarchical feature selection algorithm. Their final architecture achieved a 91% accuracy using a relatively small number of images. Hsu et. al. [59] showed the importance of domain knowledge to differentiate exudates from other brighter lesions. Marker controlled watershed algorithm was applied on preprocessed images for the detection of exudates in [60]. Li et. al. [61] localized optic disc using principal component analysis (PCA). Then exudates were extracted by a combination of the techniques- region growing and edge detection. They reported sensitivity and specificity of 100% and 71% for exudate detection. Color retinal images were segmented using fuzzy c-means clustering in [62]. After extracting features from the segments, a multilayer neural network classifier was trained to identify exudates from these segments. They achieved a sensitivity of 93.5% on 300 images. Like blood vessel detection, convolutional neural networks have also been used recently for exudate detection [63, 64].

Microaneurysms detection is very crucial for DR detection because they are the earliest recognizable feature of DR. Since their texture and color is very similar to Haemorrhages, many existing works have employed similar techniques for their detection. Ege et.

al. [65] used a Bayesian, a Mahalanobis and k nearest neighbor classifier for identifying microaneurysms, haemorrhages and exudates. The Mahalanobis classifier attained the best performance where sensitivity was 69%, 83% and 99% for microaneurysms, haemorrhages and exudates respectively. Linear structuring element after local contrast normalization was used in [66]. They attained a sensitivity of 85.4% and specificity of 83.1% for microaneurysm detection. Sinthanayothin et. al. [67] developed a fully automated system to detect exudates, haemorrhages and microaneurysms using of a new technique, called 'Moat Operator'. They considered hemorrhages and microaneurysms (HMA) as one group, and hard exudates as another group. The sensitivity and specificity for exudates detection were 88.5% and 99.7% respectively and sensitivity and specificity for HMA detection was 77.5% and 88.7% respectively. Mizutani et. al. [68] selected candidates for microaneurysm using double ring filter as well as circular Hough transform. They used rule based classifier and artificial neural network on the candidates to detect microaneurysms. Larsen et. al. [69] used image processing for the detection of both haemorrhages and microaneurysms where they reported a specificity of 71.4% and a sensitivity of 96.7%. Some of the recent works have also used machine learning techniques for detecting hemorrhages and microaneurysms [70, 71]. Convolutional Neural Network (CNN) was employed to detect micoaneurysms and hemorrhages in [72]. They achieved area under the ROC curve of 0.894 and 0.972 on two different data sets.

2.3.2 DR classification without feature extraction

Due to the recent drastic development of deep learning (DL) techniques, image classification has attracted a lot of attraction in the last few years. Especially the unprecedented success of convolutional neural networks (CNNs or ConvNets) has revolutionized the field of image classification. CNNs have been recently used on fundus images for detecting Diabetic Retinopathy. Although DL based approaches generally require a large dataset to perform well, this approach has the inherent advantage of applying classification algorithms directly on the images without any feature engineering. Performance of some of these studies are extremely good compared to the performances of the studies reported in the previous subsection.

Abràmoff et. al. [23] developed a hybrid deep learning model (IDX-DR X2.1) and evaluated

it on the Messidor-2 dataset[1]. This system used a CNN inspired by AlexNet [73] models. The network was trained with lesions of DR and it provided three outputs for DR severity: no DR (or mild DR), referable DR or rDR (moderate nonproliferative DR or worse) and vision-threatening DR or vtDR (severe NPDR and PDR). They reported a sensitivity of 96.8% and specificity of 87% for rDR detection. For the vtDR output, the reported specificity was 90.8%.

Extensive feature engineering was performed using CNN in [21] for DR detection. The feature extraction by the CNN was optimized by selecting regions of interests (ROIs). Dimensionality reduction was applied on these features to select the more significant features. They trained their CNN on the 35,126 images from the Kaggle dataset[2] for five classes of DR severity. However, they converted their output to binary classification where accuracy of 97.28%, sensitivity of 100% and specificity of 99% was reported. Although these two works have used feature engineering, they are presented in this subsection because their classification model has used deep convolutional neural network.

Binary classification (DR vs no DR) was also implemented in [20]. They used the Inception-v3 model [74] and stochastic gradient descent algorithm was implemented for optimizing their system. The model was trained on 128,175 images and it was validated on the EyePACS[3] and Messidor-2 dataset. The system attained a sensitivity of 90.3% and a specificity of 98.1% for EyePACS and a sensitivity of 87% and a specificity of 98.5% for Messidor-2 and AUC scores of 0.991 and 0.990 respectively at the operating point selected for high specificity.

Quellec et. al. [75] proposed a method to detect referable DR as well as Lesions with CNNs. They used one of the solutions from the Kaggle Diabetic Retinopathy Competition[4] to detect referable DR. Their proposed model was mainly based on visualization methods of CNN. Heatmap generation modifications were performed to improve the quality of DR and lesion detection. They validated their model on Kaggle, DiaretDB1[5], and E-Ophtha[6] datasets and achieved AUC of 0.954, 0.955 and 0.949 respectively.

[1] http://latim.univ-brest.fr/indexfce0.html
[2] https://www.kaggle.com/c/diabetic-retinopathy-detection/data
[3] http://www.eyepacs.com/data-analysis
[4] https://www.kaggle.com/c/diabetic-retinopathy-detection
[5] http://www2.it.lut.fi/project/imageret/diaretdb1/
[6] http://www.adcis.net/en/third-party/e-ophtha/

Gargeya et. al. [35] developed a deep learning architecture for a binary DR classification. They employed a technique called deep feature learning using the principles of deep residual learning. From the CNN, they developed a feature vector with 1027 features. Later, a second level tree-based gradient boosting classifier was implemented on the popular EyePACS dataset consisting of 75,137 images. They reported a sensitivity of 94% and a specificity of 98% and an AUC of 0.97 on their local validation dataset. Messidor-2 dataset was also used to validate the robustness of the system where the model achieved a sensitivity of 96%, a specificity of 87% and an AUC of 0.940.

Colas et. al. [76] also proposed a system to grade 2 stages of DR (no DR and referable DR). Their algorithm was trained on over 70,000 images from the Kaggle dataset. Although the original dataset was labelled in 5 classes, they converted them into two classes. No DR and mild DR was grouped as non-referable and moderate, severe NPDR and proliferative DR was grouped as referable DR. They reported 94.6% area under the cure with a sensitivity of 96.2% and a specificity of 66.6%.

Ramachandran et. al. [77] used a third-party DR screening system incorporating a deep neural network to identify referable DR. They evaluated their model on 485 eye images from the Otago database. Their network achieved AUC of 0.901 with 84.6% sensitivity and 79.7% specificity for Otago dataset. On the other hand, AUC of 0.980 with a sensitivity of 96.0% and a specificity of 90.0% was attained for the Messidor dataset.

Takahashi et. al. [78] used GoogleNet [74] architecture to grade 4 stages of DR using 9,939 images. They used 496 images for their validation set and the rest was used for training. The CNN was trained using two different methods, one with manual staging of three photographs (AI1) and the other with manual staging of one photograph (AI2). The mean accuracy for AI1 and AI2 was 81% and 77% respectively and the final mean accuracy for 20 fold cross validation was 80%.

Ting et. al. [79] used two CNNs to detect rDR (mild NPDR and moderate NPDR) and vtDR (severe NPDR and PDR). The model was validated on the dataset of Singapore National Diabetes Retinopathy Screening Program (SIDRP) 2014-2015 which contained 71,896 images from 14,880 patients as well as ten external datasets consisting of 40,752 images. An AUC of

0.879 with sensitivity and specificity of 89.56% and 83.49% was achieved for detecting rDR. On the other hand, the model attained a sensitivity of 100%, a specificity of 81.4% and an AUC for 0.908 for grading vtDR.

Different architectures of CNN was used for DR detection in [80] where they achieved maximum accuracy of 83.68% on the EyePACS dataset. Raju et al. [81] evaluated the performance of a CNN for DR detection on the test dataset of Kaggle having 53,576 images. They presented the performance as a binary classification, reporting a sensitivity of 80.28% and a specificity of 92.29%. Lam et. al. [82] explored multinomial classification models and their model achieved peak test set accuracies of only 57.2% for a five class DR severity detection. Pratt et. al. [83] designed a CNN to predict the exact DR stage for a five-class DR detection task. The network was trained on the publicly available Kaggle dataset. Their proposed technique achieved an accuracy of 75% and sensitivity of 95%. Rakhlin et. al. [84] proposed a deep CNN based model where they achieved 99% sensitivity and 71% specificity on Messidor-2 dataset for binary classification. They reported average sensitivity of 86% and specificity of 82% on their Kaggle validation dataset.

Most of the works mentioned in this subsection outperform the works mentioned in the previous subsection for the following reasons:

i Most of the approaches mentioned in the previous subsection have focused on identifying features rather than detecting DR severity. If the goal is to develop an automatic screening system, then the system must output the level of DR severity which is the case for these deep learning techniques.

ii Some of the approaches mentioned in the previous subsection have classified DR based on a subset of the features. However, the successful detection DR must need to consider all of the features. Since deep learning techniques are employed on the whole image, they can learn about the features implicitly, thereby can achieve a much better performance.

iii Feature engineering is an extremely cumbersome and laborious process. Deep learning techniques circumvent this step and provide more robust performance.

Chapter 3

Methodology

In this work, we have implemented a new system using deep convolutional neural network to detect the severity of DR. Our model outputs DR severity in one of five classes as proposed by Wilkinson et. al. [9] which is the preferred classification across ophthalmologists. Our CNN uses deep architecture to identify this grading of DR.

As a first step, the input images were analyzed for very poor quality images which were discarded. A test set was then separated. The rest of the images were then preprocessed to eliminate border, center image, and highlight boundaries as well as edges. Later, the images were augmented heavily to rectify the class imbalance problem as well as to encode multiple invariances in our model. Then, these images were used to train our model. Our model architecture was inspired by the popular VGGNet [85, 86] which is known to perform very well for image classification. While training the model, we used our custom loss function which is central to the performance of our model. Different hyper-parameters were used to find the optimal one. We validated the performance of our proposed model on the test set and two external datasets: E-Ophtha [28] and Messidor [27]. We also developed a simple web application that can demonstrate our system. So, the main steps of developing our system include-

i Data Analysis and Cleaning

ii Image Preprocessing

iii Image Augmentation

iv Model Development

v Model Training

vi Web Application Development

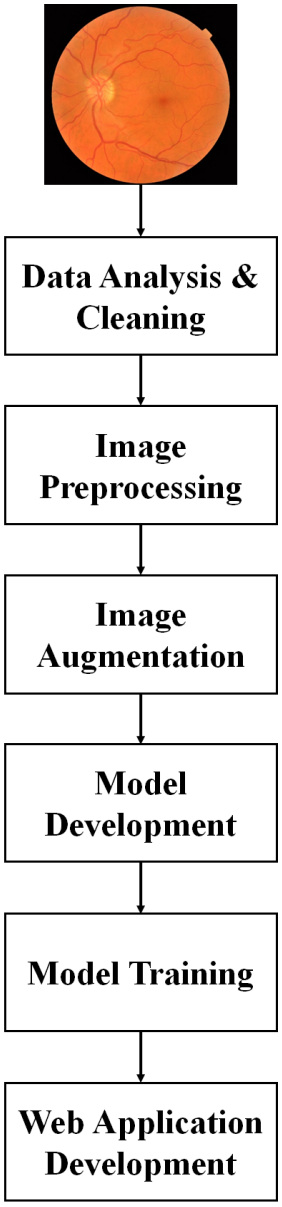


Figure 3.1: Flow Diagram of the Steps Involved

A flow diagram of the steps involved in our work for DR severity detection from fundus images is shown in Figure 3.1. In this section, we will be discussing each of these steps in details.

3.1 Data Analysis and Cleaning

In 2015, the California Healthcare Foundation sponsored a competition [87] to detect diabetic retinopathy from an existing dataset. The dataset was provided by EyePACS, a free platform for retinopathy screening. We have used this dataset extensively for both training and testing our model. This dataset is also popularly known as Kaggle dataset because Kaggle[1] hosted the competition. This dataset is a collection of high-resolution retinal images or fundus images which were captured under a variety of imaging conditions. There were a total of 88,702 images in this dataset from 44,351 patients, one image for each eye. So, images were labeled with a subject id and either left or right. For example, "1_left.jpeg" is the fundus image of the left eye of patient id 1. Each image was graded from 0 to 4 based on the severity level of DR, where 0 corresponds to the healthy state and 4 corresponds to the most severe state of DR. The meaning of this severity level along with the distribution of images among these classes are presented in Table 3.1.

Image Label	**DR Severity**	**Count**
0	No Diabetic Retinopathy (Healthy Eye)	65343
1	Mild Nonproliferative Diabetic Retinopathy	6205
2	Moderate Nonproliferative Diabetic Retinopathy	13153
3	Severe Nonproliferative Diabetic Retinopathy	2087
4	Proliferative Diabetic Retinopathy	1914

Table 3.1: Summary of the severity level and distribution of data in Kaggle dataset

The original dataset is divided into two parts where the training set comprises of 35,126 images and the test set comprises of 53,576 images. We combined the whole dataset into 88,702

[1] `https://www.kaggle.com/`

images and then selected our training and test dataset later. Total size of this dataset was roughly 89 GB. One of the major challenges of processing this dataset was the amount of noise or "bad data" present in the dataset. Generally, Kaggle datasets are very standard, error free and easy to process. However, this dataset was very challenging because some of the images were too dark or too bright or too low-contrast. Another crucial challenge was the imbalance present in the dataset. From Table 3.1, we can see that almost 73.7% of all the images are of healthy eyes where there is no diabetic retinopathy. We shall discuss about these challenges in details in the following subsections.

3.1.1 Ungradable Image Detection

The images represented a heterogeneous group of patients with different DR severity levels. The fundus images were captured with a variety of different camera models from patients of different ethnicities. As a result, there was a huge variation among the images in size, resolution, aspect ratio, color contrast, and orientation. Subtle signs of retinopathy, like microaneurysms, can be easily masked on low contrast or blurred, or low-resolution image. It is important for images to be of good quality in order to provide a reliable diagnosis. In a typical screening environment, studies have found that 10%-20% of the images suffered from inaccurate diagnosis [88]. Generally, poor patient's fixation, poor focus and camera artifacts are the main reasons behind the ungradable images. The variation among the images can be visualized in Figure 3.2. All of these images are from the Kaggle dataset and they have the same severity level. Despite all the images belong to the same class (no DR), the variation among the images are conspicuous.

Although the images of Figure 3.2 differ greatly from one another, they are still gradable. In fact, a vast number of the images of the dataset are like this. A clinician needs to identify the subtle features like microaneurysms, exudates, and haemorrhages to identify the presence and severity of DR. The features can still be seen in these images. As a result, image processing techniques to highlight boundaries and edges will be able to identify some of these lesions from the images. However, a few images of this dataset are too bad to diagnose properly as shown in Figure 3.3. As we can see, these images are in extremely bad condition, thereby ungradable.

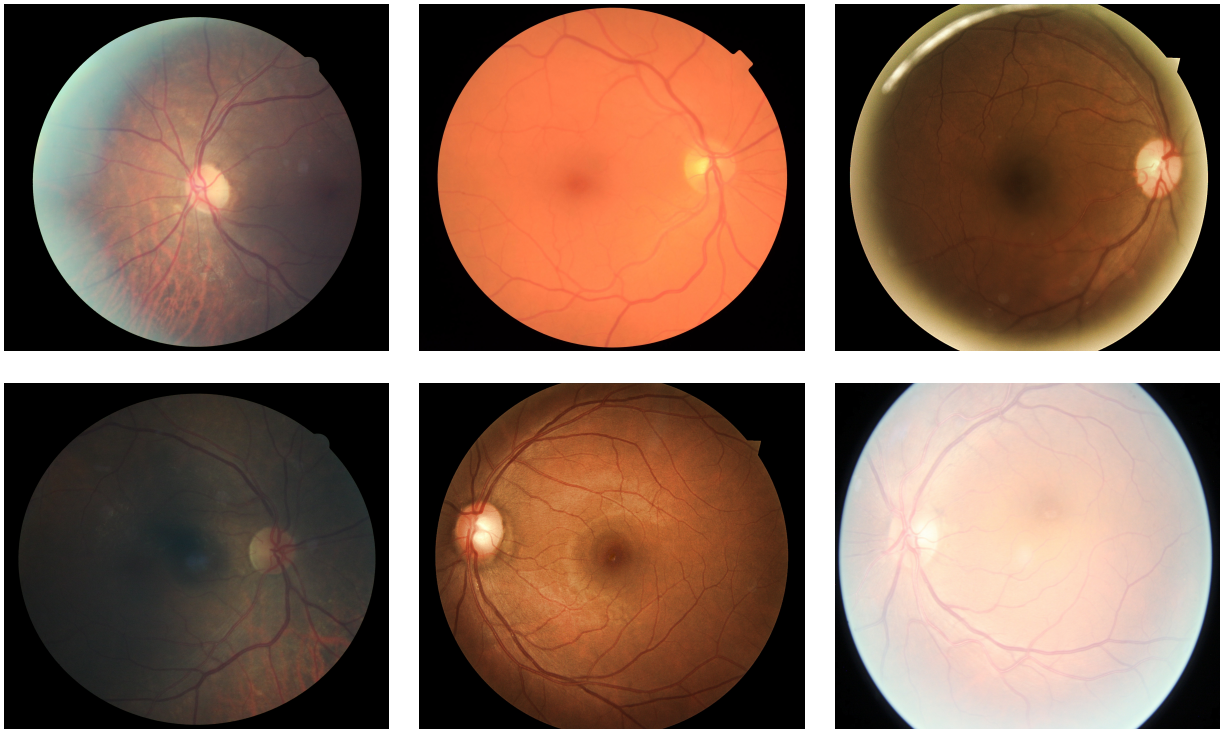


Figure 3.2: Variation among the images belonging to the same class. Here all images belong to class-0 or healthy eye.

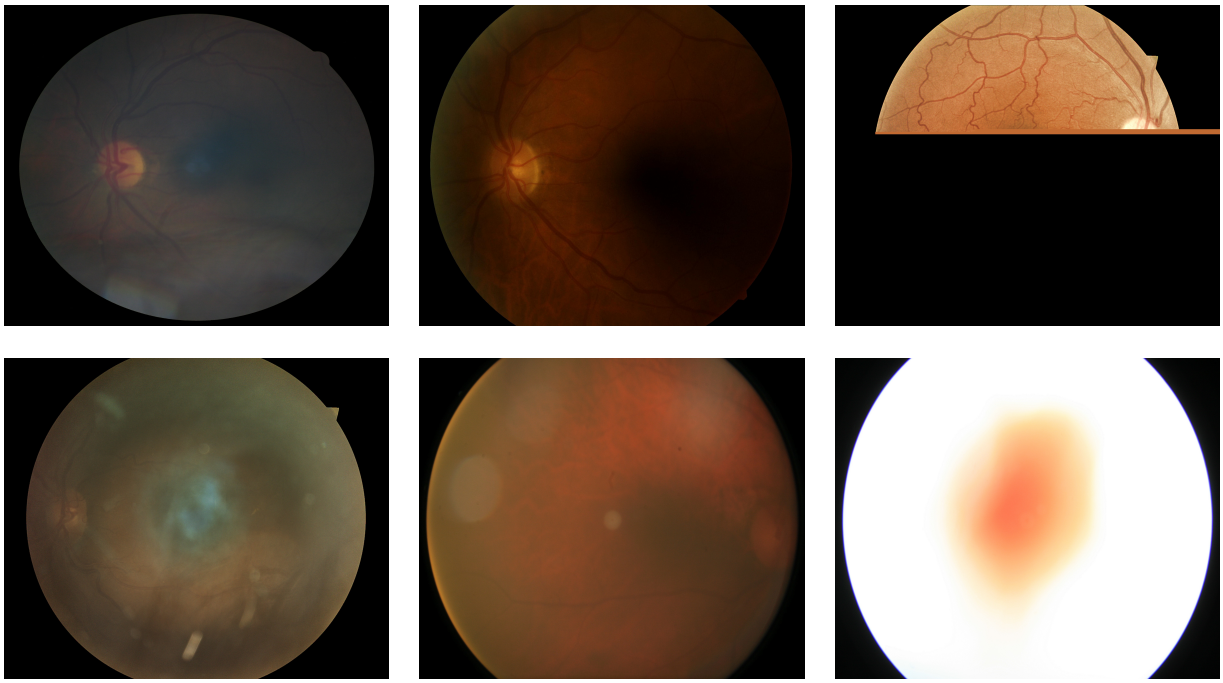


Figure 3.3: Example of some ungradable images.

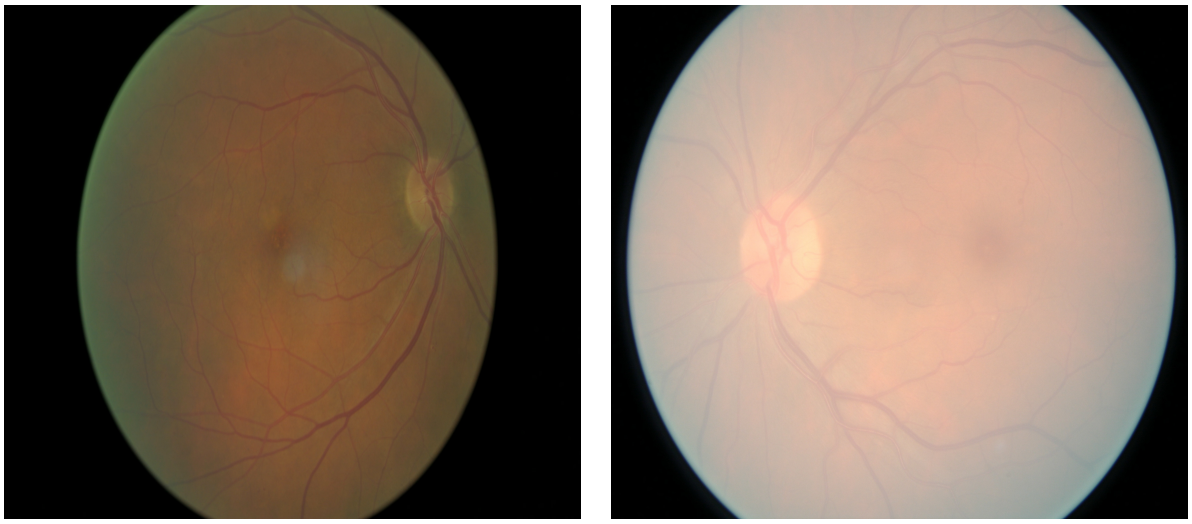


Figure 3.4: Example of some low-contrast but gradable images

Any machine learning model will suffer greatly because of the presence of this type of "bad data". When the model tries to learn its weights' parameter based on these low-quality images, the model's generalization capability can deteriorate for the high-quality images. In the end, analysis of an image of such low quality may produce unreliable results when the system labels the image as normal while lesions of DR are still present. So, we needed to discard these ungradable images.

As we can see from Figure 3.3, most of the ungradable images have very low contrast and standard deviation. Consequently, we wrote a program to identify these images based on low contrast and low standard deviation where we found 2,662 such images. It is worth mentioning that, our goal is to develop a robust system that can identify DR severity even from "reasonably bad" images. As a result, we made sure that no gradable image gets discarded. So, we examined each of these images manually and found that 38 of them are gradable despite having poor contrast. We can see a couple of such images in Figure 3.4. Finally, the number of discarded images were 2,624.

3.1.2 Training and Test Data Selection

After discarding 2,624 ungradable images, there were 86,078 images left, out of which 63,528 images (73.8%) belonged to class-0 or healthy images. We then randomly selected 5,500 images

Image Label	Initial Count	Ungradable Image Count	Gradable Image Count	Test Data Count	Training Data Count	Remaining Image Count
0	65343	1815	63528	5550	34950	23028
1	6205	91	6114	1500	4614	0
2	13153	462	12691	2000	10691	0
3	2087	91	1996	500	1496	0
4	1914	165	1749	500	1249	0
Total	**88702**	**2624**	**86078**	**10000**	**53000**	**23028**

Table 3.2: Training and test data selection

from class-0, 1,500 images from class-1, 2,000 images from class-2, 500 images from class-3 and 500 images from class-4. In this way, a test dataset of 10,000 images was prepared which was kept completely separate from the rest of the images. While preparing the training dataset from the remaining images, we selected 34,950 images randomly from class-0 and the remaining 18,050 images from the other classes to rectify the huge class imbalance to some extent. As a result, a training dataset comprising of 53,000 images were prepared. The remaining 23,028 images of class-0 was discarded. The summary of this selection and the distribution of images across different classes are shown in Table 3.2.

3.2 Image Preprocessing

The preprocessing of this heterogeneous dataset is a crucial step for the overall performance of the system because of two primary reasons. (i) Although the dataset contained images with different resolutions, colors and aspect ratios, ConvNets require a fixed input size for all the images. Hence, converting all the images to a fixed size is absolutely mandatory. (ii) Proper preprocessing of the images can aid the ConvNet to learn features and converge quickly. For example, if the lesions and other features were highlighted in the images, the model can comprehend the implicit features and tune its parameters accordingly. Our preprocessing included mainly four steps:

i Cropping

ii Downsampling

iii Sharpening

iv Clipping

Figure 3.5 demonstrates the effect of each of these steps on a healthy fundus image. We used OpenCV [89] and ImageMagick [90] for the above-mentioned steps. Details of these steps are elaborately discussed in this section.

3.2.1 Cropping

All the images of the dataset had a black background where the retina is roughly situated in the middle. The images had black extensions in all four sides (mostly in left and right) as shown in Figure 3.5a. Since our key elements lie in the retina, we cropped the images in all directions to remove the invalid black space so that only the retina is inscribed in a rectangle as shown in Figure 3.5b.

3.2.2 Downsampling

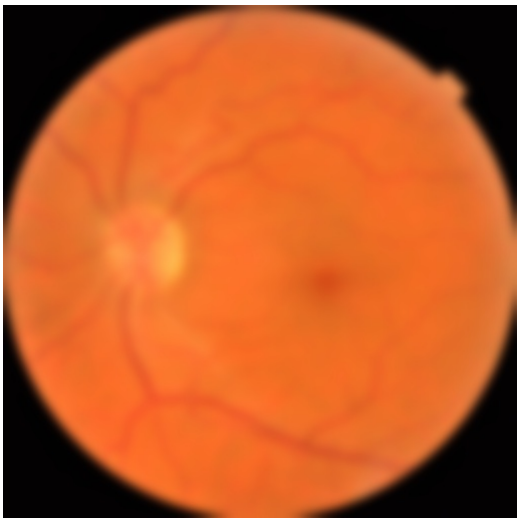
Image resolution of the dataset varied in a wide range from 5184×3486 to 1792×1184. As a result, we standardized the resolution by downsampling all the images to a fixed size in accordance with the input requirements of our model. The size of the input image for CNN is of pivotal concern for the performance of the model. Small features like microaneurysms are the earliest signs of DR. These subtle details can easily get lost when the image is shrunk too much. On the other hand, large input size requires large GPU computational time. To further illustrate the effects of image size, we downsampled all the training dataset images to two different dimensions: 512×512 and 256×256 and trained two models.



(a) Initial image



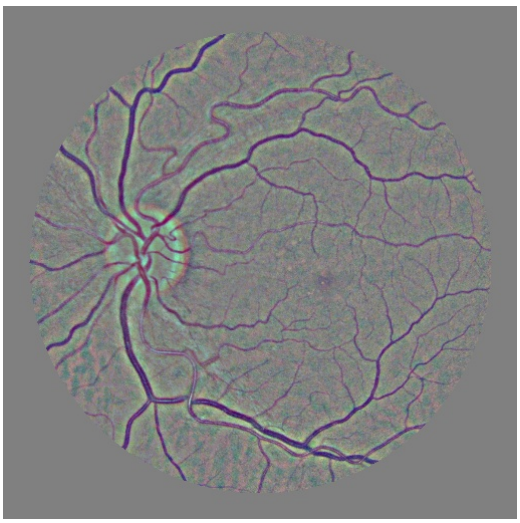
(b) Extra black region cropped



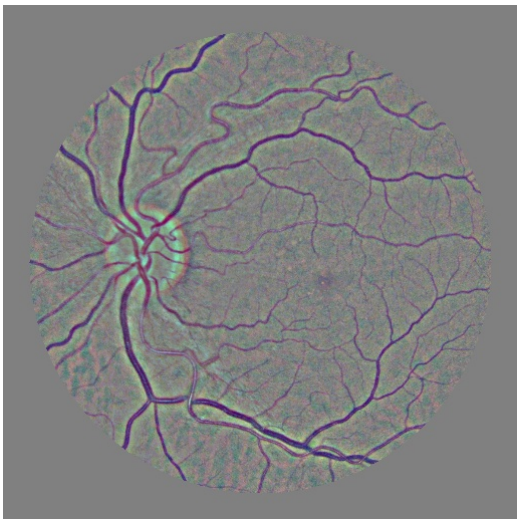
(c) Gaussian-blurred image of 3.5b



(d) 3.5c subtracted from 3.5b



(e) Boundary-effect removed from 3.5d



(f) Retina centered

Figure 3.5: Steps involved in preprocessing

3.2.3 Sharpening

Image sharpening is a general technique to increase the sharpness of an image so that small details can easily be detected. However, our goal was not to sharpen the raw fundus image. Instead, we tried to highlight only the edges and sharp changes. We used a popular technique named "Unsharp masking" to achieve our goal. This technique uses a blurred image to create a sharp image by subtracting the blurred image from the original image. Let us consider $f(x, y)$ is our original image and $\overline{f}(x, y)$ is a blurred or smoothed image of $f(x, y)$. Then, we can obtain a mask using the following equation:

$$g_{mask}(x, y) = f(x, y) - \overline{f}(x, y)$$

$g_{mask}(x, y)$ image contains the discontinuities and sharp changes across the image. When the goal is to sharpen the original images, this mask is usually added to the original image. However, we skip this step because we only need the $g_{mask}(x, y)$ image for our CNN. We used the widely popular "Gaussian Blur" filter to obtain the blurred image $\overline{f}(x, y)$. Mathematically, applying a Gaussian blur to an image is the same as convolving the image with a Gaussian function. The Gaussian function in two dimensions can be represented as:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

A Gaussian kernel is a rectangular array of pixels where the pixel values correspond to the values of Gaussian curve. Example of such a kernel is shown in Figure 3.6. When an image is convoluted with a Gaussian kernel, the resultant image is a blurred one. We used OpenCv's built-in *GaussianBlur()* function with a kernel size of 5×5. Then, we subtracted the blurred image from the original image to get our g_{mask} image which was further preprocessed as described below. A sample blurred image and the resultant sharp image can be seen in Figure 3.5c and Figure 3.5d respectively. As we can see, blood vessels and other sharp changes are quite prominently visible in the sharp image.

$$\frac{1}{273}\begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

Figure 3.6: A sample Gaussian kernel

3.2.4 Clipping

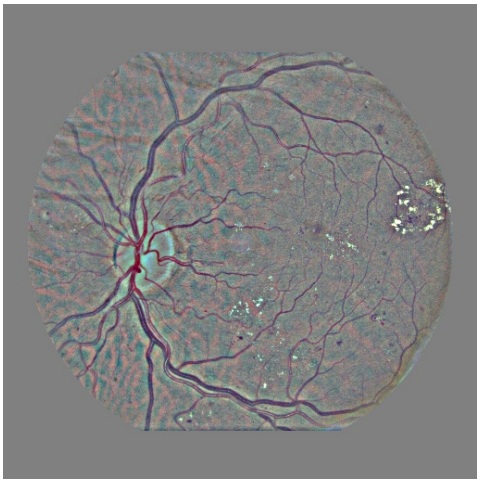
In Figure 3.5d, we can see that the boundary is quite oddly highlighted. Moreover, in some other images, this boundary depth varies significantly due to camera artifacts. If these images were used as input for our CNN, it is likely that the CNN will try to learn these variations which may lead to poor performance. So we clipped these sharp images to 90% of the size to remove this "boundary effect". Then, the retina was placed in the center of the image. Figure 3.5e and 3.5f demonstrates the effect of clipping and centering.

3.2.5 Summary

To summarize, our 53,000 training dataset images were first cropped and downsampled to two different dimensions: 512×512 and 256×256. Then, the images were convoluted with a Gaussian blurring kernel to obtain a blurred image. Later, these blurred images were subtracted from the original image and a sharp image was found where the discontinuities were highlighted. Finally, these images were clipped to remove the boundary of retina and then the retina was centered. Figure 3.5 shows the effects of applying these preprocessing steps on a healthy fundus image. However, it is essential to know how this preprocessing aids to identify the lesions of DR. Figure 3.7 shows the fundus images of two DR affected retina as well as their preprocessed versions. As we can see, lesions like microaneurysms, exudates, and haemorrhages are clearly visible in the preprocessed image. Any sharp change in the image gets highlighted through this preprocessing step.



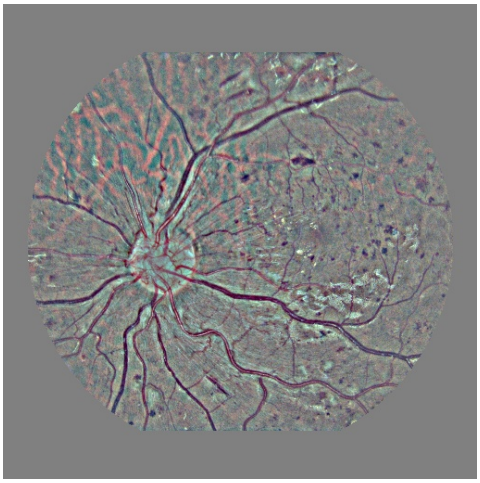
(a) A sample class-3 fundus image



(b) Preprocessed image of 3.7a

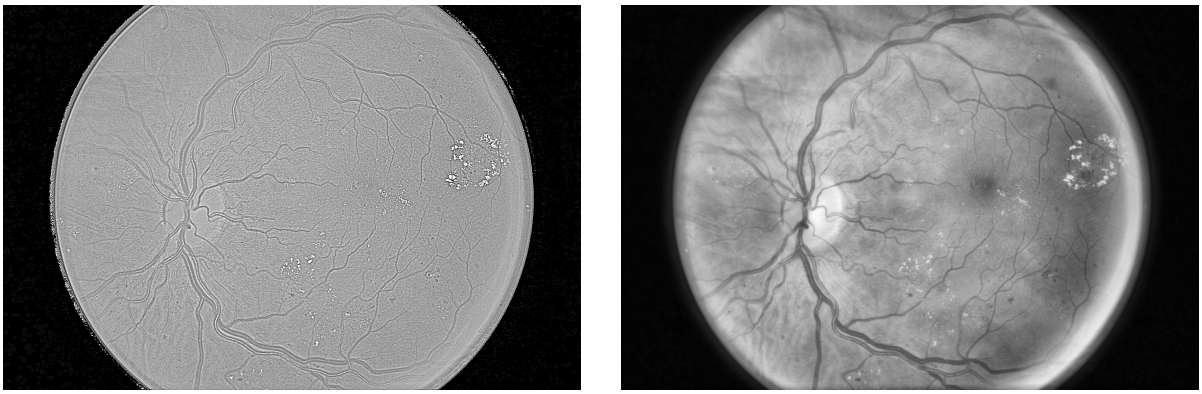


(c) A sample class-4 fundus image



(d) Preprocessed image of 3.7c

Figure 3.7: Preprocessing helps identify lesions of DR



(a) Effect of SUACE (b) Effect of CLAHE

Figure 3.8: Effect of applying SUACE and CLAHE on Figure 3.7a

We also tried a few other preprocessing techniques like *Speeded Up Adaptive Contrast Enhancement (SUACE)* [91] and *Contrast Limited Adaptive Histogram Equalization (CLAHE)* [92] in place for unsharp masking for highlighting discontinuities. Effect of applying SUACE and CLAHE on Figure 3.7a is shown in Figure 3.8a and 3.8b. In case of SUACE, small abnormalities may get lost whereas CLAHE highlights less important features. The final output of unsharp masking seems much more conducive to CNN training for DR detection.

3.3 Image Augmentation

In our training dataset of 53,000 images, 34,950 images (65.94%) belong to the class of healthy fundus images whereas only 1249 images (2.36%) belong to the class of Proliferative Diabetic Retinopathy (PDR), as shown in Table 3.2 . This imbalance ratio is also presented in Figure 3.9. Because of this highly skewed distribution of data, augmentation is a must. Image augmentation is a method of applying different image transformations across a data set to increase image heterogeneity as well as decrease class imbalance while preserving the prognostic characteristics in the image itself. Detection of DR from fundus image is rotationally invariant. This means that the identification of DR depends on the presence of different anatomical components, regardless of the orientation. Although we performed image augmentation mainly to rectify the imbalance problem, it improved the model's ability to generalize and correctly classify fundus images of various orientations. We used OpenCV to perform these transformations.

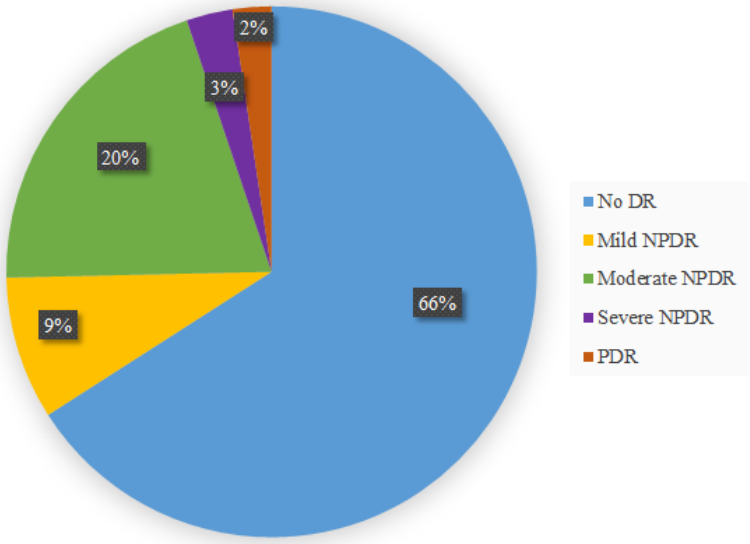


Figure 3.9: Skewed Distribution of Data

Our training dataset was augmented in a specific way so that the counts of images from different classes are similar. Since class 0 already had a very high number of images, no augmentation was applied to these images. Initially, there were 4,614 images belonging to class-1. We applied rotations of 45°, 90°, 135°, 180°, 225°and 270°on these images and generated a total of 27,684 images belonging to class-1. For class-2, we applied rotations of 90°and 180°to generate 21,382 images for this class. 10 rotations of 30°intervals were applied on the class-3 images and its horizontally flipped versions to generate 29,920 images. On the other hand, 11 rotations of 30°intervals were applied on the class-4 images and its horizontally flipped versions to generate 27,478 images. Finally, we had a total of 159,464 images (size 15.3 GB) as our training dataset where images were evenly distributed among five classes. A summary of this augmentation procedure along with the counts of images from different classes can be seen in Table 3.3. Figure 3.10 shows some example of image augmentation for a class 4 image. It is worth mentioning that image augmentation was again performed before training the model. We shall discuss that in section 3.5.

Image Label	Training Data	Augmentation Procedure	Generated Image	Total Image
0	34950	N/A	0	34950
1	4614	6 rotations (45°interval)	27684	32298
2	10691	2 rotations (90°interval)	21382	32073
3	1496	Horizontal flipping 10 rotations on both original image and flipped image (30°interval)	29920	31416
4	1249	Horizontal flipping 11 rotations on both original image and flipped image (30°interval)	27478	28727
Total	**53000**	**88702**	**106464**	**159464**

Table 3.3: Summary of augmentation procedure

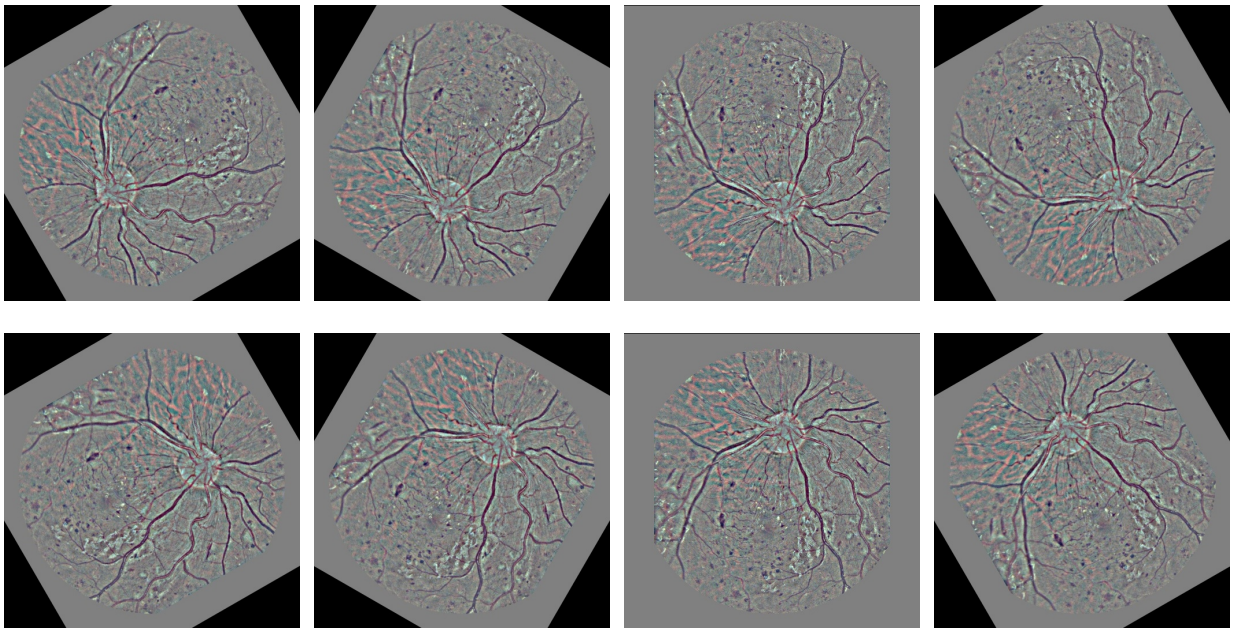


Figure 3.10: Sample augmented images of a class-4 image

3.4 Model Development

We used deep convolutional neural network (ConvNet or CNN) because of its extraordinary performance in image recognition, image classification, and computer vision. While training a ConvNet, architecture is a major concern in terms of both training time and model's performance. Extensive studies have been performed on ConvNet architectures and some of the most popular architectures are VGGNet [86], AlexNet [73], GoogleNet [74], Inception-Resnet [93] etc. Wan et. al. [94] performed a comparative analysis of these popular networks' performance on fundus images for DR classification and showed that VGGNet performs better for DR detection. As a result, our network was inspired by the architecture of VGGNet. We used Keras[2], an open-source neural-network library written in Python, on top TensorFlow[3] for our model development. In this section, we shall briefly discuss about the background of VGGNet and our proposed network architecture.

3.4.1 VGGNet

VGGNet was designed by Simonyan et. al. [86] which achieved higher accuracy and generalization using increased network depth and smaller filters. The network is an improvement of AlexNet. VGGNet has 5 convolutional layers, each followed by a max-pooling layer. Each of the convolution layers comprises of multiple convolution operations. There are 3 fully-connected layers after the convolution layers and a softmax layer for classification. Figure 3.11 shows the VGGNet architecture. It was trained on ImageNet [95] dataset.

3.4.2 Proposed Network Architecture

We developed two networks for the two different datasets differing by image sizes: 512×512 and 256×256. The network that was developed for 512×512 dataset outperformed the other network by a big margin. Hence, the better network's architecture is presented in details in this subsection. The other network differs to this by only the shapes of the layers.

[2] `https://keras.io/`
[3] `https://www.tensorflow.org/`

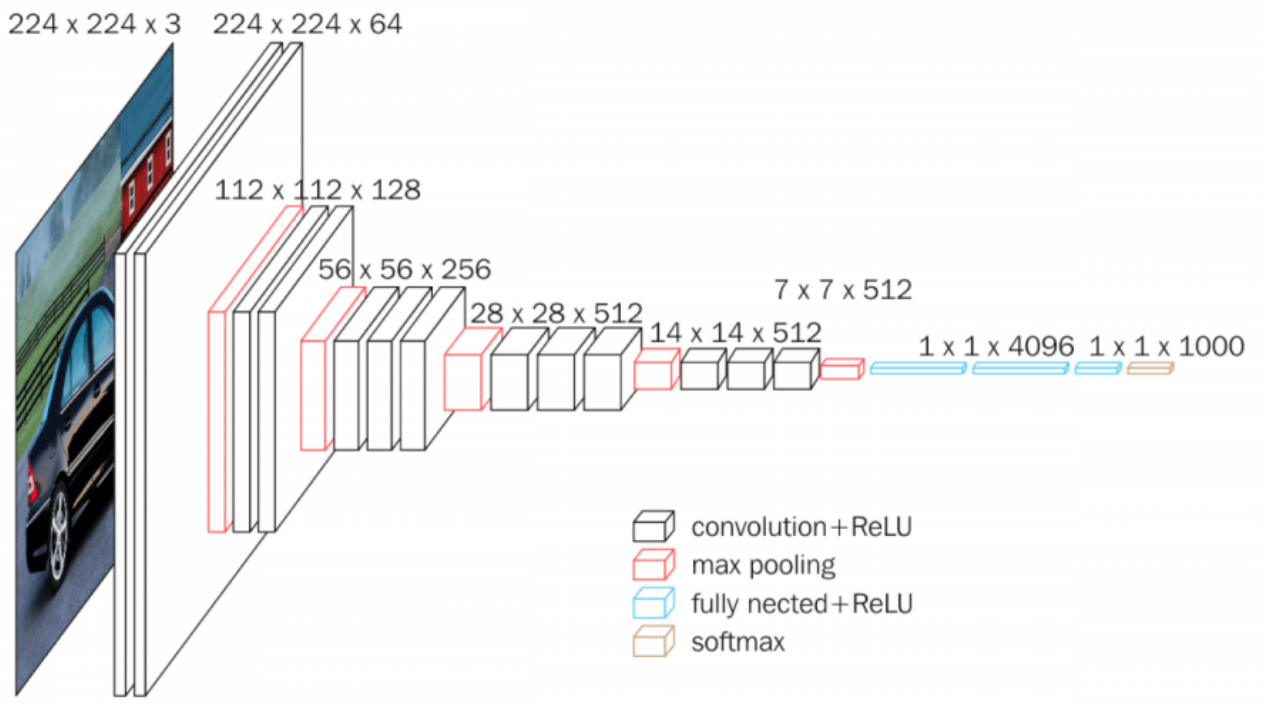


Figure 3.11: VGG-16 Architecture

The input layer of the network is 512 × 512 because the size of our input. We used seven convolution layers each consisting of two convolution operations except the last layer which consists of one convolution operation. Each convolution layer is followed by a max pooling layer. Like VGGNet, we used two fully connected layers and a softmax layer for DR classification. We tried kernel filters of sizes 3×3 and 4×4 and found the best result in kernel of size 3×3. As a result, all the convolution layers of our network have the same kernel size of 3×3 and the stride of 2. Kernel size of 2×2 is used in all the max-pooling layers. As for activation function, ReLU was used in all convolution layers for nonlinearity. The final extracted features were flattened before passing through the fully connected layers. There are two fully connected layers, having 256 and 128 neurons followed by a softmax layer of 5 neurons for classification. Dropout of 0.5 was added after all but the last fully connected layers to reduce overfitting. Table 3.4 illustrates the network architecture of our proposed DR classification system which contains a total of 2,737,765 trainable parameters.

Layer Type	**Filter Size & Number**	**Output Shape**	**Parameters**
input	N/A	(512, 512, 3)	N/A
convolution	$3 \times 3 \times 32$	(510, 510, 32)	896
convolution	$3 \times 3 \times 32$	(508, 508, 32)	9248
max-pooling	2×2	(254, 254, 32)	0
convolution	$3 \times 3 \times 64$	(252, 252, 64)	18496
convolution	$3 \times 3 \times 64$	(250, 250, 64)	36928
max-pooling	2×2	(125, 125, 64)	0
convolution	$3 \times 3 \times 96$	(123, 123, 96)	55392
convolution	$3 \times 3 \times 96$	(121, 121, 96)	83040
max-pooling	2×2	(60, 60, 96)	0
convolution	$3 \times 3 \times 128$	(58, 58, 128)	110720
convolution	$3 \times 3 \times 128$	(56, 56, 128)	147584
max-pooling	2×2	(28, 28, 128)	0
convolution	$3 \times 3 \times 192$	(26, 26, 192)	221376
convolution	$3 \times 3 \times 192$	(24, 24, 192)	331968
max-pooling	2×2	(12, 12, 192)	0
convolution	$3 \times 3 \times 256$	(10, 10, 256)	442624
convolution	$3 \times 3 \times 256$	(8, 8, 256)	590080
max-pooling	2×2	(4, 4, 256)	0
convolution	$3 \times 3 \times 256$	(2, 2, 256)	590080
max-pooling	2×2	(1, 1, 256)	0
fully connected	256	(256)	65792
fully connected	128	(128)	32896
softmax	5	(5)	645

Table 3.4: Proposed Network Architecture.

3.5 Model Training

Our proposed network in the previous section was 23 layer deep consisting of more than 2.7 million parameters which were randomly initialized. The training was performed on the training dataset of 159,464 images. The model parameters were optimized by our custom loss function. A custom metric function was also written to monitor the performance of the network. Related hyper-parameters were tuned for optimal learning and training was performed with cross-validation over 10% of the training data. Performance of the model was measured by evaluating the model on the test images. In this section, we shall briefly discuss about the steps involved in training the network.

3.5.1 Custom Loss and Metric Function

Although Keras provides various built-in loss functions like categorical cross entropy, root mean square error, mean absolute error, etc., we need a loss function suitable for ordinal data. Our output classes are an ordered set of values. For example, it is much worse to misclassify severe NPDR or PDR as healthy eye than as moderate NPDR. Although both mean square error and mean absolute error penalizes the distance from mean, these loss functions are more suitable for regression problem whereas our problem is a multi-class classification problem. On the other hand, categorical cross entropy is the benchmark of loss functions for classification problem. However, it cannot capture the essence of ordinal data. So, we blended these two concepts and developed a new loss function "Mean Absolute Error with Cross-Entropy" or MAEC. The formula for our MAEC loss function is given by:

$$MAEC = (1 + MAE) \times CCE$$

where MAE and CCE denote "mean absolute error" and "Categorical Cross Entropy". They can be expressed as:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |\widehat{y}_i - y_i|$$

$$CCE = -\frac{1}{N}\sum_{i=0}^{N}\sum_{j=0}^{J} y_j.log(\widehat{y}_j) + (1 - y_j).log(1 - \widehat{y}_j)$$

where $\widehat{y}_i$ is the predicted value and y_i is the true value.

Our MAEC loss function helped the model to recognize the learning as a multi-class ordinal classification problem. We compared its performance with both Categorical Cross Entropy and Mean Squared Error where MAEC surpassed both of them. The results will be discussed in the later chapter. We also wrote a custom metrics function *f1_ score()* using scikit-learn[4],a machine learning library, to show the f1 score of the model on validation data after every epoch. Both accuracy and f1 score was used as the metric of our model while training.

3.5.2 Hyper-parameter tuning

Hyper-parameters are those parameters in a machine learning model which are set before the training or learning process starts. The values of all the other parameters are generally learned during the training phase. Hence, proper hyper-parameters selection is a pivotal factor for the training of any classifier. Our network was trained with Stochastic gradient descent (SGD) optimization function with 0.90 Nesterov momentum for 120 epochs with data augmentation at each step. We experimented with different learning rates and found 5×10^{-4} to be the best initial learning rate. The learning rate was decreased to 10^{-5} after 50 epochs and to 10^{-6} after 80 epochs. We used batch size of 16, 32 and 64 in our training and found that batch size of 16 surpassed the other ones. The summary of our training hyper-parameters are given in Table 3.5.

3.5.3 Training our model

The training of our network was performed using back-propagation algorithm with batch stochastic gradient descent such that our MAEC function is minimized. The neural network parameters are updated by propagating the gradient of loss multiplied by learning rate backwards. We used 10% of our data for validation purposes. Since the training dataset of 159,464

[4]`https://scikit-learn.org/stable/`

Hyperparameter	**Value**
Objective Function	Mean Absolute Error with Cross-entropy (MAEC)
Optimizer	SGD
Momentum	0.9
Learning Rates	5×10^{-4} (First 50 epochs) 10^{-5} (Next 30 epochs) 10^{-6} (Next 40 epochs)
Batch Size	16
Epoch	120
ReduceLROnPlateau	monitor='val_loss', factor=0.5, patience=10, epsilon=0.001
EarlyStopping	monitor='val_loss', patience=20

Table 3.5: Hyper-parameters of our network

images were balanced, the validation set was selected randomly. As mentioned earlier in Section 3.3, data was already augmented. However, it is better to augment the dataset again while training to make the model robust. Hence, data augmentation was performed at each step of the training. We performed the following augmentations randomly:

- **Rotation:** Images were randomly rotated between 0°to 180°
- **Flip:** Images were randomly flipped horizontally or vertically
- **Crop:** Images were randomly cropped to 90-95% of their original size
- **Shear:** Images were randomly sheared between 20°to 60°

The training was conducted on FloydHub [96] using TensorFlow framework for 120 epochs which took around six days on a Tesla K80 GPU. We shall discuss about the training and validation performance metrics in the next chapter.

3.6 Web Application Development

We developed a simple web application for DR classification where the user can upload a fundus image from his local computer and the application can show its class based on the pre-trained model from Section 3.5. We used Flask[5], a micro web framework written in python, for developing this. When the image has been uploaded, we first applied the preprocessing techniques mentioned in Section 3.2. Both the uploaded images and preprocessed are saved in the server for future uses. In the backend, our previously trained model was saved as a '.h5' file and it was loaded for predicting the class of the image. Figure 3.12 shows the simple web application where prediction is performed by our trained model. This type of application will greatly aid the diagnosis procedure. In addition, the model can also be improved through the new images by means of unsupervised learning.

3.7 Software and Hardware

In this section, we provide some technical details of the implementation. We used Python 3.5 environment and used of the following libraries, platforms and hardwares:

- **Keras** v2.2.4, a deep learning library which runs on top of Theano or TensorFlow
- **TensorFlow** v1.13.1, an open source library for fast numerical computation and machine learning
- **CUDA** v9.0, a parallel GPU computing platform and programming model
- **CuDNN** v7.1, a GPU-accelerated library of primitives for deep neural networks
- **OpenCV**, a library of Python bindings designed to solve computer vision problems
- **ImageMagick**, an open-source software for image processing
- **NumPy** v1.14.3, the fundamental package for scientific computing with python

[5] `http://flask.pocoo.org/`

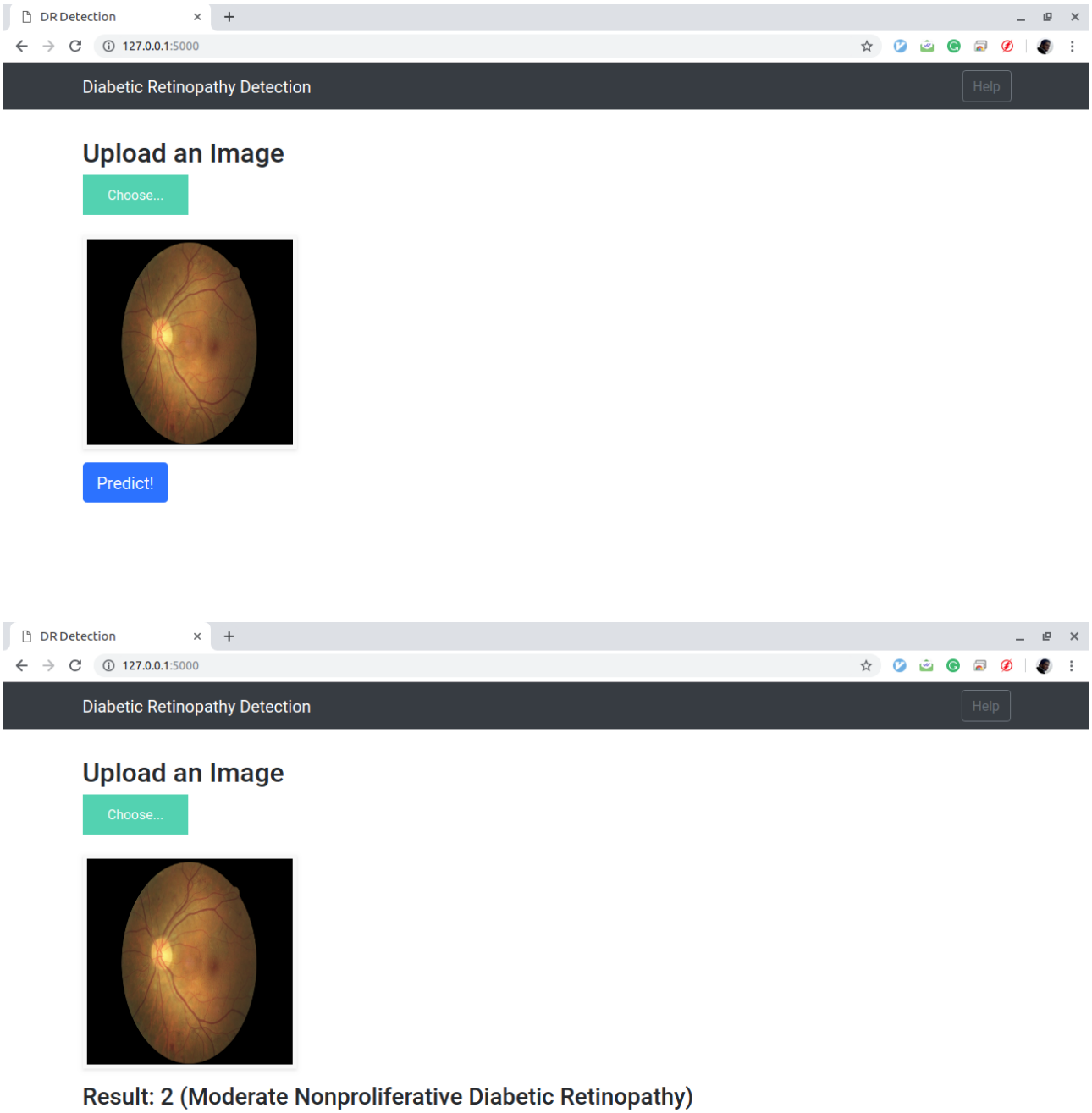


Figure 3.12: Web Application for DR Detection

- **Pandas** v0.24.1, a data analysis and manipulation library for Python
- **scikit-learn**, a machine learning library
- **Matplotlib** v3.0.3, a Python plotting library
- **Intel Core i5-6200U 2.3 GHz, 16.0 GB RAM, NVIDIA GeForce GeForce 920M**- this machine was used for initial model development and small scale testing v3.0.3, a Python plotting library
- **FloydHub**, a cloud-based deep learning platform. We used **Intel Xeon (2 core) machine (8.0 GB RAM) with Tesla K80 (12 GB) GPU** support. The final trainings of our model(s) were performed on FloydHub.

Chapter 4

Results

In this chapter, the performance of our proposed model is described elaborately. First, the training and validation performance of our model is presented. As mentioned earlier, we developed a custom loss function called MAEC and the effect of using MAEC will be discussed in details. Next, various performance metrics of our model on the test data is presented. Especially, the performance for detecting high severity DR is analyzed briefly. Later, we explore the qualitative results on the images and the reasons for misclassification. Finally, we evaluate the performance of our model on two external databases and perform a comparative analysis with some of the existing works.

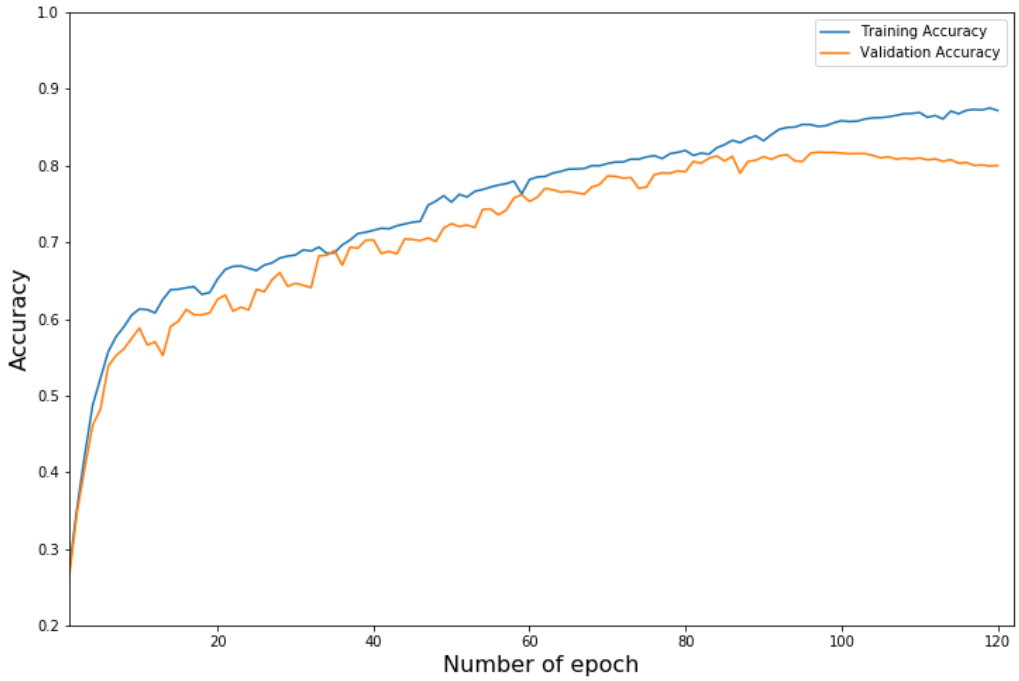
4.1 Training and Validation Performance

While training our model described in Section 3.4 with our MAEC loss function, we achieved the highest validation accuracy of 81.78% and lowest validation loss of 0.5556. On the other hands, when we trained our model with the loss function *Categorical Cross Entropy (CCE)* and *Mean Squared Error (MSE)* we attained a maximum of 74.82% and 75.88% accuracy. It is expected that the accuracy of MSE will be better than CCE because of the nature of our output. Since the output is ordered, MSE can penalize the misclassifications more than CCE. Training accuracy, validation accuracy, training loss and validation loss for our model trained with MAEC, CCE and MSE loss function can be seen if Figure 4.1, 4.2 and 4.3 respectively.

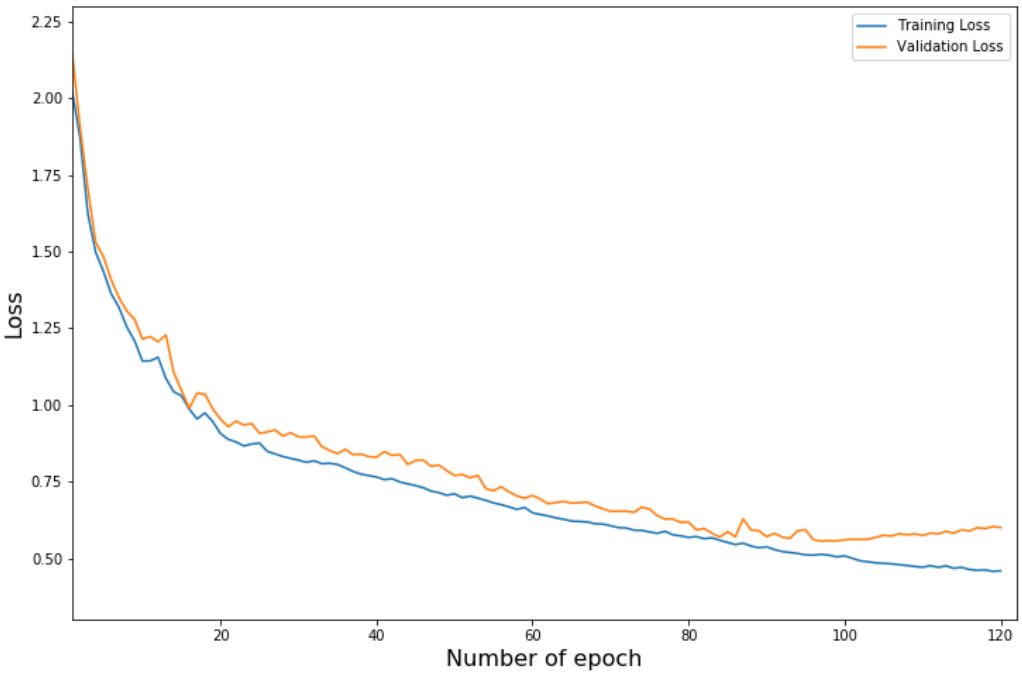
It is worth mentioning that, these models were trained with images of size 512×512. The performance of our MAEC loss function is much superior to the other ones because it penalizes the severe misclassification like MSE and considers the problem as a classification problem like CCE.

In Section 3.2.2, we mentioned that our dataset was downsampled to two different dimensions: 512×512 and 256×256. To analyze the effect of image size, we trained our model with both these datasets. Performance of the model trained with higher resolution images is much better than the other one. Figure 4.1 shows the training and validation performance of the model trained with 512×512 images where we achieved accuracy of 81.78%. On the contrary, the other model trained with 256×256 images attained a maximum of 73.77% validation accuracy. Training and loss curve can be seen if Figure 4.4. It is to be noted that while training this model, a technique called "batch normalization" [97] was used to avoid overfitting and improve robustness. Batch normalization can ameliorate the *internal covariate shift* problem [97]. The oscillation of validation accuracy and validation loss resulted because of batch normalization with high momentum.

To summarize, the performance of our model is greatly affected by the input image size and loss function. Small features like microaneurysms and haemorrhages are visible on the larger images. When downsampled, they might disappear hence causing misclassification. However, it is not feasible to train the model with image size larger than 512×512, because of the huge computational cost. The idea of using a custom loss function has been used in machine learning application before. Since the nature of the output varies from problem to problem, many data scientists have used their own custom loss function for training and achieved promising results. The results and analysis discussed in the later sections consider the model trained with 512×512 images using MAEC loss function, if not mentioned otherwise.

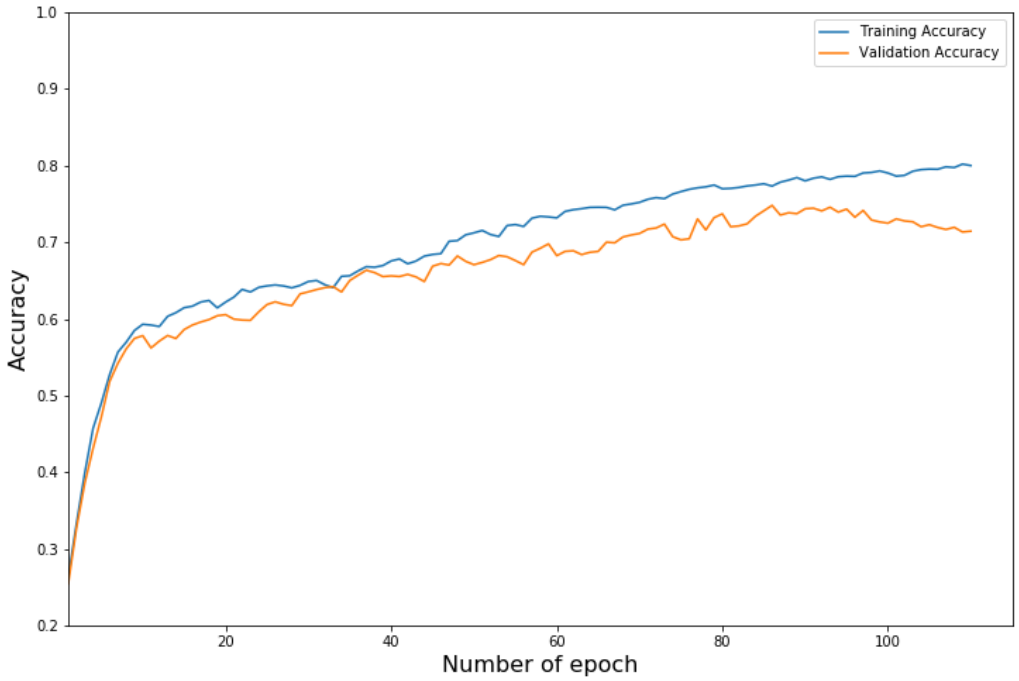


(a) Training and Validation Accuracy for MAEC

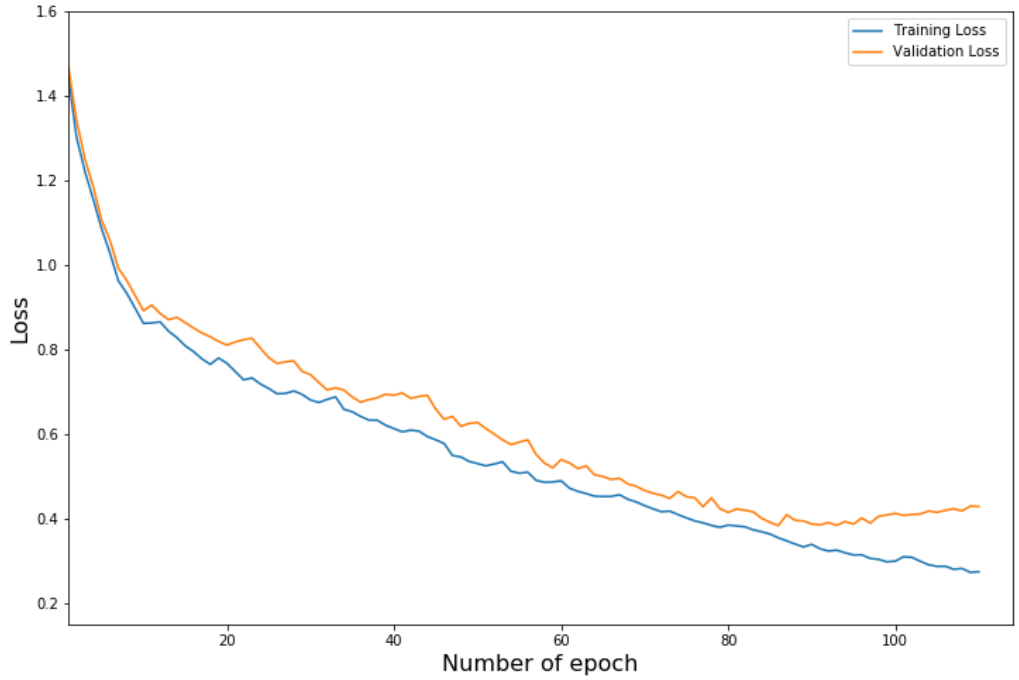


(b) Training and Validation Loss for MAEC

Figure 4.1: Accuracy and Loss curve of the model for MAEC loss function

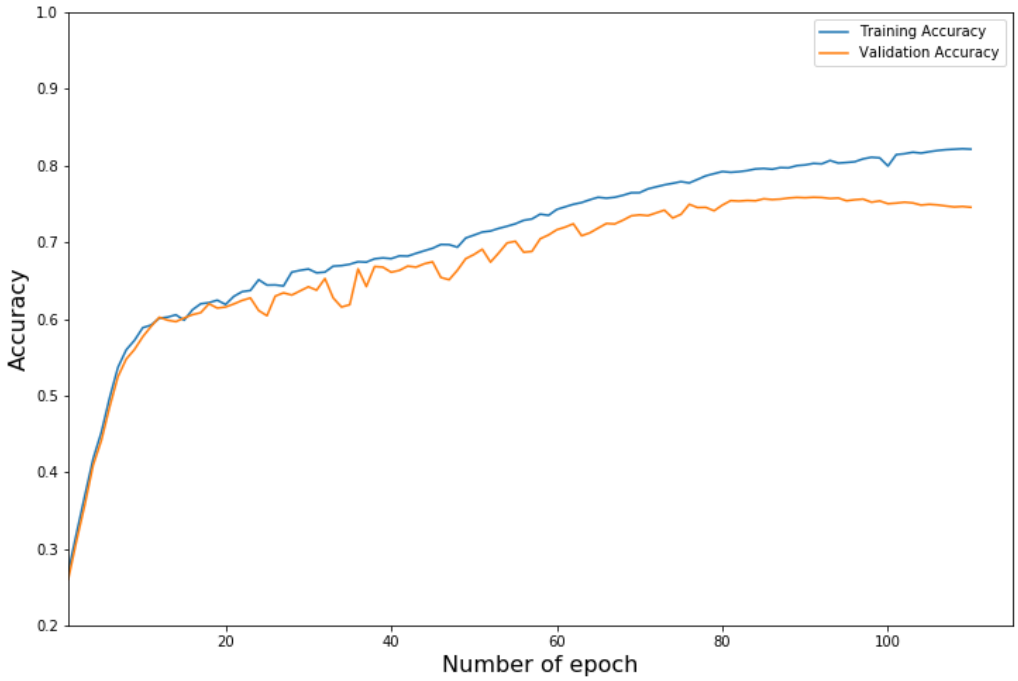


(a) Training and Validation Accuracy for CCE

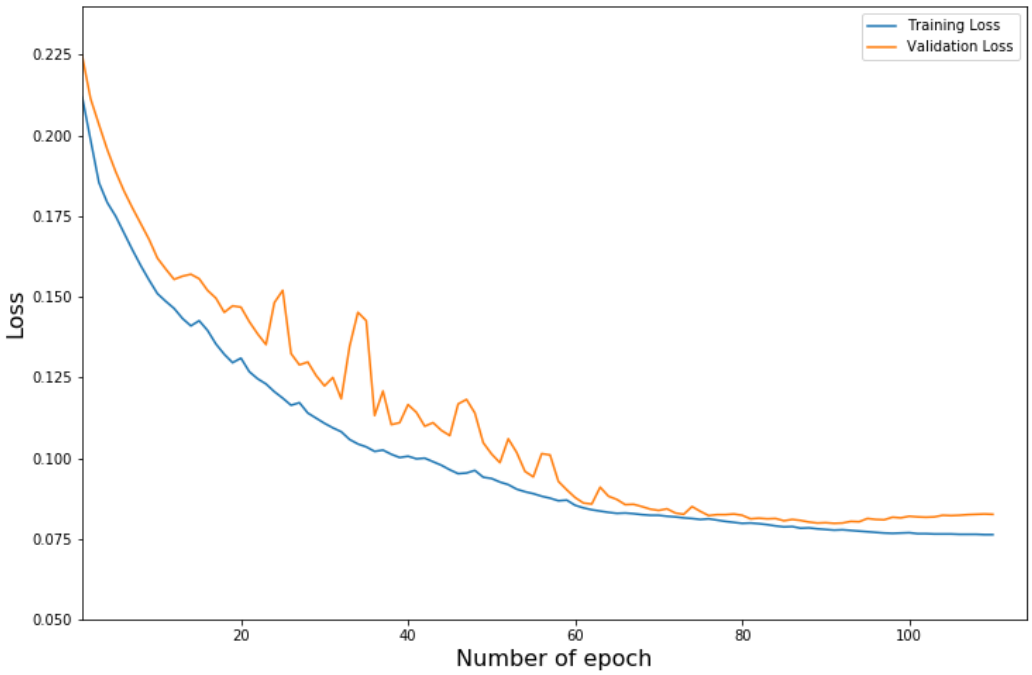


(b) Training and Validation Loss for CCE

Figure 4.2: Accuracy and Loss curve of the model for CCE loss function

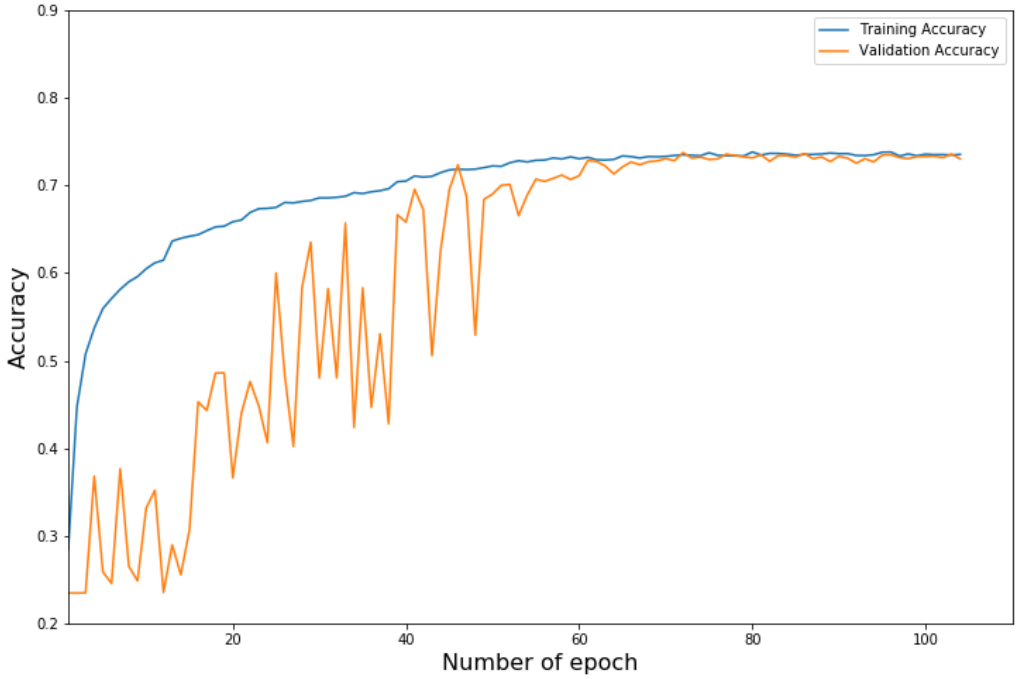


(a) Training and Validation Accuracy for MSE

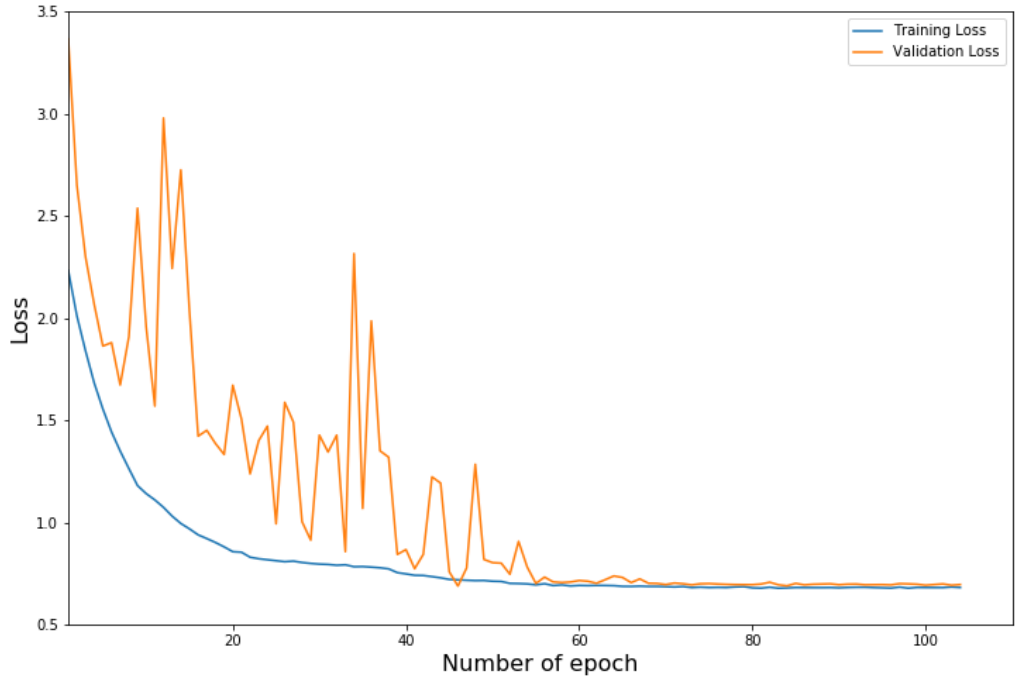


(b) Training and Validation Loss for MSE

Figure 4.3: Accuracy and Loss curve of the model for MSE loss function



(a) Training and Validation Accuracy for 256×256 images



(b) Training and Validation Loss for 256×256 images

Figure 4.4: Accuracy and Loss curve of the model trained with 256×256 images

4.2 Performance on Test Data

The model was validated on our test dataset of 10,000 images which was kept separate from the training dataset as mentioned in Section 3.1.2. The images were preprocessed as per described in Section 3.2. Out of 10,000 images belonging to five classes, our model could correctly classify 7,957 images which imply an accuracy of 79.57%. Table 4.1 shows the confusion matrix of our model for our test dataset. Since *accuracy* is not the most widely used metric for multi-class classification problem, we used four other metrics: *Sensitivity* (also known as *Recall*), *Specificity*, *Precision* and *F1 Score* which can be defined as:

$$Sensitivity = \frac{TruePositive}{TruePositive + FalseNegative}$$

$$Specificity = \frac{TrueNegative}{TrueNegative + FalsePositive}$$

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

$$F_1 = \frac{2 \times TruePositive}{2 \times TruePositive + FalsePositive + FalseNegative}$$

		Predicted Label				
		0	1	2	3	4
Actual Label	0	5366	39	93	0	2
	1	829	593	77	0	1
	2	390	128	1441	18	23
	3	18	1	204	236	41
	4	35	0	137	7	321

Table 4.1: Confusion matrix for DR classification on 10000 test images

Since this is a multi-class problem, these values were calculated for each class and are

Class	Sensitivity	Specificity	Precision	F1 Score
0	0.9756	0.717	0.8084	0.8841
1	0.3953	0.9802	0.7792	0.5245
2	0.7205	0.9361	0.7382	0.7293
3	0.472	0.9974	0.9042	0.6202
4	0.642	0.9929	0.8273	0.723

Table 4.2: Performance metrics of the model on test data

presented in Table 4.2. For, calculating the global Sensitivity, Specificity, Precision and F1 Score we used a general formula:

$$Metric = \frac{\sum |Class_i| \, Metric_i}{\sum |Class_i|}$$

where $Metric_i$ denotes the value of the metric (Sensitivity, Specificity, Precision and F1 Score) for *Class i* and $|Class_i|$ denotes the total number of instances of *Class i*. Using the above-mentioned formula the Sensitivity, Specificity, Precision and F1 Score of our model was found to be 0.7958, 0.8281, 0.7957 and 0.778 respectively. The summary of various performance metrics of our model is presented in Table 4.3.

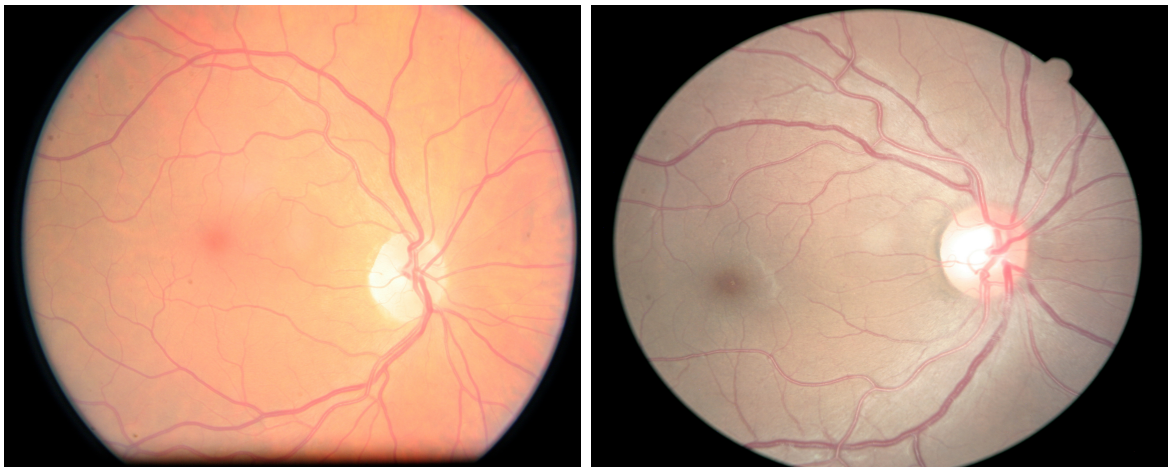
Metric	Value
Validation Accuracy	81.78%
Validation Loss	0.5556
Test Accuracy	79.57%
Sensitivity	0.7958
Specificity	0.8281
Precision	0.7957
F1 Score	0.778

Table 4.3: Global performance metrics of the model on test data (multi-class)

4.3 A Closer Look at Performance

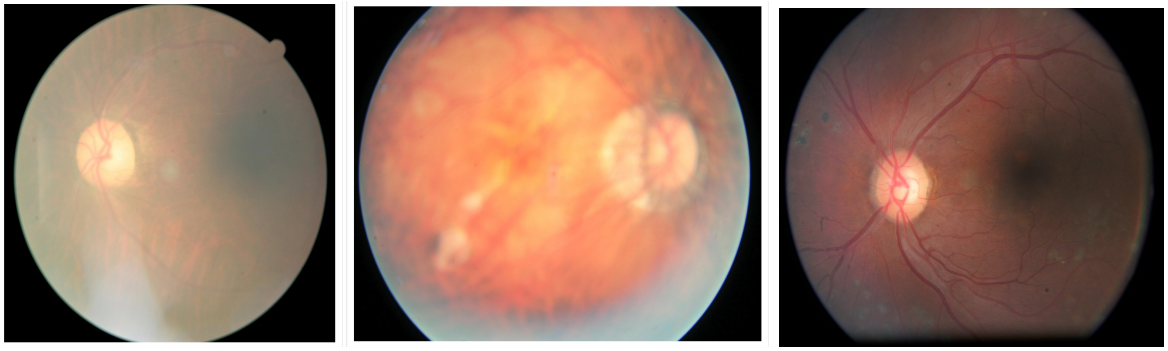
In this section, we shall analyze the performance of our model on the test data. In addition, some misclassified images along with the possible reasons behind misclassification will also be discussed. Since this is a 5-class classification problem, accuracy of 79.57% is quite promising. From our best knowledge, no other work has achieved such accuracy for a 5-class DR severity detection task. In general, human performance for successful DR grading is around 75%. From Table 4.1, we can see that our model is unable to correctly identify DR severity for 2,043 images. Most of these misclassifications are caused by camera artifacts, poor image quality and wrong actual labels. We shall now discuss about the misclassifications of each class in more details.

The model's ability to correctly identify class-0 images is quite remarkable which is reflected by a sensitivity value of 97.56%. Only 39 images belonging to class-0 was wrongly misclassified as class-1 images. However, 829 images of class 1 are wrongly classified as class 0. This is because in general, there is very little visual difference between a class-0 image and class-1 image. The first stage of DR (class-1) usually has very few microaneurysms which are very hard to identify even for specialists. Figure 4.5 demonstrates this scenario. Both the images seem like a class-0 image as there are no prominent visual lesions present. However, the left image belongs to class-0 which is classified as class-1 by our model and the right image belongs to class-1 which is classified as class-0 by our model. A vast majority of these 829 images belong to this category where there are no visual lesions of DR present, but still, they are graded as class-1 image. These misclassifications may have occurred because of downsampling the original image, unreliable actual grading and due to the noise present in the images. The overall performance metrics of our model are greatly affected due to these misclassifications. Many classification techniques combine these two classes as one because of these reasons. There are 2 class-0 images predicted as class-4 images and 1 class-1 image predicted as class-4 image. From Figure 4.6a, we can see that both of the class-0 images have such poor quality that they are almost ungradable. However, it is surprising that the image of Figure 4.6b is actually graded as a class-1 image. Neovascularization and exudates are clearly visible in the



(a) Class-0 image classified as class-1 image (b) Class-1 image classified as class-0 image

Figure 4.5: Very little anatomical difference between a class-0 and class-1 image



(a) Poor quality class-0 images classified as class-4 (b) Class-1 classified as class-4

Figure 4.6: Class-0 and class-1 images misclassified as class-4 images

image and as such, the image is classified as a class-4 image by our classifier.

The effect of classifying moderate/severe nonproliferative DR or proliferative DR as no DR is much more severe than the above-mentioned cases. Our model's performance on these cases is quite promising. From Table 4.1, we can see that 390 images of class-2 were wrongly misclassified as class-0. However, after examining these images, we see that many of them have camera artifacts present like Figure 4.7a. As a result, despite the symptoms of class-2 DR present in these images, our classifier cannot classify them properly. Some of these misclassified images have no lesions of DR present as shown in Figure 4.7b. We assume that the actual labels are faulty in these cases.

Among the 1000 images belonging to class-3 and class-4, 53 was wrongly classified as class-

0 and 1 was wrongly classified as class-1. Surprisingly, some of these images had no prominent symptoms of DR as shown in Figure 4.7c, 4.7e and 4.7h. On the other hand, there are some rare images in the test set like Figure 4.7g. As we can see, the retina is severely damaged. However, this type of images was not present in the training set. As a result, our classifier cannot identify the severity level of DR. Figure 4.7d and 4.7f shows two images which were misclassified even though visual symptoms of DR is quite conspicuous. Extensive training and ensembling multiple models may improve the performance of the model in such cases.

To summarize, although our model could not correctly classify 2,043 images to their exact severity level, overall performance is quite intriguing. Poor quality of the images, faulty actual labels and indistinguishable difference between class-0 and class-1 images are the main reasons behind these misclassifications. In the next section, we shall see the performance of our model on some other datasets where the images are more noise-free and the actual gradings are more consistent.

4.4 Comparison with Retrospective Models

Since most of the existing works have focused on binary classification (DR vs no DR) and used different datasets, it is not possible to perform apples to apples comparison. Still, we tried to compare our model with some existing studies. Hence, we had to convert our multi-class problem to two-class problem for some comparisons. We also validated our model on two other datasets: Messidor and E-Ophtha. By using our model on these datasets we can validate the generalization performance of our model as well as compare it with some studies. In this section, we shall briefly present our model's performance as a two-class problem, our model's generalization capability and a comparison with some retrospective studies.

4.4.1 Two-class Performance

We converted our problem into two types of binary classification: affected (class 1,2,3,4) vs healthy (class 0) and vision-threatening DR (vtDR) (class 2,3,4) vs referable DR (rDR) (class 0, 1). The confusion matrix for these two classifications can be seen at Table 4.4 and 4.5.



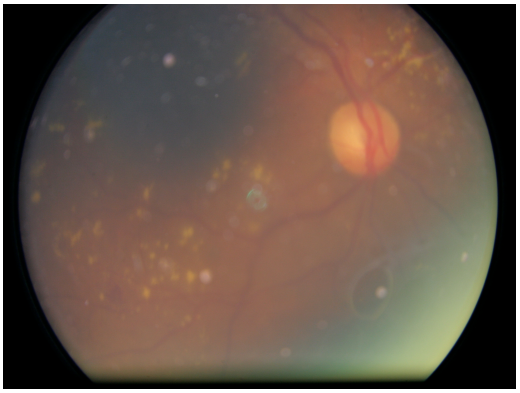
(a) Class-2 classified as class-0



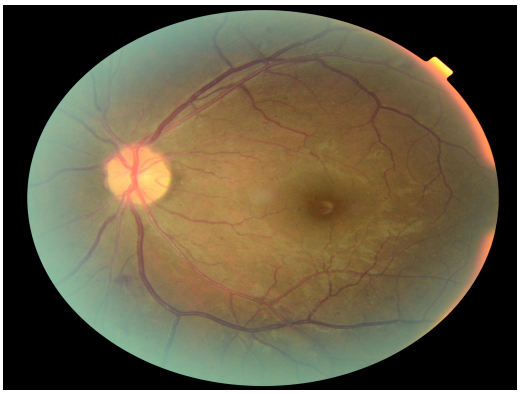
(b) Class-2 classified as class-0



(c) Class-3 classified as class-0



(d) Class-3 classified as class-0



(e) Class-3 classified as class-1



(f) Class-4 classified as class-0



(g) Class-4 classified as class-0



(h) Class-4 classified as class-0

Figure 4.7: Examples of some severe misclassifications

		Predicted 1	Predicted 0
Actual	1	3228	1272
	0	134	5366

Table 4.4: Confusion matrix for affected (class 1,2,3,4) vs healthy (class 0) classification

		Predicted 1	Predicted 0
Actual	1	2428	572
	0	173	6827

Table 4.5: Confusion matrix for vtDR (class 2,3,4) vs rDR (class 0, 1) classification

We also calculated various performance metrics for these classifications and found out that performance is high for the later classification scheme which is expected. The performance metrics are presented in Table 4.6.

Classification Scheme	**Accuracy**	**Sensitivity**	**Specificity**	**Precision**	**F1 Score**
Affected (1, 2, 3, 4) vs Healthy (0)	85.94%	0.717	0.9756	0.9601	0.8211
vtDR (2, 3, 4) vs rDR (0, 1)	92.55%	0.8093	0.9753	0.9335	0.867

Table 4.6: Performance metrics for two binary classification schemes

4.4.2 Generalization Capability and Comparison

We tested our model using the public E-Ophtha and Messidor databases for external validation. We used our second binary classification scheme (vision-threatening DR vs referable DR) for both of these databases. The E-Ophtha database contains 463 images, out of them 268 belong to healthy classes and 195 belong to DR affected retina. Our model correctly classified 434 images, thus achieving an accuracy of 93.74%, sensitivity of 88.72% and specificity of 98.12%. The Messidor database contains 1200 images with 297 DR affected images. Accuracy, sensitivity and specificity of our model on this dataset was 96.83%, 91.58% and 98.23%, respectively.

In general, the performance on these databases is far more superior than the test set. This is because Messidor and E-Ophtha databases consist of images of high quality and almost 100% of them are gradable where Kaggle's quality is more heterogeneous and estimated as around 75% gradable. Our model's remarkable performance on these databases indicates high robustness and generalization capability of the model.

As mentioned earlier, it is very difficult to perform apples to apples comparison with the other studies. We attempted to compare the results of our model with some of the existing studies using the same dataset: Kaggle, Messidor and E-Ophtha. However, it is to be noted that, although we used the same dataset for validation, some of the studies used only a subset of the datasets. As a result, the comparison does not reflect the actual performance, rather it is an approximation. Table 4.7 shows the comparison of our model with five existing works. Our model was developed focusing on the 5-class DR severity level. Among the retrospective works, only one such work (Pratt et. al. [83]) is present that is comparable with our original model. Although they achieved a higher specificity, our model's accuracy and sensitivity is much better than theirs. While comparing with the other four models, we found that in general, our model has a lower sensitivity, but a higher specificity. Had our model been originally trained for such binary classification, we could have achieved higher sensitivity by sacrificing specificity. In general, table 4.7 shows that our model has comparable or better results in comparison with the state-of-the-art works of literature.

Model	**Classification Scheme**	**Test/Validation Dataset**	**Performance**	**Our Model's Performance**
Pratt et. al. [83]	5-Class	Kaggle	Accuracy: 75% Sensitivity: 30% Specificity: 95%	Accuracy: 79.57% Sensitivity: 79.58% Specificity: 82.81%
Gargeya et. al. [35]	2-Class	E-Ophtha	Sensitivity: 90% Specificity: 94%	Sensitivity: 88.72% Specificity: 98.12%
Ramachandran et. al. [77]	2-Class	Messidor	Sensitivity: 96% Specificity: 90%	Sensitivity: 92.59% Specificity: 98.23%
Rakhlin et. al. [84]	2-Class	Kaggle	Sensitivity: 86% Specificity: 82%	Sensitivity: 82.93% Specificity: 97.53%
Colas et. al. [76]	2-Class	Kaggle	Sensitivity: 96.2% Specificity: 66.6%	Sensitivity: 82.93% Specificity: 97.53%

Table 4.7: Comparison with retrospective studies

Chapter 5

Conclusion

This work describes the development and validation of a novel deep learning based Diabetic Retinopathy severity detection model. There are very few works existing in the literature that have attempted to identify the five stages of DR and to our best knowledge, none of them has achieved better performance than our model. The major challenge was to develop such a model that can adapt to the huge variation present in a real dataset. Finding the proper preprocessing techniques and designing the proper network architecture along with tuning the hyperparameters of the model was pivotal to boost the performance of the model. Another considerable challenge was the availability of high-performance GPU for training the model. The key strength of this study includes the use of a large heterogeneous dataset (53,000 training images), use of large image size (512×512), development of a custom loss function (MAEC) and the robust performance of our model when tested on two external validation data sets.

Some limitations still exist implying possible room for improvement. Our model misclassified a significant number of DR affected images as healthy images in the test set. Although many of these misclassifications were caused by poor image quality and noisy actual label, some of the misclassifications were due to the model's inability to identify the proper severity level. Had the model been trained with more DR affected images, the performance of identifying proper severity level might have increased by a significant margin. Recently, ensemble learning has gained unprecedented popularity because of its ability to improve the overall performance of a machine learning model. Ensembling multiple networks may enhance our model's per-

formance by some margin. Another significant improvement can be the integration of both feature extraction and deep learning. Very recently, some works have focused on extracting a subset of features and then using a CNN for detecting DR severity. As part of the future plan, we hope to identify all the necessary anatomical components related to DR and then use deep learning techniques on the features for better performance and visualization. In this work, we have considered each eye separately. However, more often both eyes have the same DR severity level and by blending the information of both eyes, performance can be improved to some degree.

To conclude, this deep learning Diabetic Retinopathy severity detection system shows very promising and robust performance. Such technology offers great potential to improve the efficacy and accessibility of the DR screening programs, particularly in underdeveloped and developing nations. However, further investigation is needed before clinically deploying this system. Proper implementation of such a system coupled with telemedicine, can improve the overall standard of eye care environments.

Bibliography

[1] Image Source: `https://seeclearkalamazoo.com/services/diabetic-eye-care/diabetic-retinopathy/`. Last accessed 11:37 pm, April 15, 2019.

[2] AS Krolewski, JH Warram, LI Rand, AR Christlieb, EJ Busick, and CR Kahn. Risk of proliferative diabetic retinopathy in juvenile-onset type i diabetes: a 40-yr follow-up study. *Diabetes care*, 9(5):443–452, 1986.

[3] Sourya Sengupta, Amitojdeep Singh, Henry A Leopold, and Vasudevan Lakshminarayanan. Ophthalmic diagnosis and deep learning–a survey. *arXiv preprint arXiv:1812.07101*, 2018.

[4] AR Bhavsar, GG Emerson, and MV Emerson. Epidemiology of diabetic retinopathy: In: Browning dj, editor. diabetic retinopathy: Evidence-based management, 2010.

[5] Janet L Leasher, Rupert RA Bourne, Seth R Flaxman, Jost B Jonas, Jill Keeffe, Kovin Naidoo, Konrad Pesudovs, Holly Price, Richard A White, Tien Y Wong, et al. Global estimates on the number of people blind or visually impaired by diabetic retinopathy: a meta-analysis from 1990 to 2010. *Diabetes care*, 39(9):1643–1649, 2016.

[6] Yingfeng Zheng, Mingguang He, and Nathan Congdon. The worldwide epidemic of diabetic retinopathy. *Indian journal of ophthalmology*, 60(5):428, 2012.

[7] E Simon Barriga, Victor Murray, Carla Agurto, Marios Pattichis, Wendall Bauman, Gilberto Zamora, and Peter Soliz. Automatic system for diabetic retinopathy screening based on am-fm, partial least squares, and support vector machines. In *2010 IEEE*

International Symposium on Biomedical Imaging: From Nano to Macro, pages 1349–1352. IEEE, 2010.

[8] Carla Agurto, S Barriga, Víctor Murray, Sergio Murillo, Gilberto Zamora, Wendall Bauman, M Pattichis, and Peter Soliz. Toward comprehensive detection of sight threatening retinal disease using a multiscale am-fm methodology. In *Medical Imaging 2011: Computer-Aided Diagnosis*, volume 7963, page 796316. International Society for Optics and Photonics, 2011.

[9] CP Wilkinson, Frederick L Ferris III, Ronald E Klein, Paul P Lee, Carl David Agardh, Matthew Davis, Diana Dills, Anselm Kampik, R Pararajasegaram, Juan T Verdaguer, et al. Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales. *Ophthalmology*, 110(9):1677–1682, 2003.

[10] Oliver Faust, Rajendra Acharya, Eddie Yin-Kwee Ng, Kwan-Hoong Ng, and Jasjit S Suri. Algorithms for the automated detection of diabetic retinopathy using digital fundus images: a review. *Journal of medical systems*, 36(1):145–157, 2012.

[11] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[12] Xiangyu Zhang, Jianhua Zou, Kaiming He, and Jian Sun. Accelerating very deep convolutional networks for classification and detection. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):1943–1955, 2016.

[13] Hoo-Chang Shin, Holger R Roth, Mingchen Gao, Le Lu, Ziyue Xu, Isabella Nogues, Jianhua Yao, Daniel Mollura, and Ronald M Summers. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5):1285–1298, 2016.

[14] Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Brian Kingsbury, et al. Deep

neural networks for acoustic modeling in speech recognition. *IEEE Signal processing magazine*, 29, 2012.

[15] Daniel SW Ting, Paul H Yi, and Ferdinand Hui. Clinical applicability of deep learning system in detecting tuberculosis with chest radiography. *Radiology*, 286(2):729, 2018.

[16] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115, 2017.

[17] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes Van Diest, Bram Van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen AWM Van Der Laak, Meyke Hermsen, Quirine F Manson, Maschenka Balkenhol, et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *Jama*, 318(22):2199–2210, 2017.

[18] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. Efficient object localization using convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 648–656, 2015.

[19] *Convolution*. `https://en.wikipedia.org/wiki/Convolution`. Last accessed 12:51 am, April 13, 2019.

[20] Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama*, 316(22):2402–2410, 2016.

[21] Romany F Mansour. Deep-learning-based automatic computer-aided diagnosis system for diabetic retinopathy. *Biomedical engineering letters*, 8(1):41–57, 2018.

[22] Katrine B Nielsen, Mie L Lautrup, Jakob KH Andersen, Thiusius R Savarimuthu, and Jakob Grauslund. Deep learning-based algorithms in screening of diabetic retinopathy: A systematic review of diagnostic performance. *Ophthalmology Retina*, 2018.

[23] Michael David Abràmoff, Yiyue Lou, Ali Erginay, Warren Clarida, Ryan Amelon, James C Folk, and Meindert Niemeijer. Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning. *Investigative ophthalmology & visual science*, 57(13):5200–5206, 2016.

[24] Zhixi Li, Yifan He, Stuart Keel, Wei Meng, Robert T Chang, and Mingguang He. Efficacy of a deep learning system for detecting glaucomatous optic neuropathy based on color fundus photographs. *Ophthalmology*, 125(8):1199–1206, 2018.

[25] Philippe M Burlina, Neil Joshi, Michael Pekala, Katia D Pacheco, David E Freund, and Neil M Bressler. Automated grading of age-related macular degeneration from color fundus images using deep convolutional neural networks. *JAMA ophthalmology*, 135(11):1170–1176, 2017.

[26] James M Brown, J Peter Campbell, Andrew Beers, Ken Chang, Susan Ostmo, RV Paul Chan, Jennifer Dy, Deniz Erdogmus, Stratis Ioannidis, Jayashree Kalpathy-Cramer, et al. Automated diagnosis of plus disease in retinopathy of prematurity using deep convolutional neural networks. *JAMA ophthalmology*, 136(7):803–810, 2018.

[27] Etienne Decencière, Xiwei Zhang, Guy Cazuguel, Bruno Lay, Béatrice Cochener, Caroline Trone, Philippe Gain, Richard Ordonez, Pascale Massin, Ali Erginay, et al. Feedback on a publicly distributed image database: the messidor database. *Image Analysis & Stereology*, 33(3):231–234, 2014.

[28] Etienne Decencière, Guy Cazuguel, Xiwei Zhang, Guillaume Thibault, J-C Klein, Fernand Meyer, Beatriz Marcotegui, Gwénolé Quellec, Mathieu Lamard, Ronan Danno, et al. Teleophta: Machine learning and image processing methods for teleophthalmology. *Irbm*, 34(2):196–203, 2013.

[29] Gerald Liew, Michel Michaelides, and Catey Bunce. A comparison of the causes of blindness certifications in england and wales in working age adults (16–64 years), 1999–2000 with 2009–2010. *BMJ open*, 4(2):e004015, 2014.

[30] David J Browning. *Diabetic retinopathy: evidence-based management.* Springer Science & Business Media, 2010.

[31] *International Diabetes Federation.* IDF Diabetes Atlas - 8th Edition. Website:`https://diabetesatlas.org/`. Last accessed 06:12 pm, April 23, 2019.

[32] Sobha Sivaprasad, Bhaskar Gupta, Roxanne Crosby-Nwaobi, and Jennifer Evans. Prevalence of diabetic retinopathy in various ethnic groups: a worldwide perspective. *Survey of ophthalmology*, 57(4):347–370, 2012.

[33] Donald S Fong, Lloyd Aiello, Thomas W Gardner, George L King, George Blankenship, Jerry D Cavallerano, Fredrick L Ferris, and Ronald Klein. Retinopathy in diabetes. *Diabetes care*, 27(suppl 1):s84–s87, 2004.

[34] Richard Donnelly and Edward Horton. *Vascular complications of diabetes: current issues in pathogenesis and treatment.* John Wiley & Sons, 2008.

[35] Rishab Gargeya and Theodore Leng. Automated identification of diabetic retinopathy using deep learning. *Ophthalmology*, 124(7):962–969, 2017.

[36] Chirag A Shah. Diabetic retinopathy: A comprehensive review. *Indian journal of medical sciences*, 62(12):500–519, 2008.

[37] *Screening for Diabetic Retinopathy in Europe 15 years after the St. Vincent Declaration.* Retrieved from: `http://reseau-ophdiat.aphp.fr/Document/Doc/confliverpool.pdf`. Last accessed 05:24 pm, April 23, 2019.

[38] R John Winder, Philip J Morrow, Ian N McRitchie, JR Bailie, and Patricia M Hart. Algorithms for digital image processing in diabetic retinopathy. *Computerized medical imaging and graphics*, 33(8):608–622, 2009.

[39] C Aravind, M Ponnibala, and S Vijayachitra. Automatic detection of microaneurysms and classification of diabetic retinopathy images using svm technique. *International Journal of Computer Applications*, 975:8887, 2013.

[40] Chanjira Sinthanayothin, James F Boyce, Helen L Cook, and Thomas H Williamson. Automated localisation of the optic disc, fovea, and retinal blood vessels from digital colour fundus images. *British Journal of Ophthalmology*, 83(8):902–910, 1999.

[41] James Lowell, Andrew Hunter, David Steel, Ansu Basu, Robert Ryder, Eric Fletcher, Lee Kennedy, et al. Optic nerve head segmentation. *Medical Imaging, IEEE Transactions on*, 23(2):256–264, 2004.

[42] Saiprasad Ravishankar, Arpit Jain, and Anurag Mittal. Automated feature extraction for early detection of diabetic retinopathy in fundus images. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 210–217. IEEE, 2009.

[43] Thomas Walter and Jean-Claude Klein. Segmentation of color fundus images of the human retina: Detection of the optic disc and the vascular tree using morphological techniques. In *International Symposium on Medical Data Analysis*, pages 282–287. Springer, 2001.

[44] Shirin Hajeb Mohammad Alipour, Hossein Rabbani, and Mohammad Reza Akhlaghi. Diabetic retinopathy grading by digital curvelet transform. *Computational and mathematical methods in medicine*, 2012, 2012.

[45] Subhasis Chaudhuri, Shankar Chatterjee, Norman Katz, Mark Nelson, and Michael Goldbaum. Detection of blood vessels in retinal images using two-dimensional matched filters. *IEEE Transactions on medical imaging*, 8(3):263–269, 1989.

[46] Muhammad Moazam Fraz, Paolo Remagnino, Andreas Hoppe, Bunyarit Uyyanonvara, Alicja R Rudnicka, Christopher G Owen, and Sarah A Barman. Blood vessel segmentation methodologies in retinal images–a survey. *Computer methods and programs in biomedicine*, 108(1):407–433, 2012.

[47] AD Hoover, Valentina Kouznetsova, and Michael Goldbaum. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. *IEEE Transactions on Medical imaging*, 19(3):203–210, 2000.

[48] RA Welikala, Muhammad Moazam Fraz, Jamshid Dehmeshki, Andreas Hoppe, V Tah, S Mann, Thomas H Williamson, and Sarah A Barman. Genetic algorithm based feature selection combined with dual classification for the automated detection of proliferative diabetic retinopathy. *Computerized Medical Imaging and Graphics*, 43:64–77, 2015.

[49] Junichiro Hayashi, Takamitsu Kunieda, Joshua Cole, Ryusuke Soga, Yuji Hatanaka, Miao Lu, Takeshi Hara, and Hiroshi Fujita. A development of computer-aided diagnosis system using fundus images. In *Proceedings Seventh International Conference on Virtual Systems and Multimedia*, pages 429–438. IEEE, 2001.

[50] José Ignacio Orlando, Marcos Fracchia, Valeria del Río, and Mariana del Fresno. Retinal blood vessel segmentation in high resolution fundus photographs using automated feature parameter estimation. In *13th International Conference on Medical Information Processing and Analysis*, volume 10572, page 1057210. International Society for Optics and Photonics, 2017.

[51] Lili Xu and Shuqian Luo. A novel method for blood vessel detection from retinal images. *Biomedical engineering online*, 9(1):14, 2010.

[52] Carmen Alina Lupascu, Domenico Tegolo, and Emanuele Trucco. FABC: retinal vessel segmentation using adaboost. *IEEE Transactions on Information Technology in Biomedicine*, 14(5):1267–1274, 2010.

[53] Deepashree Devaraj Nagaveena and SC Prasanna Kumar. Vessels segmentation in diabetic retinopathy by adaptive median thresholding. *The International Journal Of Science & Technoledge (ISSN 2321–919X)*, 2013.

[54] Huazhu Fu, Yanwu Xu, Damon Wing Kee Wong, and Jiang Liu. Retinal vessel segmentation via deep learning network and fully-connected conditional random fields. In *Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on*, pages 698–701. IEEE, 2016.

[55] Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Pablo Arbeláez, and Luc Van Gool. Deep retinal image understanding. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 140–148. Springer, 2016.

[56] Paweł Liskowski and Krzysztof Krawiec. Segmenting retinal blood vessels with deep neural networks. *IEEE transactions on medical imaging*, 35(11):2369–2380, 2016.

[57] Huan Wang, Wynne Hsu, Kheng Guan Goh, and Mong Li Lee. An effective approach to detect lesions in color retinal images. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 2, pages 181–186. IEEE, 2000.

[58] Andrew Hunter, James Lowell, Jonathan Owens, Lee Kennedy, and David Steele. Quantification of diabetic retinopathy using neural networks and sensitivity analysis. In *Artificial Neural Networks in Medicine and Biology*, pages 81–86. Springer, 2000.

[59] Wynne Hsu, PMDS Pallawala, Mong Li Lee, and Kah-Guan Au Eong. The role of domain knowledge in the detection of retinal hard exudates. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–II. IEEE, 2001.

[60] C Eswaran, Ahmed Wasif Reza, and Subhas Hati. Extraction of the contours of optic disc and exudates based on marker-controlled watershed segmentation. In *Computer Science and Information Technology, 2008. ICCSIT'08. International Conference on*, pages 719–723. IEEE, 2008.

[61] Huiqi Li and Opas Chutatape. Fundus image features extraction. In *Proceedings of the 22nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (Cat. No. 00CH37143)*, volume 4, pages 3071–3073. IEEE, 2000.

[62] Alireza Osareh, Bita Shadgar, and Richard Markham. A computational-intelligence-based approach for detection of exudates in diabetic retinopathy images. *IEEE Transactions on Information Technology in Biomedicine*, 13(4):535–545, 2009.

[63] Pavle Prentašić and Sven Lončarić. Detection of exudates in fundus photographs using deep neural networks and anatomical landmark detection fusion. *Computer methods and programs in biomedicine*, 137:281–292, 2016.

[64] Oscar Perdomo, John Arevalo, and Fabio A González. Convolutional network to detect exudates in eye fundus images of diabetic subjects. In *12th International Symposium on Medical Information Processing and Analysis*, volume 10160, page 101600T. International Society for Optics and Photonics, 2017.

[65] Bernhard M Ege, Ole K Hejlesen, Ole V Larsen, Karina Møller, Barry Jennings, David Kerr, and David A Cavan. Screening for diabetic retinopathy using computer based image analysis and statistical classification. *Computer methods and programs in biomedicine*, 62(3):165–175, 2000.

[66] Alan D Fleming, Sam Philip, Keith A Goatman, John A Olson, and Peter F Sharp. Automated microaneurysm detection using local contrast normalization and local vessel detection. *IEEE transactions on medical imaging*, 25(9):1223–1232, 2006.

[67] Chanjira Sinthanayothin, James F Boyce, Tom H Williamson, Helen L Cook, Evelyn Mensah, Shantanu Lal, and David Usher. Automated detection of diabetic retinopathy on digital fundus images. *Diabetic medicine*, 19:105–112, 2002.

[68] Atsushi Mizutani, Chisako Muramatsu, Yuji Hatanaka, Shinsuke Suemori, Takeshi Hara, and Hiroshi Fujita. Automated microaneurysm detection method based on double ring filter in retinal fundus images. In *Medical Imaging 2009: Computer-Aided Diagnosis*, volume 7260, page 72601N. International Society for Optics and Photonics, 2009.

[69] Michael Larsen, Jannik Godt, Nicolai Larsen, Henrik Lund-Andersen, Anne Katrin Sjølie, Elisabet Agardh, Helle Kalm, Michael Grunkin, and David R Owens. Automated detection of fundus photographic red lesions in diabetic retinopathy. *Investigative ophthalmology & visual science*, 44(2):761–766, 2003.

[70] Meindert Niemeijer, Bram Van Ginneken, Joes Staal, Maria SA Suttorp-Schulten, and Michael D Abràmoff. Automatic detection of red lesions in digital color fundus photographs. *IEEE Transactions on medical imaging*, 24(5):584–592, 2005.

[71] Li Tang, Meindert Niemeijer, Joseph M Reinhardt, Mona K Garvin, and Michael D Abràmoff. Splat feature classification with application to retinal hemorrhage detection in fundus images. *IEEE Transactions on Medical Imaging*, 32(2):364–375, 2013.

[72] Mark JJP van Grinsven, Bram van Ginneken, Carel B Hoyng, Thomas Theelen, and Clara I Sánchez. Fast convolutional neural network training using selective data sampling: application to hemorrhage detection in color fundus images. *IEEE transactions on medical imaging*, 35(5):1273–1284, 2016.

[73] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[74] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[75] Gwenolé Quellec, Katia Charrière, Yassine Boudi, Béatrice Cochener, and Mathieu Lamard. Deep image mining for diabetic retinopathy screening. *Medical image analysis*, 39:178–193, 2017.

[76] E Colas, A Besse, A Orgogozo, B Schmauch, N Meric, and E Besse. Deep learning approach for diabetic retinopathy screening. *Acta Ophthalmologica*, 94, 2016.

[77] Nishanthan Ramachandran, Sheng Chiong Hong, Mary J Sime, and Graham A Wilson. Diabetic retinopathy screening using deep neural network. *Clinical & experimental ophthalmology*, 46(4):412–416, 2018.

[78] Hidenori Takahashi, Hironobu Tampo, Yusuke Arai, Yuji Inoue, and Hidetoshi Kawashima. Applying artificial intelligence to disease staging: Deep learning for improved staging of diabetic retinopathy. *PloS one*, 12(6):e0179790, 2017.

[79] Daniel Shu Wei Ting, Carol Yim-Lui Cheung, Gilbert Lim, Gavin Siew Wei Tan, Nguyen D Quang, Alfred Gan, Haslina Hamzah, Renata Garcia-Franco, Ian Yew San Yeo, Shu Yen Lee, et al. Development and validation of a deep learning system for diabetic retinopathy and related eye diseases using retinal images from multiethnic populations with diabetes. *Jama*, 318(22):2211–2223, 2017.

[80] Gabriel García, Jhair Gallardo, Antoni Mauricio, Jorge López, and Christian Del Carpio. Detection of diabetic retinopathy based on a convolutional neural network using retinal fundus images. In *International Conference on Artificial Neural Networks*, pages 635–642. Springer, 2017.

[81] Manoj Raju, Venkatesh Pagidimarri, Ryan Barreto, Amrit Kadam, Vamsichandra Kasivajjala, and Arun Aswath. Development of a deep learning algorithm for automatic diagnosis of diabetic retinopathy. In *MedInfo*, pages 559–563, 2017.

[82] Darvin Yi Carson Lam, Margaret Guo, and Tony Lindsey. Automated detection of diabetic retinopathy using deep learning. *AMIA Summits on Translational Science Proceedings*, 2017:147, 2018.

[83] Harry Pratt, Frans Coenen, Deborah M Broadbent, Simon P Harding, and Yalin Zheng. Convolutional neural networks for diabetic retinopathy. *Procedia Computer Science*, 90:200–205, 2016.

[84] Alexander Rakhlin. Diabetic retinopathy detection through integration of deep learning classification framework. *bioRxiv*, page 225508, 2018.

[85] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, et al. Deep face recognition. In *bmvc*, volume 1, page 6, 2015.

[86] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. URL: `http://arxiv.org/abs/1409.1556`.

[87] *Kaggle Diabetic Retinopathy Detection Competition*. `https://www.kaggle.com/c/diabetic-retinopathy-detection`. Last accessed 05:20 pm, April 25, 2019.

[88] Ronald Klein, Barbara EK Klein, Scot E Moss, and Karen J Cruickshanks. The wisconsin epidemiologic study of diabetic retinopathy xv: the long-term incidence of macular edema. *Ophthalmology*, 102(1):7–16, 1995.

[89] *OpenCV*. Website: `https://opencv.org/`. Last accessed 03:41 pm, April 21, 2019.

[90] *ImageMagick*. Website: `https://imagemagick.org/index.php`. Last accessed 10:28 am, April 22, 2019.

[91] AMRR Bandara and PWGRMPB Giragama. A retinal image enhancement technique for blood vessel segmentation algorithm. In *2017 IEEE International Conference on Industrial and Information Systems (ICIIS)*, pages 1–5. IEEE, 2017.

[92] Stephen M Pizer, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.

[93] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[94] Shaohua Wan, Yan Liang, and Yin Zhang. Deep convolutional neural networks for diabetic retinopathy detection by image classification. *Computers & Electrical Engineering*, 72:274–282, 2018.

[95] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[96] *FloydHub, a cloud-based deep learning platform.* Website: `https://www.floydhub.com`. Last accessed 11:02 pm, April 26, 2019.

[97] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.