Detection of Diabetic Retinopathy using Convolutional Neural Networks

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

BACHELOR OF ENGINEERING IN

INFORMATION SCIENCE AND ENGINEERING

ARJUN M	(USN: 01JST16IS008)
ASHISH N	(USN: 01JST16IS010)
ROHAN K R	(USN: 01JST16IS034)
PRASHANTH	(USN: 01JST16IS029)

Supervisor

Dr. Anand Raj Ulle

JSS SCIENCE AND TECHNOLOGY UNIVERSITY

MYSURU

2020

DECLARATION

We declare that this written submission represents our ideas in our own words and where other's ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission.

PROJECT GROUP

ARJUN M

USN: 01JST16IS008

ASHISH N

USN: 01JST16IS010

ROHAN K R

USN: 01JST16IS034

PRASHANTH

USN: 01JST16IS029

JSS SCIENCE AND TECHNOLOGY UNIVERSITY

Department of Information Science and Engineering

CERTIFICATE

This is to certify that the following students ARJUN M, ASHISH N, ROHAN K R, and PRASHANTH of JSS SCIENCE AND TECHNOLOGY UNIVERSITY, Department of Information Science and Engineering have successfully completed the project work titled "Detection of Diabetic Retinopathy using Convolutional Neural Networks " during the year 2020.

Name of the students

ARJUN M	(USN: 01JST16IS008)
ASHISH N	(USN: 01JST16IS010)
ROHAN K R	(USN: 01JST16IS034)
PRASHANTH	(USN: 01JST16IS029)

Internal Examiner	**External Examiner**	**Head of the Department**

ACKNOWLEDGEMENT

An ambition is successful only when it is carried out under proper guidance and blessings. We would like to thank few people who helped us in carrying this work by lending invaluable assistance. We express our sincere regards to His Holiness Jagadguru Sri Shivaratri Deshikendra Mahaswamiji who has showered his blessings on us for framing our career successfully.

We hereby thank Dr. T. N. Nagabhushan, Principal, JSS S & TU, Mysuru and Dr D S Vinod, HOD, Information Science and Engineering Department, JSS S&TU, Mysuru who encouraged us at this venture.

It is our foremost duty to thank our project supervisor Dr. Anand Raj Ulle, Professor, JSS S & TU, Mysuru for his encouragement, effective guidance and valuable suggestions right from the beginning of this project till its completion. We also extend our regards to all the teaching and non-teaching members of Department of Information Science and Engineering for their direct or indirect support towards the completion of this project.

ABSTRACT

Diabetic Retinopathy (DR) is one of the major causes of blindness . Increasing life expectancy, indulgent lifestyles and other contributing factors mean the number of people with diabetes is projected to continue rising. Regular screening of diabetic patients for DR has been shown to be a cost-effective and important aspect of their care. The accuracy and timing of this care is of significant importance to both the cost and effectiveness of treatment. If detected early enough, effective treatment of DR is available, making this a vital process.Classification of DR involves the weighting of numerous features and the location of such features. This is highly time consuming for clinicians. Computers are able to obtain much quicker classifications once trained, giving the ability to aid clinicians in real-time classification. The efficacy of automated grading for DR has been an active area of research in computer imaging with encouraging conclusions . We use CNN to predict DR or no DR. We develop a network with CNN architecture and data augmentation which can identify the intricate features involved in the classification task.

Diabetic Retinopathy (DR) grade classification has been regarded as a critical step for evaluation and management of diabetes retinopathy. Because of damages of the retina blood vessels caused by the high blood glucose level, different extent of microstructures, such as microanuerysms, hard exudates, and neovascularization, could occupy the retina area thus damaging it.Most grading protocols are based on classification systems for DR which track the appearance and progression of disease. Here grading helps the patients in knowing the severity of DR and to take required treatment.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1 Problem Statement

Create a reliable automated Diabetic Retinopathy Detection system which is based on Convolution Neural Networks and compare the performance with the previously available algorithms and models.

1.2 Objectives

1. Find and collect different datasets that can be used for detection of DR
2. Create a reliable automated Diabetic Retinopathy detection system.
3. Optimise the created model as efficiently as possible by tuning the number of layers, the number of features, parameters, etc.
4. Compare the created model's performance with other models available.

We, humans, have five senses but the most important of them all is the sense of vision. This is evident from the fact that out of the cerebral cortex, almost 30 percent goes to the visual processing which if we compare with other senses, just 8 percent for touch and a mere 3 percent for the processing of hearing [8]. The number of optic nerves that we have is in millions but auditory nerves count only about 30,000. Therefore, the most important sense organ that we have is the human eye, and like every other organ of the body, there are numerous ways in which its functioning can be inhibited.

The general functioning of a human eye is like an advanced camera. Light passes

through the pupil which contracts and dilates to control the intensity of light entering and then it goes to the gelatin-based lens that again has the power of expansion and contraction aided by the ciliary muscles to focus on distant and near objects respectively This power is called the power of accommodation. And then from there, the light passes through a gel-like liquid called vitreous humor which holds the retina and then falls on the film or the surface which we call Retina which is filled with cells called rods and cones that convert the coming light into electrical signals which are then transmitted through the optic nerve to the brain for further visual processing and thus, we see. Now, one important thing to note here is that out of all the parts of the eye, if we get some problem in outer parts like the cornea, the lens, and others, they are generally treatable and can be completely cured. But if there is some problem with the Retina, then it is more difficult to deal with the problem there. And the chances of completely curing the problem is minimalistic.

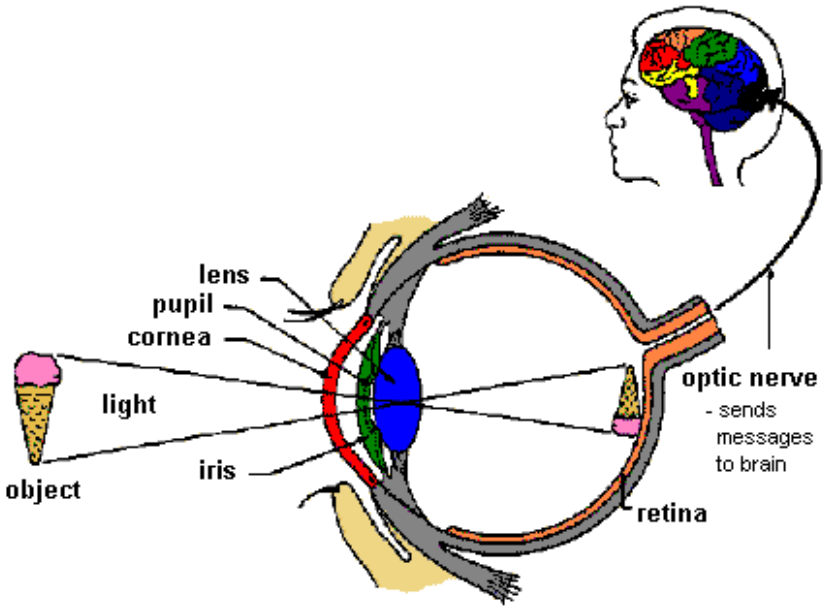


Figure 1.1: Working of The human eye [4]

The Retina can have many diseases on its one, but there are many times when it can be affected by other diseases occurring in different parts of the body. And one of the latter types is Retinopathy. According to Harvard edu [9], Retinopathy is said to be there when the disease has damaged the retina. There can be a partial or complete loss of vision And it can develop suddenly, or slowly and can get better on its own or can get worse leading to permanent damage. Many diseases can cause retinopathy and there are many causes, but one of the major causes is abnormalities in the blood vessels surrounding the retina. The different types of Retinopathy are:

- **Retinopathy of Prematurity (ROP):** It occurs in infants born prematurely or having a low birth weight. The retinal blood vessels are not fully developed yet. In the beginning stages, there are minute subtle changes but in the advanced stages, the retinal cells might not get enough energy and Oxygen and so, the retina can get completely detached leading to permanent damage. The most unfortunate fact is, there are no outward physical signs and symptoms, and therefore, only an experienced ophthalmologist can find the signs.

- **Hypertensive retinopathy:** This type of retinopathy happens because of hypertension and high blood pressure. Abnormalities due to high blood pressure include the thickening of small arteries and blockages of retinal blood vessels. There can lead to rupture of the vessels and bleeding leading to the coagulation of blood over the intercellular area. This type of retinopathy doesn't have any particular initial symptoms and can be found only during routine check-ups of the eye.

- **Central serous retinopathy:** This type of retinopathy occurs when because of any reason, there is fluid build-up and accumulation between the different layers of the retina causing blurry vision and poor night vision in the beginner phase and more advanced cases, complete detachment leading to a permanent sight loss.

- **Diabetic Retinopathy:** This type of retinopathy is found in people suffering from any of the two types of diabetes, 1 or 2. This is one of the major causes of retinopathy worldwide and after having diabetes for more than 10 years, 98 percent of people having type 1 diabetes and 78 percent of the people having type 2 diabetes are found to have some forms of diabetic retinopathy. The detection of this type of retinopathy is the topic of the paper and that's why we are going to discuss diabetic retinopathy in detail in the immediate sections.

1.3 Diabetic Retinopathy

Diabetes is a common disease these days and the trends of diabetes are increasing everywhere especially Indian Society. Diabetes has affected 422 million people worldwide; India orders among the top 3 countries in terms of the diabetic population. In recent years it has jumped from 108 million - 422 million. Out of which, half of the population are living in India, China, USA, Brazil, and Indonesia. It is a disease where the blood glucose level increases beyond a limit. The reasons can be many, either it can happen because the chemical called insulin which acts as the key to convert blood glucose to glycogen which goes inside the cells stops or reduces in production, or the cells because of some other reasons, stop or reduce consuming enough glucose. Whatever the reason is, the retention of glucose in the blood beyond a level is toxic and with diabetes, comes its complications and other associated problems. One of the problems related to diabetes is **Diabetic Retinopathy** (DR).

Diabetic Retinopathy, as discussed earlier is one of the major types of retinal problems. In Diabetic Retinopathy, broadly what happens is retinal nerve fiber may swell, leading to blurry vision and white spots in the retina. With the time, new and weaker blood vessels grow adding to the uneven surface and changes in vision blur patterns and threatening the vision, but ultimately too much blood sugar can block up many of the

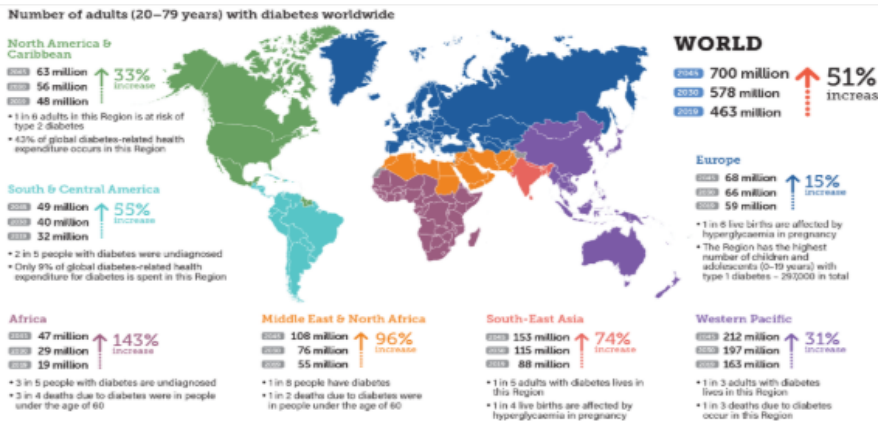


Figure 1.2: Global Trends in Diabetes [5]

tiny blood vessels and lead to non-functioning of the retinal cells and apart from that, many of the vessels might burst leading to fluid leakages in between the retinal layers and chronic retention of the condition can lead to permanent vision loss. In the meantime evaluation of contemporary medical imageries remains complex for the manual detection of the disease and hence, in recent years, Computer Vision is emerging as the helping hand towards early detection.

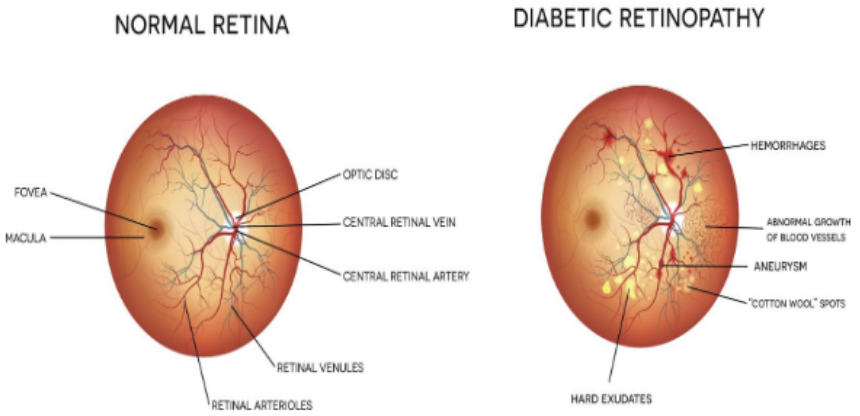


Figure 1.3: Comparison of a normal retina and DR retina

1.3.1 Types and stages of Diabetic Retinopathy

Depending on the conditions, there are different types or stages of Diabetic Retinopathy.

- **Non-proliferative DR:** In non-proliferative diabetic retinopathy, the blood vessels originally present in retina deteriorate. They can become blocked, deformed, or leak. Different substances including, blood plasma and corpuscles, proteins, fats, sugar, cholesterol, etc. will leak out and spread in between the layers of the retina or even into the vitreous humor and this condition seriously obscures the vision. The collected fluid and other particles can swell the macula and sharply impairs the vision. The formation of microaneurysms (MAs) is generally seen in this stage.

- **Proliferative DR:** This can act as the repercussion of Non-proliferative DR. In proliferative diabetic retinopathy, new weaker, abnormal, and structurally unstable blood vessels called IrMAs (Intraretinal Microvascular Abnormalities) start forming on the surface of the retina to cope up with the lessened supply of blood into the retinal cells and these unstable blood vessels keep rupturing causing minor bleeding, coagulation, blurry vision spots, irritation, and scarring. Proliferative retinopathy can cause scathing consequences like detachment of retinal layers leading to permanent vision loss.

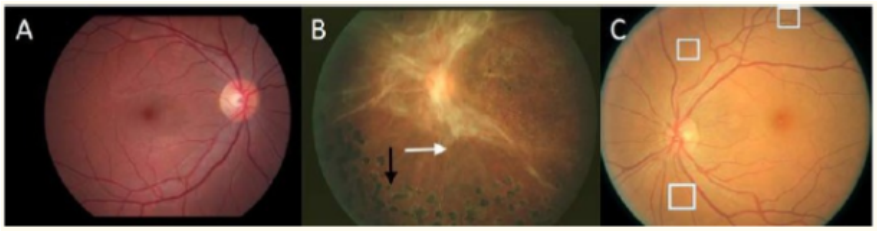


Figure 1.4: Different stages of development of DR. A- Non-proliferative Stage, B- Proliferative stage, C- Early stage

1.3.2 Symptoms

The sad truth about Diabetic Retinopathy is the symptoms may not be visible until the late stages of the disease and therefore, the only way to stop it is its early detection for which manual routine checkup is required. The general symptoms in the late stages are:

- Sudden loss of vision in one or both eyes
- Blurred vision
- Black spots
- Flashing lights
- Difficulty reading or seeing detailed work

1.3.3 Prevention

The general methods of prevention are the same as that of checking and prevention of diabetes. Keeping the blood sugar level in control, blood pressure needs to be in check, and regular doses of the required medicines are necessary. Apart from these, the most crucial part is the routine checkup of the symptoms. It can be done yearly or half-yearly. Early detection of the disease can seriously inhibit the growth of the disease and if treatment is started before the sight is affected, vision loss can be prevented.

1.3.4 Detection and Diagnosis

Early detection which is required for a good prognosis, is usually dependent on skilled and experienced ophthalmologists and hence, is prone to human errors. Therefore, to solve the above problem, we have come up with our model for the detection of Diabetic Retinopathy using Convolutional Neural Network. The general steps are **taking**

images of the retina after dilating the pupil of the affected eye and then passing those images through the pre-trained model to classify.
Once there is early detection of the disease, there are numerous methods for the diagnosis of the same. Some of the methods are:

- **Laser Photocoagulation:** Using lasers, scars are created around the IrMAs (the newer blood vessels) to inhibit their growth.
- **Vitrectomy:** Removal of all or part of the affected vitreous humor after it contains fluids that obscure the vision.
- **Reattachment of Macula:** Surgical reattachment of retinal layers after they get detached from each other.

Apart from these methods, there still are other ways like injecting corticosteroids and other VEGF(Vascular Endothelial Growth Factors) inhibitors into the eye to keep the problem in check and control.

1.3.5 Importance of detection in the early stages

This is the most important logic around which our whole project and every other work on detection or diagnosis of Diabetic Retinopathy revolves. As we have seen in the previous subsection, the diagnosis of the ailment is not exactly a diagnosis but rather controlling the symptoms. Whether laser coagulation to form scars that slow down the growth of IrMAs or Vitrectomy to remove the translucent humor or reattachment of the macula or injection of other substances to inhibit the growth of VEGFs and IrMAs, these all are different approaches to control the symptoms and hence, different ways to postpone the inevitable, which is morbidity and in this case, loss of vision.
But we also know that the aforementioned controls on different symptoms can be only applied once the symptoms are detected which is possible only when it is too late.

Hence, if there are ways by which the ailment is detected in its infancy and earlier phases, it can be controlled even before it starts imparting the symptoms on the patients. Diabetic Retinopathy takes more than ten years to develop completely in general cases and hence if diabetic people keep getting routined monitoring of their eyes, it might be detected and hence with the knowledge of the existence of this ailment, they will keep their diabetes in check and thus leading to safety of their vision as well.

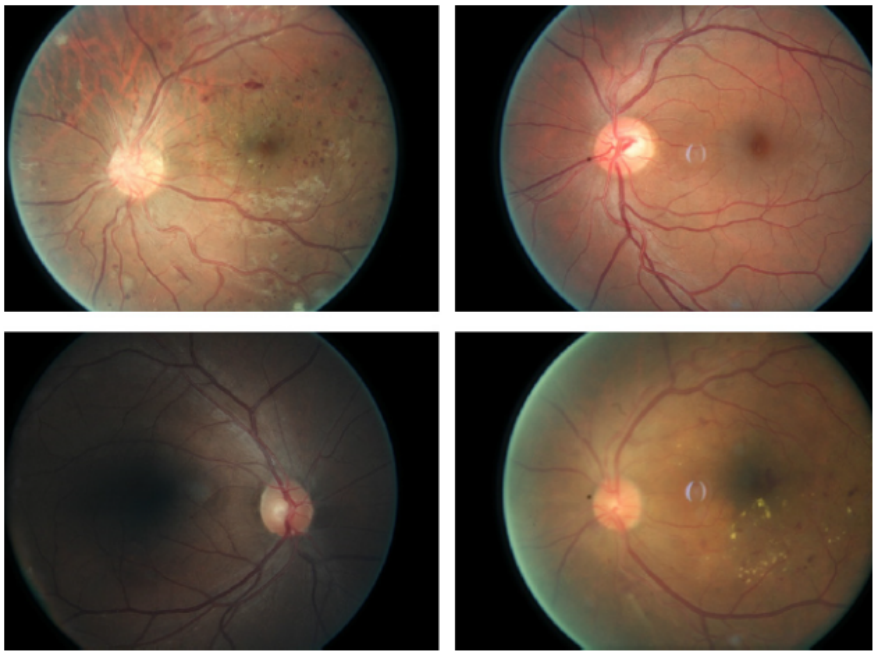


Figure 1.5: The above two images are of a proper retina and the lower two images of DR retina

1.4 Introduction to Deep Learning Approach

Deep Learning, as a branch of Machine Learning, employs algorithms to process data and imitate the thinking process, or to develop *abstractions*. Deep Learning (DL) uses layers of algorithms to process data, understand human speech, and visually recognize objects. Information is passed through each layer, with the output of the previous layer

providing input for the next layer. The first layer in a network is called the input layer, while the last is called an output layer. All the layers between the two are referred to as hidden layers. Each layer is typically a simple, uniform algorithm containing one kind of activation function. Apart from these, feature extraction is one of the major aspects of Deep Learning. Different features are extracted from the input using various methods to use it for further processing. There are various deep learning approaches but we have used the approach of **CNN (Convolutional Neural Network)**.

1.4.1 Convolutional Neural Network (CNN)

A Convolutional Neural Network (ConvNet/CNN) is a class of Deep Learning algorithm that is commonly applied to images or data that possess Spatial and temporal dependencies. The architecture of a CNN is similar to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields' overlaps to cover the entire visual area. The main reasons for the success of Convolutional Neural Network over Feed-Forward Neural Networks is its ability to capture the Spatial and Temporal dependencies and better fitting to the dataset due to less number of parameters involved. They are applied to several different tasks like image classification, image segmentation, object detection, image captioning, image colorization, etc. They are now used in different fields as well e.g. medical, security, automation, anomaly detection, etc. The different components of a Convolutional Neural Network are:

- **The Convolution Layers:** In image processing, an input image is convoluted using a feature detector, which is also called a kernel, and the output is generated which is known as Feature Map. Figure 1.6 shows the convolution between an input image matrix and a feature detector and explains the generation of the

Feature Map. The feature detectors or kernels are used to detect specific features in an image e.g. horizontal edge, vertical edge, color, shape, etc. Then these features are used for the further training of the CNN model.

- **Activation function:** After the convolution layer, activation functions are applied to the feature map to introduce non-linearity in the system. There are many different activation functions in deep learning like sigmoid, ReLU (Rectified Linear Unit), Tanh, softmax, sigmoid, etc.

- **The Pooling Layers:** The pooling layer helps to reduce the size of the feature map while maintaining the information in the feature map. It will reduce the number of parameters when the images are two large. The reduction in the size of the feature map will lead to a reduction in the computation power to calculate the next feature map as the number of convolution operations has decreased. In pooling, we take a kernel square kernel of m x m size and stride greater than 1 and we convolve that kernel over the input feature map. There are two types of pooling layers, Max Pooling where we select the maximum element from the portion of the image covered by the kernel and Average Pooling where we select to take the average of the elements from the portion of the image covered by the kernel.

- **The flattening Layer:** After the above-explained operations are performed on an input image, the next operation is flattening of the feature maps. The feature maps are flattened to make a one-dimensional array. Figure 1.6 shows an example of the flattening of a feature map. The process of flattening is performed because of the need to insert the data into the fully connected layer later on. As we can see, we have multiple pooled feature maps from the previous operation. After flattening, we end up with a long vector of input data that we then pass

through the fully connected layer to have it processed further.

- **Fully Connected Layer:** The fully connected layer consists of layers such that each neuron in the last layer is connected to each neuron in the next layer. The fully connected layer is an ANN (Artificial Neural Network) which does the job of predicting the output from among a set of classes.

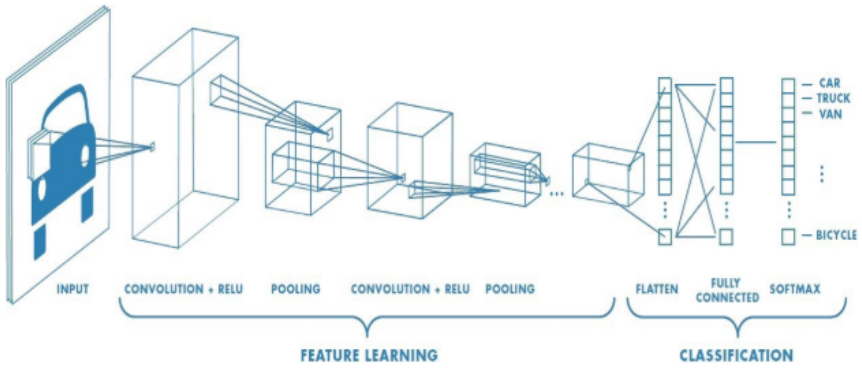


Figure 1.6: A Convolutional Neural Network Depiction [6]

1.5 Related Works

During recent years, there have been numerous studies and works on the automated detection or diagnosis of Diabetic Retinopathy. [10] proposed a method for automated detection of DR using various deep learning approaches like DNN (Deep Neural Network) and CNN (Convolutional Neural Network) after preprocessing the images and extracting certain features beforehand. [11] proposed several machine learning techniques like PNN (Probabilistic Neural Network), Bayesian Classifier, and SVM for the automated detection of DR. C. [12] worked on recognition of DR using CNN on retinal fundus image datasets. [13] have worked on the identification and recognition of DR using Deep Neural Networks on raw fundus image datasets. [2] have worked on

the production of the IDRID (Indian Diabetic Retinopathy Image Dataset) database for the further use of these images for the same purpose. [14] have worked on automatic detection of diabetic retinopathy exudates from non-dilated retinal images using mathematical morphology methods. [15] presented a method to classify diabetic retinopathy subjects from changes in visual evoked potential spectral components. According to [16] exudates are found using their high grey level variation, and their contours are determined through morphological reconstruction techniques. [17] focus on developing a computer-aided detection mechanism for finding abnormality of the retinal imaging while detecting the existence of abnormal features from the retinal fundus images. Their proposed methodology focuses on enhancing images, filtering of the noise, detection of the blood vessels and identifying optic disc, extracting exudates and the micro-aneurysms (MA), extracting features and classifying various stages of the diabetic retinopathy as mild, the moderate, the severe NPDR(Non-Proliferative Diabetic Retinopathy) and the PDR(Proliferative Diabetic Retinopathy) by the use of machine learning techniques. [18] have performed a secondary study and compared multiple methods of DR image detection. From the paper, a full overview of multiple techniques regarding DR image detection has been found. The paper tabulated possible technique and their accuracy results very precisely.

1.6 Observation

From the literature review, what we observed is that there has been a plethora of work on this particular topic, and there have been outstanding results in some cases as well. There are papers and researches on detection and diagnosis of Diabtic retinopathy using various methodologies, mathematical, intuitive, illustrative, using AI, machine Learning, Deep Learning, even in the last two, there has been a combination of all the methodologies possible and present now. Recently with the boom in GPU and hardware tech-

nology and speed and thereby public usage increase in Deep Learning, and with the help of free cloud services like Google Colab, github, etc., the development in every possible software field has increased manifolds. So, what we have done which we felt others have not done is:

1. We have performed preprocessing in the used datasets: The publically available datasets are highly heterogeneous as they are from different people all over the world, and they are taken from different kinds of cameras, different qualities, different angles, resolutions, sizes,etc and so, if they are used raw, it might lead to erroneous results as it is seen in different papers. Therefore, we have performed preprocessing, that is data augmentation, resizing and uniform downsizing.

2. The second thing that we have done is experimenting with our own CNN model rather than directly going for pretrained models like VGG 16. We have done work on our own designed convolutional network layer model and hence, there were a lot of trial and error tuning layers, parameters, etc, before we reached a good recall , precision and accuracy value. Additionally, we have performed the work on different other algorithms as well.

1.7 Motivation

As it has been discussed earlier, the disease that we were discussing about and are going to research over, is a blindingly dangerous one, but the only silver lining is that it takes ages to grow to its morbid form and if detected earlier, there are very high chances that it can be controlled or even completely treated. The downside is that the detection of disease, in the olden times had to be done manually by experienced ophthalmologists and that was a very hectic process, both for the doctor as well as the physician, and by the time there were disease symptoms starting to be shown, the disease already would

Table 1.1: Comparison of all the papers and our work

Name of paper	**Preprocessing**	**Model**	**Accuracy %**
[10]	yes	Artificial Neural Network	89.6
[14]	yes	Convolutional Neural Network over VGG 16	94.5
[19]	no	SVM, heatmap	83.4
[12]	yes	CNN, Transfer Learning	71.2
	no	Artificial Neural Network	78
[3]	no	ANN and Boltzmann Machines	68
[20]	yes	Vessel segmentation(semantic)	94.69
[21]	no	Quantitative Analysis	94.8
[22]	No	Feature Extraction	96.16
[11]	no	Bayes	87.18
[23]	yes	Semantic Segmentation	93.0
Our Model	**Yes**	**CNN 36 layers, 2 filters, SOFTmax**	**100.0**

have reached a deadly stage.

But, with the advancements of AI, Machine Learning, and now with the rapid increase in the quality of image processing and computer vision by the use of Deep Learning, there are very high chances that this disease can be detected automatically without waiting for months for the result to come out.And in this way, it can help the humankind save a lot of potential loss of vision, hence lives and property. In one sentence, our motivation is "**Computer Vision is saving Human Vision**".

Chapter 2

Methodology

As discussed earlier, there has been an ample amount of work done on this particular topic but the problem with the previous works is that due to the lack of preprocessing steps since the data is highly heterogeneous in format, there are more conclusions with erroneous results. Heterogeneity means that the obtained images are of different sizes, resolutions, formats, and types since the affected eyes are of many patients belonging to different creeds and races, and the cameras would also have been different.

Hence, to bring everyone to train under the same model, preprocessing is required. And therefore realizing this, we have proceeded initially with preprocessing steps. Hence, our general workflow is, taking the heterogeneous images from FUNDUS and IDRID datasets, preprocessing the images and making them homogenous, then extracting features from the figures for further processing and classification and then passing these features through our architectural model and finally getting the classes prediction as the outputs. Our architectural model consists of a Convolutional Neural Network whose components have been changed time and again and have tuned the parameters until we have reached the best accuracy we could. A tabular methodology flow is given below.which will be followed by a flow block diagram of the whole project. Then we will elaborate on each section and algorithm. Let's briefly look at the different components involved in the project.

Table 2.1: Methodology Flow

Steps	**Description**
Step 1	Preprocessing of the heterogeneous images dataset. The preprocessing steps involved resizing the images and downsizing all images to 256 x 256 images. and then Contrast adjustment using the histogram equalization filtering algorithm CLAHE.
Step 2	Feature extraction, section, and classification. We have used GLCM and HOG to extract features. So, based on this method, we have got features for the entire image.
Step 3	Passing the extracted features through our initial architectural model of CNN.
Step 4	Performed Convolutional Neural Network (CNN) on processed images. (VGG16, VGG 19 models)
Step 5	Tuning the architecture, number of layers, and updating the models repeatedly until the desired accuracy is achieved.
Step 6	Analysis of all the results of Step 3, Step 4, and Step 5 has been done as per performance and accuracy measurements with the testing image set.

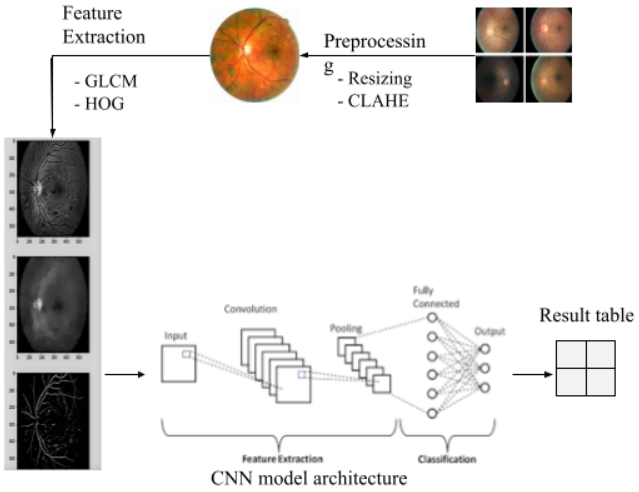


Figure 2.1: The project flow diagram

Let's go over each step in brief.

2.1 Datasets

We have used two very heterogeneous sets of datasets, the description of each of them are given below.

2.1.1 Fundus Retinal Dataset [1]

This database has been established by a collaborative research group to support comparative studies on automatic segmentation algorithms on retinal fundus images. The total number of patients taken for the same was 18 and so, 18 image pairs of the same eye from 18 human subjects using a Canon CR-1 fundus camera with a field of view of 45° and different acquisition setting. For each pair, the first image has poor quality and thus the examination had to be repeated. Both images share approximately the same field of view, whereas small shifts were caused by eye movements between the acquisitions. There were 1000 images out of which 500 were taken for the experimen-

tation.And there were 39 different labels on the photos. We converted them to two labels: DR or No DR. The reason to do this was that many other category images were of no use.

2.1.2 IDRID Dataset [2]

IDRID stands for Indian Diabetic Retinopathy Image Dataset. It is an open and free dataset for research on Diabetic Retinopathy. It was published in July 2018. Images were acquired using a Kowa VX-10 alpha digital fundus camera with a 50-degree field of view (FOV), and all are centered near to the macula. The images have a resolution of 4288×2848 pixels and are stored in jpg file format. The size of each image is about 800 KB.The number of images were:

Table 2.2: IDRID dataset

	Normal	DR
Training	136	279
TestingTesting	29	90

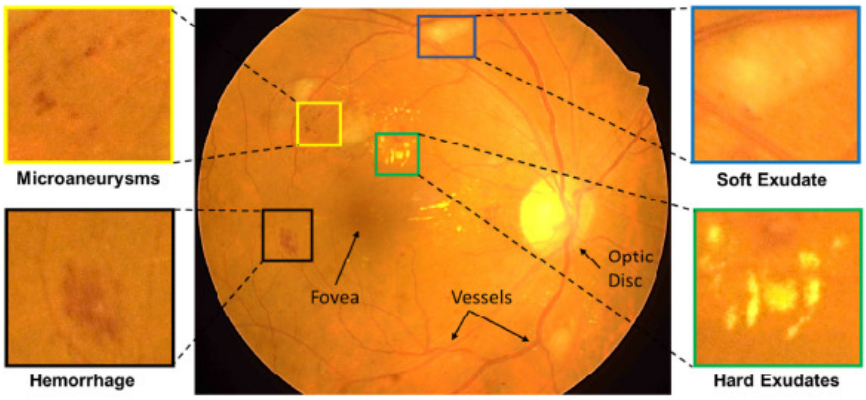


Figure 2.2: Color fundus photograph containing different retinal lesions associated with diabetic retinopathy. Enlarged parts illustrating the presence of Microaneurysms, Soft Exudates, Hemorrhages, and Hard Exudates

Therefore, a total of 469 images were there combining both the IDRID and FUNDUS images. The training and testing split was based on 70:30 ratio.

2.2 Preprocessing

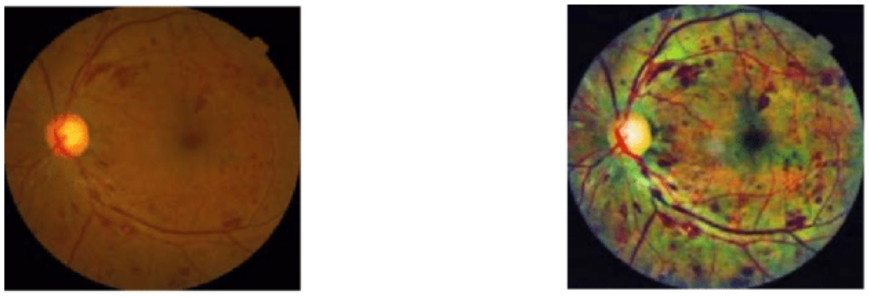
This is the part which we have done and many other works have not done. Preprocessing is an important part of Deep Learning as not doing it leads to erroneous results.Since the images were highly homogenous in their resolution, that of IDRID images being in the range of 800KB and that of other fundus images more variable in size, hence, they needed to be resized by downsizing. Hence, the different steps in preprocessing are:

2.2.1 Resizing

Every image was downsized to 256 x 256 pixels in resolution to maintain the homogeneity of the dataset. For the resizing, we used python-resize-image form the PyPI.

2.2.2 Contrast Adjustment

The contrast adjustment was done using the CLAHE [3] histogram equalization algorithm.CLAHE stands for Contrast limited adaptive histogram equalization. CLAHE is a variant of *Adaptive histogram equalization (AHE)* which takes care of over amplification of the contrast. CLAHE operates on small regions in the image, called tiles, rather than the entire image. The neighboring tiles are then combined using bilinear interpolation to remove the artificial boundaries. This algorithm can be applied to improve the contrast of images. We can also apply CLAHE to color images, where usually it is applied on the luminance channel and the results after equalizing only the luminance channel of an HSV image are much better than equalizing all the channels of the BGR image.



(a) (b)

Figure 2.3: FUNDUS image before and after applying CLAHE

2.2.3 Augmentation

Data augmentation is a strategy that enables practitioners to significantly increase the diversity of data available for training models, without actually collecting new data. Data augmentation techniques such as cropping, padding, and horizontal flipping are commonly used to train large neural networks. Initially Fundus consisted of 1000 im-

ages which had 39 categories in that only 500 images was taken .Later we made it into two categories that is DR and no DR because some of the categories were of no use.

2.3 Feature extraction

Whenever a data scientist plays with a dataset made of a high number of variables, there are pretty good chances that some of them are redundant or noisy. That means such variables do not really carry any signal that might come useful as a predictor. Noise, as always, will affect the overall accuracy of any predictive model. In such scenarios, feature extraction allows data scientists to construct a new representation of the original data. This facilitates the detection of potentially interesting patterns. A deep neural network trained to recognize people from a large set of images, will show a number of features in its layers. From the first layers such features become more and more complex and abstract. In computer vision such a complexity goes from pixels, blobs, eyes, noses, faces, until clothes and entire scenes. Of course specific neurons will activate for each of these abstract concepts. Some of the benefits of feature extraction in general are:

- **Fast training:** Training a simple model can take minutes instead of hours or days. This allows to quickly test hypotheses and increase productivity
- **Good performance:** One can build accurate models on relatively small datasets.
- **Adaptability to existing pipelines:** A specific modelling framework can improve model performance without changing their pipelines entirely. They would only need a preprocessing step performed with the *pre-trained* model.

Keeping in mind the above features, we decided to move on with feature extraction based classification. We used two methods, GLCM and HOG for this purpose. Let's look at both of these in brief.

2.4 GLCM [3]

GLCM stands for Grey-Level Co-occurrence Matrix. It is a statistical method of examining texture that considers the spatial relationship of pixels as the gray-level co-occurrence matrix (GLCM), also known as the gray-level spatial dependence matrix. The GLCM functions characterize the texture of an image by calculating how often pairs of pixels with specific values and in a specified spatial relationship occur in an image, creating a GLCM, and then extracting statistical measures from this matrix. (The texture filter functions, described in Calculate Statistical Measures of Texture cannot provide information about shape, that is, the spatial relationships of pixels in an image.) erpretation difficulty. A GLCM is a matrix where the number of rows and columns is equal to the number of gray levels, G, in the image. The matrix element $P(i, j \mid \triangle x, \triangle y)$ is the relative frequency with which two pixels, separated by a pixel distance $(\triangle x, \triangle y)$, occur within a given neighborhood, one with intensity 'i' and the other with intensity 'j'. The matrix element $P(i, j \mid d, \theta)$ contains the second order statistical probability values for changes between gray levels 'i' and 'j' at a particular displacement distance d and at a particular angle θ. Using a large number of intensity levels G implies storing a lot of temporary data, i.e. a $G \times G$ matrix for each combination of $(\triangle x, \triangle y)$.

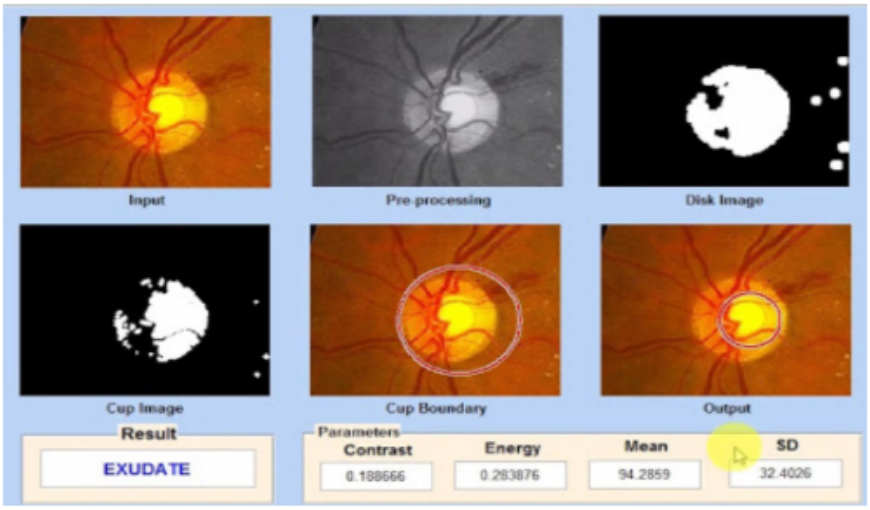


Figure 2.4: GLCM texture analysis and feature extraction

2.4.1 HOG

HOG stands for Histogram of Oriented Gradients. HOG, or Histogram of Oriented Gradients, is a feature descriptor that is often used to extract features from image data.The HOG descriptor focuses on the structure or the shape of an object.In the case of edge features, we only identify if the pixel is an edge or not. HOG is able to provide the edge direction as well. This is done by extracting the **gradient and orientation** (or you can say magnitude and direction) of the edges. Additionally, these orientations are calculated in **'localized' portions**. This means that the complete image is broken down into smaller regions and for each region, the gradients and orientation are calculated. We will discuss this in much more detail in the upcoming sections. Finally the HOG would generate a **Histogram** for each of these regions separately. The histograms are created using the gradients and orientations of the pixel values, hence the name 'Histogram of Oriented Gradients'. Hence, The HOG feature descriptor counts the occurrences of gradient orientation in localized portions of an image. The different steps involved in HOG are:

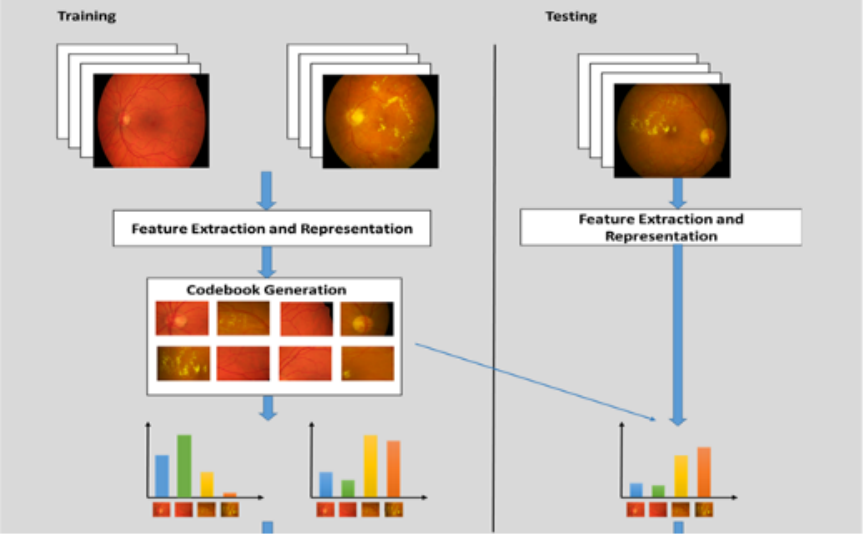


Figure 2.5: Feature extraction using HOG [7]

1. Calculation of x and y gradients
2. Calculation of magnitude and orientation
3. Calculate Histograms of cells using the magnitude
4. Normalise the gradients in 16 x 16 cells
5. Features for the complete image

2.5 Our CNN model and the results in different models

After extracting the features and creating the histograms from the HOG feature extraction. Now, it was time to send them to the model for training. The convolutional neural network (CNN) belongs to the feed-forward artificial neural network (ANN), which is very similar to ordinary neural networks. The CNN is a well-known deep learning architecture, in which individual neurons are tiled in such a way that they respond to overlapping regions in the visual fields. CNNs are an important class of learnable

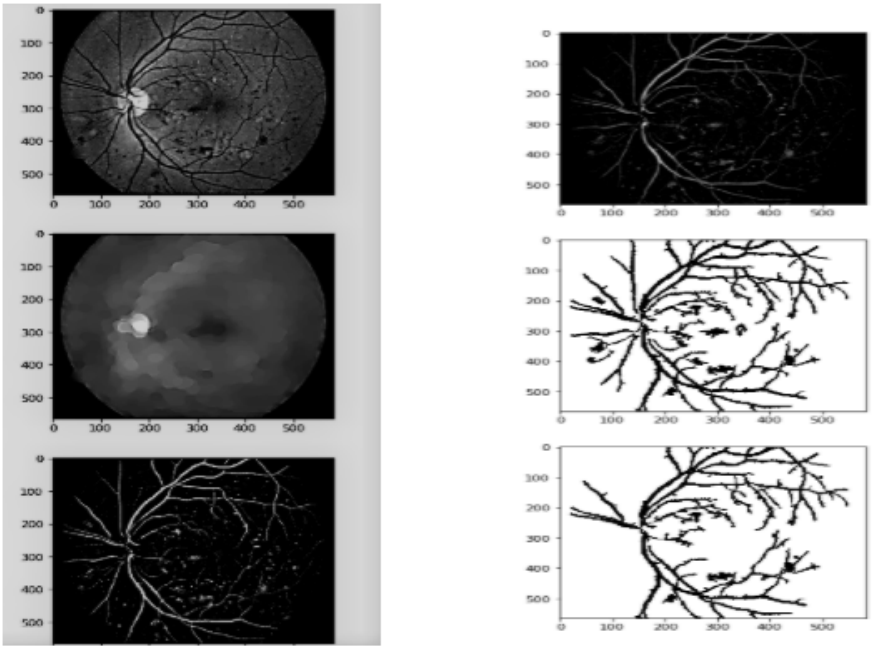


Figure 2.6: Feature extraction using GLCM and HOG in our project

representation applications, and they were inspired by biological neural networks. Numerous variants have been proposed over the last several years. However, the basic components are very similar. CNNs consist of alternating convolution and pooling operations. Typically, the convolutional layers are interspersed with pooling layers to reduce computation time, and build up further spatial and configuration invariance; the last few layers (close to the outputs) will be fully connected 1-dimensional layers. In more detail, a feed-forward neural network can be viewed as a function f of mapping data x:

$$f(x) = f_L(\cdots f_2(f_1(x_1, w_1), w_2)\cdots, w_L) \tag{2.1}$$

Each function f_l takes x_l (x_1 is the input data x) as input with a learnable parameter vector w_l. L denotes the depth of the neural network. Although the type and sequence

of functions are usually handcrafted, the parameters can be discriminatively learned from example data such that the resulting function f realizes a useful mapping. Formally, in a CNN, each x_l will be a $M \times N \times C$ array. As our problem can be simplified as a binary classification problem, we can define the loss function of the CNN as:

$$L(w) = \frac{1}{n}(n\dot{a}i = 1l(z_i, f(x_i; y))) \tag{2.2}$$

where n is the number of samples, z_i is the true label of sample i. The training problem can be converted to training a neural network to minimize the loss function L. Figure (**REF**) gives the general architecture of a CNN network, which consists of multiple layers of small neurons. The results of these collections are then tiled so that they overlap to obtain a better representation of the original image (such as edges in the image). Convolutional layers consist of a rectangle grid of neurons, which takes a rectangle region of the previous layer as input. Moreover, there may be several grids in each convolutional layer, using potentially different filters. Typically, there is a pooling layer after each convolutional layer, which are subsampled from the previous convolutional layer. This pooling can be carried out in several ways, such as the average, maximum, etc. Finally, after several convolutional layers and max pooling layers, a fully connected layer (or several layers) will be built using outputs from previous layers (maybe the fully connected, pooling or convolutional layer), which is used as a compact feature to describe the whole input image. The network is optimized by backpropagation and stochastic gradient descent.Note that the forward and backward propagations may differ depending on the type of the layer. Several different CNN architectures have been proposed and evaluated in our experiments.The depth of the tested neural network ranges from 9–18, and the convolution kernel size ranges from 1 to 5. To fit the input size of the CNN, we resize the image size to 224 224 3. The final architecture of the network used in our work is given in Table (**REF**). For a given input, the network outputs two

probabilities that sum up to 1, one for each class (our problem is a binary classification problem). In our experiment, 800 labeled images are used to train the neural network, while 200 images are used to evaluate the performance of the trained neural network. The architecture of the CNN model that we used was:

Table 2.3: Initial CNN model

Output Shape	**Description**
256 x 256 x 3	input
254 x 254 x 32	3 x 3 Convolution Layer, 32 filters
4096	1 Batch Normalization layer
1024	1 Activation Layer
1	1 Dense Layer 90 neurons

But since the results were poor,hit and trials and tuning our models and parameters were done,and did augmentation of the datasets.The whole result is summarised below.
The following table (2.4) shows the results of the initial model. The architecture of the network was having three convolutional layers having 64 filters, 32 filters and 16 filters each having a kernel size of 3 alternately with 2 x 2 pooling layers and the dataset was Fundus. The output came as two classes 0 and 1 specifying not DR and DR and the other values accuracy, precision, recall, f1-score and support accordingly.

Table 2.4: Outputs on CNN Fundus (64,32,16)

	Precision	Recall	f1-score	Support
Class 0	0.15	0.88	0.26	8
Class 1	0.89	0.27	0.29	47
Accuracy	98.97	98.97	0.27	55
Macro avg	0.52	0.52	0.27	55
Weighted avg	0.78	0.27	0.28	55

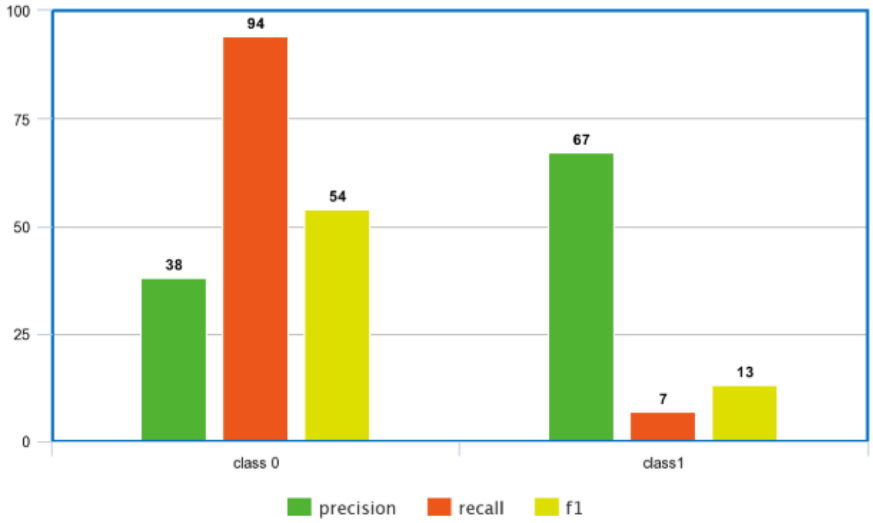


Figure 2.7: Outputs on CNN Fundus (64,32,16)

Table 2.5: Confusion Matrix for CNN Fundus(64,32,16)

	Predicted DR	Predicted not DR
Actual DR	7	1
Actual not DR	39	8

Since, some of the values of the results are not very promising in the initial set, hence, we tried with different architectures .

The following table (2.6) shows the results on our initial model. The difference is the dataset is augmented in this case. The architecture of the network was having three convolutional layers having 64 filters, 32 filters and 16 filters each having a kernel size of 3 alternately with 2 x 2 pooling layers and the dataset was Fundus. The output came as two classes 0 and 1 specifying not DR and DR and the other values accuracy, precision, recall, f1-score and support accordingly.

Table 2.6: Outputs on CNN Fundus augmented(64,32,16)

	Precision	Recall	f1-score	Support
Class 0	0.38	0.94	0.54	34
Class 1	0.67	0.07	0.13	56
Accuracy			0.40	90
Macro avg	0.52	0.51	0.34	90
Weighted avg	0.56	0.40	0.29	90

Table 2.7: Confusion Matrix for CNN on IDRID(64,32,16)

	Predicted DR	Predicted not DR
Actual DR	32	2
Actual not DR	52	4

Apart from CNN, we have trained and tested the datasets over different other algorithms, so that a comparison can be made with our own model.
In (2.8), there are outputs from logistic regression algorithms on the IDRID datasets.

Table 2.8: Confusion Matrix for Logistic Regression on IDRID(64,32,16)

	Predicted DR	Predicted not DR
Actual DR	5	20
Actual not DR	11	47

After this, we tuned the CNN model itself and kept on updating it until a desirable result was found. We mixed both the FUNDUS and IDRID. The outputs of various hit and trials on CNN are given below:
Here, in (2.9), there is comparison of the outputs of CNN model layer variations and dataset variation. We have used two layers of CNN, three layers and then the combination of IDRid and FUNDUS datasets over the same.

Table 2.9: Outputs on CNN layer trials

Algorithm	Accuracy	Precision	Recall
FUNDUS 2 layers (filters: 16,32)	81	14.54	100
IDRID 2 layers (filters: 16,32)	68.21	60.8	73.52
FUNDUS and IDRID 2 layers (16,32)	74.09	56.52	76.19
FUNDUS (filter 64,32,16)	98.97	88.88	87.5
IDRID (filters 32, 32, 16)	81.03	100	100
IDRID and FUNDUS (filters 32, 32,16)	75	73.8	50.81

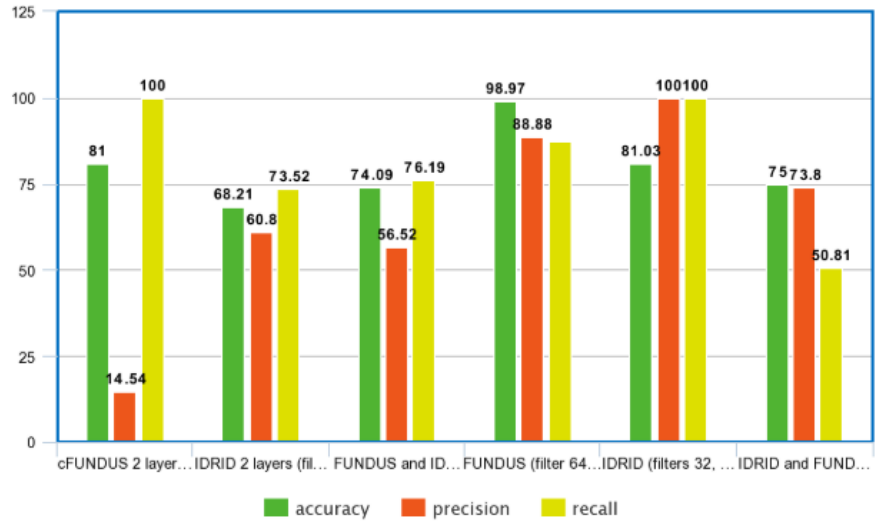


Figure 2.8: Outputs on CNN Layer Trials

In (2.10), we have shown the output of CNN 2 layers that are two convolutional layers each having 32 and 16 filters respectively in alternation with 2 x 2 pooling layers. The number of neuron layers is the same i.e. 100 and the number of epochs is 25.

Table 2.10: FUNDUS 2 LAYERS (FILTERS 16,32)

	Precision	Recall	f1-score	Support
Class 0	0.15	1.00	0.25	8
class1	0.00	0.00	0.00	47
accuracy			0.15	55
Macro avg	0.07	0.50	0.13	55
Weighted avg	0.02	0.15	0.04	55

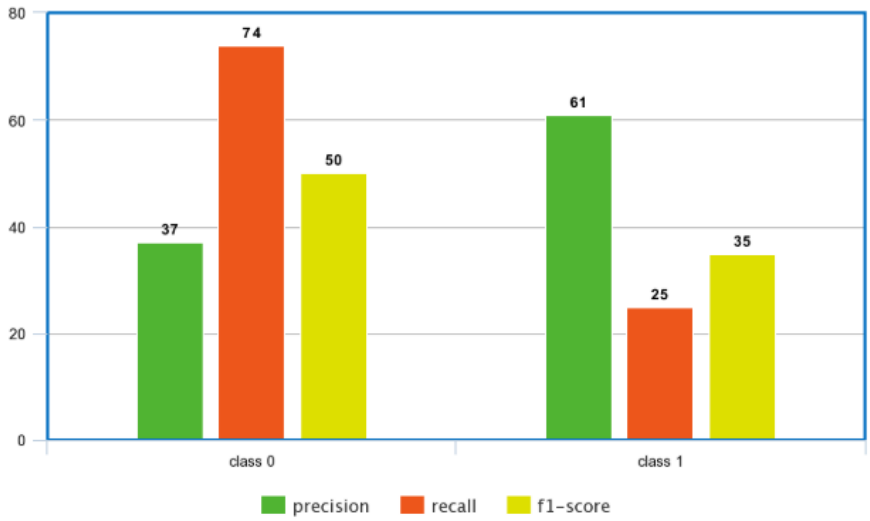


Figure 2.9: IDRID 2 LAYERS (FILTERS16,32)

Table 2.11: Confusion Matrix for FUNDUS 2 LAYERS (FILTERS 16,32)

	Predicted DR	Predicted not DR
Actual DR	8	0
Actual not DR	47	0

Similarly, in (2.12), we have shown the outputs of the CNN model on two convolution layers like in the tabe 17, on the IDRID dataset.

Table 2.12: IDRID 2 LAYERS (FILTERS 16,32)

	Precision	Recall	f1-score	Support
Class 0	0.37	0.74	0.50	34
class1	0.61	0.25	0.35	56
accuracy			0.43	90
Macro avg	0.49	0.49	0.42	90
Weighted avg	0.52	0.43	0.41	90

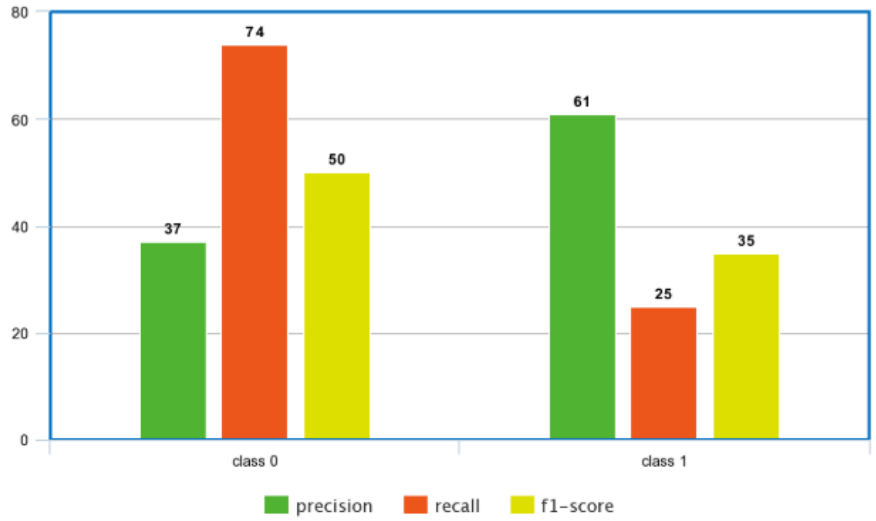


Figure 2.10: IDRID 2 LAYERS (FILTERS 16,32)

Table 2.13: Confusion Matrix for IDRID 2 LAYERS (FILTERS 16,32)

	Predicted DR	Predicted not DR
Actual DR	25	9
Actual not DR	42	14

After going for them datasets IDRID and FUNDUS separately, we combined both and performed CNN two layers on them, with the same specifications of the model. And the accuracy has decreased a bit.

Table 2.14: FUNDUS AND IDRID 2 LAYERS (FILTERS 16,32)

	Precision	Recall	f1-score	Support
Class 0	0.36	0.76	0.49	42
class1	0.57	0.19	0.28	69
accuracy			0.41	111
Macro avg	0.46	0.48	0.39	111
Weighted avg	0.49	0.41	0.36	111

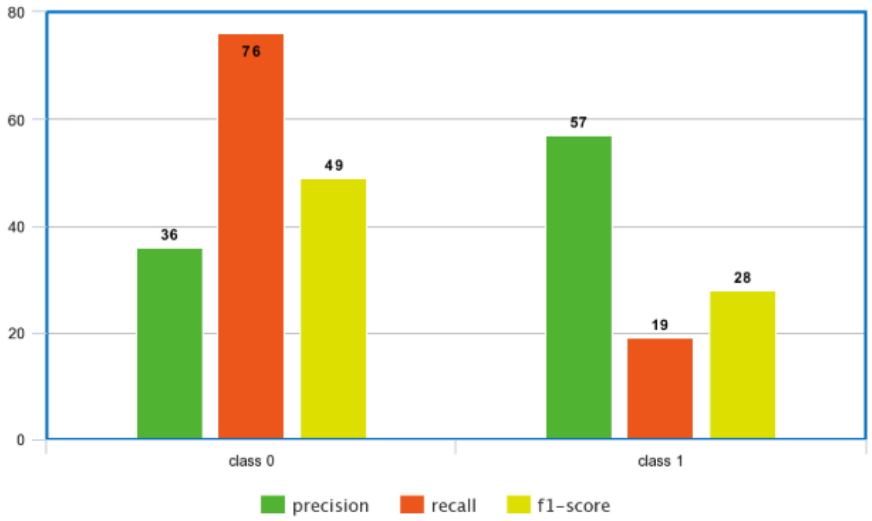


Figure 2.11: FUNDUS AND IDRID 2 LAYERS (FILTERS 16,32)

Table 2.15: Confusion Matrix for IDRID and FUNDUS 2 LAYERS (FILTERS 16,32)

	Predicted DR	Predicted not DR
Actual DR	32	10
Actual not DR	56	13

After two layers, we again moved to CNN 3 layers i.e. convolutional layers having 64,32, and 16 filters each with kernel size 3 separated by 2 x 2 pooling layers. The (2.16) shows the outputs for the fundus image datasets on the same architecture.

Table 2.16: FUNDUS (FILTERS 64,32,16)

	Precision	Recall	f1-score	Support
Class 0	0.15	0.88	0.26	8
class1	0.89	0.17	0.29	47
accuracy			0.27	55
Macro avg	0.52	0.52	0.27	55
Weighted avg	0.78	0.27	0.28	55

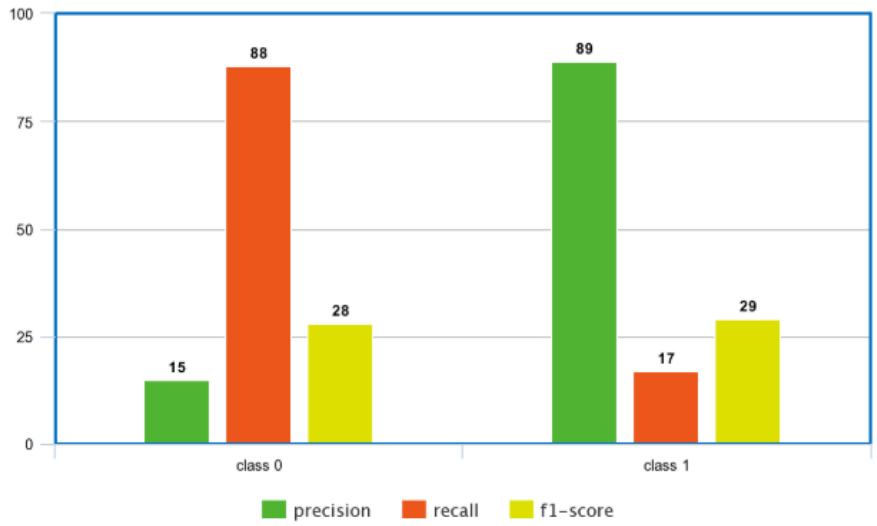


Figure 2.12: FUNDUS (FILTERS 64,32,16)

Table 2.17: Confusion Matrix for FUNDUS (FILTERS 64,32,16)

	Predicted DR	Predicted not DR
Actual DR	7	1
Actual not DR	39	8

Again, there is a change of architecture. Instead of 64 filters, we applied only 43 filters in the first layer and kept the remaining architecture the same. The (2.18) shows output for the IDRID datasets.

Table 2.18: IDRID (32,32,16)

	Precision	Recall	f1-score	Support
Class 0	0.39	1.00	0.56	34
class1	1.00	0.05	0.10	56

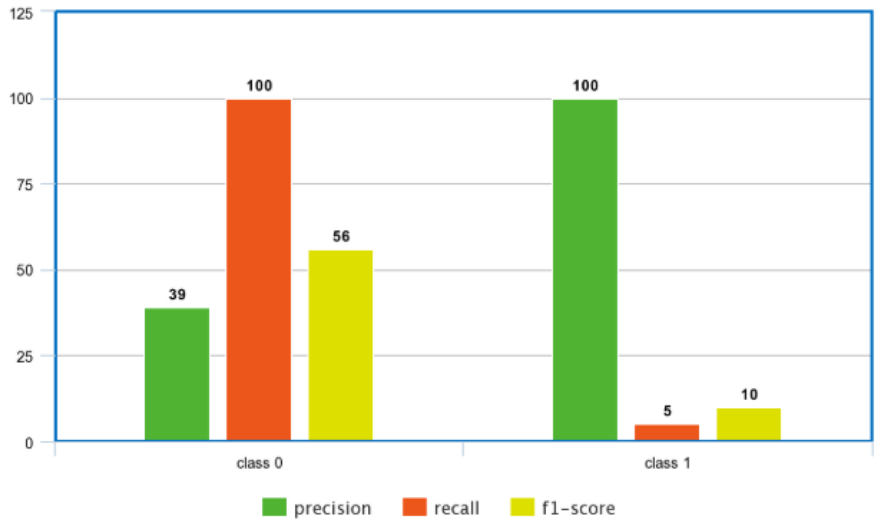


Figure 2.13: IDRID (32,32,16)

Table 2.19: Confusion Matrix for IDRID (FILTERS 32,32,16)

	Predicted DR	Predicted not DR
Actual DR	34	0
Actual not DR	53	3

Here, the same architecture that is three convolutional layers having 32, 32, and 16 filters each separated by 2 x 2 pooling layers is performed on a combination of IDRID

and FUNDUS images.

Table 2.20: IDRID AND FUNDUS (FILTERS 32,32,16)

	Precision	Recall	f1-score	Support
Class 0	0.39	0.74	0.51	42
class1	0.65	0.29	0.40	69
accuracy			0.46	111
Macro avg	0.52	0.51	0.45	111
Weighted avg	0.55	0.46	0.44	111

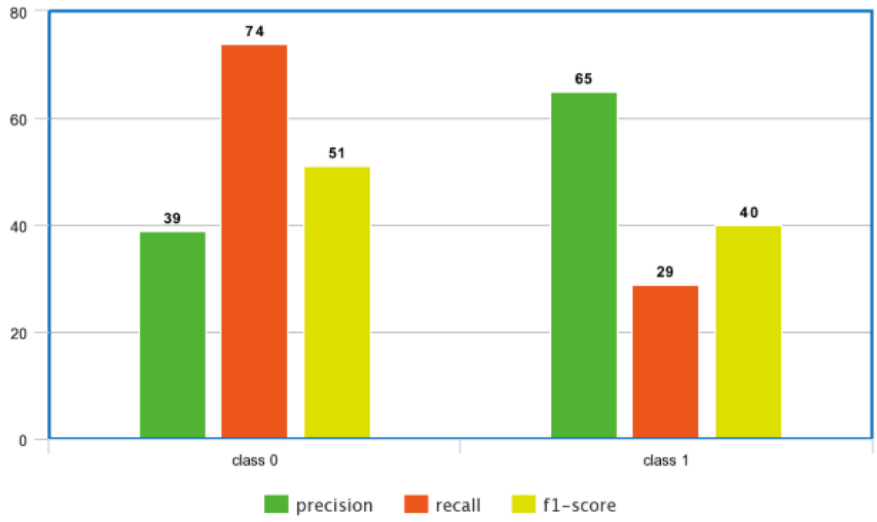


Figure 2.14: IDRID AND FUNDUS (32,32,16)

Table 2.21: IDRID AND FUNDUS (FILTERS 32,32,16)

	Predicted DR	Predicted not DR
Actual DR	32	11
Actual not DR	49	20

As already told, apart from CNN, we have applied other algorithms as well so that a fair comparison can be done. In table 8, the results of the algorithm Random Forest with mtry = 1 applied on the FUNDUS dataset.

Table 2.22: Confusion Matrix for Random Forest on Fundus

	Predicted DR	Predicted not DR
Actual DR	395	5
Actual not DR	11	191

Here in table 7, there is the confusion matrix of application on kNN (k nearest neighbors) where k=3 on FUNDUS dataset

Table 2.23: Confusion Matrix for kNN on Fundus

	Predicted DR	Predicted not DR
Actual DR	390	10
Actual not DR	102	100

Again, the next table i.e. (2.24) shows the confusion matrix of Logistic Regression on

the FUNDUS dataset.

Table 2.24: Confusion Matrix for Logistic Regression on Fundus

	Predicted DR	Predicted not DR
Actual DR	396	4
Actual not DR	4	198

As in several cases, there was a case of overfitting encountered, that ‘s why we needed to augment our data, and so, on the augmented FUNDUS dataset, we applied Naive Bayes’ algorithm to check what accuracy comes. (2.25) shows the output of the same.

Table 2.25: Confusion Matrix for Logistic Regression on Fundus

Algorithm	**Accuracy**	**Precision**	**Recall**
Naive Bayes	92.19	81.12	1

Table 2.26: Confusion Matrix for Naive Bayes on Augmented Fundus

	Predicted DR	Predicted not DR
Actual DR	353	47
Actual not DR	0	202

The (2.27) shows the output of different algorithms on the IDRID dataset. The different algorithms used are Random Forest, kNN with k=3, Naive Bayes and Logistic Regression.

Table 2.27: Outputs on IDRID images

Algorithm	Accuracy	Precision	Recall
Random Forest (mtry=1)	63.85	69.44	86.2
kNN (k=3)	69.87	69.87	100
Naive Bayes	59.03	69.35	74.13
Logistic Regression	62.65	70.14	81.03

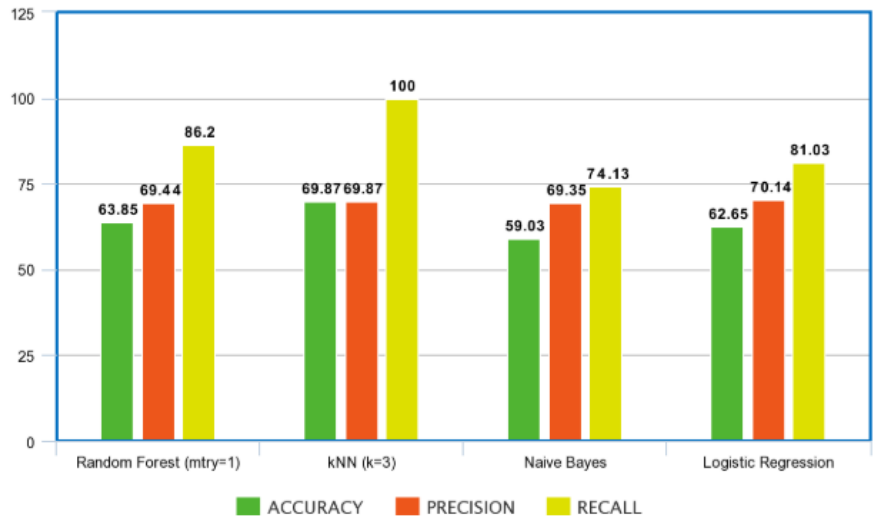


Figure 2.15: Output on IDRID Images

Similarly, there are different algorithms which we have applied on FUNDUS database and get the results for comparison. The different algorithms are kNN with $k = 3$, Random Forest with mtry = 1, Logistic Regression and Naive Bayes. (2.28) summarises the same.

Table 2.28: Outputs on FUNDUS dataset

Algorithm	Accuracy %	Precision %	Recall %
CNN (initial Model) (64,32,16)	98.97	88.88	87.5
kNN (k=41)	81.13	90.9	49.5
Random forest (mtry=1)	97.34	97.44	94.55
Logistic regression	98.67	98.01	98.01
Naive Bayes	90	84	99

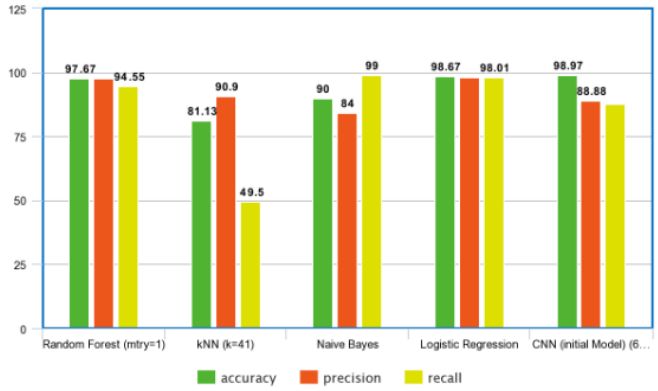


Figure 2.16: Outputs on FUNDUS dataset

Even after augmenting the data, the results were improper. The accuracy increased in Naive Bayes' Classifier but the recall decreased a lot. Again, we went for the Artificial Neural Network but it showed an accuracy of 100 percent and that was nothing but overfitting even after the augmentation. Hence, we needed more datasets and also, we added IDRID dataset images as well.

Chapter 3

Enhanced Architecture and Results

In this chapter, we are going to talk about the enhanced architecture and the comparison of the updated model with all the other algorithms. Previously, we had dealt with a CNN model whose architecture has been discussed earlier. But since the results were very poor, hence there was a lot of hit and trial involved. Ultimately, we reached this architecture, where we added 4 more layers of convolution, ran the whole thing for 100 epochs instead of 25 in the previous case and we got a very good accuracy of 100 percent. Then we had a comparison with other pretrained models, kNN, VGG 16, etc. as well. Now, we will discuss the enhanced architecture in the following table and thereafter we will move towards the final results. The enhanced architecture is as follows:

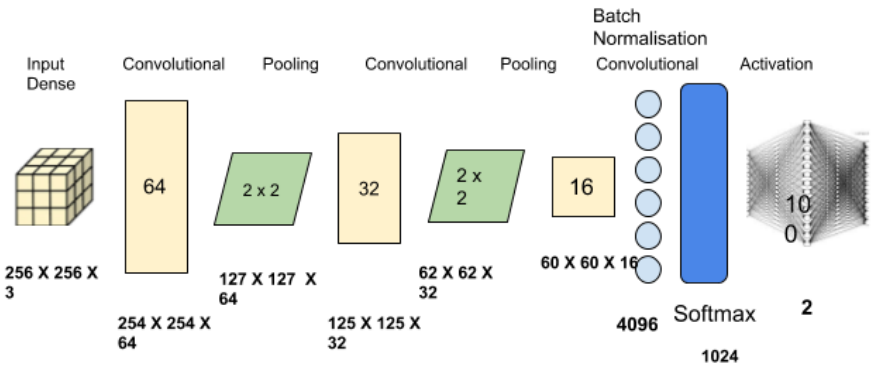


Figure 3.1: The enhanced architecture of our model

The final architecture consists of 3×3 convolution layer with 64 filters. Next comes the 2×2 max pooling layer. After that again 3×3 convolution layer but with 32 filters. Next layer will again be the max pooling layer and again a convolution layer but with 16 filters. This architecture also consists of one batch normalization layer , one activation

layer that is softmax and 100 neurons fully connected layer. By using 100 epochs with this architecture, 100 percent accuracy was achieved. Also epochs of 25 were also tried and 98.26 accuracy was achieved and with 15 epochs 92 accuracy was achieved.

Table 3.1: Enhanced Network Architecture

Output Shape	**Description**
256 x 256 x 3	input
254 x 254 x 64	3 x 3 Convolution Layer, 64 filters
127 x 127 x 64	2 x 2 Max Pooling Layer
125 x 125 x 32	3 x 3 Convolution Layer, 32 filters
62 x 62 x 32	2 x 2 Max Pooling Layer
60 x 60 x 16	3 x 3 Convolution Layer, 16 filters
4096	1 Batch Normalization Layer
1024	1 Activation Layer: Softmax
2	100 neurons fully connected layer

Using the same dataset machine learning algorithms like KNN, Naive bayes, Logistic Regression and Random forest were done, pretty good accuracy was achieved. Table below(3.2) shows different machine learning algorithms and CNN(enhanced model) accuracies.

Table 3.2: Comparison of different algorithms on the updated model

Algorithm	**Accuracy %**	**Precision %**	**Recall %**
kNN	81.17	94.26	67.43
Naive Bayes	90.53	84.19	99.74
Logistic Regression	97.82	99.46	96.15
Random Forest	96.54	98.4	94.61
CNN	100	100	100
VGG	86	14.54	100

Table 3.3: Variance of the result with number of epochs on the updated model

Neurons	**Features**	**Number of Epochs**	**Accuracy**
100	39	100	100
100	39	25	98.26
90	39	15	92

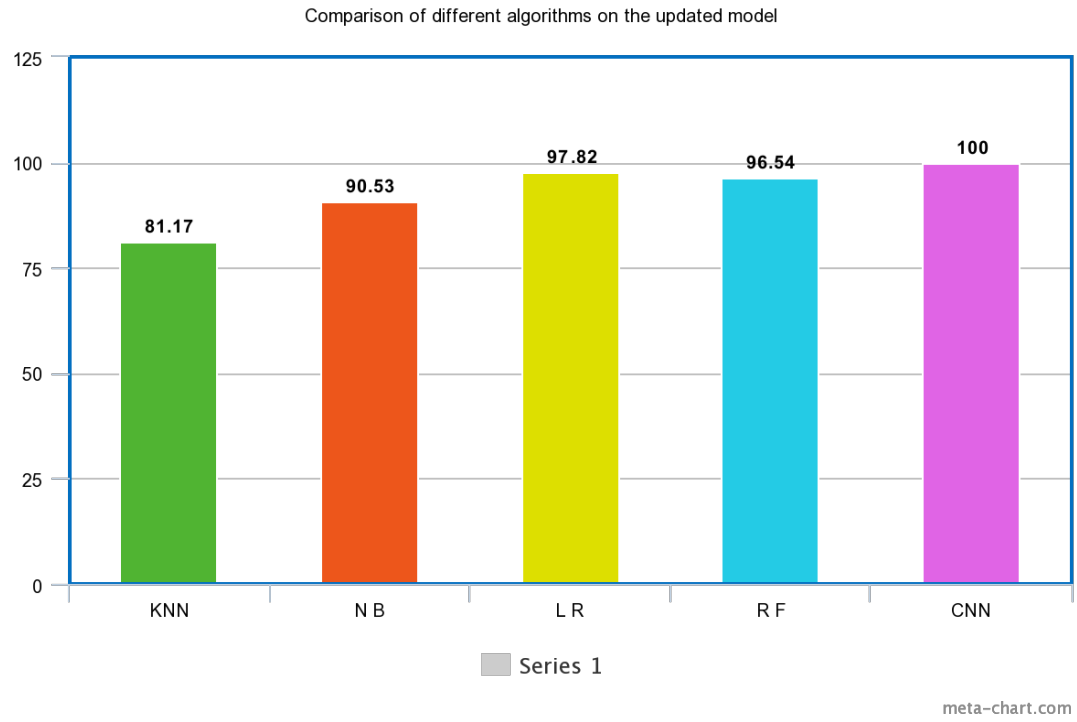


Figure 3.2: Graphs of accuracy

Chapter 4

Conclusion and Future Work

we have done pre-processing and then have tuned our model just for better accuracy. Hence, this model can be used time and again with reliability. The project is about proposing an optimal model for Diabetic Retinopathy detection. Processing of Retinopathy images is very essential to get proper features.

Statistical values can predict level of severity properly but in case of noisy images the chances of getting poor data will lead to lower accuracy. For getting better result selecting proper features out of the image is also important . Both CNN and DNN models are effective in term for image, because of CPU training time of CNN getting affected in the study, in this case DNN outperforms CNN for training accuracy as well as validation accuracy. For future work model can train with GPU system, with more number of processed data for getting higher accuracy results. A standalone application will be good for identification of retinopathy images. Also, the proposed model can be integrated with existing NPDR screening algorithms in enhanced prioritization and resourcefulness of the present day eye-care Delivery.

There is a scope of future work as well.Detection and grading can be combined into a single model to make a robust model. Since, the system is completely reliable which is evident from the 100 percent accuracy of the system in testing. It can be used in real time systems and if properly moduled into a framework like Apple EyeNet, general masses can use it to detect whether they have Diabetic Retinopathy or not and thus, our work will be fruitful.

References

[1] High-resolution fundus (hrf) image database. [Online]. Available: https://www5.cs.fau.de/research/data/fundus-images/

[2] P. Porwal, S. Pachade, R. Kamble, M. Kokare, G. Deshmukh, V. Sahasrabuddhe, and F. Meriaudeau, "Indian diabetic retinopathy image dataset (idrid): A database for diabetic retinopathy screening research," *Data*, vol. 3, p. 25, 07 2018.

[3] B. Krismono Triwijoyo and Y. Pradipto, "Detection of hypertension retinopathy using deep learning and boltzmann machines," *Journal of Physics: Conference Series*, vol. 801, p. 012039, 01 2017.

[4] J. Garvin. The physics of vision. [Online]. Available: http://ffden-2.phys.uaf.edu/211_fall2013.web.dir/jessica_garvin/homepage.htm

[5] Idf diabetes atlas. [Online]. Available: https://www.diabetesatlas.org/data/en/world/

[6] S. Saha. A comprehensive guide to convolutional neural networks — the eli5 way. [Online]. Available: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way/3bd2b1164a53

[7] R. Gargeya and T. Leng, "Automated identification of diabetic retinopathy using deep learning," *Ophthalmology*, vol. 124, 03 2017.

[8] D. Grady. The vision thing: Mainly in the brain. [Online]. Available: https://www.discovermagazine.com/mind/the-vision-thing-mainly-in-the-brain

[9] Retinopathy. [Online]. Available: https://www.health.harvard.edu/a_to_z/retinopathy-a-to-z

[10] S. Dutta, "Classification of diabetic retinopathy images by using deep learning models," *IEEE*, vol. 3, pp. 891–921, 2017.

[11] R. Kandan and P. Aruna, "Diagnosis of diabetic retinopathy using machine learning techniques," *ICTACT Journal on Soft Computing*, vol. 03, pp. 563–575, 07 2013.

[12] C. Lam, D. Yi, M. Guo, and T. Lindsey, "Automated detection of diabetic retinopathy using deep learning," *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, vol. 2017, pp. 147–155, 05 2018.

[13] K. Xu, D. Feng, and H. Mi, "Deep convolutional neural network-based early automated detection of diabetic retinopathy using fundus image," *Molecules*, vol. 22, p. 2054, 11 2017.

[14] A. Sopharak, B. Uyyanonvara, S. Barman, and T. Williamson, "Automatic detection of diabetic retinopathy exudates from non-dilated retinal images using mathematical morphology methods," *Computerized medical imaging and graphics : the official journal of the Computerized Medical Imaging Society*, vol. 32, pp. 720–7, 11 2008.

[15] R. Sivakumar, G. Ravindran, M. Muthayya, S. Lakshminarayanan, and C. Velmurughendran, "Diabetic retinopathy classification," 11 2003, pp. 205 – 208 Vol.1.

[16] T. Walter, J.-C. Klein, P. Massin, and A. Erginay, "A contribution of image processing to the diagnosis of diabetic retinopathy - detection of exudates in color

fundus images of the human retina," *IEEE transactions on medical imaging*, vol. 21, pp. 1236–43, 11 2002.

[17] V. Raman, P. Then, and P. Sumari, "Proposed retinal abnormality detection and classification approach: Computer aided detection for diabetic retinopathy by machine learning approaches," 06 2016, pp. 636–641.

[18] M. R. K. Mookiah, U. R. Acharya, K. Chua, C. Lim, E. Ng, and A. Laude, "Computer-aided diagnosis of diabetic retinopathy: A review," *Computers in Biology and Medicine*, vol. 43, pp. 2136–2155, 12 2013.

[19] G. Yadav, S. Maheshwari, and A. Agarwal, "Contrast limited adaptive histogram equalization based enhancement for real time video system," in *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2014, pp. 2392–2397.

[20] M. Akram and S. Khan, "Multilayered thresholding-based blood vessel segmentation for screening of diabetic retinopathy," *Engineering with Computers*, vol. 29, 04 2013.

[21] M. Fraz, R. Welikala, A. Rudnicka, C. Owen, D. Strachan, and S. Barman, "Quartz: Quantitative analysis of retinal vessel topology and size – an automated system for quantification of retinal vessels morphology," *Expert Systems with Applications*, vol. 42, 05 2015.

[22] S. Sil Kar and S. Maity, "Blood vessel extraction and optic disc removal using curvelet transform and kernel fuzzy c-means," *Computers in Biology and Medicine*, vol. 70, 01 2016.

[23] R. Sundaram, R. KS, P. Jayaraman, and V. B, “Extraction of blood vessels in fundus images of retina through hybrid segmentation approach,” *Mathematics*, vol. 7, p. 169, 02 2019.