A Deep Learning Based Approach for Grading of Diabetic Retinopathy Using Large Fundus Image

A PROJECT REPORT

Submitted by

T.KEERTHANA

In partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

SSM INSTITUTE OF ENGINEERING AND TECHNOLOGY, DINDIGUL - 624003

ANNA UNIVERSITY: CHENNAI-600025

MAY -2023

BONAFIDE CERTIFICATE

Certified that this project report **"A Deep Learning Based Approach for Grading of Diabetic Retinopathy using Large Fundus Image"** is the bonafide work of **"T.KEERTHANA (922119106039), B.MADHUMITHA (922119106051), R.M.MAHALAKSHMI (922119106052), S.MALINI (922119106053)"** who carried out the project work under my supervision.

SIGNATURE OF HOD **SIGNATURE**

Dr. S. KARTHIGAI LAKSHMI,M.E.,Ph.D., **Mr.A.MANIKANDAN,M.E.,**

HEAD OF THE DEPARTMENT **ASSISTANT PROFESSOR**

Department of Electronics and Communication Engineering, SSM Institute of Engineering and Technology, Dindigul - 624002

Department of Electronics and Communication Engineering, SSM Institute of Engineering and Technology, Dindigul - 624002

Submitted for the VIVA-VOCE Examination held on …….

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

First and foremost, we would like to express our deep sense of gratitude to our most honorable Chairman and Management Trustee **Mr.K.Shanmugavel** of SSM Institute of Engineering and Technology for providing us with necessary facilities during the course of study.

We feel immensely pleased to express our sincere thanks to our Principal **Dr.D.Senthil Kumaran** for encouragement and support.

We extend our solemn gratitude to **Dr.S.Karthigai Lakshmi**, Head of the Department, Electronics and Communication Engineering, for timely support to all our activities.

We express our sincere thanks to our Project Coordinator **Dr.K.Rajesh, ASP/ECE** and our guide **Mr.A.Manikandan, AP/ECE** for his guidance throughout our project work.

Also we express our thanks to the Faculty members of our Department, nonteaching staff members and my dear friends for their moral support, help and encouragement towards the successful completion of the project. We are most indebted to our parents, with whose support, resulted in making our dreams of becoming successful graduates, a reality.

.

ABSTRACT

Diabetic Retinopathy affects one-third of all diabetic patients and may cause vision impairment. It has four stages of progression, i.e., mild non-proliferative, moderate non-proliferative, severe non-proliferative and proliferative Diabetic Retinopathy. The disease has no noticeable symptoms at early stages and may lead to chronic destruction, thus causing permanent blindness if not detected at an early stage. The proposed research provides deep learning frameworks for autonomous detection of Diabetic Retinopathy at an early stage using fundus images. The first framework consists of cascaded neural networks, spanned in three layers where each layer classifies data into two classes, one is the desired stage and the other output is passed to another classifier until the input image is classified as one of the stages. The second framework takes normalized, HSV and RGB fundus images as input to three Convolutional Neural Networks, and the resultant probabilistic vectors are averaged together to obtain the final output of the input image. Third framework used the Long Short Term Memory Module in CNN to emphasize the network in remembering information over a long time span. Proposed frameworks were tested and compared on the large-scale Kaggle fundus image dataset EYEPAC. The evaluations have shown that the second framework outperformed others and achieved an accuracy of 98.06% and 83.78% without and with augmentation, respectively.

TABLE OF CONTENT

LIST OF FIGURE

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 DIABETES MELLITUS

Diabetes Mellitus (DM) is a complex disease resulting in severe complications in various parts of the body. Nevertheless, good control of DM will avoid or delay various complications, including Diabetic Retinopathy(DR). Thus, screening and early treatment can avoid significant loss of vision. Such efforts to control this enduring disease as well as the early complications detection such as DR should be strengthened, because DR is an asymptomatic condition in its initial stage. But, slowly it becomes threatening to the vision of the patients. Considering these complications and the rising numbers of diabetic patients, the screening of DR is vital to prevent the complications. To achieve this, significant resources will be required for the management of the condition including human resources, to increase the current workload within the field of disease diagnostics.

1.2 DIABETIC RETINOPATHY

Diabetic Retinopathy (DR) is a chronic eye disease, commonly found in elderly people (age 50 or above), and can cause severe visual impairments or even blindness if not treated at an early stage. The progression of DR can be categorized into four stages (as per the clinical standards), where the perceptible symptomatic appearances of the disease can only be visualized in the last stages when it becomes nearly impossible to fully recover the vision loss.

DR is a serious sight-threatening complication of diabetes. Diabetes interferes with the body's ability to use and store sugar (glucose). The disease is characterized by too much sugar in the blood, which can cause damage throughout the body, including the eyes. Over time, diabetes damages small

blood vessels throughout the body, including the retina. Diabetic retinopathy occurs when these tiny blood vessels leak blood and other fluids. This causes the retinal tissue to swell, resulting in cloudy or blurred vision.

Diabetic retinopathy usually affects both eyes. The longer a person has diabetes, the more likely they will develop diabetic retinopathy. If left untreated, diabetic retinopathy can cause blindness. When people with diabetes experience long periods of high blood sugar, fluid can accumulate in the lens inside the eye that controls focusing. This changes the curvature of the lens, leading to changes in vision. However, once blood sugar levels are controlled, usually the lens will return to its original shape and vision improves. Patients with diabetes who can better control their blood sugar levels will slow the onset and progression of diabetic retinopathy.

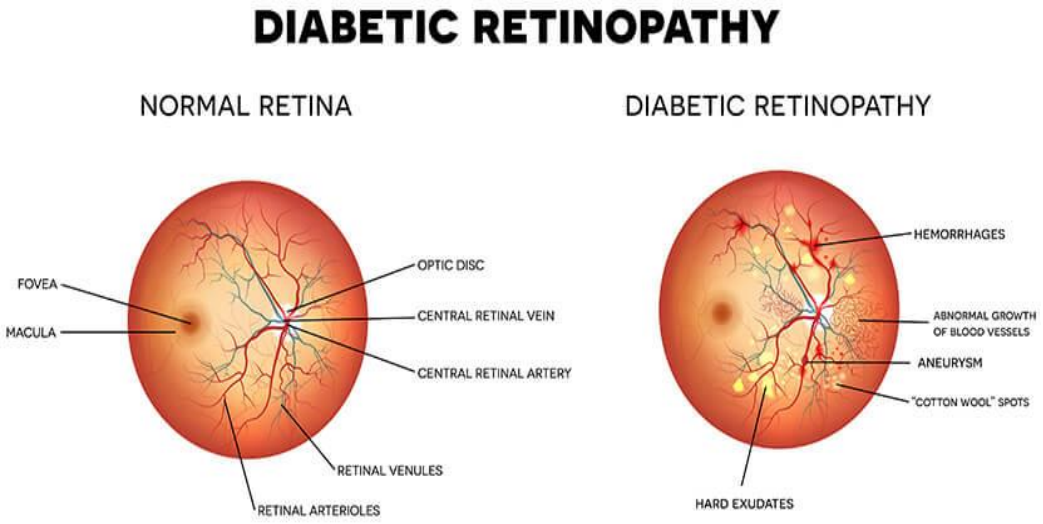

Figure 1.1. NORMAL VS DR

1.2.1 CAUSES

DR is caused by the blood vessels rupturing due to high blood sugar levels. These leaky vessels produce fluid clots and oxygen deficiency, leading to severe visual impairments.

Anyone who has diabetes can develop diabetic retinopathy. The risk of developing the eye condition can increase as a result of:

- Having diabetes for a long time
- Poor control of your blood sugar level
- High blood pressure
- High cholesterol
- Pregnancy
- Tobacco use
- Being Black, Hispanic or Native American

1.2.2 COMPLICATION:

Diabetic retinopathy involves the growth of abnormal blood vessels in the retina. Complications can lead to serious vision problems:

- **Vitreous haemorrhage** the new blood vessels may bleed into the clear, jellylike substance that fills the centre of your eye. If the amount of bleeding is small, you might see only a few dark spots (floaters). In more-severe cases, blood can fill the vitreous cavity and completely block your vision.
- **Retinal detachment** the abnormal blood vessels associated with diabetic retinopathy stimulate the growth of scar tissue, which can pull the retina away from the back of the eye. This can cause spots floating in your vision, flashes of light or severe vision loss.

- **Glaucoma** new blood vessels can grow in the front part of your eye (iris) and interfere with the normal flow of fluid out of the eye, causing pressure in the eye to build. This pressure can damage the nerve that carries images from your eye to your brain (optic nerve).
- **Blindness** Diabetic retinopathy, glaucoma or a combination of these conditions can lead to complete vision loss, especially if the conditions are poorly managed.

1.3 CLASSIFICATION:

Clinically, DR is classified into two types, i.e.,

- ✓ **Proliferative Diabetic Retinopathy (PDR)**
- ✓ **Non-Proliferative Diabetic Retinopathy (NPDR)**

1.3.1 NON PROLIFERATIVE :

NPDR is further subdivided based on retinal findings:

Mild NPDR – At least one microaneurysm present on retinal exam.

Moderate NPDR – Characterized by multiple microaneurysms, dot-and-blot haemorrhages, venous beading, and/or cotton wool spots.

Severe NPDR – In the most severe stage of NPDR, you will find cotton wool spots, venous beading, and severe intraretinal microvascular abnormalities (IRMA). It is diagnosed using the "4-2-1 rule." A diagnosis is made if the patient has any of the following: diffuse intraretinal haemorrhages and microaneurysms in 4 quadrants, venous beading in ≥2 quadrants, or IRMA in ≥1 quadrant. Within one year, 52-75% of patients falling into this category will progress to PDR.

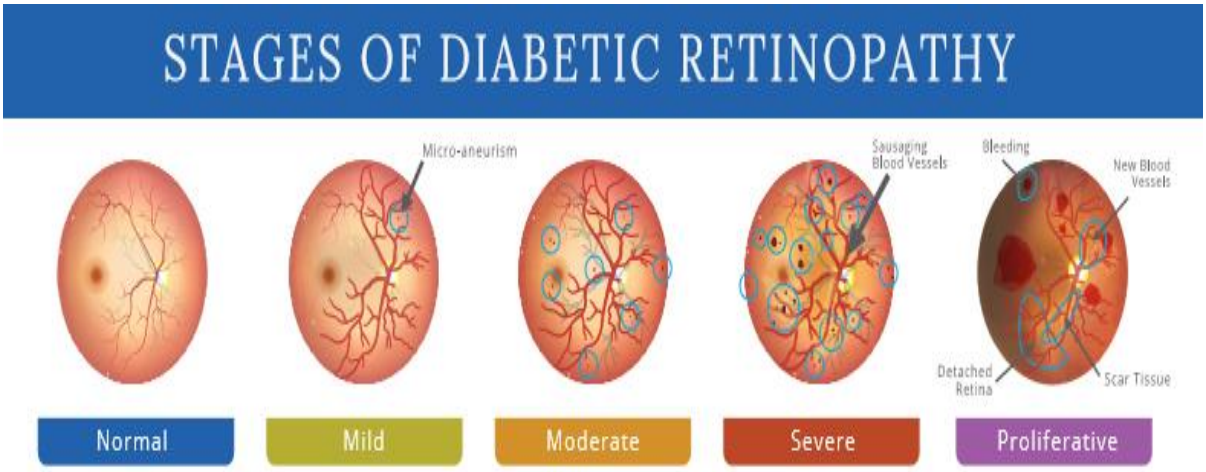

Figure 1.2. STAGES OF NPDR

1.3.2 PROLIFERATIVE

The retina has a high metabolic requirement, so with continued ischemia, retinal cells respond by releasing angiogenic signals such as vascular endothelial growth factor (VEGF). Angiogenic factors, like VEGF, stimulate growth of new retinal blood vessels to bypass the damaged vessels. This is referred to as neovascularization. In PDR, the fibrovascular proliferation extends beyond the ILM. This may sound like a good idea, but the new vessels are leaky, fragile, and often misdirected. They may even grow off the retina and into the vitreous. As the vitreous shrinks with age, it pulls on these fragile vessels and can cause them to tear, resulting in a vitreous haemorrhage and sudden vision loss. These vessels may also scar down, forming strong anchors between the retina and vitreous causing traction on the retina. If enough force is created, a tractional retinal detachment may occur. This is another mechanism by which DR can cause sudden vision loss. If the retina is not re-attached soon, especially if the macula is involved, vision may be permanently compromised.

While the effects of neovascularization in PDR can be devastating, the most common cause of vision loss in diabetics is macular edema. Macular

edema can occur in NPDR, but it is more common in more severe cases of DR due to the leakiness of the new blood vessels.

Diabetics can also have problems located more anteriorly in the eye. The angiogenic molecules that are produced by the retina may float anteriorly, causing neovascularization of the iris. These vessels can grow into the angle of the anterior chamber where the trabecular meshwork, the drain of the eye, resides. This can obstruct outflow of aqueous fluid, raising intraocular pressure and causing acute glaucoma.

1.4 DEEP LEARNING

Deep Learning (DL) is a class of Artificial Intelligence (AI) methods inspired by the structure of human brain and is based on artificial neural networks. Essentially, DL refers to methods learning the mathematical representation of the latent and intrinsic relations of the data in an automatic manner. Unlike traditional machine learning methods, deep learning ones require much less human guidance, since they are not based on the generation of hand-crafted features, a task that can be very laborious and time consuming, but instead learn appropriate features directly from the data. In addition, DL methods scale much better than traditional ML methods as the amount of data increases. By using this method, we can classify the different grades of Diabetic Retinopathy.

Deep learning is also subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behaviour of the human brainalbeit far from matching its abilityallowing it to “learn” from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.

1.5 METHODOLOGY

A classification technique named Deep Learning-based classification system is proposed for automatic detection of retinal lesions and classification of DR. The proposed system utilizes different image processing methods to detect abnormal retinal lesions automatically, and then predict the severity of DR disease to get early treatment and prevent complete blindness.

The proposed technique consists of pre-processing of fundus images, detection of candidate retinal lesions, formulation of feature set, and classification of DR stages. In the pre-processing phase, the technique eliminates background noises and detects optic disc from the retinal fundus image. Four leading lesions; blood vessels, microaneurysms, haemorrhages, and exudates are extracted using different image processing approaches in lesion extraction phase. A feature set is formulated based on the pixel area of each candidate lesion which is further used in classifying that region.

We have designed deep learning framework to accurately classify the detected fundus image into one of the five stages of DR. The major research contributions covered are as follows:

1. A custom lightweight CNN is proposed to handle a complex multi-class problem.

2. A new pre-processing pipeline including different colour spaces for fundus images is proposed to compliment CNN architecture, which has lowered the burden from CNN architecture.

3. A new ensemble approach is presented to handle inter-class similarities in a robust manner.

4. Deep features are complimented with a random forest classifier to reduce trainable parameters

1.6 CONCLUSION:

Deep Learning classification system has been developed to predict the DR for cost-effective screening process. The simulation result on publicly standard fundus image datasets exhibits that the intended technique gives promising results in identifying retinal lesions and it has better capability of classifying several stages of DR compared with other existing automatic diagnosing system.

CHAPTER 2

LITERATURE SURVEY

2.1 TITLE: Diabetic Retinopathy Image Classification Using Support Vector Machine

AUTHOR: Mk et al.

YEAR: 2021

DESCRIPTION: MK et al. used a different approach that focused only on important features and exudates on retina to predict the condition of the disease, since they play a significant role in detecting the severity of disease. They used scale-invariant feature transform (SIFT) and speed-up robust features (SURF) to extract the features of exudate region. Later applied support vector machine (SVM) for the detection of DR and achieved sensitivity of 94%.

2.2 TITLE: Diabetic Retinopathy Grade Classification Based On Fractal Analysis and Random Forest

AUTHOR: F.Alzami et al.

YEAR: 2020

DESCRIPTION: F. Alzami et al. used fractal dimensions for retinopathy detection. Fractal dimensions are used to observe the vascular system of the retinal part of the eye, it not only detects the DR, but also the severity of the disease. The proposed algorithm, evaluated on the Messidor dataset using a random forest classifier, provided satisfactory results for the detection of DR, but the results were not very promising for the classification. They further concluded that other features such as univariate, multivariate, etc., can be used for testing the proposed algorithm.

2.3 TITLE: Classification of Diabetic Retinopathy Based On Hybrid Neural Network

AUTHOR: Alexander Rakhlin

YEAR: 2020

DESCRIPTION: Alexander Rakhlin represented his work for Diabetic Retinopathy Detection (DRD) by designing a Deep Learning framework. He trained and tested the proposed algorithm on the Kaggle dataset, and achieved sensitivity, and specificity of 92% and 72%, respectively. The network was trained for binary classification and lacks the categorization of DR with respect to severity level. The architecture used in the project originated from famous VGG-Net family, winner of ImageNet Challenge ILSVRC-2014, designed for large scale natural image classification. As a pre-processing step, retinal images were normalized, scaled, centered, and cropped to 540 × 540 pixels.

2.4 TITLE: Automated Grading of Diabetic Retinopathy using DenseNet-169 Architecture

AUTHOR: Rahman et al.

YEAR: 2019

DESCRIPTION: Rahman et al. [19] proposed a deep learning framework to grade fundus images as mild, moderate, or severe based on the lesions and exudates. Fundus images are passed to DenseNet-169 after resizing followed by data augmentation. Extracted deep features were used to classify the image among one of the four stages of diabetic retinopathy. Training and testing of the proposed framework were performed on Kaggle APTOS 2019. The proposed model was compared with pre-trained models DenseNet-121 and ResNet-50, and resulted in the highest accuracy of 96.54%.

2.5 TITLE: Diabetic Retinopathy Detection using Ensemble Machine Learning

AUTHOR: Margaret greven

YEAR: 2021

DESCRIPTION: An ensemble-based machine learning algorithm proposed in 2021 [28] incorporated three different classifiers, i.e., Random Forest, SVM, and Neural Networks, followed by a meta classifier to reach a decision. Using an ensemble-based approach, robustness and performance have been enhanced. The proposed algorithm was tested on the Messidor dataset and yielded an accuracy of 0.75. Another ensemble-based algorithm was proposed for diabetic retinopathy screening in 2021.

2.6 TITLE: Detection and Prognosis Evaluation of Diabetic Retinopathy using Ensemble Deep Convolutional Neural Networks

AUTHOR: Quan Dong Nguyen

YEAR: 2020

DESCRIPTION: A two staged classifier has been proposed, where the first stage was comprised of outputs from six classifiers, i.e., SVM, KNN, Multilayer perceptron, Naive Bayes, Decision Trees and Logistic Regression, followed by the second stage, i.e., a neural network, which utilized the output from classifiers to reach the final decision. The proposed algorithm resulted in a test accuracy 76.40% when tested on the Messidor dataset. Another ensemble-based deep neural network architecture was proposed using ResNet in 2020.

2.7 TITLE: Automatic Grading of Diabetic Retinopathy Using Deep Learning and Principal Component Analysis

AUTHOR: Emanate al.

YEAR: 2021

DESCRIPTION: Emanate al. proposed an algorithm to detect various severity levels of diabetic retinopathy using retinal image dataset. The proposed framework takes a retinal image and computes PCA for each channel, followed by classification of each channel using neural network architecture. Final results were concluded based on majority voting by accumulating the results of all three channels. The algorithm yielded an accuracy of 85% when tested on DRD (Kaggle Competition dataset) with retinal images belonging to all severity levels

2.8 TITLE : Automatic Diagnosis of Different Grades of Diabetic Retinopathy and Diabetic Macular Edema Using 2-D-FBSE-FAWT

AUTHOR: Robert kleinman

YEAR: 2022

DESCRIPTION: In 2022 [20], a novel deep learning-based algorithm was proposed to accurately classify a fundus image among different stages of diabetic retinopathy and Macular Edema using order one and two 2D Fourier–Bessel series expansion-based flexible analytic wavelet transforms. Local binary patterns and variance were extracted from the selected bands to represent the statistical properties of the extracted regions using a feature vector.

CHAPTER-3

SYSTEM STUDY

3.1 EXISTING SYSTEM:

- There is various research gap in diabetic retinopathy autonomous detection system
- The existing process to screen DR is time consuming and is hampered by the lack of trained ophthalmologists.
- Because of Data distortions, it is difficult to deal with all five classes at the same time.
- Some research algorithms yielding problem of binary classification that is HEALTHY vs NON-HEALTHY images and thus might result in the incorrect classification of severe cases
- Mild non-Proliferative DR is very difficult to detect as the symptoms include slowing of retinal blood flow, increased leukocytes adhesion & loss of Retinal Pericytes

3.2 PROPOSED SYSTEM

In this chapter, we propose an automated diagnosis technique of Diabetic Retinopathy (DR) using convolution neural network and cascaded classification system that is capable of identifying different retinal lesions; blood vessels, micro aneurysms (MAs),haemorrhages (HMs), exudates (EXs) and also predicting the severity of DR based on detected lesions features. The overall block diagram of the proposed technique is illustrated in the below figure.

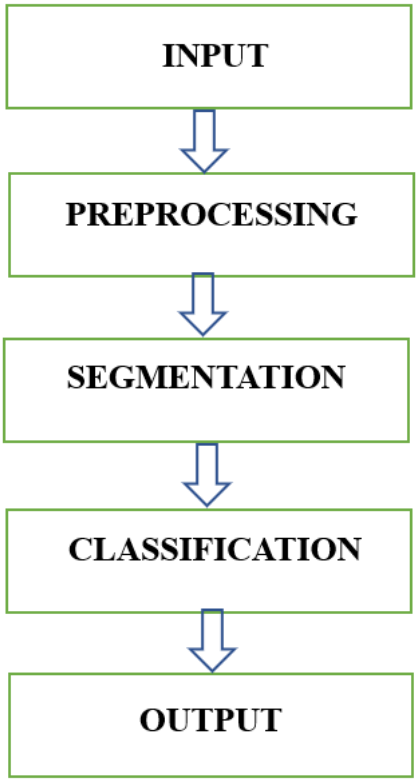

Figure 3.1 BLOCK DIAGRAM OF PROPOSED SYSTEM

The proposed technique is partitioned into two phases: **Detection phase** and **Classification phase.**

In the **Detection phase**, the retinal fundus images are passing through different image processing techniques and morphological operations as adaptive histogram equalization, opening, closing, erosion, top-that transformation, dilation etc.

The fundus images were pre-processed with the above techniques to remove the background noise and generate a high-intensity image. Then, the optic disk and normal blood vessels were detected and eliminated from the image to extract only abnormal components (fragile blood vessels, MAs, HMs, and EXs). Next, a feature set is formulated from eachof the candidate lesions to feed to the classifier.

At last, in the **Classification phase** the image is classified into five stages i.e. normal, mild NPDR, moderate NPDR, severe NPDR, PDR.

CHAPTER 4

SYSTEM REQUIREMENT

4.1 SOFTWARE REQUIREMENT

TOOLS USED – MATLAB

4.1.1 MATLAB:

MATLAB to organize, clean, and analyse complex data sets from diverse fields such as climatology, predictive maintenance, medical research, and finance.

MATLAB provides:

- Datatypes and pre-processing capabilities designed for engineering and scientific data
- Interactive and highly customizable data visualizations
- Apps and Live Editor tasks that helps with interactive data cleaning, preparation, and code generation
- Thousands of prebuilt functions for statistical analysis, machine learning, and signal processing
- Extensive and professionally written documentation
- Accelerated performance with simple code changes and additional hardware
- Expanded analysis to big data without big code changes
- Automatic packaging of analysis into freely distributable software components or embeddable source code without manually recoding algorithms
- Sharable reports automatically generated from your analysis

4.2 DATASET

- Two Public Dataset INDIAN DIABETIC RETINOPATHY IMAGE DATASET(IDRiD) and KAGGLE DATASET
- INDIAN DIABETIC RETINOPATHY IMAGE DATASET(IDRiD) Contains 516 Retina images
- KAGGLE DATASET Contains 1748 Retina images

SAMPLE DATASET

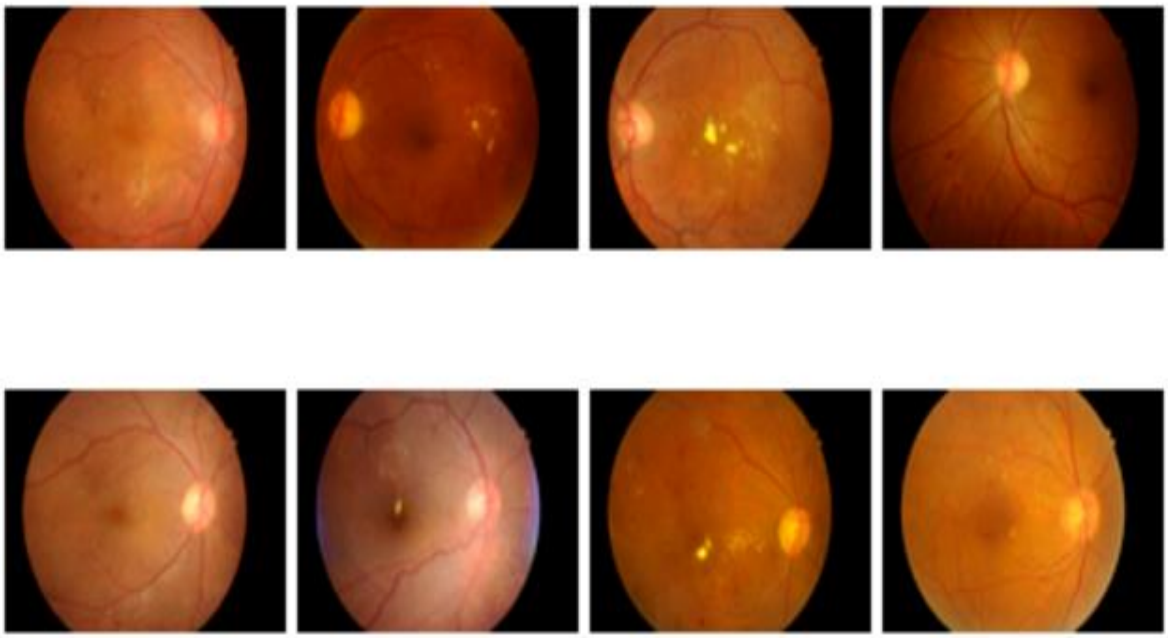

FIG 4.1 KAGGLE DATASET

CHAPTER 5

METHODOLOGY

5.1 INTRODUCTION:

The proposed architecture is comprised of three phases, i.e., image pre-processing, feature extraction followed by classification. We have used the Kaggle dataset to train/test the proposed frameworks. The flow chart diagram of the proposed work is given. To extract deep features, we trained Deep Convolutional Networks. Heat Maps extracted from the proposed framework highlight the presence of any exudates, microaneurysms, hemorrhages, cotton wool spots, or new build vessels; which indicate the extraction of features from the affected region, thus yielding high accuracy.

The deep CNN is capable of taking unknown images as input and extracting problem-specific features, thereby generating an appropriate response. CNN, improving its feature extraction during every iteration of backpropagation, converges closer to the optimum solution for the specific problem under consideration. Based on the decision-making of CNN, it is decided to update the parameter weights in case of false positives or to retain the parameter weights in case of true positives. Invariably, the working capabilities of CNN are dependent on the quality and nature of images; therefore, we resorted to pre-processing instead of feeding the system with raw images.

During the pre-processing, we have enhanced the minute details in order to make exudates and micro-aneurysms more prominent. We have further carried out light equalization as the images were taken in light conditions of varying intensities. To conserve the computational resources, images were resized and the background was removed.

The pre-processed images were then forwarded to our deep CNN architecture, which extracted the best-defined features to signify the objects of our interest within the images. These collected features are fed to a classifier for DR classification into normal, mild, moderate, severe, or proliferative. As highlighted in Figure 1, we have proposed three different CNN-based frameworks to categorize an image among five classes. Lastly, the accuracy of all the frameworks has been compared to analyse the performance. Below are the details of each of the CNN-based proposed frameworks.

PROPOSED MODULE

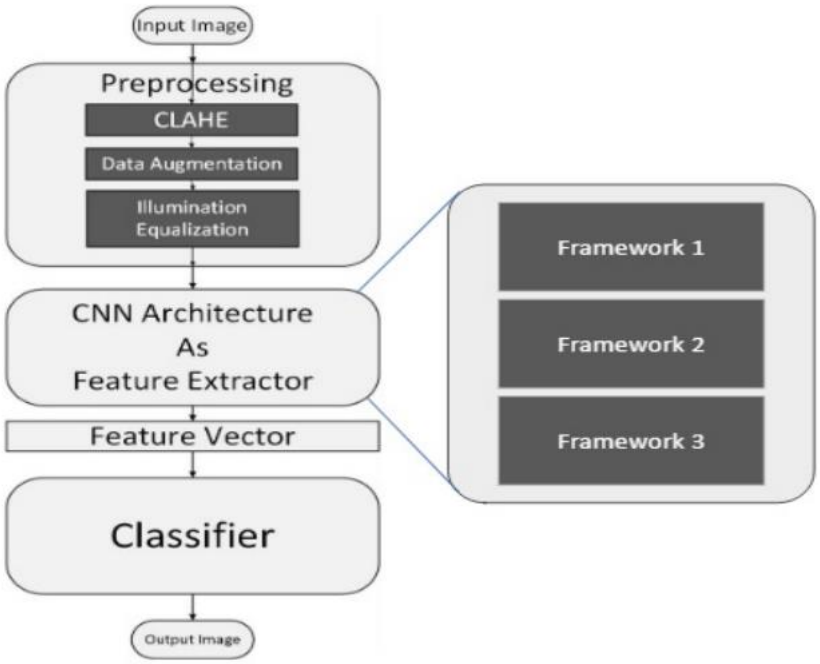

Figure 5.1. BLOCK DIAGRAM OF PROPOSED MODULE

1. **Framework 1** used a basic CNN Architecture for feature extraction; these deep features are defined to represent our objects of interest within the images and are classified using a conventional Random forest classifier (showed the best accuracy among many other classifiers), and designed a cascaded architecture. In this architecture, multiple cascaded layers are used to classify an image into one of the five classes.

2. **Framework 2** performed multiple pre-processing techniques on dataset to compare their results for five class classifications and created an improved ensemble result using them. We applied CNN Architecture of Framework 1 on RGB, HSV and normalized images, categorized as CNN-1, CNN-2, and CNN-3, respectively.

3. **Framework 3** classified images by creating a series of patches (9 patches from each image), and classification was done using LSTM-based CNN Architecture.

5.2 PRE-PROCESSING

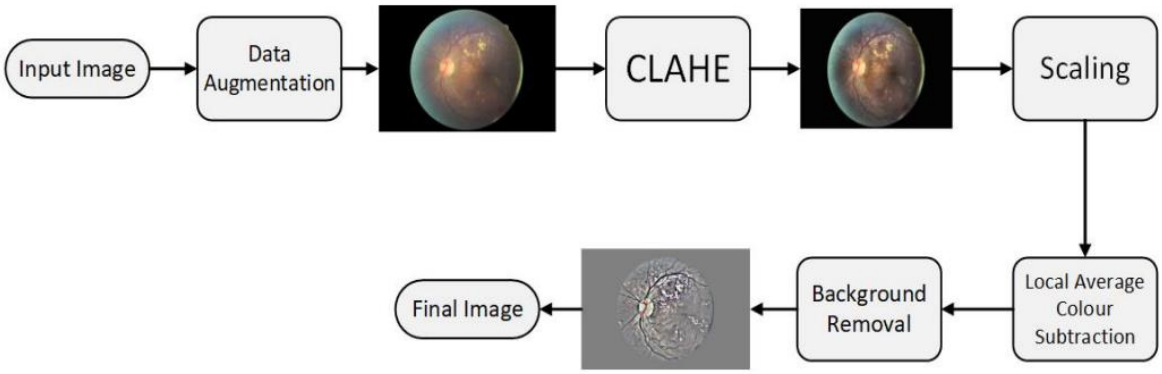

Figure 5.2. PRE-PROCESSING PHASE OF INPUT IMAGE

The first stage of the proposed architecture is the pre-processing phase. The input images are not standardized as they have an un-wanted black background, noise, different aspect ratios, varying light conditions, different color averages, and imbalanced classes. An algorithm shown in figure has been designed to pre-process the raw images and make it optimal. The steps included:

5.2.1 Data Augmentation:

In order to solve the issue of class imbalance, we applied data augmentation to the dataset. A huge class imbalance existed among the dataset.

With class 0 representing more than 50% of the dataset, portions of the dataset represented by the classes in terms of percentage are shown in Table 1. There was a great possibility of over-fitting. As the system will be seeing normal class images way more than any of the other 4 classes, ultimately the trained parameters will learn to lean towards normal class. We over-sampled the images of less dominant classes, using the regeneration of new images from the existing ones by changing a few details in them. We have applied the following augmentations (transformations) during our research:

(a) Flipping to 90 degrees.

(b) Rotating images by [0, 180] degrees with a step size of 15 degrees.

(c) Translating (on scale x and y dimensions).

Table 5.1 distribution of classes

Label	Grade	Percentage	Train Images	Test Images	Train Images Augmented
0	Normal	73%	25,642	39,110	25,642
1	Mild	6.96%	2445	3729	22,005
2	Moderate	15.57%	5469	8342	21,876
3	Severe	2.46%	864	1318	14,688
4	Proliferative	2.01%	706	1077	12,002

To up-sample Class 1 and Class 2 samples, the flip transformation has been applied. Since classes 3 and 4 differ from the rest of the classes by a large percentage, therefore both of the classes were up-sampled by applying all of the transformation steps, i.e., flip, rotation, and translation to reduce the skewed output of the trained model. The classwise data distribution before and after augmentation is shown.

5.2.2 CLAHE

We used Contrast Limited Adaptive Histogram Equalization (CLAHE) technique for the enhancement of an image. It performs a very clear and detailed contrast improvement in an image by equalization of lighting effects. The enhancement results are remarkable even for low-contrast images (underwater images), evident in Figure 3. It can be seen that applying CLAHE on retinal images has enhanced the visibility of minute details. The working of CLAHE is as follows:

(a) Step 1 Division of image into small equal-sized partitions.

(b) Step 2 Performing histogram equalization on every partition.

(c) Step 3 Limiting the amplification by clipping the histogram at some specific value.

5.2.3 Re-scaling:

We have then re-scaled the images by standardizing the size of the optic region to have a common radius (300 pixels or 500 pixels).

5.2.4 Background Removal, Colour Normalization and Gray Scale Mapping:

We then removed the background, keeping a specific radius to focus on ROI, subtracted the local average colour, thereby suppressing the unwanted details, and mapped the local average to 50% grayscale image.

5.2.5 Resizing:

Finally, we resized the image to 256 × 256 according to the requirement of CNN.

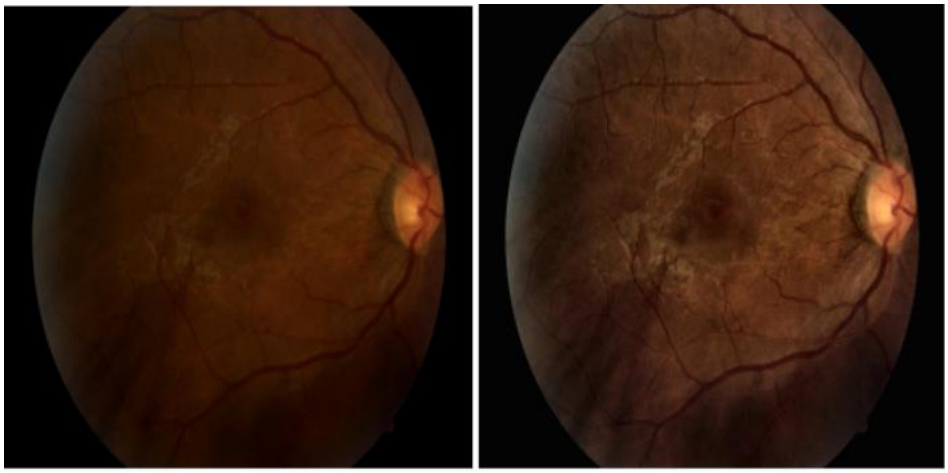

Figure 5.3. Effect of applying CLAHE; (a) original; and (b) CLAHE processed.

5.3 CNN Architecture for Deep Feature Extraction

In order to improve the accuracy, we may add architectural layers to a level, after which there is a drop in accuracy due to inherent factors like over-fitting, etc. The CNN proposed in our research is comprised of layers, namely; input map, convolution layer, activation layer, and max pooling layer. The parameters of CNN, depicting the layer, number of feature maps in each layer (Activation Shape), number of features in each layer (Activation size), and weights (parameters to be trained) are shown in Table 2. The architectural diagram for custom CNN is also shown

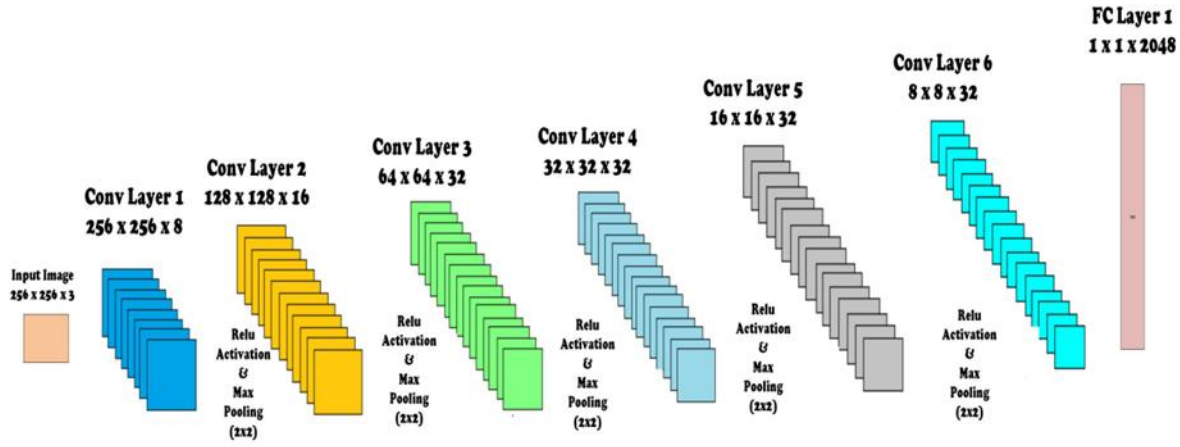

Figure 5.4. CUSTOM DESIGN LIGHTWEIGHT CNN ARCHITECTURE

TABLE 5.2. HYPER PARAMETERS OF THE CNN

Layer	Activation Shape	Activation Size	Parameter
Input	256 × 256 × 3	0	0
Conv 1	256 × 256 × 8	524,288	80
Conv 2	128 × 128 × 16	262,144	160
Conv 3	64 × 64 × 32	131,072	320
Conv 4	32 × 32 × 32	32,768	320
Conv 5	16 × 16 × 32	8192	320
Conv 6	8 × 8 × 32	2048	320
FC 1	2048 × 1	2048	4,194,304

Detailed architecture of the layers are as follows:

5.3.1 Convolution:

This layer has a set of filters (kernel), as our first layer has eight filters. Each filter should have the same depth as the depth of input, since our first layer input image is RGB with depth 3, so we have eight filters of depth 3. The input image becomes convolved with each filter and forms eight feature maps on each convolution layer. Comprising filters (kernel), feature maps are created for the input image based on the number of filters, e.g., if the first layer has 8× filters, this layer will produce 8× feature maps as a result of input image convolution with each filter. For an image of size $k * l$ and filter of size $m * n$, each feature map will have size (with 2 × 2 zero padding and stride = 1):

$$[(k - m + 1) + 2] * [(l - n + 1) + 2]$$

For 8× filters of size $m * n$, the trainable parameters to collect features from image will be:

$$(m * n * 8) + 1$$

where 1 is added for bias.

5.3.2 Maxpooling Layer:

Maxpooling is used to reduce the size (down sample) of the coming feature maps intelligently so that most of the information remains intact. With the reduced dimensions, parameters are also reduced as we move forward in the network. Therefore, it reduces the chances of over-fitting and becomes an efficient system with less computational cost. We used a kernel of size 2 × 2 in each maxpooling layer, and it selects the maximum number from 2 × 2 frame and the frame moves with a 2 × 2 stride. This layer has no parameters that are to be trained.

5.3.3 Activation Layer:

In order to make an effective system for complicated problem solving, an activation function has been added. We used a rectified linear unit (ReLU) after every convolutional and Fully Connected layer since it resulted in the highest accuracy among other activation functions and prevented the vanishing gradient problem effectively. We have designed ReLU within the convolution layer in our architecture, and it has improved the accuracy of classification.

5.3.4 Dropout Layer:

We have avoided the over-fitting by using the dropout layer, as it prevents the network dependence on single node. Probability is assigned to each node, which decides its value.

5.3.5 Fully Connected (FC) Layer:

After passing through many convolutions, activation and max-pooling layers, the most important feature is concentrated. Feature maps of the last layer are then flattened, enabling them to be fed into the FC layer, where every

feature value is connected to every neuron. We have used an FC layer consisting of 2048× neurons. The number of parameters to be trained for this layer are represented as:

$$\text{Number of features} * \text{Number of neurons}$$

5.3.6 Feature Vector:

After this FC layer, we collected these 2048× features of every image as labelled feature vectors for subsequent training of our classifier.

5.4 Classification Using Classifiers:

We have tested our system with various classifiers, i.e., Naive Bayes Logistic Regression, Simple Logistic, SVM, 3NN, Bagging and Random Forest. Among all, Random Forest Tree (RFT) has proven to be most suited classifier for grading of DR. As suggested by its name, RFT is structured on the idea of decision trees in which a decision is made at every node for an unknown problem. The nature of questions in the unknown problem should be discriminating among various classes. At every stage, data is traversed towards its destined end-node (related class) based on the decisions.

5.4.1 Cascaded Classifier

Instead of designing one Classifier of five classes, we have designed a Cascaded Classifier using four binary classifiers, since cascaded classifiers are computationally less expensive and require less training time [32,33]. Our Cascaded Classifier Network (CCN) is a novel deep learning architecture comprised of multiple CNNs merged with classifiers, each predicting a specific feature-set to further divide the data. The division process continues until we obtain leaf nodes equal to the number of classes. Our cascaded classifier is shown. This design allowed us to train all the neural networks simultaneously.

We were able to reduce the possibility of error and over-fitting due to the binary nature of the proposed classifier.

Cascaded Classifier Architecture:

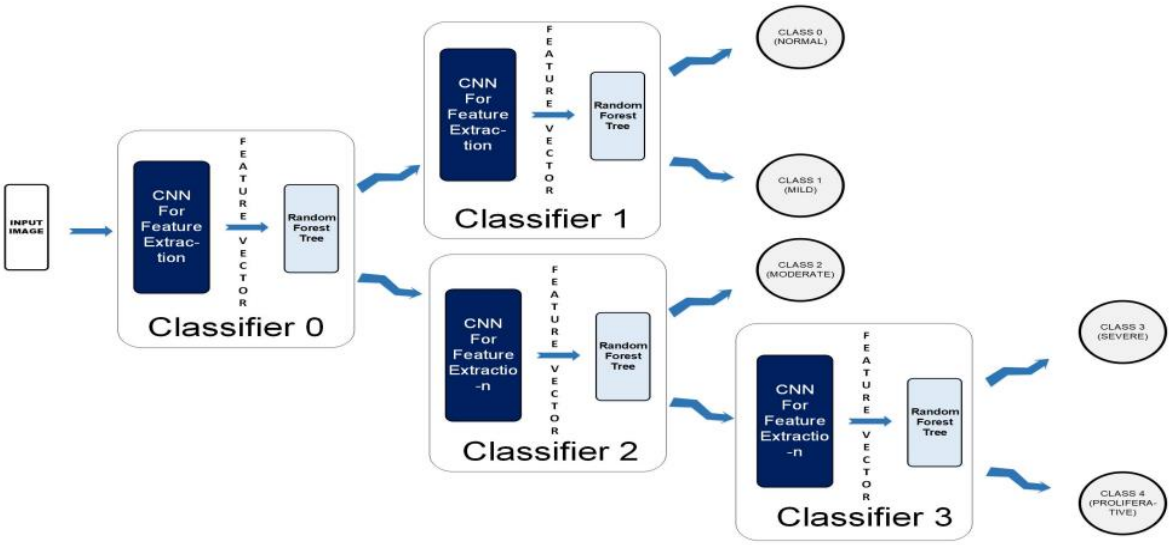

Figure 5.5. CASCADED CLASSIFIER ARCHITECTURE

1. A coloured fundus image (256 × 256 × 3) is fed to the CCN as input.

2. The output map is sequentially forwarded from the previous one to the next four classifiers within the CCN.

3. The new test colour fundus image moves node-wise and ultimately reaches one of the five leaf nodes (terminal nodes) for final decision-making.

4. Finally, it can be classified among one the five classes: Normal, Mild, Moderate, Severe, and Proliferative.

Level of Cascaded Classifier:

1. **Level 0**: The unknown test image is partially classified at the first stage. In Classifier 0, the image goes to class 0, representing Normal and Mild Classes, or Class1, which characterizes Moderate, Severe, and Proliferative stages of DR.

2**. Level 1**: At level-1, Classifier-1 further classifies class-0 from level-1 by labelling them into Normal (Label0) or Mild (Label1).

3. **Level 2**: Using Classifier-2, class-1 can further be divided into two classes where class-0 is Label 2 (Moderate), and class 1 is Label 3 (Severe) and Label4 (Proliferative). At this level, we have acquired end nodes for the classes Normal, Mild and Moderate; however, Class 1 output from Classifier 2 needs further classification.

4. **Level 3**: During this stage, Class 1 output from Classifier 2 is further segregated and labeled as Severe (Label3) or Proliferative (Label4).

5. **Level 4**: Finally, we were able to acquire five separate classes as leaf nodes with subsequent labels as Label 0 (Normal), Label 1 (Mild), Label 2 (Moderate), Label 3 (Severe) and Label 4 (Proliferative). In our design, by making use of the simultaneous training technique, we were able to save the time originally required in case of independent classifier training.

5.4.2 ENSEMBLED SYSTEM DESIGN

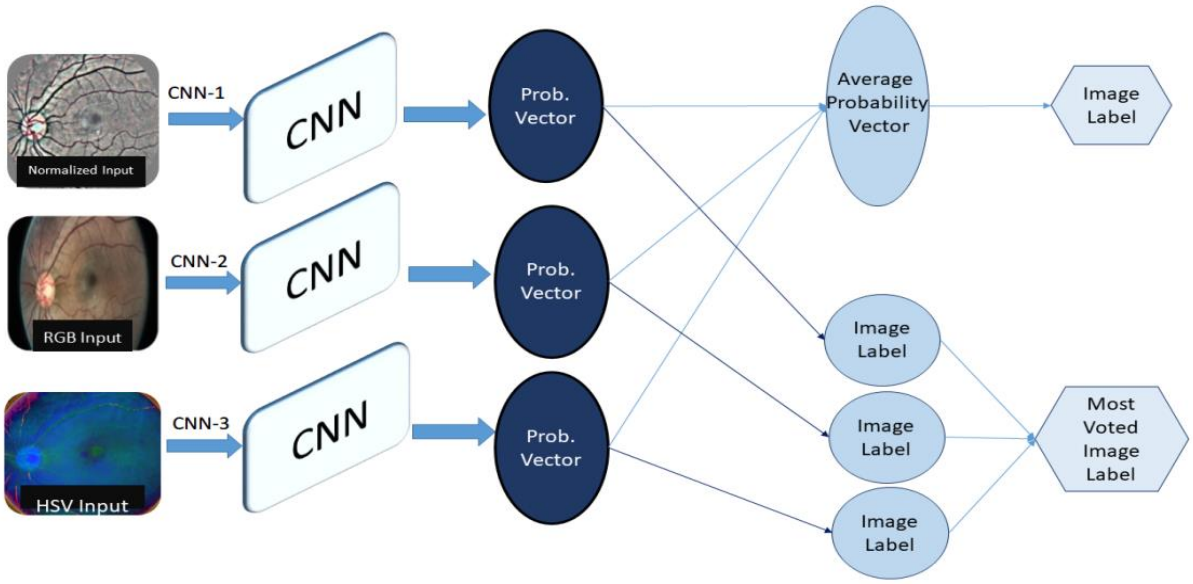

Figure 5.6. ENSEMBLED SYSTEM DESIGN

In the ensembled system, we combined different systems to get our final results. We used our CNN Architecture for the classification of three different types of pre-processed datasets, i.e., Normalized images, RGB images, and HSV images, categorized as CNN-1, CNN-2, and CNN-3, respectively.

1. **Normalized (CNN-1):** We Pre-processed Images in order to suppress unwanted information from images. This dataset is pre-processed with the same algorithm as we used in Framework 1.

2. **RGB (CNN-2):** It is additive colour representation of images. Every pixel in the image represents the saved information in the terms of colour. The intensity value of the three primary colours, red, green, and blue, combined shows one colour. The name RGB is also derived from the initials of the three primary colours.

3. **HSV (CNN-3):** It is a representation of an image in terms of hue, saturation and value, and the three channels represent each of these. Hue is an angle on a coloured spherical surface; it gives colour. Saturation is the measure of how light or dark that colour is; it is a point on the radius at the angle represented by hue. Value is a measure of brightness or intensity of colour; it works simultaneously with saturation.

Separate results of these three different systems are observed and we also observed ensemble results in two different ways for improved accuracy as shown.

1. Most Voted results from three CNN classifiers were collected.

2. Average Probability vector from probability vectors of three CNNs was used to collect results

5.4.3 LSTM CNN

Long Short Term Memory networks, usually just called LSTM, share a special kind of RNN, capable of learning long-term dependencies. LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn. The learning mechanism of LSTM is shown in Figure 7, and these working steps of an LSTM architecture are as follows:

1. The first step in our LSTM [34] is to decide what information we are going to throw away from the cell state. This decision is made by a sigmoid layer called the forget gate layer. It looks at ht−1 and xt , and outputs a number between 0 and 1 for each number in the cell state Ct−1. A '1' represents completely keep this, while a '0' represents completely get rid of this.

2. The next step is to decide what new information we are going to store in the cell state. It has two parts.

• First, a sigmoid layer, called the input gate layer, decides which values we will update.

• Next, a tanh layer creates a vector of new candidate values, Ct , that could be added to the state. In the next step, we will combine these two to create an update to the state.

3. It's now time to update the old cell state, Ct−1, into the new cell state Ct . The previous steps have already reached a decision; we just need to propagate it. We multiply the old state by ft , forgetting the things we decided to forget earlier. Then, we add Ct to it. This is the new candidate value, scaled by how much we decided to update each state value. Finally, we need to decide what we are propagating to the output. This output will be based on our cell state, but will be a filtered version. First, we run a sigmoid layer, which decides

what parts of the cell state we are passing to the output. Then, we put the cell state through tanh (to push the values to be between −1 and 1) and multiply it by the output of the sigmoid gate, so that we only output the parts we decided to propagate

CHAPTER 6

SIMULATION AND RESULT

```
function
imgProb =
exDetect(
rgbImgOrig,
removeON,
onY, onX )
                %exDetect: detect exudates
                % V. 0.2 - 2010-02-01
                % make compatible with Matlab2008
                % V. 0.1 - 2010-02-01
                %                                                    source:
/mnt/data/ornl/lesions/exudatesCpp2/matlab/exudatesCpp3
                   addpath('misc');
                   %-- Parameters
                   showRes = 0; % show lesions in image
                   %--
                   % if no parameters are given use the test image
                   if( nargin == 0 )
                      rgbImgOrig = imread( 'misc/img_ex_test.jpg' );
                      removeON = 1;
                      onY = 905;
                      onX = 290;
                      showRes = 1;
                   End
                   %
                   imgProb = getLesions( rgbImgOrig, showRes, removeON,
                onY, onX );
                End
                function [lesCandImg] = getLesions( rgbImgOrig, showRes,
                removeON, onY, onX )
                   % Parameters
                   winOnRatio = [1/8,1/8];
                   %
                   % resize
                   origSize = size( rgbImgOrig );
                   newSize = [750 round(  750*(origSize(2)/origSize(1)) ) ];
                   %newSize = newSize-mod(newSize,2); % force the size to be
                even
```

```
   newSize = findGoodResolutionForWavelet(newSize);
   imgRGB = imresize(rgbImgOrig, newSize);
   imgG = imgRGB(:,:,2);
   % change colour plane
   imgHSV = rgb2hsv( imgRGB );
   imgV = imgHSV(:,:,3);
   imgV8 = uint8(imgV.*255);

%    %--- normalise
%    imgV = [];
%    if( isempty( forBgImg ) )
%                [imgVfor, imgVnorm, forN, forTrimSize] =
getForacchiaBg2( imgV, 10, 1 );
%       %create an image with the original size
%       imgVforOs = zeros(newSize);
%                                imgVforOs(forTrimSize:newSize(1)-
forTrimSize,forTrimSize:newSize(2)-forTrimSize) = imgVfor;
%    else
%       imgVforOs = imresize(forBgImg, newSize);
%    end
%    %---
   %--- Remove OD region
   if( removeON )
      % get ON window
      onY = onY * newSize(1)/origSize(1);
      onX = onX * newSize(2)/origSize(2);
      onX = round(onX);
      onY = round(onY);
      winOnSize = round(winOnRatio .* newSize);
      % remove ON window from imgTh
      winOnCoordY = [onY-winOnSize(1),onY+winOnSize(1)];
      winOnCoordX = [onX-winOnSize(2),onX+winOnSize(2)];
      if(winOnCoordY(1) < 1), winOnCoordY(1) = 1; end
      if(winOnCoordX(1) < 1), winOnCoordX(1) = 1; end
      if(winOnCoordY(2) > newSize(1)), winOnCoordY(2) =
newSize(1); end
      if(winOnCoordX(2) > newSize(2)), winOnCoordX(2) =
newSize(2); end
   %    imgThNoOD = imgTh;
   %             imgThNoOD(winOnCoordY(1):winOnCoordY(2),
winOnCoordX(1):winOnCoordX(2)) = 0;
      End
   %---
```

```
    % Create FOV mask
    imgFovMask = getFovMask( imgV8, 1, 30 );
    imgFovMask(winOnCoordY(1):winOnCoordY(2),
winOnCoordX(1):winOnCoordX(2)) = 0;
%     %--- Calculate threshold using median Background
%     x=0:255;
%     offset=4;
%                         subImg = double(imgVforOs) -
double(medfilt2(imgVforOs, [round(newSize(1)/30)
round(newSize(1)/30)] ));
%     subImg = subImg .* double(imgFovMask);
%     subImg(subImg < 0) = 0;
%     histImg=hist(subImg(:),x);
%     histImg2 = histImg(offset:end);
%     xPos = x(offset:end);
%     pp = splinefit( xPos, histImg2 );
%     splineHist = ppval( pp, xPos );
% %     figure;plot(xPos,splineHist);
%     splineHistDD = [diff(diff(splineHist)) 0 0];
%     zcList = crossing(splineHistDD);
%     th = xPos(zcList(1));
%     imgThNoOD = subImg >= th;
%     %---
%     %--- fixed threshold using median Background (normal)
%         subImg = double(imgV8) - double(medfilt2(imgV8,
[round(newSize(1)/30) round(newSize(1)/30)] ));
%     subImg = subImg .* double(imgFovMask);
%     subImg(subImg < 0) = 0;
%     imgThNoOD = uint8(subImg) > 10;
%     %---

    %--- fixed threshold using median Background (with
reconstruction)
    medBg = double(medfilt2(imgV8, [round(newSize(1)/30)
round(newSize(1)/30)] ));
    %reconstruct bg
    maskImg = double(imgV8);
    pxLbl = maskImg < medBg;
    maskImg(pxLbl) = medBg(pxLbl);
    medRestored = imreconstruct( medBg, maskImg );
    % subtract, remove fovMask and threshold
    subImg = double(imgV8) - double(medRestored);
    subImg = subImg .* double(imgFovMask);
```

```
    subImg(subImg < 0) = 0;
    imgThNoOD = uint8(subImg) > 0;
    %---

%       %--- create mask to remove fov, on and vessels, hence
enhance lesions
%     se = strel('disk', 5);
%     imgVess = imdilate(imgVess,se);
%     imgMask = imgFovMask & ~imgVess;
%     %---

    %--- Calculate wavelet background
%     imgWav = preprocessWavelet( imgV8, imgMask );
%     imgWav = preprocessWavelet( imgVforOs, imgMask );
    %---

    %--- Calculate edge strength of lesions
    imgKirsch = kirschEdges( imgG );
    img0 = imgG .* uint8(imgThNoOD == 0);
    img0recon = imreconstruct(img0, imgG);
    img0Kirsch = kirschEdges(img0recon);
    imgEdgeNoMask = imgKirsch - img0Kirsch; % edge strength
map
    %---
    % remove mask and ON (leave vessels)
    imgEdge = double(imgFovMask) .* imgEdgeNoMask;

%       %--- Calculate edge strength for each lesion candidate
(Matlab2009)
%     lesCandImg = zeros( newSize );
%     lesCand = bwconncomp(imgThNoOD,8);
%     for idxLes=1:lesCand.NumObjects
%       pxIdxList = lesCand.PixelIdxList{idxLes};
%           lesCandImg(pxIdxList) = sum(imgEdge(pxIdxList)) /
length(pxIdxList);
%     end
%     %---
    %--- Calculate edge strength for each lesion candidate
(Matlab2008)
    lesCandImg = zeros( newSize );
    lblImg = bwlabel(imgThNoOD,8);
    lesCand = regionprops(lblImg, 'PixelIdxList');
    for idxLes=1:length(lesCand)
```

```
        pxIdxList = lesCand(idxLes).PixelIdxList;
        lesCandImg(pxIdxList) = sum(imgEdge(pxIdxList)) /
length(pxIdxList);
    End
    %---

%     %--- Calculate edge strength for each lesion candidate (for
wavelet)
%     lesCandImg = zeros( newSize );
%     lesCandImg2 = zeros( newSize );
%     lesCand = bwconncomp(imgThNoOD,8);
%     for idxLes=1:lesCand.NumObjects
%         pxIdxList = lesCand.PixelIdxList{idxLes};
%         if( length(pxIdxList) > 4 )
% %             lesCandImg(pxIdxList) = sum(imgWav(pxIdxList))
/ length(pxIdxList); %mean
%                                         lesCandImg(pxIdxList) =
std(double(imgWav(pxIdxList))); %std
%             lesCandImg2(pxIdxList) = max(imgWav(pxIdxList))-
min(imgWav(pxIdxList));
%         end
%     end
%     %---

    % resize back
    lesCandImg = imresize( lesCandImg, origSize(1:2), 'nearest' );

    if( showRes )
        figure(442);
        imagesc( rgbImgOrig );
        figure(446);
        imagesc( lesCandImg );
    End
End
function sizeOut = findGoodResolutionForWavelet( sizeIn )
    % Parameters
    maxWavDecom = 2;
    %
    pxToAddC                =                2^maxWavDecom                -
mod(sizeIn(2),2^maxWavDecom);
    pxToAddR                =                2^maxWavDecom                -
mod(sizeIn(1),2^maxWavDecom);
```

```
   sizeOut = sizeIn + [pxToAddR, pxToAddC];
End
function imgOut = preprocessWavelet( imgIn, fovMask )
   % Parameters
   maxWavDecom = 2;
   %
%    % add pixel to allow wavelet decomposition
%                        pxToAddC      =      2^maxWavDecom      -
mod(size(imgIn,2),2^maxWavDecom);
%                        pxToAddR      =      2^maxWavDecom      -
mod(size(imgIn,1),2^maxWavDecom);
%    if(pxToAddC > 0 && pxToAddC <= 2^maxWavDecom)
%       imgIn( :,end+1:end+pxToAddC ) = 0;
%       fovMask( :,end+1:end+pxToAddC ) = 0;
%    end
%    if(pxToAddR > 0 && pxToAddR <= 2^maxWavDecom)
%       imgIn( end+1:end+pxToAddR,: ) = 0;
%       fovMask( end+1:end+pxToAddR,: ) = 0;
%    end

   [imgA,imgH,imgV,imgD] = swt2( imgIn, maxWavDecom,
'haar' );
   imgRecon                              =                              iswt2(
zeros(size(imgA(:,:,2))),imgH(:,:,2),imgV(:,:,2),imgD(:,:,2),
'haar' );
   imgRecon(imgRecon < 0) = 0;
   imgRecon = uint8( imgRecon );
   imgRecon = imgRecon .* uint8(fovMask);
   imgOut = imgRecon * (255 / max(imgRecon(:)));
End
function f = gauss1d( x, mu, sigma )
   f = exp( -(x-mu).^2 / (2*sigma^2) ) / (sigma * sqrt(2*pi) );
End
```

Results

Prior to ascertaining the efficacy and accuracy of any deep learning framework, It requires extensive training using a valid and diverse dataset. In our system design, we have performed training and testing using Kaggle's Diabetic retinopathy dataset. We have evaluated the proposed architecture using accuracy, sensitivity, and specificity as evaluation metrics.

Kaggle Dataset

The Kaggle dataset [35] (provided by EyePACS), has for diabetic retinopathy, a collection of fundus eye images, publicly available for a comparison of detecting Diabetic Retinopathy using Neural Networks, and has been used for training, testing, and validation of the proposed algorithm. It consists of a total of 88,702 images of which 35,126 are training images and 53,576 are testing images. These high-resolution images consisting of left and right eye images are taken under different conditions. Clinicians have rated the images with presence (detection) and severity (grading) of DR on a scale of 0 to 4, where 0 represents Normal or No DR Disease, 1 signifies Mild disease, 2 specifies Moderate disease, 3 represents Severe disease and finally, 4 signifies Proliferative DR disease. In this dataset about 73% of the total images belongs to class 0 (Normal or No DR Disease). Whereas, 708 images are labeled as Proliferative DR, i.e., the highest stage on the severity level, which might lead to blindness if left untreated. In order to resolve the class imbalance problem in the dataset, we have applied augmentation on the classes having the least contribution in training the model. Table 1 elaborates the class-wise effect on training dataset size before and after augmentation, resulting in an increase of training data samples from 35,126 to 96,213. Classes with fewer training samples are up-sampled by applying Flip, Rotation, and Translation along the x and y planes.

Cascaded Design

1. After obtaining feature vectors from the trained neural network, we have performed testing using more than one classifier and have resorted to Random Forest Tree (RFT) based on its accuracy. The performance metrics for different classifiers are given in Table 6.1. The highest accuracy of 74% has been achieved using Random Forest Classifier.

Table 6.1. PERFORMANCE COMPARISON BETWEEN CLASSIFIERS TESTED FOR CASCADED CLASSIFIER.

Classifier	Accuracy	Sensitivity	Specificity
Naive Bayse	62%	78%	44%
Logistic Regression	69%	85%	50%
Simple Logistic	70%	93%	37%
SVM	72%	93%	42%
3NN	66%	88%	36%
Random Forest	94%	94%	31%
CCN	98.5%	97%	27%

2. After the confirmation of the best classifier for the problem under consideration, we used it in our cascaded design as depicted in Figure 5. Table 4 shows how each stage is divided into various sub-classes, i.e., class 0 vs class 1. Accuracy for each level through the Cascaded Classifier along with confusion

matrices are shown. Using this architecture, we overcame the problem of class imbalance in the dataset. Stage 3 shows the lowest accuracy as the two classes were close to each other and intra-class variance is quite low. Whereas, stage 1 shows the highest accuracy since there is a significant difference between Normal and Mild PDR. This architecture helped make class-wise observations and highlighted the accuracy achieved by grouping the classes.

3. By making use of our detection and classification system, we were able to achieve an average accuracy of 98.05%.

CHAPTER 7

CONCLUSION

The automatic classification of diabetic retinopathy (DR) is a growing research area intended to minimize the workload of traditional diagnosing process. The proposed algorithm categorizes a fundus image into one of the five classes of Diabetic Retinopathy. Among the proposed frameworks, an ensemble-based classification algorithm resulted in the highest accuracy. Following the given system of pre-processing and architecture design, we achieved significantly good accuracy. The result shows that the cascaded classifier of the convolution neural network and the random forest is a promising tool for the automatic Five Class Classification of Diabetic retinopathy in optimal resources. Using the cascaded system, training becomes less time-consuming, as training can be done in parallel. A good pre-processing algorithm for a challenging conditioned dataset enhances the quality measures. Moreover, compared with other existing detection and classification approaches, the proposed technique is able to detect maximum DR lesions and classify the unknown stages of DR more accurately. The experimental results validate that the overall performance of the proposed technique is effective in grading DR stages and the system can provide assistance to an ophthalmologist for detecting DR (and its severity level) in a more efficient, reliable and faster way.

7.1 FUTURE ENHANCEMENT

- The disease’s global prevalence in estimated to rise at an exponential rate, reaching 529 million by 2030.
- This is concerning for Worldwide National Health care, as affects people's ability to work, putting the economy in risk.
- It greatly reduce the consumption of human resources and provide a powerful reference to the doctors to make an accurate diagnosis and detect the type of Diabetic Retinopathy

7.2 REFERENCE

1. Gonzalez, R.C. Deep convolutional neural networks [Lecture Notes]. IEEE Signal Process. Mag. 2018, 35, 79–87.

2. Eluyode, O.S.; Akomolafe, D.T. Comparative study of biological and artificial neural networks. Eur. J. Appl. Eng. Sci. Res. 2013, 2, 36–46.

3. Abudawood, M. Diabetes and cancer: A comprehensive review. J. Res. Med. Sci. 2019, 24, 94.

4. Forouhi, N.G.; Wareham, N.J. Epidemiology of diabetes. Medicine 2018, 42, 698–702.

5. Shukla, U.V.; Tripathy, K. Diabetic Retinopathy. Updated 2022 Aug 22. StatPearls 2022. Available online: https://www.ncbi.nlm. nih.gov/books/NBK560805/ (accessed on 1 January 2020).

6. Wang, Y.; Wang, G.A.; Fan, W.; Li, J. A Deep Learning Based Pipeline for Image Grading of Diabetic Retinopathy. In Proceedings of the International Conference, ICSH, Shenzhen, China, 1–2 July 2018.

7. Behera, M.K.; Chakravarty, S. Diabetic Retinopathy Image Classification Using Support Vector Machine. In Proceedings of the 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), Gunupur, India, 13–14 March 2020; pp. 1–4.

8. Sudarmadji, P.W.; Pakan, P.D.; Dillak, R.Y. Diabetic Retinopathy Stages Classification using Improved Deep Learning. In Proceedings of the 2020 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS), Jakarta, Indonesia, 19–20 November 2020; pp. 104–109.

9. Alzami, F.; Megantara, R.A.; Fanani, A.Z. Diabetic Retinopathy Grade Classification based on Fractal Analysis and Random Forest. In Proceedings of

the 2019 International Seminar on Application for Technology of Information and Communication (iSemantic), Semarang, Indonesia, 21–22 September 2019; pp. 272–276.

10. Boral, Y.S.; Thorat, S.S. Classification of Diabetic Retinopathy based on Hybrid Neural Network. In Proceedings of the 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 8–10 April 2021; pp. 1354–1358.

11. Elswah, D.K.; Elnakib, A.A.; Moustafa, H.E. Automated Diabetic Retinopathy Grading using Resnet. In Proceedings of the 2020 37th National Radio Science Conference (NRSC), Mansoura, Egypt, 8–10 September 2020; pp. 248–254.

12. Alexander, R. Diabetic Retinopathy detection through integration of Deep Learning classification framework. BioRxiv 2018.

13. Harry, P.; Frans, C.; Deborah, B.; Simon, P.H.; Yalin, Z. Convolutional Neural Networks for Diabetic Retinopathy. In Proceedings of the Procedia Computer Science, Chiang Mai, Thailand, 2–4 March 2016.

14. Ghosh, R.; Ghosh, K.; Maitra, S. Automatic detection and classification of diabetic retinopathy stages using CNN. In Proceedings of the 2017 4th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 2–3 February 2017; pp. 550–554.

15. Kang, Z.; Zaiwang, G.; Wen, L.; Weixin, L.; Jun, C.; Shenghua, G.; Jiang, L. Multi-Cell Multi-Task Convolutional Neural Networks for Diabetic Retinopathy Grading. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Honolulu, HI, USA, 18–21 July 2018.

16. Mansour, R.F. Deep-learning-based automatic computer-aided diagnosis system for diabetic retinopathy. Biomed. Eng. Lett. 2017, 8, 41–57.

17. Guo, L.; Shibao, Z.; Xinzhe, L. Exudate Detection in Fundus Images via Convolutional Neural Network. 2018. Available online: https://link.springer.com/chapter/10.1007/978-981-10-8108-8_18 (accessed on 1 January 2020).

18. Aljehane, N.O. An Intelligent Moth Flame Optimization with Inception Network for Diabetic Retinopathy Detection and Grading. In Proceedings of the 2022 2nd International Conference on Computing and Information Technology (ICCIT), Tabuk, Saudi Arabia, 25–27 January 2022; pp. 370–373.

19. Rahman, M.T.; Dola, A. Automated Grading of Diabetic Retinopathy using DenseNet-169 Architecture. In Proceedings of the 2021 5th International Conference on Electrical Information and Communication Technology (EICT), Khulna, Bangladesh, 17–19 December 2021; pp. 1–4.

20. Chaudhary, P.K.; Pachori, R.B. Automatic Diagnosis of Different Grades of Diabetic Retinopathy and Diabetic Macular Edema Using 2-D-FBSE-FAWT. IEEE Trans. Instrum. Meas. 2022, 71, 1–9.

21. Indian Diabetic Retinopathy Image Dataset (IDRiD). Available online: https://idrid.grand-challenge.org/ (accessed on 1 January 2020).

22. Farag, M.M.; Fouad, M.; Abdel-Hamid, A.T. Automatic Severity Classification of Diabetic Retinopathy Based on DenseNet and Convolutional Block Attention Module. IEEE Access 2022, 10, 38299–38308.

23. Li, X.; Hu, X.; Yu, L.; Zhu, L.; Fu, C.W.; Heng, P.A. CANet: Cross-disease attention network for joint diabetic retinopathy and diabetic macular edema grading. IEEE Trans. Med. Imaging 2019, 39, 1483–1493.

24. Yue, T.; Yang, W.; Liao, Q. CCNET: Cross Coordinate Network for Joint Diabetic Retinopathy and Diabetic Macular Edema Grading. In Proceedings of the 2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Glasgow, UK, 11–15 July 2022; pp. 2062–2065.

25. Wang, H.; Sun, Y.; Cao, Y.; Ouyang, G.; Wang, X.; Wu, S.; Tian, M. Classification for diabetic retinopathy by using staged convolutional neural network. In Proceedings of the 2022 Asia Conference on Algorithms, Computing and Machine Learning (CACML), Hangzhou, China, 25–27 March 2022; pp. 228–233.

26. Qian, Z.; Wu, C.; Chen, H.; Chen, M. Diabetic retinopathy grading using attention based convolution neural network. In Proceedings of the 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 12–14 March 2021; Volume 5, pp. 2652–2655.

27. Mohamed, E.; Abd Elmohsen, M.; Basha, T. Improved Automatic Grading of Diabetic Retinopathy Using Deep Learning and Principal Component Analysis. In Proceedings of the 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Virtual, 1–5 November 2021; pp. 3898–3901.

28. Alkasassbeh, O.M.; Alauthman, M. Diabetic Retinopathy Detection using Ensemble Machine Learning. In Proceedings of the 2021 International Conference on Information Technology (ICIT), Shanghai, China, 23–26 December 2021; pp. 173–178.