VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JNANA SANGAMA", BELAGAVI-590018.

A

Project Work Phase -II
Report on

"Integrated Image Framework for Diagnosing Diabetic Retinopathy"

SUBMITTED IN PARTIAL FULFILMENT FOR THE AWARD OF THE DEGREE OF
BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

By

SIMRAN DUA [1JB19CS127]
SNEHA C M [1JB19CS131]
SONALI SINGH [1JB19CS132]
SRUSHTI G [1JB19CS138]

UNDER THE GUIDANCE OF

Internal Guide
Dr. Prakruthi M K
Associate Professor
Dept. of CSE SJBIT,
Bengaluru

SJBIT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SJB INSTITUTE OF TECHNOLOGY
#67, BGS HEALTH AND EDUCATION CITY, Dr Vishnuvardhan Road, Kengeri, Bengaluru-560060.

2022-2023

||Jai Sri Gurudev||

Sri Adichunchanagiri Shikshana Trust ®

SJB INSTITUTE OF TECHNOLOGY

#67, BGS Health & Education City, Dr Vishnuvardhan Road, Kengeri, Bengaluru-560060.

SJBIT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the Project work entitled **"INTEGRATED IMAGE FRAMEWORK FOR DIAGNOSING DIABETIC RETINOPATHY"** carried out by **SIMRAN DUA [1JB19CS127], SNEHA C M [1JB19CS131], SONALI SINGH [1JB19CS132], SRUSHTI G [1JB19CS138]** is a bonafide student of SJB Institute of Technology in partial fulfillment for the award of **"BACHELOR OF ENGINEERING"** in **COMPUTER SCIENCE AND ENGINEERING** as prescribed **VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI** during the academic year 2022-2023. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the said degree.

Signature of Guide	Signature of HOD	Signature of Principal
Dr. Prakruthi M.K	Dr. Krishna A.N.	Dr. K.V. Mahendra Prashanth
Associate Professor	Professor & Head	Principal, SJBIT
Dept. of CSE, SJBIT	Dept. of CSE, SJBIT	

Principal
SJB Institute of Technology
67, BGS Health & Education City,
Dr. Vishnuvardhan Road,
Kengeri, Bengaluru - 560 060.

EXTERNAL VIVA

Name of the Examiners	Signature with Date
1. Dr. Pradeep HK	24/5/23
2. L. KalaChandraSekhar	Kala-L 24/5/23

ACKNOWLEDGEMENT

We would like to express my profound gratefulness to His Divine Soul Jagadguru Padmabhushan **Sri Sri Sri Dr. Balagangadharanatha Maha Swamiji** and His Holiness Jagadguru **Sri Sri Sri Dr. Nirmalanandanatha Maha Swamiji** for providing me an opportunity to complete my academics in this esteemed Institution.

We would like to thank **Revered Sri Sri Dr. Prakashnath Swamiji**, Managing Director, SJB Institute of Technology, for his continuous support in providing amenities to carry out this project in this admired institution.

We express my gratitude to **Dr. K. V. Mahendra Prashanth**, Principal, SJB Institute of Technology, for providing me an excellent facilities and academic ambiance, which has helped me complete the project work satisfactorily.

We sincerely thank **Dr. Krishna A N**, Professor and Head of the Department of Computer Science and Engineering, SJB Institute of Technology, for providing me with invaluable support throughout the period of the project.

We express my deepest gratitude and sincere thanks to my guide **Dr. Prakruthi M.K**, **Associate Professor**, Department of Computer Science and Engineering, SJB Institute of Technology Bengaluru, we sincerely thank **Chaitra M, Project Coordinator**, Department of CSE, for her valuable support.

Finally, we take this opportunity to extend my earnest gratitude and respect to my parents, the teaching & non-teaching staff of the department, the library staff, and all my friends, who have directly or indirectly supported me during the period of my project work.

Regards,

Simran Dua [1JB19CS127]

Sneha C M [1JB19CS131]

Sonali Singh [1JB19CS132]

Srushti G [1JB19CS138]

DECLARATION BY THE STUDENT

We, **Simran Dua [1JB19CS127], Sneha C M [1JB19CS131], Sonali Singh [1JB19CS132], and Srushti G [1JB19CS138]** students of 8th-semester Computer Science and Engineering, **SJB INSTITUTE OF TECHNOLOGY**, Bangalore, hereby declare that the project entitled "**Integrated Image Framework for Diagnosing Diabetic Retinopathy**" submitted to the **Visvesvaraya Technological University, Belagavi** during the academic year **2022-23** is a record of an original work done by us under the guidance **Dr. Prakruthi M K**, Associate Professor, Department of Computer Science and Engineering, SJB Institute of Technology, Bangalore. This project dissertation report is submitted in partial fulfillment of the Computer Science and Engineering award. The results embodiedin this report have not been submitted to any other University or Institute for the award of any degree.

Date:

Place: Bangalore

Simran Dua [1JB19CS127]
Sneha C M [1JB19CS131]
Sonali Singh [1JB19CS132]
Srushti G [1JB19CS138]

ABSTRACT

Diabetic retinopathy is a microvascular complication of diabetes mellitus that is a significant cause of new-onset blindness. The Diabetic Retinopathy Study was the first multi-centred, randomized, clinical trial in ophthalmology. It is the most common cause of blindness in the developed countries. A literature review is conducted to assess the effectiveness of existing approaches to find that Convolution Neural Network (CNN) has been frequently adopted for analysing the fundus retinal image for detection and classification. However, fundus images are usually related to uneven lighting and some visual distortion factors. Choosing a suitable pre-processing mechanism is crucial for improving fundus image quality to highlight hidden information and weigh the essential features such as retinal blood-vessels and exudates to analyse diabetic retinopathy. However, selecting an appropriate pre-processing mechanism is challenging to perform a comprehensive enhancement operation. It cannot be achieved using single techniques because each denoising and enhancement techniques are associated with a few advantages and limitations. Therefore, the resulting images may associate with other different artifacts, even after pre-processed. To address such issues, modelling a universal framework is designed to integrate different pre-processing techniques to evaluate fundus image enhancement. In recent time, its diagnosis and classification are done by using machine learning techniques with acceptable results as Computer Vision and Image Processing techniques are being used effectively and efficiently on medical images like on retina images in modern medical science. Use of machine learning algorithms and deep learning algorithms have increased enormously in the medical image analysis and processing. Deep learning techniques like Deep Neural Networks, Convolution Neural Networks have been used to Diabetic Retinopathy detection by using retina images. This project uses image pre-processing and deep learning techniques to determine various stages of DR i.e Normal eye, Mild NPDR, Moderate NPDR, Severe NPDR, Proliferative DR. A user-friendly, interpretable, User Interface (UI) was developed with the system. The major goal of the proposed project is to have the best performance outcomes, and a practical choice to improve the quality of life in people.

TABLE OF CONTENTS

List of Figures

CHAPTER 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 Overview

Diabetes or diabetes mellitus is a metabolic disease in which the person’s body produces an inadequate amount of insulin to produce high blood sugar. It is a life-threatening and rapidly growing chronic disease. Diabetic retinopathy develops when high blood sugar levels associated with diabetes damage the tiny blood vessels in the retina[1].

Over time, the damaged blood vessels can leak or become blocked, leading to various changes in the retina, including swelling, hemorrhages, and the growth of abnormal blood vessels. These changes can impair vision and, if left untreated, may result in permanent vision loss or even blindness.

Diabetic retinopathy is a progressive condition that typically affects both eyes and may not cause noticeable symptoms in its early stages. As the condition advances, symptoms such as blurred vision, floaters, difficulty seeing in low light, and even complete vision loss may occur. Diabetic retinopathy is a serious eye condition that requires early detection and appropriate management to prevent vision loss. Regular eye exams, good blood sugar control, blood pressure management, and lifestyle changes, such as maintaining a healthy diet and exercising, are crucial in preventing or managing diabetic retinopathy. Treatment options for diabetic retinopathy include laser therapy, injections of anti-VEGF medications, and in severe cases,

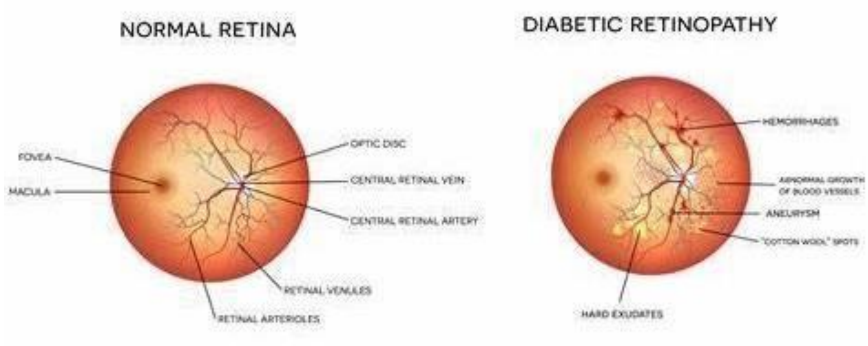

Fig 1.1 : Diabetic Retinopathy

surgical interventions like vitrectomy. Timely diagnosis and intervention are key to preserving vision and reducing the impact of diabetic retinopathy on the quality of life for individuals with diabetes. As the statistics of diabetics around the world is growing enormously and its pervasiveness is growing anything like black night or the pandemic, its complications also creating problems in large numbers for people suffering from Diabetes. In India itself, more than 62 million people are suffering from diabetes and likely to cross 130 million by 2045. People who are suffering from diabetes for more than 20 years have 80% chance of causing diabetic retinopathy. According to the International Diabetes Federation, the number of adults with diabetes in the world is estimated to be 366 million in 2011 and by 2030 this would have risen to 552 million. The number of people with type 2 diabetes is increasing in every country 80% of people with diabetes live in low-and middle-income countries. India stands first with 195% (18 million in 1995 to 54 million in 2025). Previously, diabetes mellitus (DM) was present, largely, among the urban population in India. Recent studies clearly show an increasing prevalence in rural areas as well. Indian studies show a 3-fold increase in the presence of diabetes among the rural population over the last decade or so (2.2% in 1989 to 6.3% in 2003).

In India, Study shows the estimated prevalence of type 2 diabetes mellitus and diabetic retinopathy in a rural population of south India are nearly 1 of 10 individuals in rural south India, above the age of 40 years. The earlier work in the detection of varies stages DR based on explicit feature extraction & classification by using various Image Processing techniques & Machine learning algorithm respectively. Though high accuracy can be achieved using these methods but diagnosing diabetic retinopathy based on the explicit extraction of features is an intricate procedure. Due to the development of Computer vision in recent times & availability of large datasets, it is now possible to use a deep Neural network for the detection & classification of Diabetic retinopathy.

The method proposed in this paper aims at detecting the various stages of Diabetic Retinopathy by using U-Net segmentation with region merging & Convolutional Neural Network. Retinal segmentation is the process of automatic detection of boundaries of blood vessels within the retina. This allows the classifier to learn important features such as retinal proliferation and retinal detachment.

1.2 Symptoms of Diabetic Retinopathy

Blurry vision, dark spots, impairment in vision with color identification problems, fluctuation problems, night vision problems, problem while driving and reading, etc. are the few signs and symptoms of diabetic retinopathy. Diabetic retinopathy may not cause noticeable symptoms in its early stages, which is why regular eye exams are crucial for individuals with diabetes. However, as the condition progresses, the following signs and symptoms may occur:

Floaters: Floaters are tiny specks or spots that seem to "float" in the field of vision. These can be caused by bleeding into the vitreous, the gel-like substance that fills the inside of the eye, as a result of diabetic retinopathy.

- **Fluctuating vision:** Vision may fluctuate, with periods of clear vision followed by blurry vision or vice versa. This can be a sign of changes in the retina due to diabetic retinopathy.
- **Impaired color vision:** Colors may appear faded or less vibrant than usual, which can be a sign of diabetic retinopathy affecting the retina.
- **Poor night vision:** Difficulty seeing in low light conditions, such as at night or in dimly lit areas, may occur as a result of diabetic retinopathy affecting the retina's ability to transmit visual signals to the brain.
- **Dark or empty areas in vision:** Dark or empty areas may appear in the visual field, indicating areas of retinal damage or loss of vision due to advanced diabetic retinopathy.

It's important to note that these signs and symptoms can also be indicative of other eye conditions, and only a comprehensive eye exam by an eye care professional can accurately diagnose diabetic retinopathy. If you have diabetes or are at risk of developing diabetes, regular eye exams and close monitoring of your blood sugar levels are essential for early detection and management of diabetic retinopathy[2].

1.3 Risk Factors of Diabetic Retinopathy

- **Duration of diabetes:** The longer an individual has diabetes, the higher the risk of developing diabetic retinopathy.
- **Poor blood sugar control:** Poorly controlled blood sugar levels over time can increase the risk of diabetic retinopathy.
- **High blood pressure:** Hypertension, or high blood pressure, is a risk factor for diabetic retinopathy as it can damage the blood vessels in the retina.
- **High cholesterol levels:** Elevated levels of cholesterol, particularly LDL (low-density lipoprotein) cholesterol, can increase the risk of diabetic retinopathy.
- **Pregnancy:** Pregnant women with diabetes, particularly gestational diabetes, are at an increased risk of developing diabetic retinopathy.
- **Type of diabetes:** Individuals with type 1 diabetes or type 2 diabetes have a higher risk of developing diabetic retinopathy.

- **Ethnicity:** Certain ethnic groups, such as African Americans, Hispanics, and Native Americans, have a higher risk of diabetic retinopathy.
- **Family history:** Having a family history of diabetic retinopathy or other diabetic complications may increase the risk of developing the condition.
- **Smoking:** Smoking can increase the risk of diabetic retinopathy by damaging the blood vessels in the retina.
- **Kidney disease:** Diabetic nephropathy, which is a complication of diabetes affecting the kidneys, is also a risk factor for diabetic retinopathy.

It's important to note that having one or more of these risk factors does not guarantee that an individual will develop diabetic retinopathy, but it increases the likelihood. Regular eye exams and appropriate management of diabetes and associated risk factors are crucial in reducing the risk of diabetic retinopathy and its complications[3].

1.4 Prevention of Diabetic Retinopathy

While diabetic retinopathy cannot be completely prevented, there are several measures that can be taken to reduce the risk of developing the condition or delaying its progression. Some preventive strategies for diabetic retinopathy include:

- **Strict blood sugar control:** Maintaining optimal blood sugar levels through a healthy diet, regular exercise, and appropriate use of diabetes medications or insulin as prescribed by a healthcare provider can help reduce the risk of diabetic retinopathy.
- **Regular eye exams:** Diabetes individuals should undergo comprehensive eye exams at least once a year, or as recommended by their eye care specialist. Early detection and treatment of diabetic retinopathy can help prevent or slow down its progression.
- **Blood pressure management:** Keeping blood pressure within the target range, as advised by a healthcare provider, can reduce the risk of diabetic retinopathy. This may involve lifestyle modifications, medications, and regular blood pressure monitoring.
- **Cholesterol management:** Maintaining healthy cholesterol levels, particularly LDL (low-density lipoprotein) cholesterol, through lifestyle modifications.
- **Smoking cessation:** Quitting smoking or avoiding exposure to secondhand smoke is crucial in preventing diabetic retinopathy, as smoking can damage the blood vessels in the retina and exacerbate the condition.

- **Regular physical activity:** Engaging in regular physical activity, as advised by a healthcare provider, can help manage blood sugar levels, blood pressure, and cholesterol levels, which in turn can lower the risk of diabetic retinopathy.
- **Medication adherence:** Following prescribed medication regimens for diabetes, blood pressure, and cholesterol management as recommended by a healthcare provider can help reduce the risk of diabetic retinopathy.
- **Healthy lifestyle:** Maintaining a healthy lifestyle that includes a balanced diet, regular exercise, adequate sleep, stress management, and avoiding excessive alcohol consumption can contribute to overall health and reduce the risk of diabetic retinopathy.
- **Pregnancy management:** Pregnant women with diabetes should work closely with their healthcare provider to manage blood sugar levels during pregnancy and undergo regular eye exams to monitor for any signs of diabetic retinopathy.

It's important to note that these preventive measures are general guidelines and may need to be tailored to an individual's specific health condition and medical history. Consulting with a qualified healthcare professional or an eye care specialist is crucial for personalized advice on preventive strategies for diabetic retinopathy[4].

1.5 DR-Grading-Based Studies

As per the International Clinical Diabetic Retinopathy (ICDR) scale, diabetic retinopathy can be graded into separate grades:

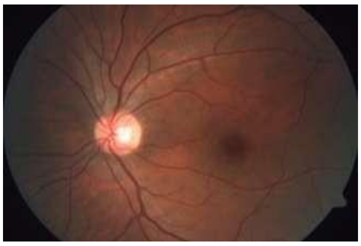

Fig 1.2 Normal DR

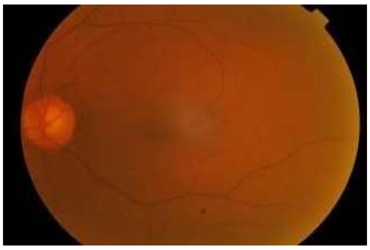

Fig 1.3 Mild DR

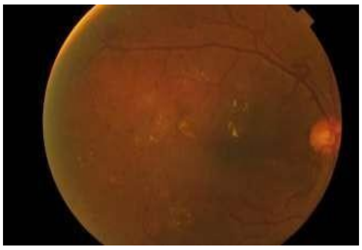

Fig 1.4 Moderate DR

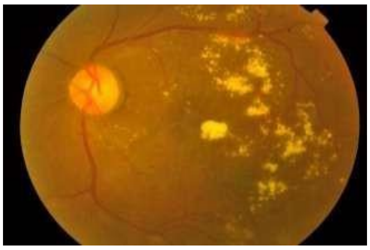

Fig 1.5 Severe DR

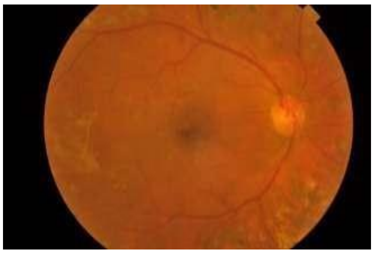

Fig 1.6 Proliferative DR

no apparent retinopathy, mild no proliferative diabetic retinopathy (NPDR), moderate NPDR, severe NPDR, and proliferative diabetic retinopathy (PDR). An example of each grade is presented. Many studies have been proposed for multi-class classification and grading of fundus images into the above-mentioned five stages. (a) Normal (b) Mild (c) Moderate (d) Severe (e) DR. The five types of diabetic retinopathy. A simple CNN model was used by the authors after applying a green channel filter to assess the stage of DR from fundus images. A CNN, which combined multi-view fundus images, was used along with attention mechanisms by the authors. It was called MVDRNet and used VGG-16 as the basic network. A locally collected dataset containing multi-view fundus images was employed for this. Another study that used a locally collected dataset from the University Hospital Saint Joan, Tarragona, Spain. The CNN model used had batch normalization followed by the ReLU function. This was followed by a linear classifier and a softmax function. Two datasets—a balanced dataset with no augmentation and another one with augmentation—were used by the authors.

A CNN was used to demonstrate the improvement in accuracy in DR grading due to the augmentation. Agustin and Sunyoto performed a comparison of different regularization methods regarding how they reduce the overfitting of CNNs when used for DR severity. Dropout regularization was found to reduce overfitting and to increase accuracy[5].

CHAPTER 2

LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

1. In "Novel Framework for Enhanced Learning-based Classification of Lesion in Diabetic Retinopathy"[6] Prakruthi M K , Kumaraswamy G the proposed scheme introduces a computational framework where a simplified feature enhancement operation is carried out, resulting in artifact-free images with better features. The enhanced image is then subjected to CNN to perform multiclass categorization of potential stages of diabetic retinopathy to see if it outperforms existing schemes.

2. In "Diabetic retinopathy detection through deep learning techniques"[7] Wejdan L. Alyoubi , Wafaa M. Shalash, Maysoon F. Abulkhair Automated systems for DR detection play an important role in detecting DR at an early stage. The DR stages are based on the type of lesions that appear on the retina. This article has reviewed the most recent automated systems of diabetic retinopathy detection and classification that used deep learning techniques.

3. Shailesh Kumar, Basant Kumar proposed a diabetic retinopathy detection scheme by extracting accurate area and ate number of microaneurysm from color fundus images[8]. For detection of microaneurysms, principal component analysis (PCA), contrast limited adaptive histogram equalization (CLAHE), morphological process, averaging ltering have been used. Classification of DR has been done by linear Support vector machine (SVM). The sensitivity and specificity of DR detection system are observed as 96% and 92% respectively.

4. In another research paper, hard exudates in retinal fundus images are employed to classify moderate and severe non-proliferative diabetic retinopathy. The hard exudates are segmented using mathematical morphology and the extracted features are classified by using soft margin SVM. The classification model achieves an accuracy of 90.54% for 75 training data and 74 testing data of retinal images. Several methods have been proposed based on the deep neural network for the classification of Diabetic retinopathy based on severity A major difficulty of fundus image classification using the deep neural network is high variability, especially in the case of retinal proliferation and retinal detachment of new blood vessels, which lowers the accuracy of the network.

5. Currently, various approaches are being evolved for classifying diabetic retinopathy stages. From this perspective, it is seen that machine learning has always played a dominant role. This can be seen in the review presented by Atwany et al[9].

6. Improving Dermoscopic Image Segmentation with Enhanced Convolutional-Deconvolutional Networks, 2017 by Yading Yuan and Yeh-Chi Lo. Automatic skin lesion segmentation on dermoscopic images is an essential step in the computer-aided diagnosis of melanoma. However, this task is challenging due to significant variations of lesion appearances across different patients. This challenge is further exacerbated when dealing with a large amount of image data. In this paper, we extended our previous work by developing a deeper network architecture with smaller kernels to enhance its discriminant capacity. In addition, we explicitly included color information from multiple color spaces to facilitate network training and thus to further improve the segmentation performance[10].

7. The recent work of Abdelsalam and Zahran used a Support Vector Machine using a multifractal-based geometry system to diagnose and classify. The method has also used lacunarity parameters for accomplishing singular decisions[11].

8. Another work carried out by Li et al. has used an attention network with a unique grading system to identify the condition of macular edema. This paper aims to learn features based on disease-specific and disease-dependent attributes selectively. The feature maps were constructed using Convolution Neural Network (CNN) with different resolutions. Study towards quantification in diabetic retinopathy is carried out by Okuwobi et al., where region-of-interest is used, followed by estimation of hyperreflective foci to obtain better segmentation. The adoption of deep learning is also reported in work carried out by Qiao et al[12].

9. The work implemented by Gayathri et al. has presented a unique multiclass classification with the automated binary system[13]. The study has used multiresolution features and different ranges of classifier e.g. J48, random trees, random forest, support vector machine, etc., over multiple datasets. The idea is to present a unique feature extraction model for assisting the binary classification of retinal fundus images.

10. Further adoption of deep learning was witnessed in the work by Wang et al, which has addressed the non-interpretability issues in its outcome. The model used the Kappa coefficient to assess the features of diabetic retinopathy. The idea is also to determine the severity score and build a relationship between severity scores and their corresponding features. The adoption of the neural network is seen in Zang et al, where a rate dropout is designed to suppress the overfitting problem during classification.

11. Deep Residual Learning for Image Recognition, 2015 by Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun[14]. Deeper neural networks are more difficult to train. We present a residual learning framework to ease the training of networks that are substantially deeper than those used previously. We explicitly reformulate the layers as learning residual functions with reference to the layer inputs, instead of learning unreferenced functions. We provide comprehensive empirical evidence showing that these residual networks are easier to optimize, and can gain accuracy from considerably increased depth. On the ImageNet dataset we evaluate residual nets with a depth of up to 152 layers—8× deeper than VGG nets but still having lower complexity.

CHAPTER 3

PROBLEM STATEMENT

CHAPTER 3

PROBLEM STATEMENT

One of the most severe effects of diabetes is diabetic retinopathy which can result in total blindness if ignored. Early detection is a major obstacle because it is crucial to the success of treatment. Millions of people can benefit from the detection of early-stage diabetic retinopathy, which is this project's main focus. Unfortunately, accurate diagnosis of the diabetic retinopathy stage is notoriously difficult and necessitates skilled human fundus imaging interpretation. Convolutional neural networks (CNN) model can be trained by using training datasets, it is used to diagnose diabetic retinopathy and it will give the probability of the eye infected with diabetics. By employing fundus retinal images, we suggest a deep-learning-based technique for diabetic retinopathy stage of severity detection.

3.1 Challenges

- **Finding the most suitable dataset**: Out of the wide variety of datasets available on the web the most suitable dataset which can give the most effective results should be chosen for pre-processing.
- **Spatial sizes and multiple aspect ratio:** As we have taken over 2,000 images for pre-processing converting all the images to a uniform size for easy comparison was another challenge.
- **Deciding the perfect Algorithm:** Out of several algorithms available for filtering, feature enhancement selecting the best combination of algorithms for our dataset was another obstacle.
- **Need for specialized equipment and expertise:** Diagnosing diabetic retinopathy requires specialized equipment such as ophthalmoscopes, fundus cameras, or optical coherence tomography (OCT) machines, as well as expertise in interpreting the results. These resources may not be readily available in all healthcare settings, especially in remote or underserved areas, which can hinder timely and accurate diagnosis.
- **Variability in Image Quality:** The quality of the retinal images obtained might not fit the desired pixel size. Poor image quality can make it difficult for healthcare providers to accurately assess the retina and detect diabetic retinopathy[15].

3.2 Motivation

Diabetic Retinopathy (DR) is one of the common eye diseases and is a diabetes complication that affects the eyes. Diabetic retinopathy may cause no symptoms or only mild vision problems. Eventually, it can cause blindness. So early detection of symptoms could help to avoid blindness. If detect diabetic retinopathy quickly and we can easily follow up treatments such as laser therapy, anti-vascular endothelial growth factor (VEGF) injections, or other interventions to be implemented. Detecting diabetic retinopathy in its early stages can increase the chances of preserving vision and maintaining a good quality of life.

The increasing number of diabetic retinopathy cases worldwide requires intensified efforts in developing tools to assist in the diagnosis of diabetic retinopathy. Technical detection of diabetic retinopathy will lead to a large amount of savings of time and effort[16].

3.3 Objectives

- To investigate and analyze the normal Diabetic Retinopathy digital fundus images from publicly available datasets.
- To explore the image pre-processing and image enhancement techniques.
- To design and develop a model which detects the presence of Diabetic Retinopathy and if present classifies whether the diabetic retinopathy is Mild, Moderate, Sever, or Proliferative.
- To evaluate the proposed model with the existing models[16].

CHAPTER 4

REQUIREMENT ANALYSIS

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Hardware Requirements

- System: Intel Core i5 processor
- Hard Disk: 120GB
- Monitor: 15.6’’ Full HD
- Input Devices: Mouse, Keyboard
- Ram: 8GB

4.2 Software Requirements

- Python Version 3.8 or above
- Anaconda Latest version
- Python libraries numpy, pandas, OpenCV, matplotlib
- Keras and TensorFlow latest versions
- Jupyter notebook to train and run the CNN model.

4.3 Software Description

4.3.1 Python:

Python is a high-level, interpreted programming language that was created in the late 1980s by Guido van Rossum. It is named after the famous British comedy group Monty Python, as Guido was a big fan of their work. Python is now one of the most popular programming languages in the world, used by developers for a wide range of tasks such as web development, data analysis, artificial intelligence, scientific computing, and more.

Python is known for its simplicity and ease of use, with a syntax that is easy to learn and read. It is also an interpreted language, which means that code can be executed directly without the need for compilation. Python is dynamically typed, meaning that you don't have to declare the data type of a variable before using it.

Python has a large and active community of developers who contribute to its development and create a wide range of libraries and tools that can be used for various tasks. Python's standard library provides a rich set of modules that can be used for tasks such as file I/O, networking, web programming, and more.

Python is known for its simplicity and ease of use, with a syntax that is easy to learn and read.

Some of the striking utilizations of Python are:

- Helps in server-side programming in Web improvement.

- Used tremendously in programming advancement where various complex issues should be settled.
- Used in insights.
- Used for System scripting.
- Python bolsters database network and can likewise peruse from, write to and annex documents.
- It can be utilized in taking care of huge information and furthermore to determine complex scientific issues.
- It is additionally utilized for programming prototyping in the product advancement process[17].

4.3.2 Anaconda

Anaconda is an integrated development environment (IDE) for the Python programming language. It is designed to provide an easy-to-use environment for scientific computing and data analysis. Spyder includes many features that make it a powerful tool for Python development, including:

- Interactive console: Spyder has an interactive console that allows you to run Python code and see the results immediately. This is useful for experimenting with code and debugging.
- Code editor: Spyder's code editor includes many features to make coding easier, such as syntax highlighting, code completion, and code folding.
- Variable explorer: Spyder's variable explorer allows you to view and manipulate the variables in your code. This is useful for debugging and data analysis.
- Debugger: Spyder includes a debugger that allows you to step through your code and find and fix errors.
- Plots and graphs: Spyder include a built-in plotting library called matplotlib, which allows you to create plots and graphs of your data.
- Integration with other tools: Spyder can be used with other Python libraries and tools, such as NumPy, SciPy, and pandas.

Anaconda is available for Windows, macOS, and Linux, and is free to download and use. It is a popular choice for scientific computing and data analysis in the Python community.

Anaconda was made by Pierre Raybaut in 2009 and post 2015. Spyder has been kept up and unendingly improved by a gathering of intelligent pythin engineers and a submitted network for the equal.

- PyFlakes

- Pylint
- Rope

For GUI, Spyder utilizes Qt. It is integrated to utilize both PyQt or PySide Python ties. A slim reflection layer created by the Spyder venture and later embraced by various bundles and gives the opportunity to utilize either backend is called QtPy. It is a python system used to create basic frontend GUIs for python activities and projects created in Python.

As a cross-stage IDE it is accessible through:

- Anaconda
- Windows
- macOS through MacPorts

and on significant Linux conveyances, for example:

- Debian
- Arch Linux
- Gentoo Linux
- Fedora
- openSUSE
- Ubuntu

Anaconda can be extensible with first gathering and outsider modules and has the game plan for instinctive gadgets for data audit and embeds python unequivocal code quality confirmation and thought fulness instruments[18].

CHAPTER 5

DESIGN AND ARCHITECTURE

CHAPTER 5

DESIGN AND ARCHITECTURE

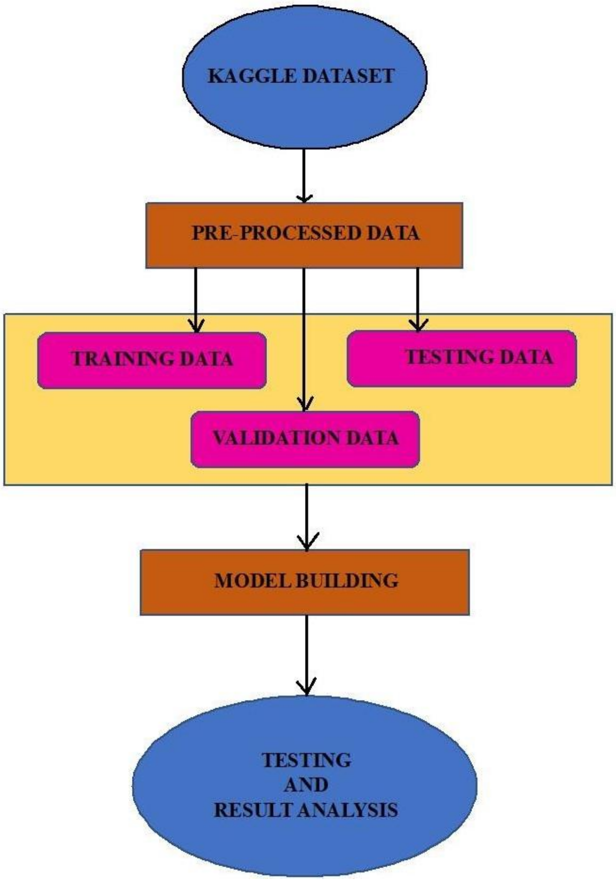

Fig 5.1: Flowchart of proposed model

5.1 Dataset

The primary aim of this proposed model is to develop a “Integrated Image Framework. For Diagnosing Diabetic Retinopathy” which gives the good accuracy score and efficiency as compared to previously designed model. In this model input is taken in the form of fundus images from Kaggle dataset i.e APTOS 2019 blindness detection[19].

5.2 Pre-processing Data

Input images undergo image processing which results in proper enhance images. Contrast enhanced image act as the input for the CNN and results in the various stages of diabetic retinopathy from which the patient is suffering from in the form of graph[20].

5.3 Training, Testing and Validation of Data

Training: The training step is where a machine learning model is developed using a labeled dataset. The labeled dataset consists of input data and corresponding output labels. The model is trained to predict the output labels for new input data based on patterns it learns from the

labeled dataset. During training, the model adjusts its internal parameters to minimize the difference between its predicted output labels and the actual output labels in the training dataset.
Validation: The validation step is used to evaluate the performance of the trained model on new, unseen data. The validation dataset is a subset of the labeled dataset that is not used during training. The model's performance is evaluated on the validation dataset to ensure that it can generalize well to new data that it has not seen before. The validation dataset is also used to tune hyperparameters, which are parameters that are not learned during training but affect the behavior of the model.

Testing: The testing step is used to evaluate the final performance of the model after it has been trained and validated. The testing dataset is another subset of the labeled dataset that is not used during training or validation. The model's performance is evaluated on the testing dataset to provide an unbiased estimate of its performance on new, unseen data[21].

5.3 Model Building, Testing and Result Analysis

Training model is all set and ready to test for new input data. Different ML techniques are applied and based on the predictions metrics and accuracy best model is decided[21].

CHAPTER 6

METHODOLOGY

CHAPTER 6

METHODOLOGY

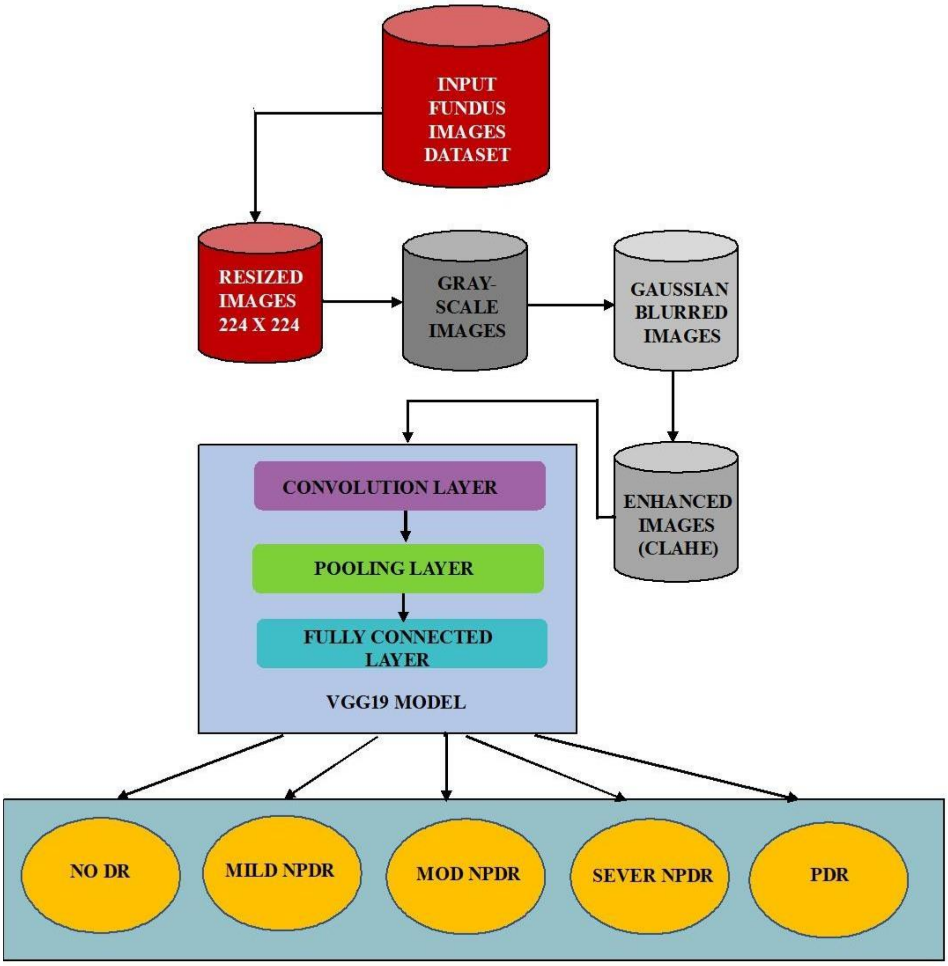

Fig 6.1: Proposed Method for Diabetic Retinopathy Diagnosis

A range of techniques is used for the picture pre-processing on the selected image informative gathering inside the application areas of image handling. The main goal of this study is to establish a strategy for dealing with ways to make the Dental image more comprehensible for subsequent conclusions. The most successful method for detecting cancer early on is the chosen automated MRI. The MRI images used in this investigation's work are regular images compiled from Gemini Scans. The images are created using two MRIs, each of which focuses on the patient's right lung and the opposite side. The informative list has two open categories of images: trouble and begin. A picture improvement technique is a plan for handling an image with the aim of producing a result that is more appropriate than the necessary image for a certain application[22].

6.1 Dataset Collection

In this model, input is taken in the form of fundus images from Kaggle dataset i.e APTOS 2019 blindness detection.

Kaggle is a platform for data science competitions, datasets, and machine learning projects. It was founded in 2010 by Anthony Goldbloom and Ben Hamner and acquired by Google in 2017. Kaggle provides a community of data scientists and machine learning practitioners with access to high-quality datasets, tools for building and sharing models, and a platform for hosting and participating in data science competitions.

On Kaggle, data science competitions are hosted by companies and organizations looking for innovative solutions to specific problems. Competitors build and submit models to solve the problem, and the results are evaluated against a holdout dataset to determine the winner. The competitions are typically open to anyone, and the winners receive cash prizes, job offers, or other incentives.

In addition to competitions, Kaggle also provides a platform for data sharing and collaboration. Users can upload and share datasets, notebooks, and code, and can collaborate on machine learning projects with other users. Kaggle also offers a wide range of resources and tutorials for data science and machine learning, including courses, videos, and blog posts.

Kaggle has become a popular platform for data scientists and machine learning practitioners to develop and showcase their skills, and has been used to tackle a wide range of real-world problems in fields such as healthcare, finance, and education[23].

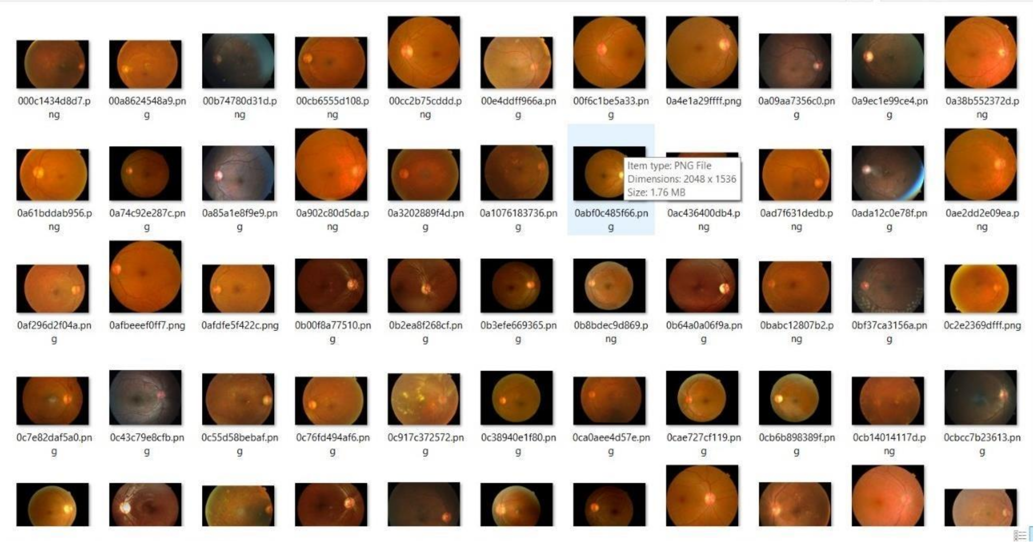

Fig 6.2 : Datasets

6.2 Image pre-processing

Image pre-processing is an essential step in deep neural network applications involving image data. It involves transforming raw input images into a form that can be fed into a deep neural network. The goal of image pre-processing is to standardize the data and remove any irrelevant information that may hinder the performance of the deep neural network.

Here are some common image pre-processing techniques used in deep neural network applications:

- Resizing and cropping: Images are often resized and cropped to a standard size to reduce computational complexity and make training more efficient.
- Normalization: Image pixel values are typically normalized to ensure that they fall within a certain range (usually between 0 and 1). This is important for ensuring that the deep neural network can learn the underlying patterns in the data.
- Data augmentation: Data augmentation is the process of artificially increasing the size of the training dataset by applying random transformations to the images, such as rotations, translations, and flips. This can help prevent overfitting and improve the generalization performance of the deep neural network.
- Color space conversion: Images may be converted from one color space to another (such as from RGB to grayscale) to reduce computational complexity and remove irrelevant information.
- Noise reduction: Pre-processing techniques such as denoising can help reduce the impact of noise in images and improve the performance of the deep neural network.

  Overall, image pre-processing is an important step in deep neural network applications involving image data. It can help standardize the data and remove any irrelevant information that may hinder the performance of the deep neural network[24].

6.2.1 Resizing and Gray Scaling the Image

The creation and reduction of the specified image size in pixel location is a crucial task in picture maintenance strategy. Picture contribution may be broken down into two unique categories: picture up- and down-examining. Both processes are crucial when resizing the data to organize either the correspondence channel or the output display. While sending low resolution forms to clients is more effective, providing the final visual data may need estimating the initial high resolution. The precise scaling of image data is a crucial advancement in many applications, from simple consumer items to critical constraints in the medicinal security and defense sectors. The speed of resizing can be improved with the use of approach experiences since the next image frequently has block artefacts, which are not immediately obvious but commonly also have a negative impact on the techniques that are being compared. For noise elimination, filtering techniques like Gaussian and median channel are employed. By converting colored photographs to grayscale ones, noise may be easily removed[25].

6.2.2 Filtering Image – Gaussian Blur

The most common filtering techniques for enhancing images focus on either the high frequencies of the image, such as smoothing it, or the low frequencies of the image, such as enhancing or detecting its edges. For example, you may filter a photograph to emphasize details or remove specific highlights. There are several approaches, therefore the quality that can be anticipated relies on the image and how it will be used. In some applications, such as edge recognition, noise elimination, sharpening, and edge smoothing, picture filtering is crucial. filtration techniques Since input photographs are affected by many types of noise, such as Gaussian channel, Median channel, etc., noise picture processing is controlled to enhance image quality through filtering.

Gaussian: In both applications and hypotheses, Gaussian filtering is expected to play the most important role. Gaussian sifting, which may be a WAP with weight defined as Where decides the separation of the fast rot, is a frequently used image isolating approach. In essence, Gaussian smoothing is a close-by filtering technique. Gaussian channel is seen to overly smooth photographs when used as a picture denoise estimate, resulting in the loss of genuine detail, particularly edge sharpness. The purpose of the Gaussian channel is to remove noise from the input. The fall time and rise time are getting shorter since this channel has an input that works stepwise without overshooting. So, according to this movement of conduct, the filter will wait as little as possible.

According to science, a Gaussian filtering method responds to the direction of contribution to the methodology for drawing a curve from a Gaussian capacity. As a result of Gaussian haze, the Gaussian capacity is now used to obstructing images. The effect of illustration programming is frequently known to reduce visual noise. The Gaussian distribution (Gx) with standard deviation (σ) for one estimation and two estimations is addressed in the following two equations, where (x) is an image point for one estimation and (x,y) is a pixel for the other[26].

6.2.3 Contrast Enhancement – CLAHE

Contrast enhancement is a type of image processing technique that is used to improve the contrast of an image by adjusting the intensity of its pixels. The main goal of contrast enhancement is to make the image more visually appealing, by increasing the difference between the light and dark areas of the image.

There are various techniques that can be used for contrast enhancement, including linear and nonlinear methods. Linear methods, such as histogram equalization, adjust the intensity of the pixels in a linear manner. Nonlinear methods, such as adaptive histogram equalization and contrast stretching, adjust the intensity of the pixels in a nonlinear manner, which can produce more natural-looking images.

Contrast enhancement is commonly used in a wide range of image processing applications, including medical imaging, satellite imaging, and computer vision. It can be used to improve the visibility of important features in an image, enhance the appearance of textures and patterns, and improve the overall quality of the image.

Different enhancement techniques are evaluated to adjust contrast and non-linear illumination in the input image. The power law-based contrast correction is extensively adopted in the existing literature. Novel Image Framework for Diagnosing Diabetic Retinopathy 7 December 2022-2023 www.retino.org It has the ability of contrast adjustment by control ling the gamma correction factor and constant C. For image enhancement, the power law-based contrast correction mechanism is given as: FIs = C × FI γ (8) Where, FIs is the high contrast intensity output image obtained from power law factor γ , where if $\gamma < 1$ mapping towards high brightness, r>1 towards darker. The next technique evaluated is CLAHE ("Contrast Limited Adaptive Histogram Equalization").

CLAHE (Contrast Limited Adaptive Histogram Equalization) is a popular image processing technique used for contrast enhancement in digital images. It is an adaptive method that improves the contrast of an image by redistributing the pixel intensities in a way that increases the local contrast while preventing the over-amplification of noise.

The basic idea behind CLAHE is to divide the image into small regions called tiles, and then perform histogram equalization on each tile separately. However, to prevent over-enhancement of the contrast, the pixel intensities are clipped at a certain level, which is based on the maximum value of the histogram. This limits the amount of enhancement that can be performed on each tile, which results in a more natural-looking image.

One advantage of CLAHE over other contrast enhancement techniques is its adaptiveness to local image features. This means that the amount of contrast enhancement applied to each tile is based on the image content within that tile, rather than being applied uniformly across the entire image.

CLAHE is commonly used in medical imaging applications such as X-ray and MRI images, as well as in satellite and aerial imaging, where improving the contrast of images can be crucial for effective analysis and interpretation of data[27].

6.3 Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is a type of deep neural network that is widely used in computer vision tasks, such as image recognition and object detection. It is designed to process input data with a grid-like topology, such as a two-dimensional image.

CNNs use a process called convolution to extract features from an image. This involves sliding a small window, called a kernel or filter, over the image and performing element-wise multiplication and summation operations between the filter and the input image pixels. This produces a new feature map, which can then be passed through additional layers of the network for further processing.

CNN’s typically consist of multiple convolutional layers, followed by pooling layers, which reduce the dimensionality of the feature maps. The output from the last pooling layer is then flattened and passed through one or more fully connected layers, which performs classification or regression tasks.

CNNs have shown state-of-the-art performance on a wide range of computer vision tasks, such as image classification, object detection, and segmentation. They have also been applied to other domains, such as natural language processing and speech recognition, with some success.

You can combine two functions in a particular kind of linear operation to demonstrate how the form of one function can be changed by another. Simply stated, two images that are represented as two matrices are multiplied to create the output in order to extract information from an image. While CNNs are comparable to other neural networks, they complicate the equation by using a series of convolutional layers. Without convolutional layers, CNN is unable to operate.

A variety of computer vision apps have benefited greatly from CNN artificial neural networks. A backpropagation algorithm is used by a convolutional neural network to acquire spatial hierarchies of data automatically and adaptively. It has several layers, including completely connected, pooling, and convolution layers.

ConvNet's job is to keep elements necessary for obtaining an accurate prediction while condensing the images into a more manageable format. This is essential for developing an architecture that can scale to very large datasets and acquire features[28].

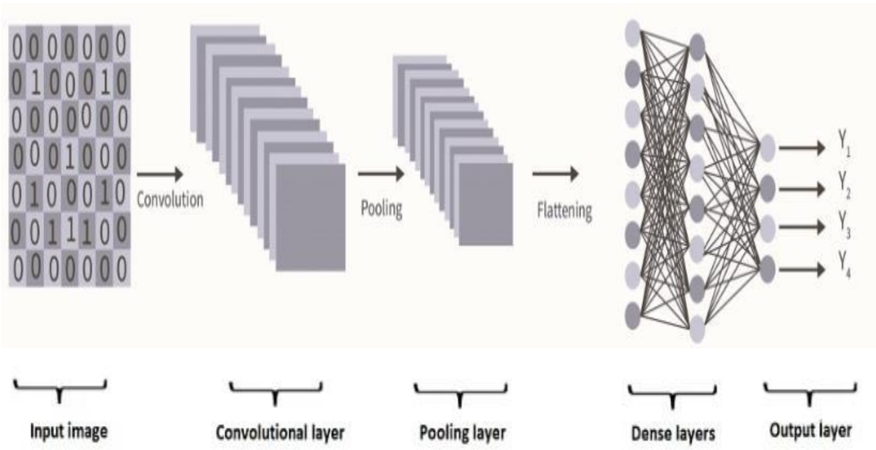

Fig 6.3: CNN Architecture

6.3.1 The Convolutional Layer (CONV)

The convolutional layer is the fundamental building block of Convolutional Neural Networks (CNNs). It performs a mathematical operation called convolution on the input image or feature map to produce a set of output feature maps.

During convolutional, the layer applies a set of learnable filters, also known as kernels or weights, to the input. Each filter slides over the input and computes the dot product between its weights and the corresponding input pixels. The result is a single value in the output feature map, which represents the filter's response to a particular feature in the input.

In a single convolutional layer, multiple filters can be applied to produce multiple output feature maps. Each output feature map captures a different aspect of the input data, such as edges, corners, or textures.

The size of the output feature maps is determined by the size of the input, the size of the filters, the stride, and the padding. Stride determines how much the filter is shifted each time it moves across the input, while padding adds additional values around the input to preserve the spatial dimensions of the output feature map.

Convolutional layers are typically followed by non-linear activation functions, such as ReLU (Rectified Linear Unit), to introduce non-linearity and increase the representational power of the network.

Convolutional layers are essential for image and video processing tasks and have been shown to be very effective in computer vision problems like image classification, object detection, and segmentation[29].

$f(x) = \max(0, x)$

6.3.2 Pooling Layer (POOL)

A pooling layer is a type of layer commonly used in Convolutional Neural Networks (CNNs) to reduce the spatial dimensionality of the input feature maps. It is typically inserted after one or more convolutional layers to help reduce the size of the data being processed, which can help improve computational efficiency and reduce overfitting.

Pooling operates on each feature map independently by dividing it into non-overlapping sub-regions (e.g. 2x2 or 3x3), and then computing a single output value for each sub-region. The output value is typically the maximum value in the sub-region (Max Pooling), the average value (Average Pooling), or the L2 norm (L2 Pooling).

The main purpose of pooling is to reduce the spatial dimensionality of the input feature map while preserving the important features. Pooling also helps to introduce translational invariance in the network by reducing the sensitivity of the output to small translations of the input.

Pooling layers are typically followed by convolutional layers to further process the reduced feature maps. They are an important component of many CNN architectures and have been shown to be effective in computer vision tasks such as image classification, object detection, and segmentation.

Pooling can be divided into two categories: maximum and average pooling. The kernel-covered area of the image has the highest number, which is returned by max pooling. The average of all the numbers in the kernel-covered region of the image is produced by average pooling[30].

6.3.3 Fully Connected Layer (FC)

A fully connected layer, also known as a dense layer, is a type of layer commonly used in neural networks, including Convolutional Neural Networks (CNNs). Unlike convolutional and pooling layers, which operate on a local region of the input data, fully connected layers process the entire input at once.

In a fully connected layer, every neuron in the layer is connected to every neuron in the previous layer. Each neuron in the layer performs a weighted sum of the inputs it receives from the previous layer, adds a bias term, and applies an activation function to produce the output. The output of a fully connected layer can be thought of as a high-level representation of the input data, which can then be used for classification, regression, or other tasks.

In a CNN, fully connected layers are often used at the end of the network to perform the final classification or regression task based on the features extracted by the convolutional and pooling layers. The output of the last pooling layer is typically flattened into a one-dimensional vector and passed through one or more fully connected layers before the final output is present.

The number of neurons in a fully connected layer and the number of layers in a network depend on the complexity of the problem being solved and the size of the input data. Deep neural networks with many fully connected layers are capable of modeling highly complex functions, but can also be prone to overfitting if the training data is limited.

Fully connected layers are a key component of many neural network architectures and have been shown to be effective in a wide range of tasks, including image classification, natural language processing, and speech recognition[31].

6.3.4 Dropout Layer

Dropout is a regularization technique that is often used in deep neural networks, including Convolutional Neural Networks (CNNs), to prevent overfitting. The dropout layer is a type of layer that implements this technique.

During training, dropout randomly “drops out” (i.e., set to zero) some of the neurons in the layer with a probability p. The neurons that are dropped out are selected randomly for each batch of data that is processed, which means that the network is forced to learn a more robust and generalizable set of features that are not dependent on specific neurons.

The dropout layer is typically inserted after a fully connected layer, but can also be used after a convolutional layer. During training, the dropout layer is applied to the output of the previous layer, and during inference (i.e., testing), the dropout layer is typically removed or turned off.

The dropout layer has been shown to be an effective regularization technique in deep neural networks and can help prevent overfitting, which is a common problem in deep learning. By randomly dropping out neurons during training, the dropout layer helps to reduce the co-adaptation of neurons, which can lead to overfitting.

The dropout layer is a widely used technique in deep learning and has been shown to be effective in a variety of applications, including image classification, speech recognition, and natural language processing[32].

6.4 Visual Geometry Group (VGG)

The abbreviation VGG, which means for Visual Geometry Group, refers to a deep CNN design that is typical in that it has several layers. The number of layers is referred to as "deep," and VGG-16 or VGG-19, accordingly, have 16 or 19 convolutional layers. Utilizing the VGG architecture, creative object recognition models are created. On several tasks and datasets outside of ImageNet, the VGGNet, developed as a deep neural network, outperforms benchmarks. Additionally, it is still one of the most popular picture recognition architectures in use today. The VGG19 model, also referred to as VGGNet-19, is conceptually like the VGG16 model except that it allows 19 layers. The weight layers of the model are denoted by the numerals "16" and "19." (Convolutional layers). VGG19 has three more convolutional layers than VGG16 does. The characteristics of the VGG16 and VGG19 networks will be covered in more depth in the essay's concluding section[33].

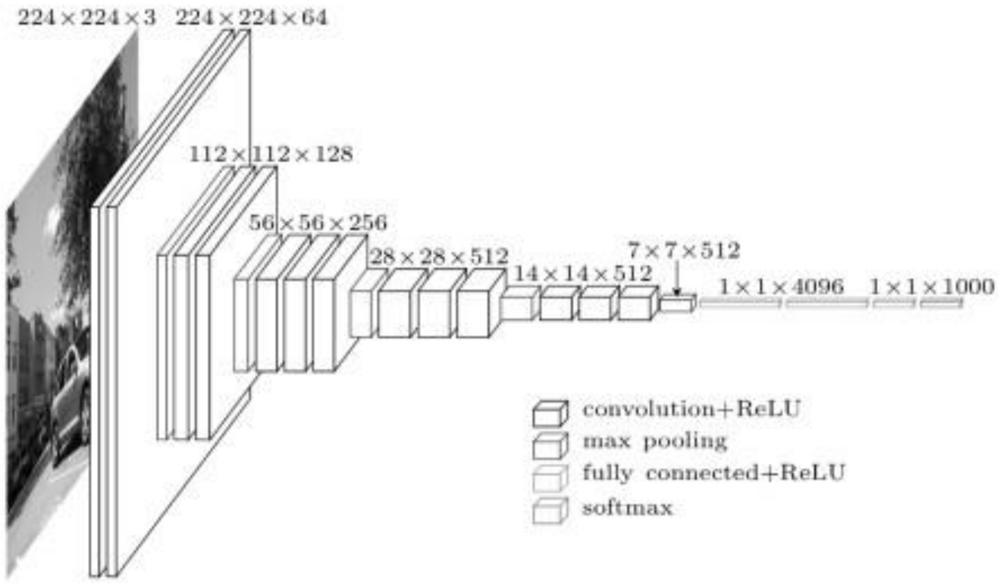

Fig 6.4: VGG Architecture

6.5 Accuracy

Accuracy is simpler to comprehend. It determines how well our model predicts by comparing the model predictions with the real values in terms of a percentage. The model is likely to make substantial errors in most of the data if accuracy is low but loss is high. However, loss and accuracy must also be kept to a minimum if the model produces small errors in the bulk of the data. However, in some instances it results in significant data errors if they are both high. The model performs best when accuracy is high and loss is low, which results in only minor mistakes on a small subset of the data[34].

	Low Loss	**High Loss**
Low Accuracy	A lot of small errors	A lot of big errors
High Accuracy	A few small errors	A few big errors

Table 6.5: Accuracy Information

6.6 Loss

It is a figure that represents all the errors in our model. It assesses how well our algorithm performed. If the errors are numerous, the model is not performing well because the loss will be large. Otherwise, the lower it is, the better our model works. The loss is computed using a cost or loss function. There are a variety of cost algorithms available. The most appropriate one to use will rely on the circumstances because they each have unique penalties for errors. Cross Entropy and Mean Squared Error, respectively, are the measures that are most frequently used for classification and regression problems. This is necessary for the model to be able to train using the loss function in deep learning. The goal of the programme is to lessen the magnitude of the loss. This is achieved by using techniques like gradient descent, which alters the model's parameters in response to the result of the loss. By monitoring the loss over time, our models can generate some intriguing findings. If the loss number oscillates instead of decreasing, the model might not be learning at all. But if the model declines in the training set but not in the validation set, it might be overfitting (Or decreases but there is a noticeable difference) In other words, it might have taken too much information from the training instances and become useless[34].

CHAPTER 7

IMPLEMENTATION

CHAPTER 7

IMPLEMENTATION

Tensorflow:

TensorFlow is an end-to-end open-source platform for machine learning. It has a comprehensive, flexible ecosystem of tools and libraries, and community resources that help developers easily build and deploy deep learning applications. It is a python library that can be used to create deep learning modules directly or by using wrapper libraries. This provides a collection of workflows to develop and train modules using python[35].

Keras:

Keras is an open-source software library that provides a python interface for artificial neural networks. It acts as an interface for the TensorFlow library. The Keras functional API provides a more flexible way of defining modules. It specifically allows defining multiple input or output modules that share layers. It not only supports convolutional networks and recurrent networks individually but also in their combination[35].

Numpy:

Numpy stands for numerical python. It is the library for python programming language adding support for large, multi-dimensional arrays and matrices along with the large collection of high-level mathematical functions to operate on these arrays. It aims to provide an array object that is up to 50x faster than the traditional python lists[36].

Pandas:

Pandas an open-source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named Numpy, which provides support for multi-dimensional arrays. Pandas is a Python library. Pandas are used to analyze data[36].

Pathlib:

Pathlib module in Python provides various classes representing file system paths with semantics appropriate for different operating systems. This module comes under Python's standard utility modules. Path classes in Pathlib module are divided into pure paths and concrete paths. Pure paths provides only computational operations but does not provides I/O operations, while concrete paths inherit from pure paths provides computational as well as I/O operations[37].

PIL:

Python Imaging Library (expansion of PIL) is the de facto image processing packagefor Python language. It incorporates lightweight image processing tools that aids in editing, creating and saving images. Support for Python Imaging Library got discontinued in 2011, buta project named pillow forked the original PIL project and added Python3.x support to it. Pillow was announced as a replacement for PIL for future usage. Pillow supports a large number of image file formats including BMP, PNG, JPEG, and TIFF. The library encourages adding support for newer formats in the library by creating new file decoders[37].

Matplotlib:

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc[37].

```
In [36]: import cv2
         import glob
         import os
         from numpy import *
         from PIL import Image
         from PIL import Image
         from PIL import ImageOps
         import numpy as np
         DirPath="Input_images"
         Files=os.listdir(DirPath)
         i=1
         for File in Files:
             imgPath=os.path.join(DirPath,File)
             img = cv2.imread(imgPath)
             imgRes=cv2.resize(img,(150,150))
             cv2.imwrite(f'./val_data/Sever_NPDR/image%i.png'%i,imgRes)
             i+=1
```

Fig 7.1: Resizing of Image

```
In [19]: from PIL import Image,ImageFilter
         import numpy as np
         DirPath="Resized_images"
         Files=os.listdir(DirPath)
         i=1
         for File in Files:
             imgPath=os.path.join(DirPath,File)
             img = cv2.imread(imgPath)
             gray_img=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
             cv2.imwrite(f'./Gray_Scale_images/image%i.png'%i,gray_img)
             i+=1
```

Fig 7.2: GrayScaling of Image

```
In [20]: import cv2
         import matplotlib.pyplot as plt
         import skimage.filters
         import numpy as np
         DirPath="Gray_Scale_images"
         Files=os.listdir(DirPath)
         i=1
         for File in Files:
             imgPath=os.path.join(DirPath,File)
             img1 = cv2.imread(imgPath)
             img_RGB=cv2.cvtColor(img1,cv2.COLOR_BGR2RGB)
             blur = cv2.GaussianBlur(img_RGB,(11,11),0)
             img_RGB1=cv2.cvtColor(img1,cv2.COLOR_BGR2RGB)
             cv2.imwrite(f'./Filtered_images/image%i.png'%i,img_RGB1)
             i+=1
```

Fig 7.3: Filtering of Image

```
In [10]: vgg = VGG19(input_shape=[224, 224]+ [3], weights='imagenet', include_top=False)

In [11]: for layer in vgg.layers:
             layer.trainable = False

In [12]: x = Flatten()(vgg.output)

In [13]: prediction = Dense(5, activation='softmax')(x)

         model = Model(inputs=vgg.input, outputs=prediction)

In [14]: model.summary()
```

Fig 7.4: Building a Model

```
In [16]: from tensorflow.keras.callbacks import EarlyStopping
         early_stop=EarlyStopping(monitor='val_loss',mode='min',verbose=1,patience=5)

In [17]: history = model.fit(
           train_x,
           train_y,
           validation_data=(val_x,val_y),
           epochs=16,
           callbacks=[early_stop],
           batch_size=32,shuffle=True)
```

Fig 7.5: Training the dataset

CHAPTER 8

RESULTS AND ANALYSIS

CHAPTER 8

RESULTS AND ANALYSIS

8.1 Input images

The standard size of each image is 150 x 150 pixels. Resizing is needed because models train faster on smaller images. The raw pictures accumulated from the CT scan and in this manner, the sites are not set for direct preparation. various noises present in these pictures.

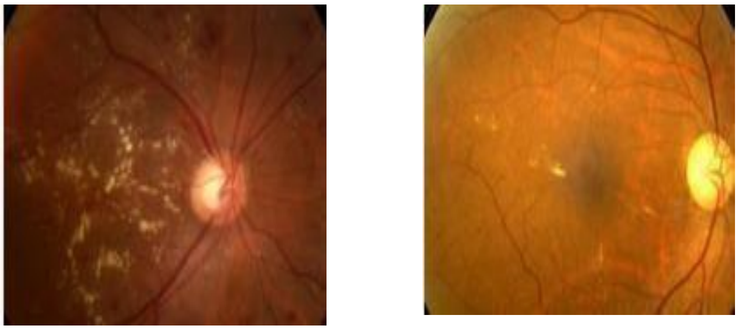

Fig 8.1: Input fundus images of the human eye

8.2 Resizing Image

The standard size of each image is 150 x 150 pixels. Resizing is needed because models train faster on smaller images. The creation and reduction of the specified image size in pixel location is a crucial task in picture maintenance strategy. Picture contribution may be broken down into two unique categories: picture up- and down-examining.

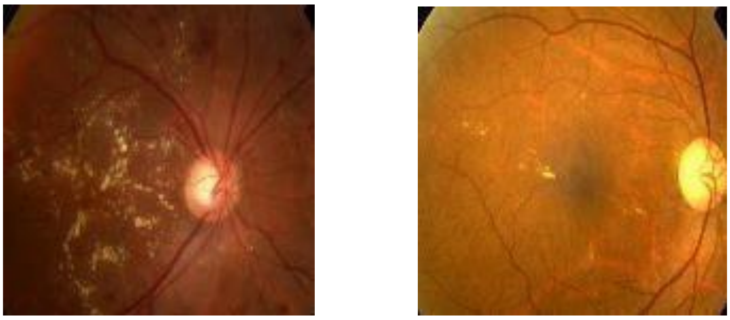

Fig 8.2: Resizing input image

8.3 Gray Scaling Image

Grayscale image which helps in simplifying algorithms as well as eliminates the complexities related to computational requirements. For noise elimination, filtering techniques like Gaussian and median channel are employed. By converting colored photographs to grayscale ones, noise may be easily removed.

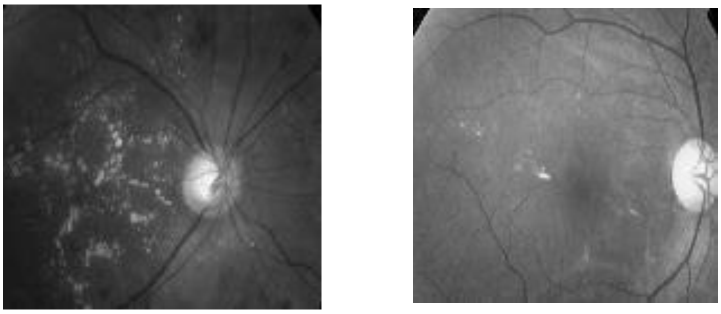

Fig 8.3: Gray scaling image

8.4 Image filteration – Gaussian Blurring

Gaussian filter Gaussian filter is a one type of noise removing technique, gaussian blur which removes the noise of image and also smoothen and sharpen the image. Below figure shows the input image and Gaussian filtered image Gaussian filtering method responds to the direction of contribution to the methodology for drawing a curve from a Gaussian capacity.

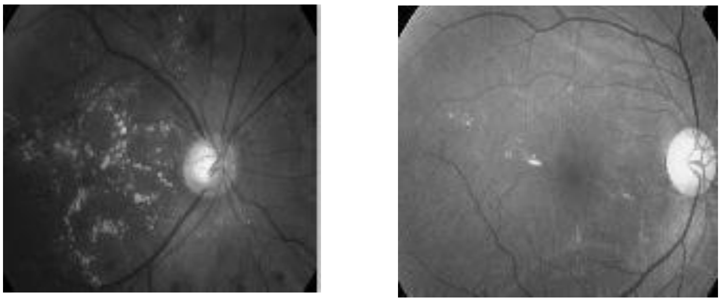

Fig 8.4: Gaussian blurring images

8.5 Contrast Enhancement - CLAHE

Using CLAHE contrast enhancement technique which adjust relative brightness and darkness of images and improves their visibility. The image is getting bright when CL is increased because input image has very low intensity and larger CL makes its histogram flatter. As the BS is bigger, the dynamic range becomes larger and the contrast of image is also increasing.

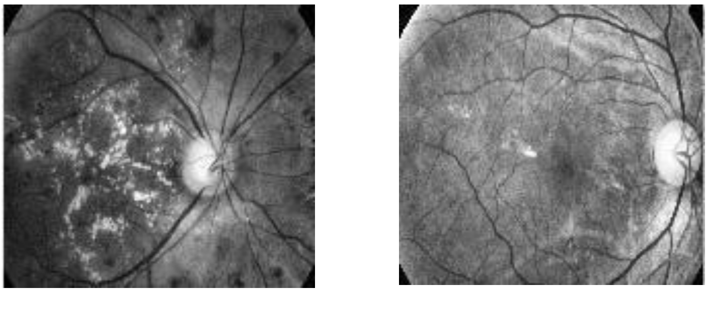

Fig 8.5: Contrast enhancement images

8.6 Resultant Analysis

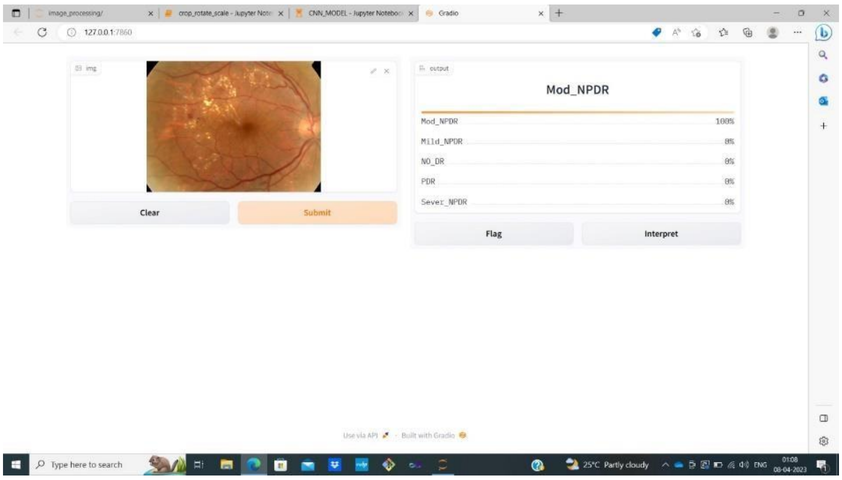

Fig 8.6 : Moderate NPDR

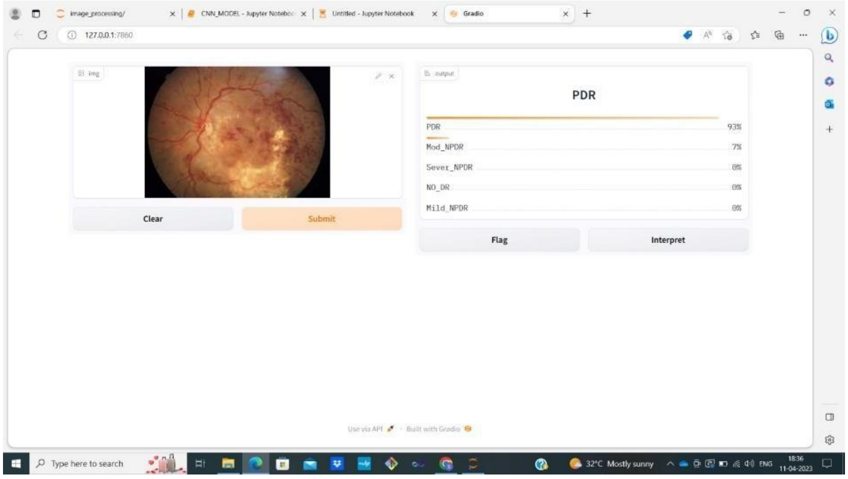

Fig 8.7 : PDR

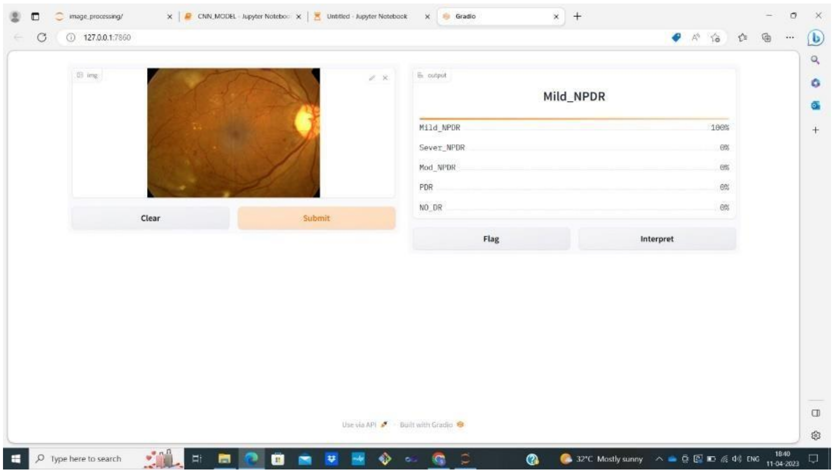

Fig 8.8 : Mild NPDR

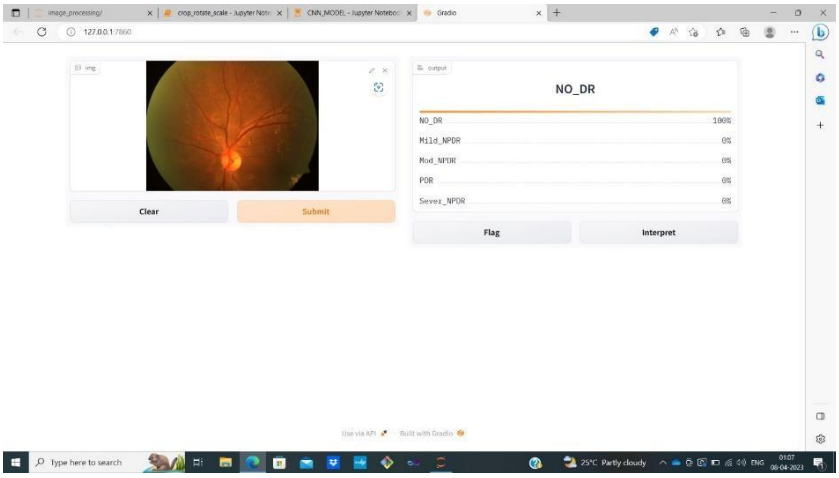

Fig 8.9 : No DR

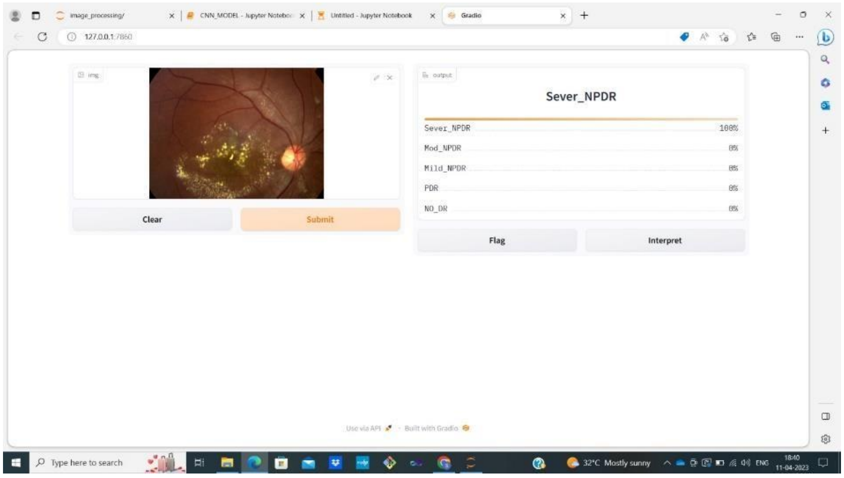

Fig 8.10: Severe NPDR

8.7 Result Graph

For an effective analysis, the proposed system is also compared with the existing works done in a similar interest research area.

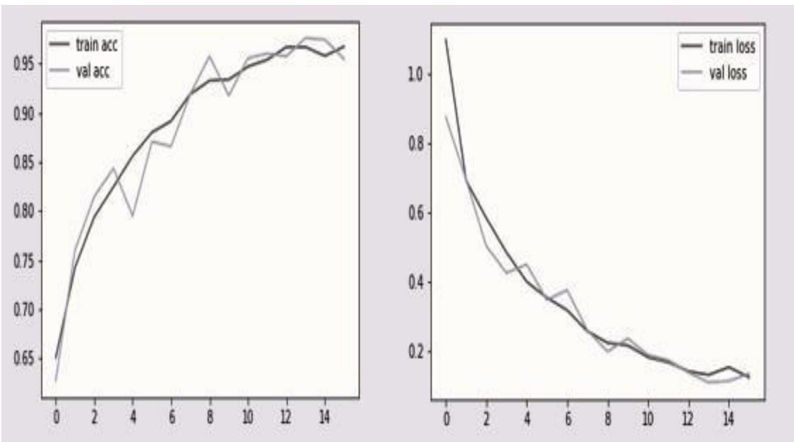

Fig 8.11: Accuracy Graph and Loss Graph

Based on the entire analysis, the proposed system significantly improves classification performance when compared to a standard CNN and that existing studies on the precision, recall, and F1 score use applied ensemble learning techniques along with traditional feature extraction methods.

	precision	recall	f1-score	support
0	0.97	1.00	0.99	149
1	0.99	0.97	0.98	576
2	1.00	0.99	1.00	398
3	0.86	0.98	0.91	81
4	0.96	0.93	0.94	108
accuracy			0.98	1312
macro avg	0.96	0.97	0.96	1312
weighted avg	0.98	0.98	0.98	1312

```
[[149   0   0   0   0]
 [  2 560   0  12   2]
 [  2   0 396   0   0]
 [  0   0   0  79   2]
 [  0   7   0   1 100]]
```

Fig 8.12: Result

Contrary to the proposed scheme, massive fundus images are not a good candidate for the application of the conventional feature extraction and image enhancement method because they are more likely to contain artefacts and have imprecise feature generalisation. This is accurate even though the ensemble learning technique has an advantage in terms of making the right choices for the classification assignment. As a result, the proposed method works better in various evaluation scenarios and offers a superior interpretation of fundus pictures.

Precision is calculated by using the below formula:

$$Precision = \frac{True\ Positive(TP)}{True\ Positive(TP) + False\ Positive(FP)}$$

In the simplest terms, precision may be described as the ratio of True Positives to All Positives. For the purposes of our issue statement, that would be the percentage of patients out of all those who have heart disease that we correctly identify as having it. It accurately foretells heart disease in patients 84% of the time, or an accuracy rate of 0.843. Additionally, precision gives us a count of the important data elements. We must take care to avoid starting treatment on a patient who does not actually have a cardiac disease but was predicted to have one by our model.

Recall is calculated by using the below formula:

$$Recall = \frac{True\ Positive(TP)}{True\ Positive(TP) + False\ Negative(FN)}$$

The recall measures the effectiveness of our algorithm's True Positive detection. How many individuals we correctly identified as having heart disease out of all those who did is revealed by the recall. Our model's return is 0.86. Recall measures how effectively our algorithm can find the relevant facts. The True Positive Rate or Sensitivity is another name for it.

False Positive Rate (FPR)

This measurement examines the ratio of true rejections to false positives. In the context of our model, where the model forecasts that the patient has a heart condition, it is a measurement of the percentage of patients among all those who did not actually have heart illness. For our data, the FPR is 0.195.

True Negative Rate (TNR)

The ratio between the actual number of negatives and the true negatives is what this term refers to. It is the proportion of instances where the model correctly predicts that the patient does not have heart disease out of all the patients who did not have heart disease. The TNR for the previously mentioned data is 0.80.

F1 Score is calculated using the below formula:

Since we want to identify as many cardiac patients as possible for our dataset, we might consider a high recall to be more important than high accuracy.

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

For some other models, like those that determine whether a bank customer is a loan defaulter or not, high accuracy is preferred because the bank would not want to lose clients who were denied a loan because the model predicted that they would default.

CONCLUSION

CONCLUSION

The proposed method involves a sequential process of image feature enhancement and representation extraction prior to classification, resulting in improved accuracy scores and reduced computational burden compared to existing classification techniques. The approach involves the use of 2D Gaussian blurring, Clahe, and image enhancement techniques for better analysis of lesions in fundus retinal images. The proposed scheme also has the potential for categorizing different DR states based on their criticality. The introduction of feature enhancement led to improved CNN classification performance and reduced computational complexity. Future work includes extending the system to analyse failed test cases due to poor visual quality of images and optimizing the CNN and feature enhancement techniques. Additionally, the proposed system can be integrated with an on-demand pre-processing Framework for better results. We can embed this technology into the hardware so it can be used by various hospitals. We can add a few pre-processing technologies to get accurate results. In this article, multi-enhancement modeling is used to offer a flexible method to work around the limitations of utilizing a single-picture enhancement approach. This work's main objective is to present a general framework that can handle different pre-processing needs for medical image analysis, enabling the user to quickly carry out appropriate image enhancement based on the evaluation of a particular medical image with different pre-processing options. Data from both qualitative and quantitative sources are used to confirm the effectiveness of each method.

REFERENCES

REFERENCES

1) Farrikh Alzami, Abdussalam, Rama Arya Megantara and Ahmad Zainul Technology of Information and Communication, 2019.
2) Dinial Utami Nurul Qomariah, Handayani Tjandrasa and Chastine Information and Communication Technology and System, 2019.
3) Deep learning for visual understanding: A review Yanming Guo , Yu Liu, Ard Oerlemans , Songyang Lao , Song Wu, Michael S. Lew.
4) P. Porwal, et al. Indian diabetic retinopathy image dataset (IDRiD): a database for diabetic retinopathy screening research.
5) G. Quellec, K. Charrière, Y. Boudi, B. Cochener, M. Lamard Deep image mining for diabetic retinopathy screening Med Image Anal, 39, 2017.
6) Novel Framework for Enhanced Learning-based Classification of Lesion in Diabetic Retinopathy Prakruthi M K Komarasamy G.
7) Diabetic retinopathy detection through deep learning techniques: A review 2020 Wejdan L. Alyoubi Wafaa M. Shalash Maysoon F. Abulkhair
8) Diabetic Retinopathy Detection by Extracting Area and Number of Microaneurysm from Colour Fundus Image 2018 Shailesh Kumar, Basant Kumar.
9) Currently, various approaches are being evolved for classifying diabetic retinopathy stages. Thiscan be seen in the review presented by Atwany et al in the year 2019.
10) Improving Dermoscopic Image Segmentation with Enhanced Convolutional-Deconvolutional Networks, 2017 by Yading Yuan and Yeh-Chi Lo.
11) The recent work of Abdelsalam and Zahran used a Support Vector Machine using a multifractal-based geometry system to diagnose and classify(2019)
12) Study towards quantification in diabetic retinopathy is carried out by Okuwobi et al., in the year 2017 where region-of-interest is used, followed by estimation of hyperreflective foci to obtain better segmentation.
13) The work implemented by Gayathri et al. has presented a unique multiclass classification with the automated binary system(2019)
14) Deep Residual Learning for Image Recognition, 2015 by Kaiming He Xiangyu Zhang Shaoqing Ren Jian Sun.
15) Diabetic Retinopathy Detection: Methods and Challenges 2019 Shivani Patil Pradnya Kulkarni.
16) R. F. Mansour, "Evolutionary Computing Enriched Computer-Aided Diagnosis System for Diabetic Retinopathy: A Survey," in IEEE Reviews in Biomedical Engineering, vol. 10, pp. 334- 349, 2017.

17) Diabetic Retinopathy Detection: Methods and Challenges 2019 Shivani Patil Pradnya Kulkarni.

18) Deep learning for visual understanding: A review 2015 Yanming Guo, Yu Liu, Ard Oerlemans Songyang Lao, Song Wu , Michael S. Lew.

19) Diabetic Retinopathy Detection by Extracting Area and Number of Microaneurysm from Colour Fundus Image 2018 Shailesh kumar,basant kumar.

20) Shi, C.; Lee, J.; Wang, G.; Dou, X.; Yuan, F.; Zee, B. Assessment of image quality on color fundus retinal images using the automatic retinal image analysis. Sci. Rep. 2022.

21) Franklin, S.W.; Rajan, S.E. An automated retinal imaging method for the early diagnosis of diabetic retinopathy. Technol. Health Care 2013, 21.

22) Li, S.; Zhao, R.; Zou, H. Artificial intelligence for diabetic retinopathy. Chin. Med. J. 2021.

23) https://www.kaggle.com/datasets/jclchan/feature-extraction

24) Pragathi, P.; Rao, A.N. An effective integrated machine learning approach for detecting diabetic retinopathy. Open Comput. Sci. 2022, 12.

25) Zhang, W.; Zhong, J.; Yang, S.; Gao, Z.; Hu, J.; Chen, Y.; Yi, Z. Automated identification and grading system of diabetic retinopathy using deep neural networks. Knowl. Based Syst. 2019.

26) Khade, S.; Ahirrao, S.; Phansalkar, S.; Kotecha, K.; Gite, S.; Thepade, S.D. Iris Liveness Detection for Biometric Authentication: A Systematic Literature Review and Future Directions. Inventions 2021.

27) Grzybowski, A.; Brona, P.; Lim, G.; Ruamviboonsuk, P.; Tan, G.S.W.; Abramoff, M.; Ting, D.S.W. Artificial intelligence for diabetic retinopathy screening: A review. Eye 2020.

28) Ribeiro, L.; Oliveira, C.M.; Neves, C.; Ramos, J.D.; Ferreira, H.; Cunha-Vaz, J. Screening for Diabetic Retinopathy in the Central Region of Portugal. Added Value of Automated 'Disease/No Disease' Grading. Ophthalmologica 2015.

29) Ipp, E.; Liljenquist, D.; Bode, B.; Shah, V.N.; Silverstein, S.; Regillo, C.D.; Lim, J.I.; Sadda, S Evaluation of an Artificial Intelligence System for Autonomous Detection of Referrable and Vision-Threatening Diabetic Retinopathy. JAMA Netw. Open 2021.

30) Lian, C.; Liang, Y.; Kang, R.; Xiang, Y. Deep Convolutional Neural Networks for Diabetic Retinopathy Classification. ACM Int. Conf. Proceeding Ser. 2018.

31) Quellec, G.; Charrière, K.; Boudi, Y.; Cochener, B.; Lamard, M. Deep image mining for diabetic retinopathy screening. Med. Image Anal. 2017.

32) Oliveira, A.; Pereira, S.; Silva, C. Retinal vessel segmentation based on Fully Convolutional Neural Networks. Expert Syst. Appl. 2018.

33) https://www.geeksforgeeks.org/vgg-16-cnn-model/
34) https://medium.com/analytics-vidhya/accuracy-precision-and-recall-in-machine-learning-classification-ae84004e86a1
35) https://www.tensorflow.org/api_docs/python/tf/keras
36) https://www.codecademy.com/article/introduction-to-numpy-and-pandas
37) https://docs.python.org/3/library/pathlib.html