

DETECTION ANALYSIS OF DIABETIC RETINOPATHY

By

DINESH GOUTHAM GANDLA : UC05573

Sai Sujith Reddy Chagam : XF15736

Akhileshwar Rao Emmanani : NI74131

Under the Supervision of

Prof. M. Ali Yousuf



UMBC

UNIVERSITY OF MARYLAND
BALTIMORE COUNTY

Abstract

Diabetes is a prevalent illness that affects the health of many people in today's society, there has been a rise in associated diseases that are affecting the community. Diabetic Retinopathy (DR) is an example of a silent disease that can occur as a result of either Type 1 or Type 2 diabetes and can lead to permanent blindness if not diagnosed promptly. However, the manual screening of DR by ophthalmologists is a time-consuming process. Therefore, this project aims to use Deep Learning (DL) to analyze various DR stages and develop a unique methodology for detecting diabetic retinopathy.

By utilizing a modified pre-trained Resnet-152 architecture and improved pre-processing techniques, our model is able to detect diabetic retinopathy with greater accuracy. The model was trained on a large dataset, consisting of 1437 high-resolution fundus images, to automatically identify the DR stage. The retinal images were acquired following a clinical procedure, utilizing the Zeiss brand's Visucam 500 camera. The dataset has been classified by expert ophthalmologists . The DR stages are classified into seven categories, ranging from 1 to 7. The input parameters for our model were patient's fundus eye images, as described in this paper. Our model performed well with an impressive accuracy of 90%.

Contents

List of Figures	v
List of Tables	1
List of Acronyms	1
List of Symbols	1
1 Introduction	1
1.1 Introduction	2
1.2 Background and Motivation	4
1.2.1 Stages of Retinopathy	4
2 A Review on Detection of Diabetic Retinopathy Methods	7
2.1 Literature Survey	8
2.2 Research Gaps	10
2.3 Objectives	11
3 Proposed Methodology	12
3.1 Proposed methodology	13
3.1.1 System Architecture	13
3.1.2 Dataset	14
3.1.3 CNN	15
3.1.4 Input layer	15
3.1.5 Convolution Layer	15
3.1.6 Pooling Layer	16
3.1.7 Fully Connected Layer	17

Contents

3.2	VGG19	18
3.3	RESENET152	20
3.4	Densnet169	20
3.5	Densnet121	21
3.5.1	DensNet Architecture	22
3.5.1.1	DenseNet Structure	22
3.5.1.2	DenseBlocks and Layers	23
3.5.2	Date Set Representation	25
3.5.3	Morphological Operations	26
3.5.3.1	Thresholding	26
3.5.3.2	Gaussian blur	26
3.5.4	CLAHE	28
3.5.4.1	Median Blur and Masking	30
4	Experiments and Results	32
4.1	Evaluation Metrics	33
4.2	Experiments	34
4.2.0.1	Experiment-1	34
4.2.0.2	Experiment-2	35
4.2.0.3	Experiment-3	37
4.2.0.4	Experiment-4	37
4.2.0.5	Experiment-5	38
4.3	Results	40
4.3.1	Discussion	41
5	Conclusions and Future Scope	42
5.1	Conclusions	43
5.2	Future Scope	43
	Bibliography	44

List of Figures

1.1	Visions of both the normal eye and the affected eye [1]	3
1.2	Diabetic Retinopathy [1]	5
1.3	Stages of Diabetic Retinopathy [2]	6
3.1	System Architecture	13
3.2	Sample fundus images of the five stages of DR [3]	14
3.3	Image of size 4X4 [3]	16
3.4	CNN Architecture [4]	17
3.5	VGG16 Architecture [5]	19
3.6	VGG19 Architecture [5]	19
3.7	DensNet Structure [6]	22
3.8	Modified DensNet-121 Architecture	23
3.9	DensNet Architecture [6]	24
3.10	Full Architecture in abstract form [6]	24
3.11	Graphical representation of images in each class	25
3.12	Before and after applying gaussian blur	27
3.13	Before and after cropping the image	27
3.14	RGB to LAB	29
3.15	Contrast-Limited Adaptive Histogram Equalization	29
3.16	Before and after applying CLAHE	29
3.17	Before and after applying median blur	30
3.18	Before and after masking	31

List of Figures

4.1	Confusion Matrix	33
4.2	Loss and Accuracy graphs of CNN MODEL	35
4.3	Loss and Accuracy graphs of VGG-19	36
4.4	Loss and Accuracy graphs of DensNet-169	37
4.5	Loss and Accuracy graphs of ResNet-152	38
4.6	Loss and Accuracy graphs of DensNet-121	39
4.7	Training and Testing Confusion Matrix of DensNet-121 Model	40

1

Introduction

1. Introduction

1.1 Introduction

Diabetic retinopathy is a condition that arises as a complication of diabetes and impacts the retina, the part of the eye responsible for vision. It leads to damage in the blood vessels within the retina, causing blockages or leakage. Additionally, it can result in the growth of abnormal blood vessels, leading to bleeding, scarring, and permanent vision loss. The most common cause of visual impairment associated with this condition is the thickening of the central part of the retina, known as diabetic macular edema, which can cause irreversible vision damage. Among adults, Diabetic Retinopathy is the primary cause of new cases of blindness and the leading cause of vision loss for individuals with diabetes. Early symptoms may not be easily noticeable, but undergoing a comprehensive dilated eye exam at least once a year can assist in the early detection and prevention of complications [7]. Managing and controlling diabetes effectively and promptly seeking medical attention for any initial symptoms are crucial measures for preventing diabetic retinopathy .

Diabetic Retinopathy is becoming an increasingly significant factor behind blindness among individuals aged 20-60 worldwide. This condition results in a decline in productivity and quality of life for affected individuals and imposes socio-economic burdens on communities. It affects approximately 2.5 million individuals out of the global population of 50 million blind individuals. A recent investigation conducted in Pakistan revealed that cataracts and uncorrected refractive error were more common causes of visual impairment compared to Retinopathy. In order to prevent blindness, the World Health Organization (WHO) recommends regular screening for individuals with diabetes. Detecting diabetic retinopathy promptly and accurately is crucial for cost-effective treatment. Since this condition does not exhibit early symptoms, early detection may pose a challenge. Nevertheless, timely intervention can prevent blindness [8].

According to recommendations from the World Health Organization (WHO), individuals who have diabetes should receive regular screenings to detect diabetic retinopathy

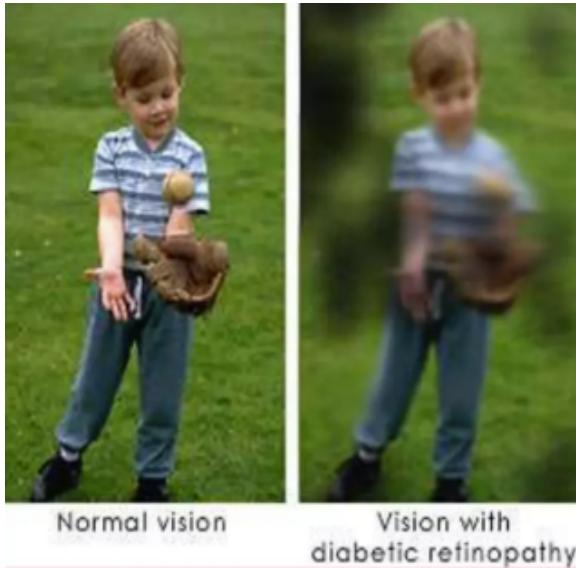


Figure 1.1: Visions of both the normal eye and the affected eye [1]

(DR) early on and prevent blindness [9]. The timing and accuracy of DR detection are crucial for effective treatment and cost-effectiveness. Since DR is asymptomatic, early detection can be difficult. However, early detection can lead to effective treatment and prevention of blindness, making screening an important tool for preventing blindness in those with diabetes.

Currently, the process of identifying DR is both laborious and manual. It relies on the expertise of a trained medical professional to carefully examine and evaluate digital color fundus photographs of the retina. Regrettably, the delayed results from human assessment, which can take up to a day or two to complete, can lead to missed follow-ups, miscommunication, and delayed treatment. Typically, clinicians diagnose DR by identifying specific abnormalities in the retinal blood vessels associated with the disease. Although this approach is effective, it demands substantial resources. Unfortunately, the necessary expertise and equipment are often unavailable in regions with high diabetes rates, where there is a pressing need for efficient DR detection. With the increasing number of individuals affected by diabetes, the existing infrastructure for preventing DR-related blindness will become even more insufficient [3].

1. Introduction

1.2 Background and Motivation

Diabetic Retinopathy (DR) is a prominent factor behind the deterioration of vision, which occurs due to harm caused to the blood vessels in the retina of people diagnosed with diabetes. It encompasses two main types: Non-Proliferative Diabetic Retinopathy (NPDR) and Proliferative Diabetic Retinopathy (PDR). NPDR comprises Mild, Moderate, and Severe stages. Mild NPDR involves a single micro-aneurysm (MA), while Moderate NPDR exhibits ruptured MAs in deeper layers and flame-shaped hemorrhages. Severe non-proliferative diabetic retinopathy (NPDR) is identified by the presence of more than 20 intraretinal hemorrhages in each quadrant of the retina. Additionally, it includes venous bleeding and notable abnormalities in the retinal microvasculature [10]. PDR represents an advanced stage where neovascularization leads to the formation of new blood vessels on the retina's inner surface. Figure 1.2 illustrates the different DR stages, with Normal and Mild stages posing challenges due to visual similarities. In India alone, over 62 million people have diabetes, and those with a diabetes duration exceeding 20 years face an 80% risk of developing DR. Elevated blood sugar damages the delicate retinal blood vessels, causing DR and potential blindness. Medication and regular eye monitoring can prevent 90% of new cases. Untreated DR affects both retinas, causing vision loss. Risk factors include uncontrolled blood sugar, high blood pressure, and elevated cholesterol levels. [11]

In the preceding section, it was elucidated that color fundus images play a crucial role in diagnosing DR. However, manual analysis of these images necessitates specialized domain expertise, making it a time-consuming and costly process. Thus, leveraging computer vision techniques to analyze the fundus images and aid physicians and radiologists automatically becomes imperative.

1.2.1 Stages of Retinopathy

Unmanaged diabetes is associated with raised levels of blood sugar, blood pressure, and cholesterol, along with increased body weight. These factors can cause harm to the delicate blood vessels in the retina, resulting in a condition known as diabetic retinopathy.

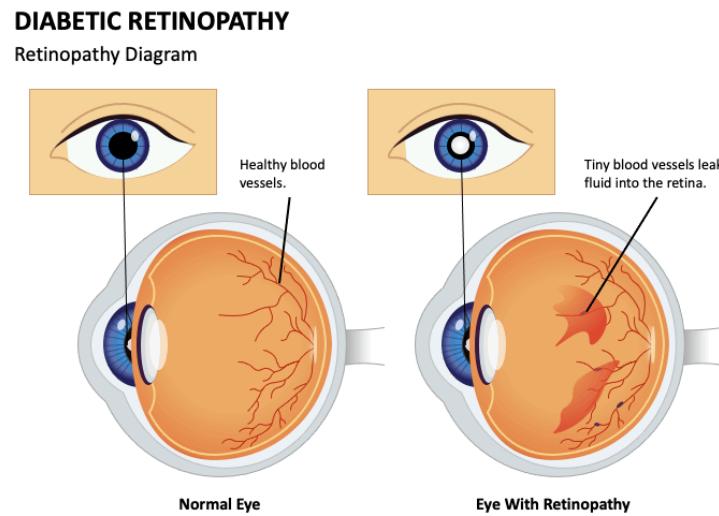


Figure 1.2: Diabetic Retinopathy [1]

Detecting diabetic retinopathy at an early stage can help prevent or minimize the loss of vision., but as the disease progresses, preventing further vision loss becomes increasingly challenging.

- **Non-Proliferative Retinopathy:** Non-Proliferative Retinopathy (NPDR) is a common stage of diabetic retinopathy, a condition affecting the retina due to uncontrolled diabetes. NPDR is characterized by the presence of microaneurysms, hemorrhages, and swelling of the retinal blood vessels. These abnormalities can lead to reduced blood flow and oxygen supply to the retina, causing vision problems. NPDR is usually asymptomatic in its early stages, highlighting the importance of regular eye exams for individuals with diabetes. Timely detection and management of NPDR are crucial to prevent its progression to more severe stages and to mitigate the risk of vision loss associated with diabetic retinopathy. [9].
- **Proliferative Diabetic Retinopathy:** Proliferative retinopathy is a serious complication that can occur in individuals with diabetes, particularly those with uncontrolled blood sugar levels. It is characterized by the abnormal growth of blood vessels in the retina, the light-sensitive tissue located at the back of the eye. These

1. Introduction

new blood vessels are fragile and prone to leaking, causing hemorrhages and leading to further complications. Proliferative retinopathy often develops in individuals who have had diabetes for many years. It is crucial to manage diabetes effectively to minimize the risk of this condition. Regular eye examinations are essential for early detection and prompt intervention. Treatment options for proliferative retinopathy include laser therapy, which helps to seal the leaking blood vessels and prevent further growth, and surgical procedures to restore retinal detachment. Timely intervention can significantly reduce the risk of vision loss and improve the overall prognosis for individuals affected by this condition [9].

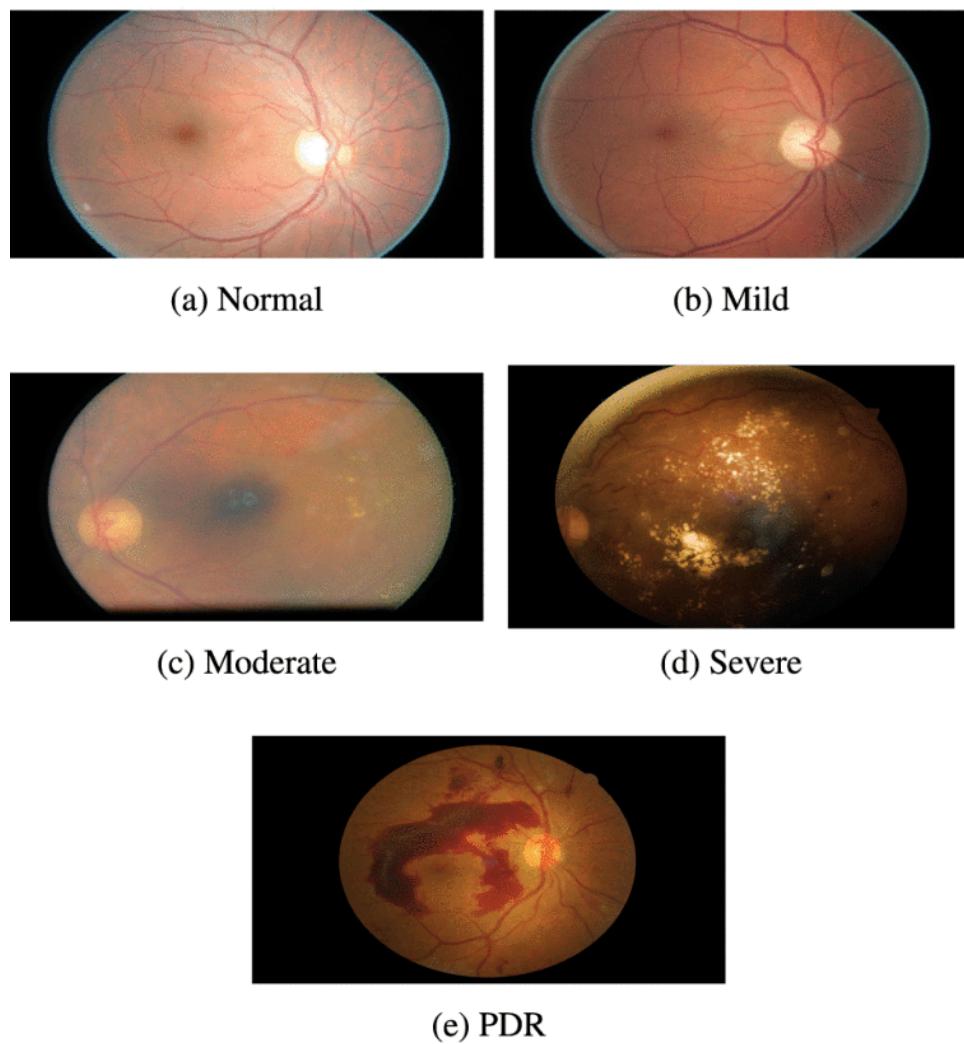


Figure 1.3: Stages of Diabetic Retinopathy [2]

2

A Review on Detection of Diabetic Retinopathy Methods

2. A Review on Detection of Diabetic Retinopathy Methods

2.1 Literature Survey

The field of computer vision has witnessed tremendous progress in recent years, with deep learning-based methods emerging as successful techniques. Convolutional neural networks (CNNs) have been suggested to address various tasks due to their ability to perform efficient feature extraction, leading to highly representative features [12].

The study conducted by [9] focused on the binary classification of diabetic retinopathy (DR) and utilized five different transfer learning models: Xception, InceptionResNetV2, MobileNetV2, DenseNet121, and NASNetMobile. The models achieved validation accuracies ranging from 80% to 96.25%, with InceptionResNetV2 yielding the highest accuracy at 96.25%. This indicates that transfer learning models can be a useful tool in the classification of DR. In this paper, the model is trained on only 1115 retinal fundus images. In this paper, the model fails to achieve good accuracy when the data set is large.

The authors of [13] conducted a study on automated diabetic retinopathy (DR) detection from color fundus images without the need for preprocessing or feature extraction. They proposed the use of a pre-trained convolutional neural network (CNN) model, DenseNet-169, and employed hyperparameter tuning and data augmentation techniques to achieve optimal classification performance. In order to evaluate the efficacy of their method (referenced as [14]), the researchers employed the openly accessible Kaggle AP-TOS 2019 Blindness Detection dataset. They conducted a comparative analysis between the DenseNet-169 model and two other pre-trained models, namely DenseNet-121 and ResNet-50. However, it is worth mentioning that the accuracy of this model is limited, primarily due to the significance of preprocessing for attaining improved outcomes.

The methodology outlined in [15] proposes a technique for classifying the severity of diabetic retinopathy (DR) by employing deep layer aggregation. This involves combining features from multiple convolutional layers of the Xception architecture, which are then input into a multi-layer perceptron (MLP) for classification. The authors evaluated their approach using four deep feature extractors: InceptionV3, MobileNet, ResNet50, and the

original Xception architecture. The deep layer aggregation effectively merged the deep features, leading to improved learning compared to the conventional Xception architecture. Additionally, researchers in [16] incorporated transfer learning and hyper-parameter tuning to further enhance classification performance. The proposed model in [15] was tested using the Kaggle APTOS 2019 contest dataset. The results demonstrated that the modified Xception deep feature extractor improved DR classification, achieving an accuracy of 83.09%, compared to 79.5% for the original Xception architecture. Furthermore, the modified model exhibited higher sensitivity (88.24% versus 82.35%) and slightly higher specificity (87.00% versus 86.32%). However, it's important to note that even with these modifications, the model's accuracy did not reach a satisfactory level.

The study by [17] utilized Deep Learning (DL) to analyze different stages of Diabetic Retinopathy (DR). They employed a dataset of around 3662 training images and trained a model called DenseNet to automatically detect DR stages and classify them into high-resolution fundus images. The dataset, available on Kaggle (APTOS), includes five DR stages (0, 1, 2, 3, and 4). The model [18] took patient fundus eye images as input and used the DenseNet Architecture to extract fundus image features. The architecture achieved an accuracy of 0.9611 and a quadratic weighted kappa score of 0.8981 in detecting DR. The study also compared VGG16 and DenseNet121 CNN architectures but did not evaluate them against other pre-trained models.

The research conducted in [19] employed pre-trained weights derived from ImageNet to initialize MobileNetV2. The training process incorporated data augmentation and resampling techniques. The combination of MobileNetV2 with an SVM classifier resulted in a highly efficient deep-learning model named MobileNetV2-SVM. The model demonstrated impressive performance, achieving a quadratic weighted kappa score of 0.925, 85% accuracy, and AUROC values of 1.00, 0.82, 0.94, 0.94, and 0.93 for the normal, mild, moderate, severe, and proliferative DR classes, respectively. However, the model did not exceed an accuracy of 85%.”

2. A Review on Detection of Diabetic Retinopathy Methods

In a study cited [8], researchers employed 26 advanced deep learning networks to develop a comprehensive model for assessing deep feature extraction and image classification of fundus images related to diabetic retinopathy (DR). They utilized the EyePACS fundus image dataset from Kaggle and concluded that ResNet50 exhibited the highest degree of overfitting, while Inception V3 demonstrated the lowest. Conversely, another study referenced as [20] identified EfficientNetB4 as the most optimal, efficient, and reliable deep learning algorithm for DR detection. InceptionResNetV2, NasNetLarge, and DenseNet169 also exhibited good performance. EfficientNetB4 achieved a training accuracy of 99.37% and the highest validation accuracy of 79.11%, while DenseNet201 achieved the highest training accuracy of 99.58% but a lower validation accuracy of 76.80% compared to the top-performing models.

2.2 Research Gaps

- There aren't many research papers that explain or mention Precision-Recall, and even fewer that mention Precision-Recall of each class.
- The pre-processing techniques that are being used in the research papers can be improved which helps the model to achieve better accuracy.
- The deep neural network faces a significant challenge in fundus image classification due to high variability, particularly in cases of retinal proliferation and retinal detachment of new blood vessels, resulting in reduced network accuracy.
- Most of the current literature focuses on the separate extraction of blood vessels or detection of lesions using different tools and techniques, which adds complexity to the system design.
- Several current approaches detect specific lesion categories that lead to diabetic retinopathy, but they cannot be directly linked to the prediction outcome of a deep learning algorithm.

2.3 Objectives

- (i) To Propose a novel architecture that is less complex and more efficient.
- (ii) To focus mainly on the pre-processing techniques to make detecting DR easy for the model.
- (iii) To train our model that can achieve better accuracy with the existing models.

3

Proposed Methodology

3.1 Proposed methodology

Input pictures for a research project in this field must first go through a preprocessing stage. A variety of image preprocessing procedures, such as morphological operations like gaussian blur, cropping, resizing, etc are used. We apply CLAHE operations as part of preprocessing. Furthermore, we apply median blur for the images to avoid noise and then apply masking to the images. Extraction of features and further classification of images can be obtained from Resnet152.

3.1.1 System Architecture

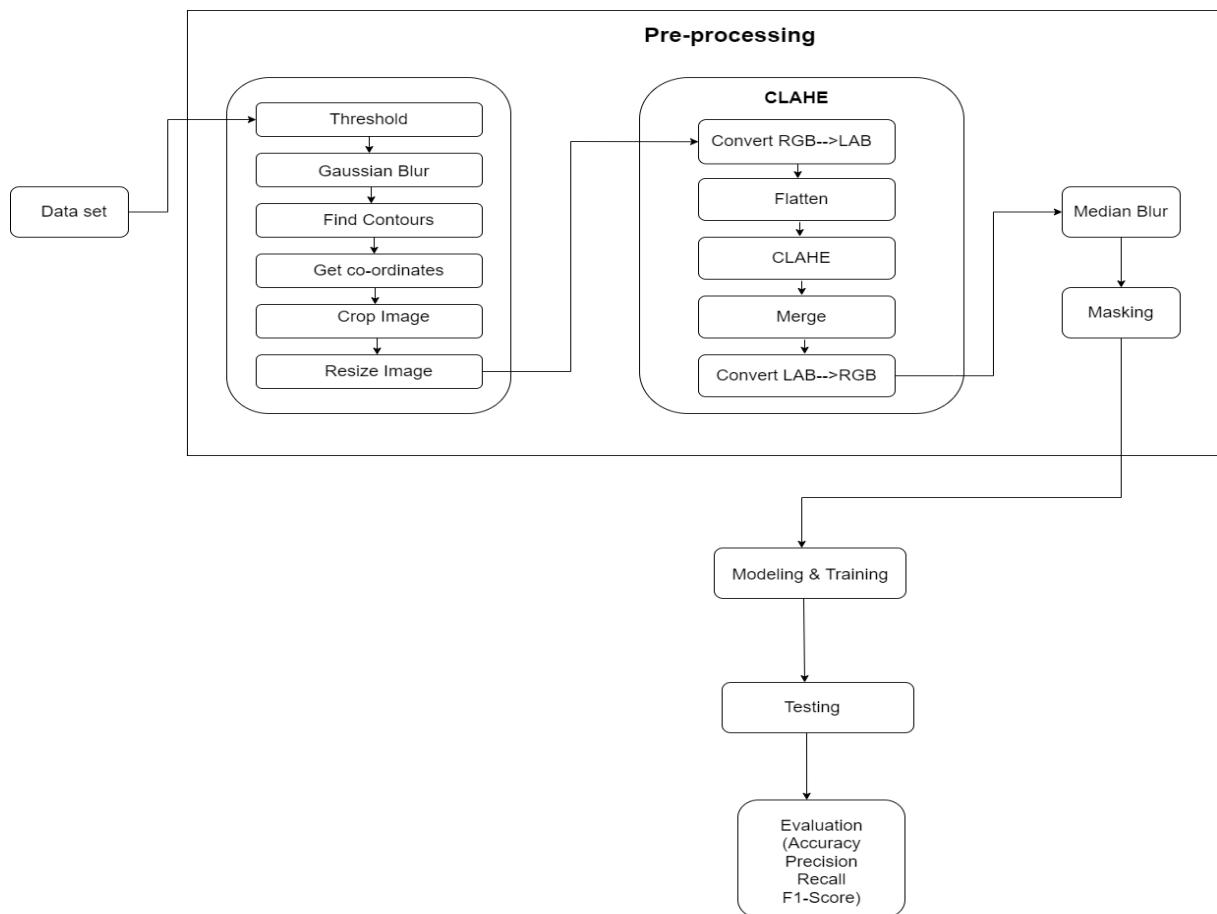


Figure 3.1: System Architecture

3. Proposed Methodology

3.1.2 Dataset

The collection includes retina scan images that have undergone Gaussian filtering to identify the presence of diabetic retinopathy. The initial dataset can be accessed from Zendos fundus images dataset. To ensure compatibility with various pre-trained deep learning models, these images have been resized to dimensions of 224x224 pixels. They are organized into separate folders based on the severity or stage of diabetic retinopathy.

(i) No-DR

(ii) Mild

(iii) Moderate

(iv) Severe

(v) Proliferate-DR

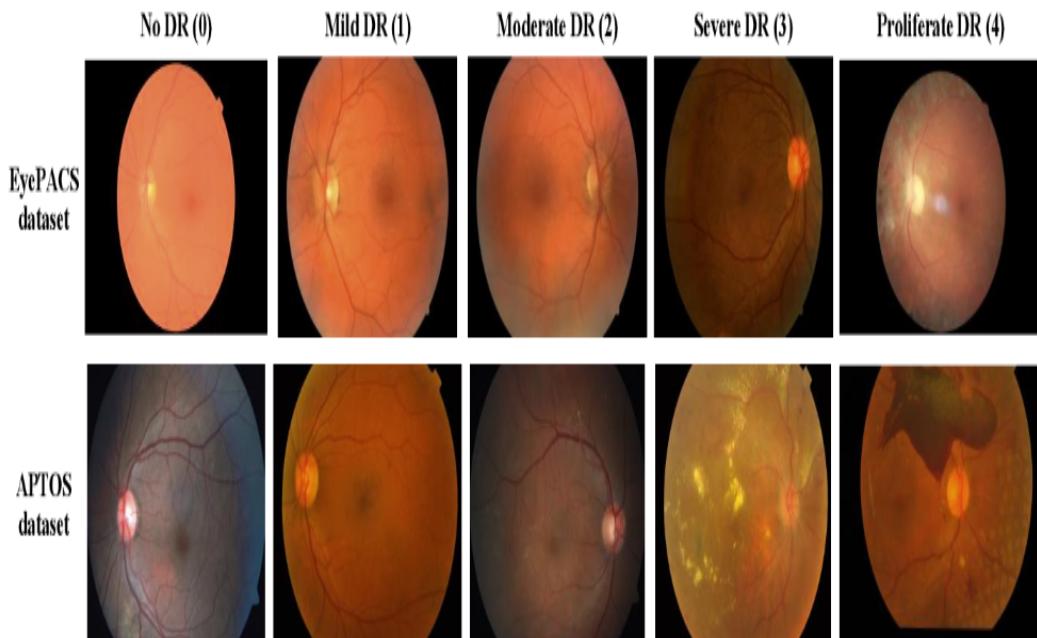


Figure 3.2: Sample fundus images of the five stages of DR [3]

3.1.3 CNN

The CNN model functions in a two-step process consisting of feature extraction and classification. In the first step, relevant information and features are extracted from the images using a combination of filters and layers. After completing this step, the data is forwarded to the next stage, known as classification. During classification, the data is sorted into different categories based on the specific target variable associated with the problem. One would generally expect a CNN model to have the following structure:

- Input layer
- Convolution layer + Activation function
- Pooling layer
- Fully Connected Layer

3.1.4 Input layer

The input layer of a CNN model takes in the image data, which can be in grayscale or RGB format. The image comprises pixels with values ranging from 0 to 255, and it is necessary to normalize these values to a range between 0 and 1 before feeding them into the model.

3.1.5 Convolution Layer

The convolution layer utilizes a filter to extract features from the input image, which justifies its name. By repeatedly applying the filter to the image, a feature map is generated, aiding in the classification of the input image. For instance, let's consider a 2D input image with normalized pixel values. Applying the filter to the image produces a 4x4 Feature Map that contains valuable information about the input image. In practical applications, multiple feature maps are typically generated.

The procedure entails applying the filter to the highlighted green section of the image depicted in the above figure. The values of the filter (represented by lines in the figure)

3. Proposed Methodology

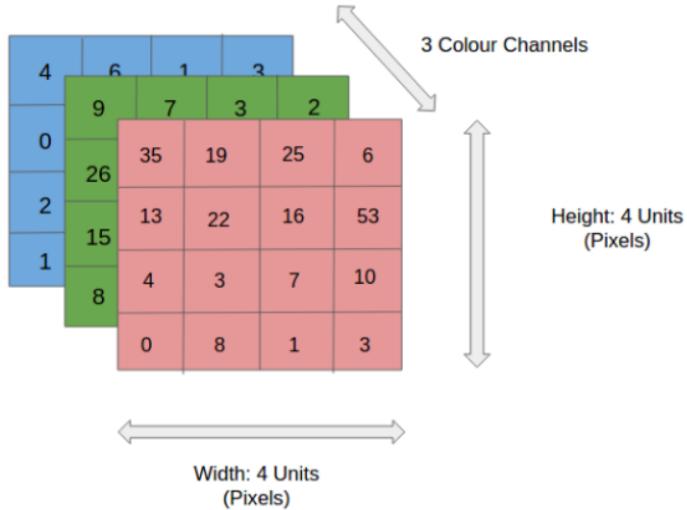


Figure 3.3: Image of size 4X4 [3]

are multiplied by the pixel values of the image pixel values and then summed to yield the ultimate value.

In the subsequent stage, the filter shifts to the neighboring column, as demonstrated in the below figure. This transition from one column or row to another is referred to as a stride. In this instance, a stride of 1 is employed, indicating that the filter moves by one column. The procedure is iterated until the complete image is covered, producing the ultimate Feature Map. Following the acquisition of the feature map, a nonlinearity is introduced by applying an activation function. It is worth mentioning that the feature map is smaller in size compared to the original image because the stride value determines its reduction. This signifies the filter's movement across the entire image with a stride of 1.

3.1.6 Pooling Layer

Following the Convolutional layer, the pooling layer is employed to decrease the dimensions of the feature map, aiding in the retention of important information or features from the input image while reducing computational requirements. Pooling generates a lower-resolution version of the input image that still maintains the significant components.

Among the most widely used types of pooling are Max Pooling and Average Pooling. The accompanying illustration demonstrates the functionality of Max Pooling, utilizing the feature map obtained from the previous example.

In this particular situation, a Pooling layer is utilized with dimensions of 2X2 and a stride of 2. It chooses the maximum value from each highlighted area, generating a fresh 2X2 representation of the original input image. This implementation of pooling successfully decreases the size of the feature map.

3.1.7 Fully Connected Layer

Up to this point, we have discussed the procedures involved in Feature Extraction. Now, we will proceed to the Classification phase. We employ a Fully Connected Layer to categorize the input image into a particular label, which resembles the one employed in Artificial Neural Networks (ANN). This layer establishes a connection between the information acquired from previous stages, namely the Convolution and Pooling layers and the output layer. As a result, the input is appropriately classified into the desired label. The accompanying image illustrates the complete process of a Convolutional Neural Network (CNN) model.

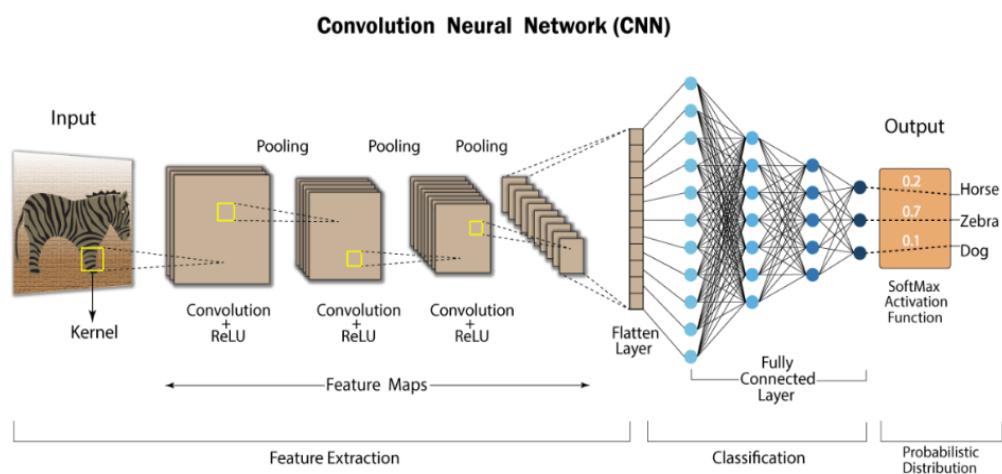


Figure 3.4: CNN Architecture [4]

3. Proposed Methodology

3.2 VGG19

Deep learning architectures have been quite effective in solving classification issues in recent years. One of them is the VGGNet architecture, which is available in two versions: 16 and 19. When looking at the general characteristics of architecture, RGB pictures of 224 times 224 pixels have been used at the entrance. The filters are 3×3 in size. There are blocks in the design that terminate with a Max Pooling layer and have batches of 2, 3, and 4 convolutional layers. VGG19 architecture features 16 convolutional layers, compared to the 13 of the VGG16 design. Each convolutional layer uses the ReLU activation function to operate. The number of filters steadily rises from 64 in the first block to 512 in the last block . Over 2.5 million celebrity photos gathered from the internet are used to train the VGGNet from scratch. With the ability to recognize objects that correspond to 1000 distinct classes as a consequence of its training, it has achieved notable success. One of the key issues here is how expensive it is in terms of time and hardware to train a model from scratch in order to get a CNN model that can categorize items that don't belong to these 1000 things. At this phase, an issue may be solved using the transfer learning approach. The technique of transferring information from a previously trained model to a newly constructed model is known as the transfer learning method [5]. Instead of completely retraining the model, transfer learning allows the features of an existing, successful model to be applied to a brand-new model.

By focusing exclusively on the final output layers of the new model in a transfer learning approach, model development becomes simpler [5]. A pre-trained CNN architecture's taught parameters of layers are mostly frozen in transfer learning. In other words, the qualities that the algorithm learns are unaltered. The layers of the network will then be fine-tuned [5]. The architecture of the VGG16 and VGG19 models are shown in Figures 3.5 and 3.6, respectively.

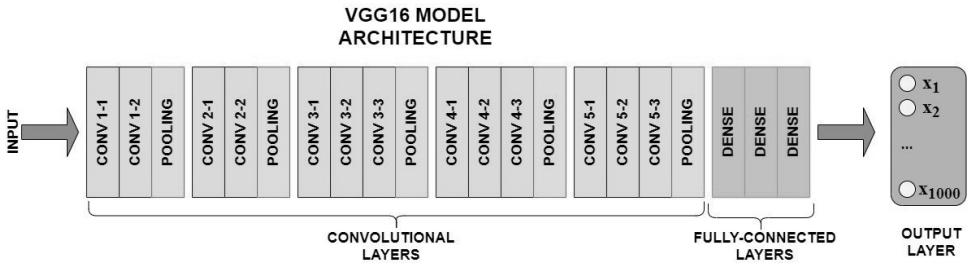


Figure 3.5: VGG16 Architecture [5]

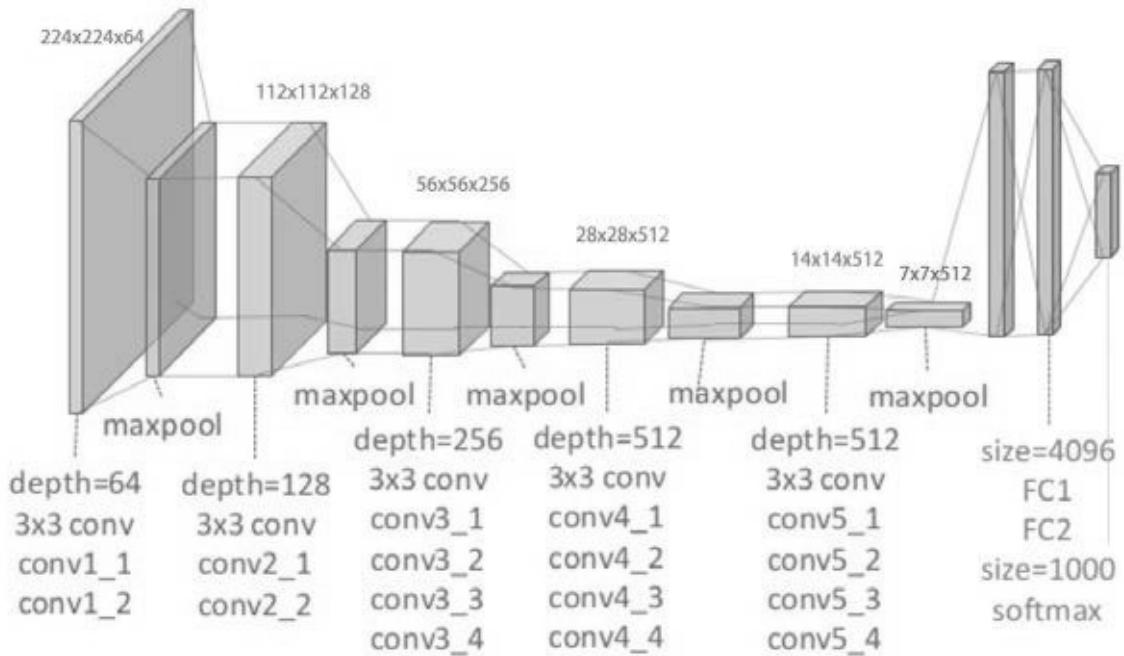


Figure 3.6: VGG19 Architecture [5]

The transfer learning approach was used in this work to freeze the convolution layers present in the feature learning sections of the VGG19 architecture. The properties that the VGGNet architecture discovered while learning to recognize objects from 1,000 distinct classes have been maintained as a result of this method. The method of extracting feature maps related to illnesses has made use of these characteristics. After the feature learning step in the VGG19 architecture, the picture has been flattened into a single dimension. 1000 artificial neurons make up a dense layer that has been defined at a later time. The last stage then employed the dropout mechanism to build a final dense layer made up of three artificial neurons.

3. Proposed Methodology

3.3 RESENET152

ResNet-152, short for Residual Network with 152 layers, is a deep convolutional neural network (CNN) architecture that was introduced by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun in their paper titled "Deep Residual Learning for Image Recognition." [21] This paper was presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

The key innovation of ResNet-152 lies in the use of residual learning blocks, which address the vanishing gradient problem that often occurs in very deep neural networks. In traditional deep networks, as the number of layers increases, training can become difficult due to the vanishing gradient problem, where the gradient of the loss function with respect to the parameters diminishes, making it challenging for the model to learn.

ResNet-152 introduces the concept of residual learning by using shortcut connections or skip connections that allow the network to learn residual functions. These shortcut connections skip one or more layers, enabling the gradient to flow more easily through the network. This architecture facilitates the training of very deep networks and allows the model to learn more complex representations.

ResNet-152 has proven to be highly effective in image recognition tasks and has been used as a pre-trained model in transfer learning scenarios. It has been employed in various competitions and benchmarks, demonstrating state-of-the-art performance in image classification tasks

3.4 Densnet169

DenseNet169 is a convolutional neural network architecture developed by the computer vision research team at Facebook AI Research (FAIR) in 2016. It is a variant of the DenseNet architecture, which is characterized by densely connected layers that pass information between all preceding layers. This approach allows for efficient feature reuse and leads to reduced computational requirements compared to traditional neural network

architectures. DenseNet169 consists of 169 layers and exhibits remarkable capabilities in diverse computer vision tasks, including image classification, object detection, and segmentation. To downsample the feature maps and enhance channel capacity, it employs a 3×3 convolutional kernel with a stride of 2, effectively reducing spatial resolution.

One of the key features of DenseNet169 is its use of dense blocks, which group multiple layers together and concatenate their feature maps before passing them on to the next block. This approach allows for more efficient use of the network's parameters and leads to improved accuracy on challenging datasets. Overall, DenseNet169 is a powerful neural network architecture that has proven to be highly effective for a wide range of computer vision tasks. Its efficient use of parameters and feature reuse make it a popular choice for researchers and practitioners alike.

3.5 Densnet121

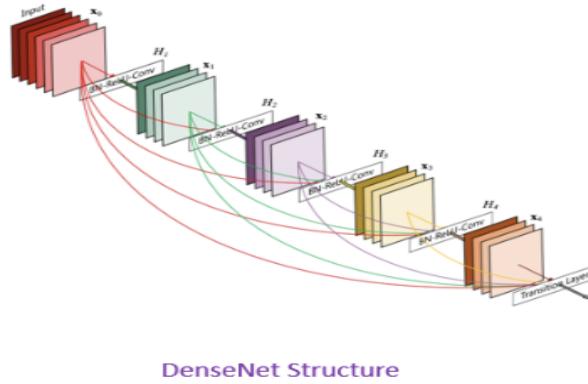
Introduced in 2017 by Huang et al. [22], DenseNet-121 is a widely used deep-learning model for image classification. It is based on a series of dense blocks, each of which is made up of several convolutional layers with shortcut connections. The model architecture has gained popularity due to its high performance in various computer vision tasks. The input to the DenseNet-121 model is a 224×224 RGB image, and the output is a probability distribution over 1000 different classes, which include various objects, animals, and scenes. One of the key features of DenseNet-121 is its ability to use feature reuse and reduce the number of parameters in the model. This is accomplished by concatenating the output feature maps of each layer in a dense block and passing them as input to the next layer. This allows for efficient use of the limited memory available in GPUs and can help to reduce overfitting.

3. Proposed Methodology

3.5.1 DensNet Architecture

3.5.1.1 DenseNet Structure

DenseNet can be categorized as a conventional network. The illustration exhibits a dense block comprising five layers, where the growth rate is $k = 4$.



$$a^{[l]} = g([a^{[0]}, a^{[1]}, a^{[2]}, \dots, a^{[l-1]}])$$

Figure 3.7: DensNet Structure [6]

In the architecture of a neural network, the output of one layer is employed as input for the subsequent layer using a composite function operation. This operation comprises several layers: convolution, pooling, batch normalization, and non-linear activation. The interconnections among these layers result in a total of $L(L+1)/2$ direct connections within the network, where L denotes the overall number of layers in the architecture.

Different versions of DenseNet, including DenseNet-121, DenseNet-160, DenseNet-201, and more, are accessible. Each version is identified by a number that represents the total number of layers in the neural network. For instance, in DenseNet-121, the number 121 is obtained through the following calculation:

$$\text{DensNet-121: } 5 + (6+12+24+16)*2 = 121$$

5- Convolution and Pooling Layer

3- Transition Layers(6,12,24)

1- Classification Layer(16)

2- DenseBlock(1X1 and 3X3 conv)

The layers in our proposed and Modified DensNet-121 are shown in Figure 3.8.

Model: "sequential"		
Layer (type)	Output Shape	Param #
densenet121 (Functional)	(None, 7, 7, 1024)	7037504
flatten (Flatten)	(None, 50176)	0
dense (Dense)	(None, 128)	6422656
dense_1 (Dense)	(None, 64)	8256
dense_2 (Dense)	(None, 32)	2080
dense_3 (Dense)	(None, 5)	165

Total params:	13,470,661
Trainable params:	6,433,157
Non-trainable params:	7,037,504

Figure 3.8: Modified DensNet-121 Architecture

3.5.1.2 DenseBlocks and Layers

Dense layers, also known as fully connected layers, are an integral component of neural networks. They connect every neuron from the previous layer to every neuron in the subsequent layer. Dense layers contribute to the model's capacity to learn complex patterns and relationships, making them a fundamental building block in deep learning. If the feature map dimensions are not the same, it is only possible to group layers using the equation mentioned above by either adding or concatenating them. However, when the dimensions are different, DenseNet solves this issue by dividing the network into DenseBlocks, where the number of filters can vary. Still, the dimensions within each block remain the same. Transition Layers are introduced to ensure the dimensions are consistent between blocks, which use downsampling and batch normalization. This step is crucial in the convolutional neural network architecture.

3. Proposed Methodology

Now, let's take a closer look at what the DenseBlock and Transition Layer contain:

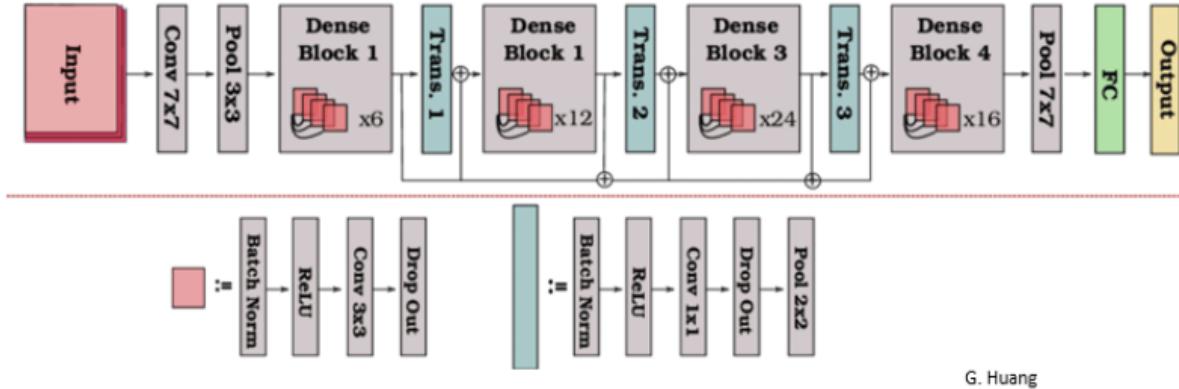


Figure 3.9: DensNet Architecture [6]

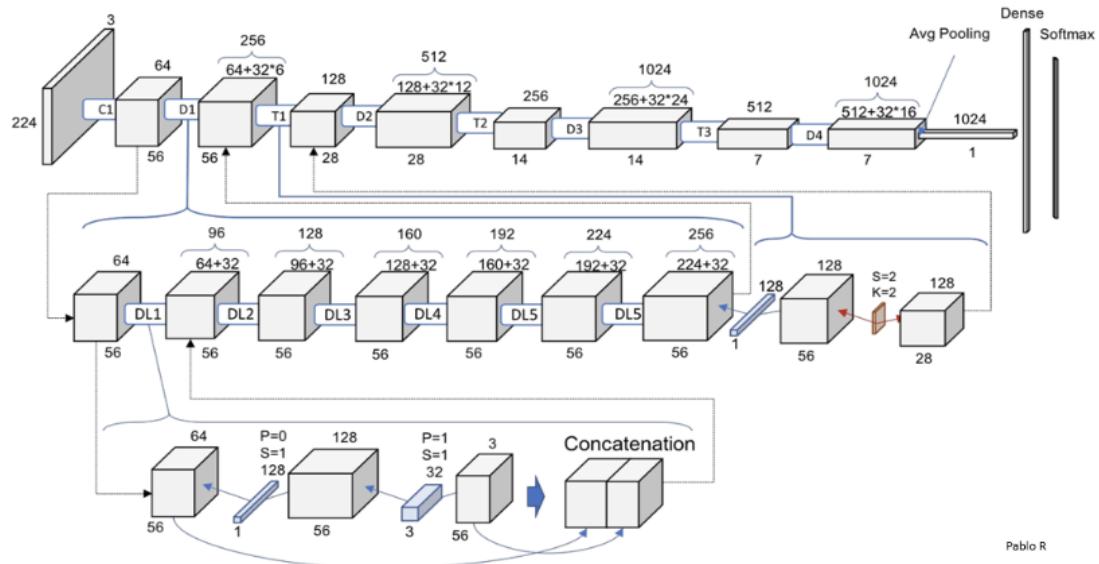


Figure 3.10: Full Architecture in abstract form [6]

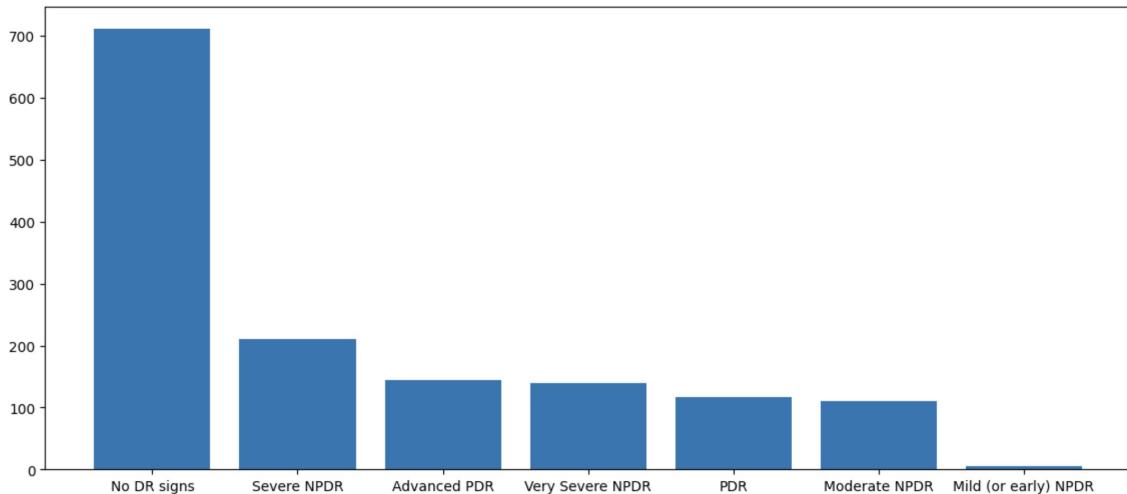


Figure 3.11: Graphical representation of images in each class

3.5.2 Date Set Representation

Data set representation plays a crucial role in machine learning and deep learning models. In this context, fundus images dataset [23], which comprises 1437 images of retinas with varying degrees of diabetic retinopathy, is widely used. When working with this dataset, it is important to ensure that the images are represented appropriately to obtain accurate results. One important step in this process is resizing the images to a uniform size to ensure consistency in the input data for the model. In our dataset, the images are resized to 224x224 pixels, which is a common size used in many deep-learning models.

Another important aspect is the representation of the color channels. The dataset images are in the BRG (Blue, Red, Green) format, which is not compatible with most deep-learning models. Therefore, converting them to the RGB (Red, Green, Blue) format is necessary, which is more widely used.

By resizing the images to a standard size and converting them to the RGB format, the dataset can be represented in a way that is compatible with many popular deep learning models, allowing for accurate and efficient analysis of diabetic retinopathy.

The count of images in each class is shown in the given table:

3. Proposed Methodology

Table 3.1: Data Set

No.	Class	No. of images
1	No-DR	711
2	Mild (or early)	6
3	Moderate	110
3	Severe	210
4	Very Severe	139
5	PDR	116
4	Advanced PDR	145

3.5.3 Morphological Operations

Morphological operations refer to a collection of image processing methods employed to alter the form and arrangement of objects within an image. These operations can be useful in analyzing and segmenting complex images, such as those found in medical imaging datasets like fundus images dataset. We will discuss how morphological operations can be used to perform a series of image processing steps on the dataset, including thresholding, Gaussian blurring, finding contours, getting coordinates, cropping images, and resizing images.

3.5.3.1 Thresholding

The first step in this process is thresholding, which is a technique used to separate the foreground and background of an image based on pixel intensity values. This is important because it allows us to isolate the areas of the image that contain the retinas of interest.

3.5.3.2 Gaussian blur

After thresholding, we apply Gaussian blurring, which is a smoothing operation that reduces noise and removes small details from the image. This helps to create a more uniform background, which can make it easier to identify and segment the areas of interest.

Next, we use the find contours function to identify the boundaries of the retinas in the image. This is done by detecting areas of the image where the pixel intensity changes abruptly. These boundaries can be represented as a set of connected points or lines, which

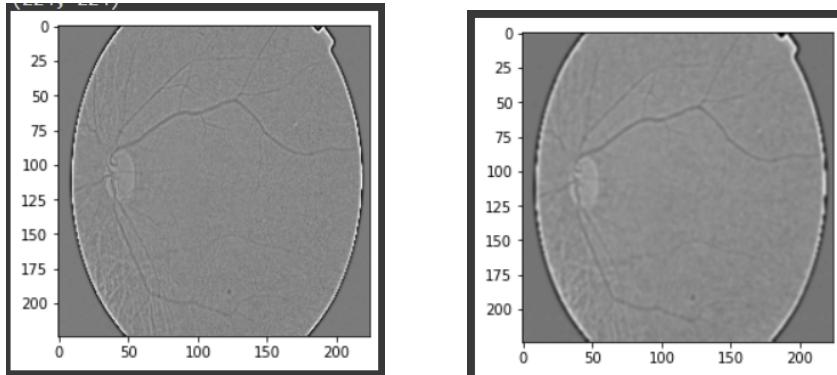


Figure 3.12: Before and after applying gaussian blur

can then be used to extract the coordinates of the retinas. Once we have the coordinates of the retinas, we can crop the images to isolate them from the surrounding background. This can be done by selecting a rectangular region of interest that contains the retina and then cropping the image to that region. Finally, we resize the images to a standardized size, which is important for training deep learning models. This can be done using a variety of methods, such as nearest neighbor interpolation or bilinear interpolation.

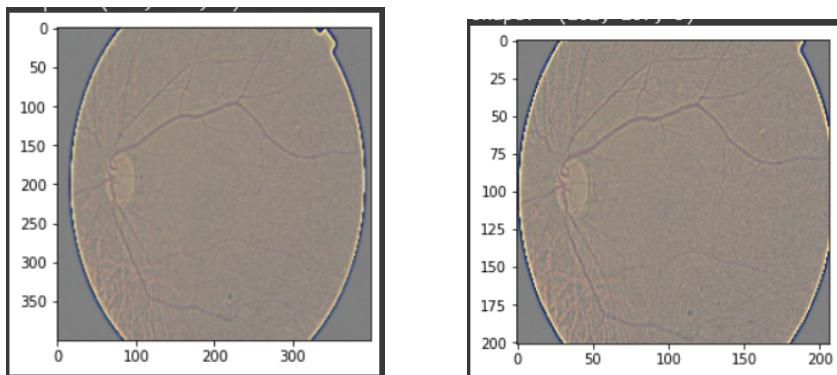


Figure 3.13: Before and after cropping the image

In conclusion, morphological operations provide a powerful set of tools for analyzing and processing medical imaging datasets like fundus images dataset. By performing thresholding, Gaussian blurring, finding contours, getting coordinates, cropping images, and resizing images in a specific order, we can extract useful information from these datasets and prepare them for analysis with deep learning models.

3. Proposed Methodology

3.5.4 CLAHE

CLAHE, or Contrast Limited Adaptive Histogram Equalization, is a popular technique for enhancing digital image contrast. In this technique, the image is divided into small regions, and the contrast of each region is enhanced separately using the adaptive histogram equalization method. This results in a more balanced distribution of pixel intensities, which can lead to better visualization and analysis of the image. We will discuss the CLAHE operations that can be applied to the fundus images dataset. The dataset consists of images in the RGB color space, which we will first convert to the LAB color space. In conclusion, applying CLAHE operations to the fundus images dataset can enhance the contrast of the retinal images and improve the accuracy of diagnosing diabetic retinopathy. By converting the images to LAB, flattening the luminance channel, applying CLAHE, merging the color channels, and converting back to RGB, we can effectively apply CLAHE to the fundus images dataset.

The first step in applying CLAHE to the fundus images dataset is to convert the images from RGB to LAB. The LAB color space is a three-dimensional color space, which separates the color information from the brightness information. This allows us to apply the CLAHE technique only to the luminance channel, which controls the brightness of the image. Once the images are converted to LAB, we will flatten the luminance channel, which converts the three-dimensional LAB image to a two-dimensional grayscale image. This is necessary because the CLAHE algorithm requires a grayscale image as input.

Next, we will apply the CLAHE algorithm to the flattened grayscale image. The CLAHE algorithm applies adaptive histogram equalization to small regions of the image, using a contrast limiting parameter to prevent over-amplification of noise. This results in a more balanced distribution of pixel intensities, which can enhance the contrast of the image. After applying the CLAHE algorithm, we will merge the processed luminance channel with the original color channels, which restores the color information to the image. Finally, we will convert the LAB image back to RGB, which produces the final processed image.

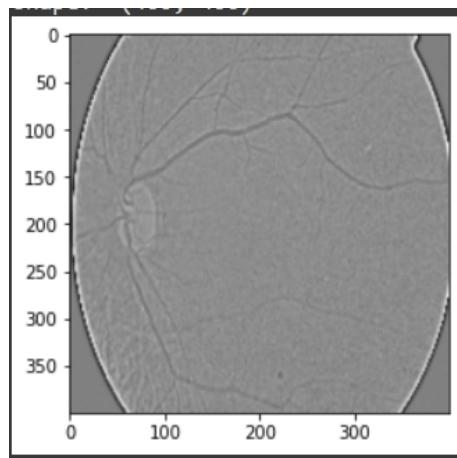


Figure 3.14: RGB to LAB

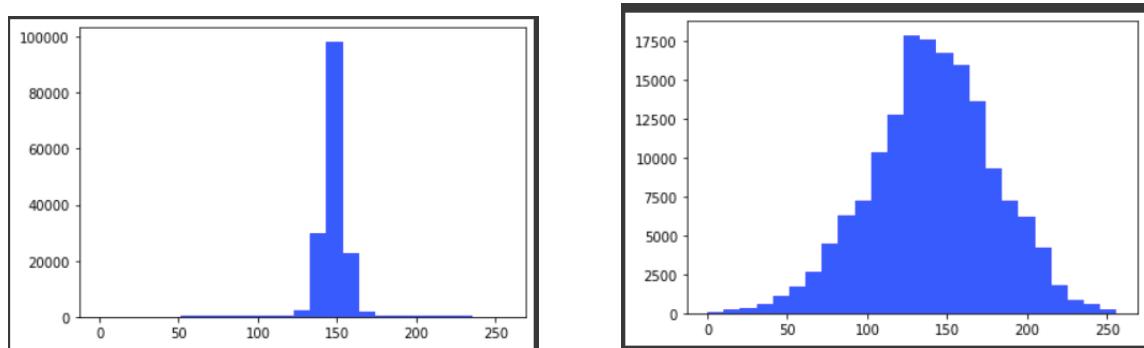


Figure 3.15: Contrast-Limited Adaptive Histogram Equalization

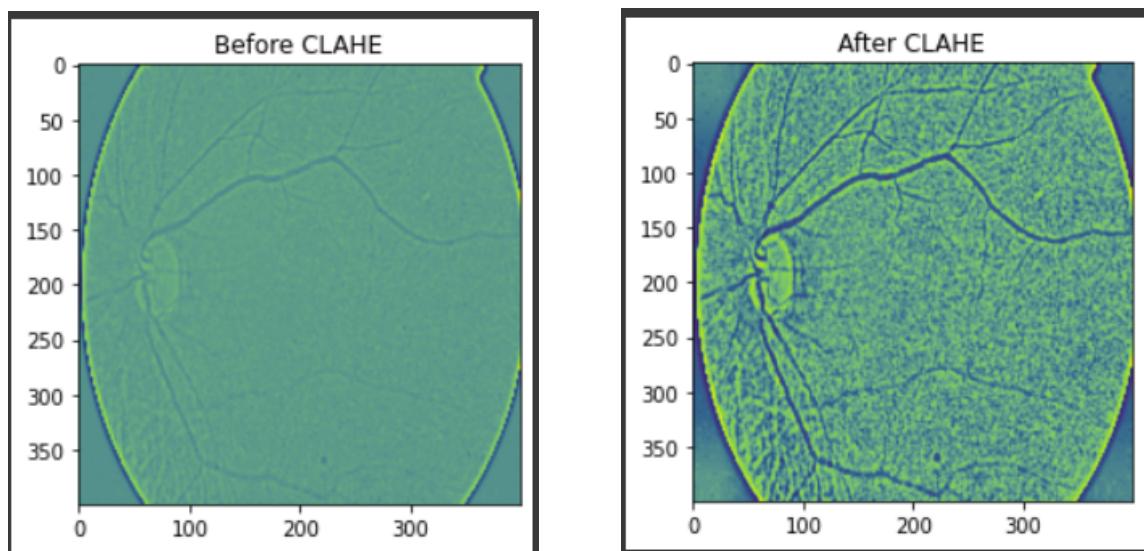


Figure 3.16: Before and after applying CLAHE

3. Proposed Methodology

3.5.4.1 Median Blur and Masking

After implementing CLAHE procedures on the fundus images dataset, it is possible to come across noise that may impact the image's accuracy. To address this issue, the widely used approach of median blur can be employed, which eliminates noise from images while retaining their edges and intricate features. The median blur method involves replacing each pixel in the image with the median value derived from its neighboring pixels. By utilizing this technique, noise can be effectively eliminated from images without causing substantial blurring.

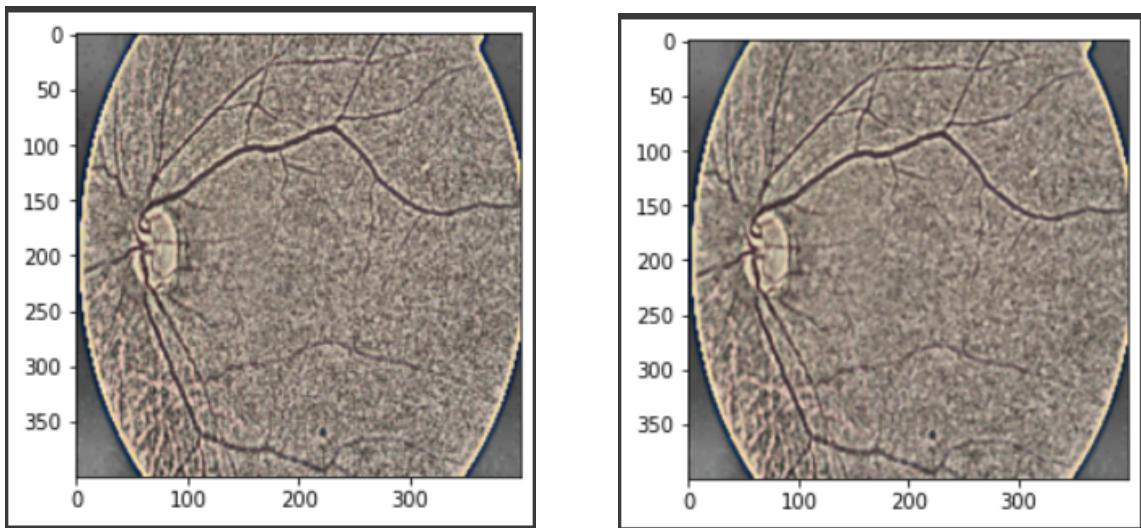


Figure 3.17: Before and after applying median blur

After applying median blur, we may still encounter some areas of the image with noise or artifacts. To address this issue, we can apply masking to the image. Masking involves creating a binary image that specifies the regions of the image that we want to preserve or modify. By using the masked image to apply additional filters or processing techniques, we can selectively enhance the quality of the image. In conclusion, applying median blur to the fundus images dataset after applying CLAHE operations can help to remove noise and artifacts from the image. Additionally, applying masking techniques can enhance the quality of the image by selectively modifying specific regions of the image.

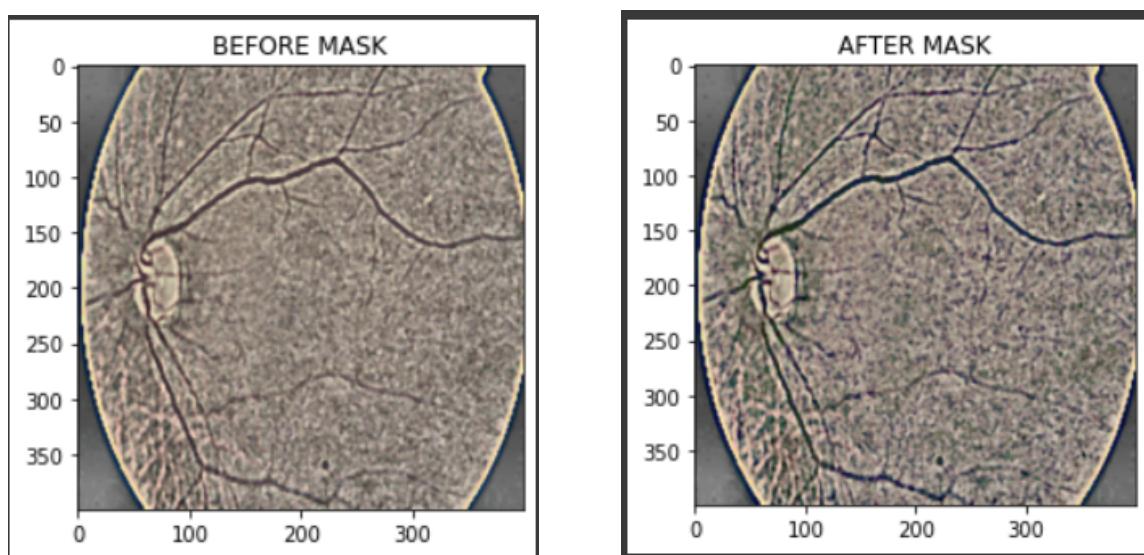


Figure 3.18: Before and after masking

4

Experiments and Results

4.1 Evaluation Metrics

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Figure 4.1: Confusion Matrix

- **True Positive (TP):** True Positive (TP) refers to the count of instances that are genuinely positive and are correctly predicted as positive by the model.
- **True Negatives (TN):** True Negatives (TN) refer to instances that are genuinely negative, and the model accurately identifies them as negative.
- **False Positive (FP):** A false positive (FP) occurs when the model wrongly identifies instances as positive when they are actually negative.
- **False Negative (FN):** False Negative (FN) refers to the count of cases where the model inaccurately identifies positive instances as negative.
- **Accuracy:** Accuracy refers to the proportion of accurate predictions out of the total number of predictions made.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.1)$$

4. Experiments and Results

- **Precision:** Precision is a measurement utilized to assess the effectiveness of a multi-class prediction model, gauges the accuracy of correctly predicted instances within a particular class relative to all instances classified by the model as belonging to that class.

$$Precision_{class} = \frac{N_{class,class}}{\sum_{i=1}^C N_{(i,class)}} \quad (4.2)$$

- **Recall:** Recall is a measure employed to assess how well a model identifies all pertinent occurrences of a particular class. It quantifies the ratio of accurate positive predictions to the total number of instances that truly belong to the positive class.

$$Recall_{class} = \frac{N_{class,class}}{\sum_{i=1}^C N_{(class,i)}} \quad (4.3)$$

- **F1-score:** F1-score, a widely employed measure for assessing model performance, is calculated as the harmonic mean of precision and recall, as defined by the following equation:

$$F1_{class} = \frac{2 \times Precision_{class} \times Recall_{class}}{Precision_{class} + Recall_{class}} \quad (4.4)$$

4.2 Experiments

After conducting our experiments, we obtained results showcasing the accuracy, precision, recall, and f1-score of our project. We tested four different architectures on the same dataset, comparing their accuracies. Our main objective was to identify the most optimal architecture for our project. Through these experiments, we aimed to determine the architecture that delivers the best performance based on the metrics mentioned.

4.2.0.1 Experiment-1

The first architecture we used was CNN Model , and we tabulated the results to analyze them. We found that CNN produced promising results, which enabled us to understand its effectiveness for our project. However, we continued with our experimentation process

to identify the most suitable architecture. The loss and accuracy graphs of the CNN model are shown in Figure 4.2.

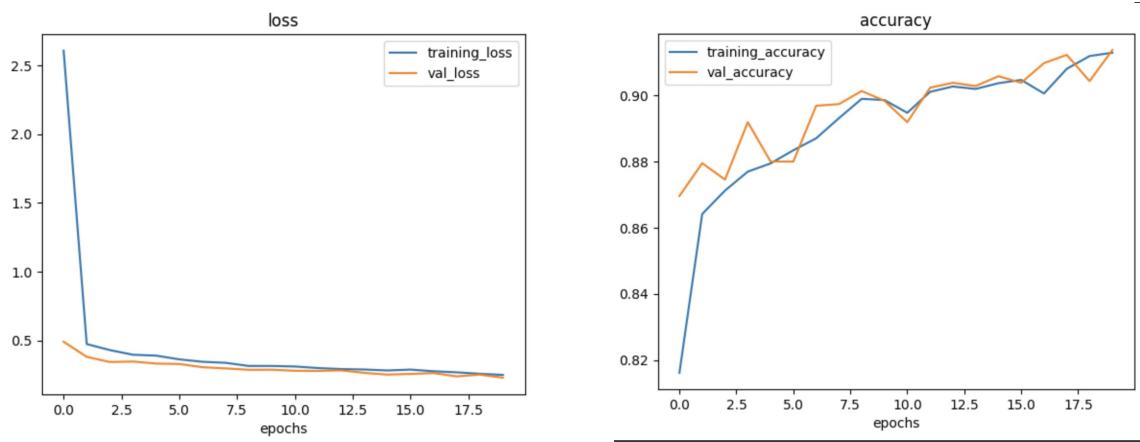


Figure 4.2: Loss and Accuracy graphs of CNN MODEL

Table 4.1: CNN MODEL Results

Class	precision	recall	f1-score
No_DR	0.93	0.98	0.95
Mild	1.00	0.72	0.84
Moderate	1.00	0.72	0.84
Severe	0.83	0.81	0.082
Very_Severe	0.93	0.76	0.084
PDR	0.74	0.78	0.76
Advanced_PDR	0.79	0.88	0.83

4.2.0.2 Experiment-2

After completing our initial experiments, we decided to further explore the performance of our project by using the VGG19 architecture. We conducted tests and obtained excellent results. To visualize our findings, we included the loss and accuracy graphs of the VGG19 model in Figure 4.3.

This comprehensive analysis guided us in making informed decisions regarding the selection of the architecture that best suited our requirements and objectives.

4. Experiments and Results

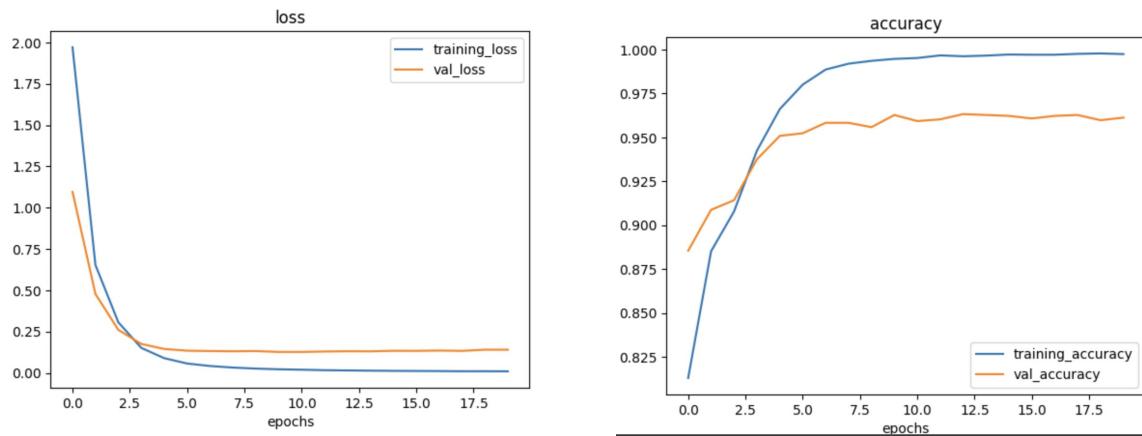


Figure 4.3: Loss and Accuracy graphs of VGG-19

Table 4.2: VGG-19 Results

Class	precision	recall	f1-score
No_DR	0.91	0.96	0.94
Mild	1.00	0.50	0.67
Moderate	0.86	0.69	0.77
Severe	0.81	0.71	0.76
Very_Severe	0.75	0.89	0.81
PDR	0.70	0.64	0.67
Advanced_PDR	0.93	0.93	0.93

4.2.0.3 Experiment-3

DenseNet-169 was employed to predict diabetic retinopathy using the fundus images dataset comprising high-resolution retinal images. The model was able to achieve high accuracy in detecting the presence and severity of diabetic retinopathy in the images. The use of DenseNet-169 proved to be effective and efficient in predicting diabetic retinopathy, demonstrating the potential for further advancements in medical image analysis. The loss and accuracy graphs of DensNet-169 are shown in the below Fig 4.4

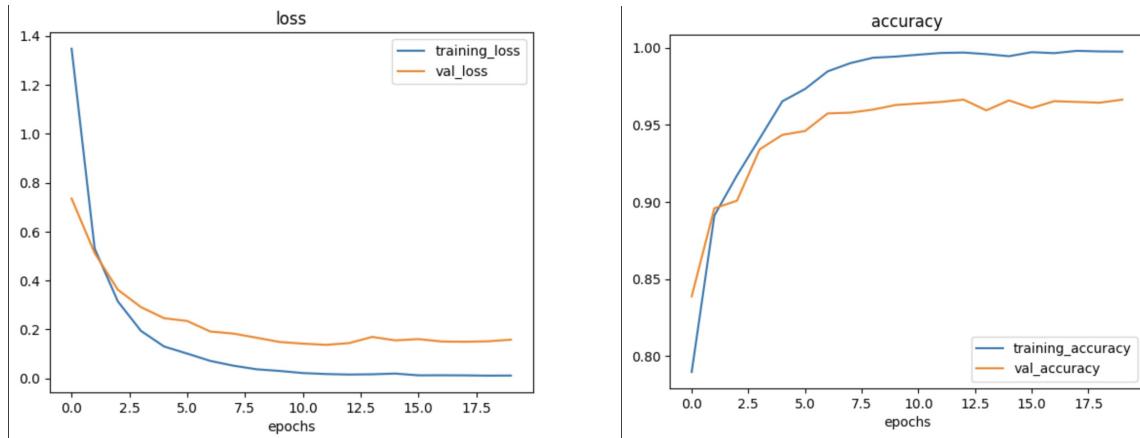


Figure 4.4: Loss and Accuracy graphs of DensNet-169

Table 4.3: DensNet-169 Results

Class	precision	recall	f1-score
No_DR	0.91	0.96	0.93
Mild	1.00	1.00	1.00
Moderate	0.86	0.73	0.79
Severe	0.73	0.76	0.74
Very_Severe	0.91	0.74	0.82
PDR	0.68	0.68	0.68
Advanced_PDR	0.90	0.90	0.90

4.2.0.4 Experiment-4

ResNet-152 was utilized to predict diabetic retinopathy by analyzing a dataset containing high-resolution retinal images. The model exhibited notable accuracy in identifying the presence and severity of diabetic retinopathy in the images. Employing ResNet-152

4. Experiments and Results

proved to be both effective and efficient in the prediction of diabetic retinopathy, showcasing promising prospects for future developments in medical image analysis. The graphical representation of the model's performance, as depicted in the figures below 4.5, illustrates the loss and accuracy metrics throughout its training process.

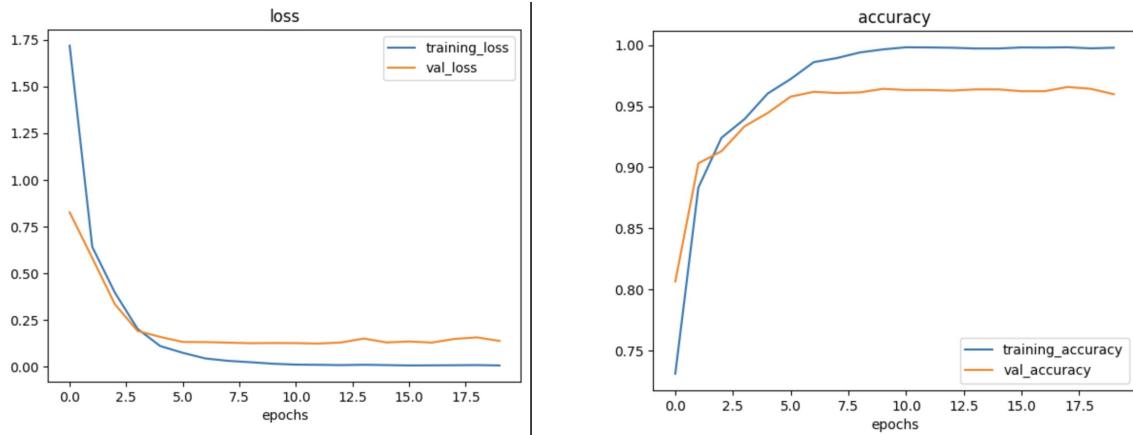


Figure 4.5: Loss and Accuracy graphs of ResNet-152

Table 4.4: ResNet-152 Results

Class	precision	recall	f1-score
No_DR	0.93	0.98	0.95
Mild	0.00	0.00	0.00
Moderate	0.90	0.69	0.78
Severe	0.75	0.89	0.82
Very_Severe	0.80	0.71	0.75
PDR	0.88	0.54	0.67
Advanced_PDR	0.83	1.00	0.91

4.2.0.5 Experiment-5

We then implemented novel pre-processing techniques before using our final Densenet-121 model. The pre-processing techniques we used improved the accuracy, precision, recall, and f1-score of our model, making it more effective for our specific needs. The precision, recall and f1-score after running the model are shown in Table 4.5. The confusion matrix is shown for the model is shown in Figure 4.7. We then implemented our Densenet-121 model and tabulated the results obtained. The model produced impressive

4.2 Experiments

results, highlighting the effectiveness of our pre-processing techniques and Densenet-121 architecture.

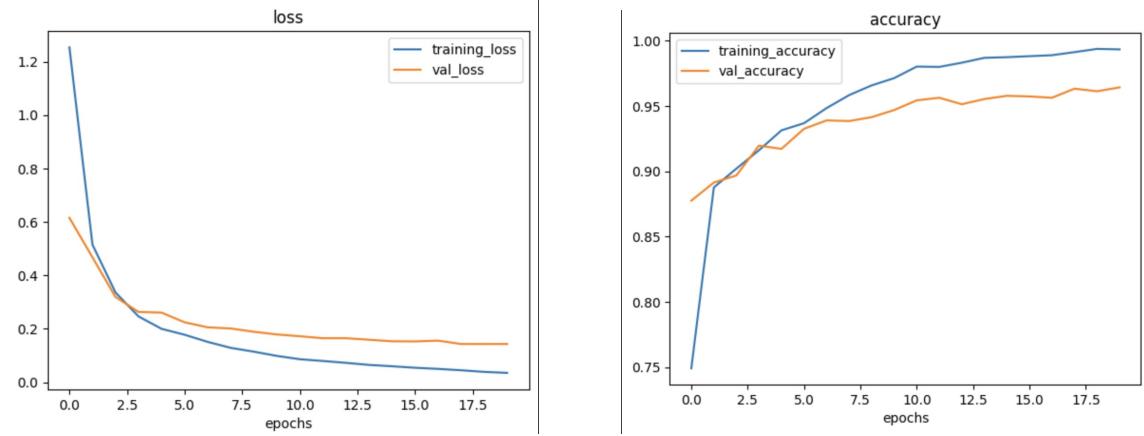


Figure 4.6: Loss and Accuracy graphs of DensNet-121

Table 4.5: DensNet-121 Results

Class	precision	recall	f1-score
No_DR	0.92	0.97	0.94
Mild	0.00	0.00	0.00
Moderate	0.87	0.77	0.72
Severe	0.91	0.76	0.83
Very_Severe	0.78	0.78	0.78
PDR	0.75	0.68	0.71
Advanced_PDR	0.78	0.93	0.85

4. Experiments and Results



Figure 4.7: Training and Testing Confusion Matrix of DensNet-121 Model

4.3 Results

We tabulated the results and compared the three models to analyze the differences. We found that the Resnet-152 model was the most optimal architecture for our project, as it produced slightly better results compared to Densenet-121, VGG-19, and DensNet-169, which are shown in Table 4.6. The model's superior performance is attributed to its architecture and the novel pre-processing techniques we used. The pre-processing techniques enabled us to improve the quality of the data, making it easier for the model to identify patterns and make accurate predictions.

Table 4.6: Accuracy Table

Model	Accuracy(%)	precision	recall	f1-score
Custom CNN	55	0.36	0.55	0.43
VGG-19	88	0.87	0.88	0.88
DensNet-169	88	0.88	0.88	0.88
DensNet-121	89	0.89	0.89	0.89
RESENET-152	90	0.91	0.90	0.90

4.3.1 Discussion

In conclusion, our experimentation process involved analyzing the accuracy, precision, recall, and f1-score of our project using four different architectures. We found our proposed Resent-152 model outperformed the other four models. The results obtained from our experiments provide valuable insights into the effectiveness of different architectures and pre-processing techniques. We believe that our findings will be useful for researchers and practitioners in machine learning and deep learning. The success of our Resent-152 model has encouraged us to continue exploring new techniques and architectures to improve the accuracy and effectiveness of our models.

5

Conclusions and Future Scope

5.1 Conclusions

In conclusion, this thesis aimed to address the problem of detecting diabetic retinopathy using deep learning models. By implementing various deep learning models, including Custom CNN model, Resnet-152, VGG-19, and DensNet-169, and using pre-processing techniques, we were able to build a Modified Resnet-152 architecture that achieved an accuracy of 90% in detecting diabetic retinopathy. This is a significant contribution to the field, as diabetic retinopathy is a common complication of diabetes that can lead to blindness if left untreated. Our findings demonstrate the potential of deep learning models in detecting diabetic retinopathy and provide insights into the efficacy of different models for this task. Furthermore, our comparison of different models highlights the importance of selecting appropriate models for specific tasks. However, our study also has limitations that need to be considered, such as the limited size of the dataset and the need for further testing on larger datasets. Nevertheless, the results of our study have significant implications for future research in the field and suggest potential applications of deep learning models in healthcare. Overall, this thesis provides valuable insights into the detection of diabetic retinopathy and highlights the potential of deep-learning models in medical image analysis.

5.2 Future Scope

The future scope of this work involves exploring the use of larger and diverse datasets to improve the robustness and generalizability of the deep learning models. Additionally, more advanced pre-processing techniques and data augmentation methods could be explored to improve the models' performance further. Furthermore, the integration of these models with telemedicine systems could provide a cost-effective and accessible way to detect DR in underserved regions. Finally, investigating the feasibility of deploying these models on mobile devices would enable on-the-spot diagnosis of DR, which could significantly improve patient outcomes.

Bibliography

- [1] E. V. Carrera, A. González, and R. Carrera, “Automated detection of diabetic retinopathy using svm,” in *2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, 2017, pp. 1–4.
- [2] S. S. Kar and S. P. Maity, “Automatic detection of retinal lesions for screening of diabetic retinopathy,” *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 3, pp. 608–618, 2018.
- [3] W. M. Gondal, J. M. Köhler, R. Grzeszick, G. A. Fink, and M. Hirsch, “Weakly-supervised localization of diabetic retinopathy lesions in retinal fundus images,” in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 2069–2073.
- [4] C. Jayakumari, V. Lavanya, and E. P. Sumesh, “Automated diabetic retinopathy detection and classification using imagenet convolution neural network using fundus images,” in *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, 2020, pp. 577–582.
- [5] A. Bagaskara and M. Suryanegara, “Evaluation of vgg-16 and vgg-19 deep learning architecture for classifying dementia people,” in *2021 4th International Conference of Computer and Informatics Engineering (IC2IE)*, 2021, pp. 1–4.
- [6] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
- [7] S. M. S. Islam, M. M. Hasan, and S. Abdullah, “Deep learning based early detection and grading of diabetic retinopathy using retinal fundus images,” *arXiv preprint arXiv:1812.10595*, 2018.
- [8] D. Das, S. K. Biswas, and S. Bandyopadhyay, “Detection of diabetic retinopathy using convolutional neural networks for feature extraction and classification (drfec),” *Multimedia Tools and Applications*, pp. 1–59, 2022.
- [9] S. Sanjana, N. S. Shadin, and M. Farzana, “Automated diabetic retinopathy detection using transfer learning models,” in *2021 5th International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*. IEEE, 2021, pp. 1–6.
- [10] N. Yalçın, S. Alver, and N. Uluhatun, “Classification of retinal images with deep learning for early detection of diabetic retinopathy disease,” in *2018 26th Signal Processing and Communications Applications Conference (SIU)*. IEEE, 2018, pp. 1–4.
- [11] V. Raja Sarobin M and R. Panjanathan, “Diabetic retinopathy classification using cnn and hybrid deep convolutional neural networks,” *Symmetry*, vol. 14, no. 9, p. 1932, 2022.

- [12] S. H. Kassani, P. H. Kassani, M. J. Wesolowski, K. A. Schneider, and R. Deters, “Breast cancer diagnosis with transfer learning and global pooling,” in *2019 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2019, pp. 519–524.
- [13] M. T. Rahman and A. Dola, “Automated grading of diabetic retinopathy using densenet-169 architecture,” in *2021 5th International Conference on Electrical Information and Communication Technology (EICT)*. IEEE, 2021, pp. 1–4.
- [14] B. Tymchenko, P. Marchenko, and D. Spodarets, “Deep learning approach to diabetic retinopathy detection,” *arXiv preprint arXiv:2003.02261*, 2020.
- [15] S. H. Kassani, P. H. Kassani, R. Khazaeinezhad, M. J. Wesolowski, K. A. Schneider, and R. Deters, “Diabetic retinopathy classification using a modified xception architecture,” in *2019 IEEE international symposium on signal processing and information technology (ISSPIT)*. IEEE, 2019, pp. 1–6.
- [16] N. Jiwani, K. Gupta, and N. Afreen, “A convolutional neural network approach for diabetic retinopathy classification,” in *2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT)*. IEEE, 2022, pp. 357–361.
- [17] S. Mishra, S. Hanchate, and Z. Saquib, “Diabetic retinopathy detection using deep learning,” in *2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, 2020, pp. 515–520.
- [18] N. Nasir, P. Oswald, O. Alshaltone, F. Barneih, M. Al Shabi, and A. Al-Shammaa, “Deep dr: Detection of diabetic retinopathy using a convolutional neural network,” in *2022 Advances in Science and Engineering Technology International Conferences (ASET)*. IEEE, 2022, pp. 1–5.
- [19] S. Taufiqurrahman, A. Handayani, B. R. Hermanto, and T. L. E. R. Mengko, “Diabetic retinopathy classification using a hybrid and efficient mobilenetv2-svm model,” in *2020 IEEE REGION 10 CONFERENCE (TENCON)*. IEEE, 2020, pp. 235–240.
- [20] C.-Y. Chen and M.-C. Chang, “Using deep neural networks to classify the severity of diabetic retinopathy,” in *2022 IEEE International Conference on Consumer Electronics-Taiwan*. IEEE, 2022, pp. 241–242.
- [21] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.
- [22] J. C. M. R. J. L. V. N. M. G.-T. J. A. D. P. P.-R. P. E. G.-S. J. F. . S. A. G. Veronica Elisa Castillo Benítez, Ingrid Castro Matto, “Densely connected convolutional networks,” 2018.
- [23] J. C. M. R. J. L. V. N. M. G.-T. J. A. D. P. P.-R. P. E. G.-S. J. F. V. E. Castillo Benítez, I. Castro Matto and S. A. Grillo, “Dataset from fundus images for the study of diabetic retinopathy,,” vol. 36, p. 107068, 2021.
- [24] P. Saranya, S. K. Devi, and B. Bharanidharan, “Detection of diabetic retinopathy in retinal fundus images using densenet based deep learning model,” in *2022 International Mobile and Embedded Technology Conference (MECON)*. IEEE, 2022, pp. 268–272.

Bibliography

- [25] E. AbdelMaksoud, S. Barakat, and M. Elmogy, “Diabetic retinopathy grading based on a hybrid deep learning model,” in *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)*. IEEE, 2020, pp. 1–6.
- [26] M. Oulhadj, J. Riffi, K. Chaimae, A. M. Mahraz, B. Ahmed, A. Yahyaouy, C. Fouad, A. Meriem, B. A. Idriss, and H. Tairi, “Diabetic retinopathy prediction based on deep learning and deformable registration,” *Multimedia Tools and Applications*, vol. 81, no. 20, pp. 28 709–28 727, 2022.
- [27] D. Rahhal, R. Alhamouri, I. Albataineh, and R. Duwairi, “Detection and classification of diabetic retinopathy using artificial intelligence algorithms,” in *2022 13th International Conference on Information and Communication Systems (ICICS)*. IEEE, 2022, pp. 15–21.
- [28] M. Ferguson, R. ak, Y.-T. Lee, and K. Law, “Automatic localization of casting defects with convolutional neural networks,” *12 2017*, pp. 1726–1735.
- [29] M. U. Emon, R. Zannat, T. Khatun, M. Rahman, M. S. Keya *et al.*, “Performance analysis of diabetic retinopathy prediction using machine learning models,” in *2021 6th International Conference on Inventive Computation Technologies (ICICT)*. IEEE, 2021, pp. 1048–1052.
- [30] A. Reethika, J. Sathish, P. K. Priya, F. D. Shadrach, and M. S. Kanivarshini, “Diabetic retinopathy detection using statistical features,” in *2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM)*, vol. 2, 2022, pp. 44–48.
- [31] S. Burewar, A. B. Gonde, and S. K. Vipparthi, “Diabetic retinopathy detection by retinal segmentation with region merging using cnn,” in *2018 IEEE 13th International Conference on Industrial and Information Systems (ICIIS)*, 2018, pp. 136–142.
- [32] L. Qiao, Y. Zhu, and H. Zhou, “Diabetic retinopathy detection using prognosis of microaneurysm and early diagnosis system for non-proliferative diabetic retinopathy based on deep learning algorithms,” *IEEE Access*, vol. 8, pp. 104 292–104 302, 2020.
- [33] A. Singh Gautam, S. Kumar Jana, and M. P. Dutta, “Automated diagnosis of diabetic retinopathy using image processing for non-invasive biomedical application,” in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, 2019, pp. 809–812.
- [34] S. Mishra, S. Hanchate, and Z. Saquib, “Diabetic retinopathy detection using deep learning,” in *2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, 2020, pp. 515–520.
- [35] R. Ghosh, K. Ghosh, and S. Maitra, “Automatic detection and classification of diabetic retinopathy stages using cnn,” in *2017 4th International Conference on Signal Processing and Integrated Networks (SPIN)*, 2017, pp. 550–554.
- [36] Y. Huang, L. Lin, P. Cheng, J. Lyu, and X. Tang, “Identifying the key components in resnet-50 for diabetic retinopathy grading from fundus images: a systematic investigation,” *ArXiv*, vol. abs/2110.14160, 2021.

Bibliography