Classification of Retinal Diseases from OCT Images Using Deep Learning Models

by

Md Saif Mokarrom
20301121
Md Anonto Shuvo
23141036
Nazmul Hasan Oyon
20101528
Rifha Hossain Munaja
20301466
Soumik Roy
20101573

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
January 2024

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Md Saif Mokarrom 20301121	Rifha Hossain Munaja 20301466
Md Anonto Shuvo 23141036	Soumik Roy 20101573

Nazmul Hasan Oyon
20101528

Approval

The thesis titled "Classification of Retinal Diseases from OCT Images Using Deep Learning Models"
submitted by

1. Md Saif Mokarrom (20301121)
2. Md Anonto Shuvo (23141036)
3. Rifha Hossain Munaja (20301466)
4. Soumik Roy (20101573)
5. Nazmul Hasan Oyon (20101528)

Of Fall, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 22, 2024.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Ashraful Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Arif Shakil
Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Table of Contents

Abstract

Retina is an important part of our vision, but it can easily get affected and create various vision problems like vision loss and others. According to the statistics provided by The World Health Organization, it is estimated that globally at least 2.2 billion people suffer from various retinal disorders. It's important to accurately classify retinal diseases since early detection can help in taking steps for treatment. In this paper, we have classified different types of retinal diseases which are based on OCT images. OCT images were used because they produce a lot of fine-grained retinal images that are useful for diagnosing and monitoring changes to the retina and optic nerve over time. For the classification, we have used Deep learning Models such as CNN models for predicting the accuracy. Moreover, we have proposed a new model for the classification. Our custom model gives an accuracy of 95.05% which is better compared to other pre-trained models. Both DME and DRUSEN class obtained maximum precision that is 97% and Normal class obtained maximum recall which is 98%. Furthermore, we have used Explainable AI (XAI) Techniques with Grad-CAM for better analysis and created a web application for live visualization of result.

Keywords: Retina classification, Proposed model, Convolutional Neural network, Grad-CAM

Chapter 1

Introduction

1.1 Background

The retina located inside the inner wall of the eye regulates light and image transmission to the brain. Regular eyesight occurs when light converges accurately on the retina. If the retinal layer sustains damage, it can result in enduring blindness due to many circumstances. Optical coherence tomography (OCT) is a sophisticated high resolution imaging technique. Using the projected laser beams, this method can create tomographic sectional images of the item under study with a high-depth resolution. Its benefits are non-contact, non-invasive, and quick imaging. OCT can provide exact information about biological tissue optical scattering and absorption. It is used to view the retina and assess ocular illnesses including age related macu lar degeneration, choroidal neovascularization glaucoma and others. OCT pictures simplify the physical characteristics such as the location, circulation of blood vessels, macular holes, cysts and drusen as disease indicators. Many of the working population has DME, a retinal problem that develops due to diabetes[1]. It occurs due to elevated glucose levels in the bloodstream and can cause significant harm to the eye's blood vessels, leading to worsening of the visuals. The enlargement of the retina as a result of the subsequent fluid leaking affects the macula's ability to operate correctly. People who have dry age related macular degeneration problem, they sometimes experience wet AMD, a medical condition where erroneous blood vessels spread into the retina and turn the retina soggy and it is called CNV. Moreover, this fluid can gradually spread and harm the retina resulting in the deterioration of light-sensitive cells known as photoreceptors. The Drusen are yellow crystals that develop beneath the retina. They are made up of lipids and proteins. It can be of small, medium or big sizes. People who are 50 or greater without age-related macular degeneration, it is typical for them. Sometimes, having many small and big Drusen can point out the symptoms of AMD. Other types of Drusen have also been found in the optic nerve, but these usually do not affect vision. The CNN methodology is an advanced Deep Learning method for diagnosing medical conditions, especially when using image-based data such as OCT images for eye infection detection and diagnosis, finding lesions and segmenting the retinal layers. CNN is a deep learning model capable of autonomously extracting multiple layers of deep features from input images in a hierarchical manner. We have also successfully applied transfer learning techniques on the training data based on OCT. The retina is a critical component in transmitting visual information to the brain. Damage to the retina can cause serious illnesses that might end in blindness. Within the realm of ophthalmology OCT is emerging as a powerful tool offering non-invasive and precise imaging capabilities for diagnosing and tracking these conditions, facilitating early intervention and enhancing patient outcomes.

1.2 Problem Statement

The world's population is increasing daily and along with it, the rate of people getting affected by retinal diseases is also increasing. According to The World Health Organization reports, at least 2.2 billion people suffer from different retinal disorders globally. These retinal diseases can occur for various reasons. Age is an essential factor as with age, conditions are seen to prevail more. For example, macular holes and macular degeneration are standard in older adults. Hypertension is also a factor, as high blood pressure can hamper retinal vessels. Besides, increase in the sugar levels in the blood can also interfere with the retinal blood vessels. Early detection of such diseases is essential for effective treatment and helping people from losing their vision. In such a situation, classification of retinal diseases is essential for detecting the diseases at an early stage and starting treatment. According to WHO, around 1 billion people's vision impairment could have been prevented if identified early. Moreover, the yearly global cost of productivity associated with vision impairment is estimated to be US 411 billion dollars. Classification helps the doctors identify the disease and take steps accordingly, like suggesting the treatment plan, monitoring and tracking its progression and making adjustments to the treatment. Early detection lowers the overall cost of the patients. Classification also helps in research for doctors to know more about the diseases. In this paper, our objective of classification of retinal disease is to enhance the affected people's quality of life. We will use optical coherence tomography (OCT) images for our research. For the classification, we want to use Deep learning Models such as CNN and Vision Transformers to predict accuracy. Also, we are planning to propose a new model for the classification. We want to use Explainable AI (XAI) Techniques with Grad-CAM for better analysis.

1.3 Research Objective

Our main goal of research is to identify diseases using optical coherence tomography (OCT) images. We want to detect this by employing advanced deep learning techniques, specifically Convolutional Neural Networks (CNNs) and Vision Transformers, to enhance the accuracy of our predictions. The research has the following objectives:

1. Creating a deep learning algorithm to categorize retinal disorders based on OCT images.
2. Assessing the effectiveness of our model by comparing it to other cutting-edge models .
3. Examining the characteristics identified by our model to gain insights into its retinal disease classification process .
4. Introducing a novel approach to elucidate the rationale behind the predictions made by our model.

Chapter 2

Related Work

In this study, the authors examined and used dataset images of CNV, Drusen, Nor mal, DME to identify ocular retinal diseases. Sertkaya et al.[2] have used VGG-16 architecture and it produced a high score which is 93.01%. Moreover, LeNet was used for retinal disease identification. Lastly, the Alexnet model results in loss reduction after all the graphs have been evaluated. They will seek to identify the distorted region by eliminating the heat map using deep learning techniques for better improvement.

Generative Adversarial Network (GAN) is a system that can forecast retinal decay while synthesizing fluorescein angiography pictures from photographs conducted in Kamran et al.[3] paper. An exogenous dye is injected into the circulation during FA in order to scan the vascular system. They described that after taking dye injection can have a huge effect like allergic shock, vomiting, nausea and even death. With Vtgan they can uniquely translate the angiogram. Contrarily, for taking pictures of the retina there is a method named color fundus imaging that is insufficiently accurate to record retina's structure. The sole noninvasive technique OCT Angiography is responsible for recording retinal vasculature. Their dataset consists of 29 sets of sick (fundus fluorescein angiograms) and 30 healthy person sets.

Baz et al.[4] have talked about a method that can preserve an image from losing its information. Moreover, it can also withstand the noise ratio from the images. That method is Automatic Segmentation. With the help of this procedure, doctors can detect and early therapeutic monitoring is possible. For identifying pathogenic changes and treating retinal illnesses, thickness measurements are essential. They conducted a study about which sectors of OCT are used to examine the structure of retina layers in order to specify ocular illnesses such as glaucoma. They proposed different layers (IRC, IS, ONL, OPL and POS) for identifying OCT retinal images. The authors have also studied some challenges and techniques for recovering the future problems among them.

In this study, Islam et al.[5] have introduced some transfer learning models like VGG16, Inception v3 and Resnet. Furthermore, some Vision Transformer like EANet, CCT and Swin also used by the authors. They constructed renal disease related detection system which deals with the 3 categories of kidney diseases. They have taken a dataset of almost twelve thousand. It consists of cysts, normal, stones and tumors which has 3709, 5077, 1377 and 2283 data respectively. After comparing them, in terms of performance the Swin Transformer showed best with an accuracy of 99.30%.

Hosain et al.[6] have described a vision transformer method with DenseNet201 for finding gastrointestinal illnesses. They have taken the curated images of the colon part and used it to identify gastrointestinal tract problems. The dataset they have obtained contains WCE pictures of the gastrointestinal(GI) tract. There were four classes and the images were of 720x576 pixels. The classes are normal, polyps, esophagitis, and ulcerative colitis. Moreover, they have encountered resource and data limitations so they used an augmentation strategy to overcome the data problem. The authors intend to work on a wider variety of gastrointestinal disorders using some methods that are based on vision transformers which will be more accurate.

He et al.[1] created a heatmap for automatically classifying retinal OCT images by applying it to the main highlighted tumor region image. The experiments took a dataset from OCT2017 and OCT-C8. Their Score CAM provided an interpretable method named SwinPoly Transformer network that can model multiscale characteristics. Because it creates a bond between nearby non intersecting windows in the last layer by adjusting window partition. The suggested method outperforms with an accuracy of 99.80% and 99.99% AUC in convolutional neural network approach .

Ma et al.[7] experimented with the viability of the ViT model for classifying retinal OCT images. They used (HCTNet) method for retinal OCT image categorization and then confirmed the viability of a Transformer based technique.It combines the benefits of Transformer and ConVNet in relating long range dependencies and extracting hierarchical abstract local analysis. The method outperforms the pure ViT and multiple classification methods with an overall accuracy of 91.56% and 86.18%. In a confirmation on these they got two retinal data from OCT 2017 and Srini vasan14.

Rasti et al.[8] describes in Optical Coherence Tomography (OCT) imaging technique, Computer Aided Diagnosis (CAD) system helps ophthalmologists to detect ocular problems early and guide them to monitor different types of treatment. A MCME ensemble model which is basically a novel CAD system helps to detect dry AMD which is Age Related Macular Degeneration and DMA including normal retina. A local and two types of public datasets were considered in this paper. OCT images were used in the dataset which contained 148 subjects and 45 acquisitions respectively. This process helped to pass over lesions detection processes, full retinal layers segmentation, and restore the true image. It introduced a new cost function and by using this MCME model precision rate was acquired 98.86%.

Three different convolutional neural models were used in Tayal et al.[9] (2021) for identifying the ocular diseases with an ADAM optimizer. Authors used OCT images for their research and there are various classes of images like diabetic macular edema, drusen and choroidal neovascularization. It was found that their model's accuracy is 0.965. Moreover the sensitivity and specificities are 0.960 and 0.986 respectively.

Some limitations were observed such as the dataset was taken from a particular region. These include limited images and shortage of different structural images. Moreover, for classifying disease accuracy, kappa value, F1 Score and losses metrics were used. The dropout regularization approach was used to avoid the problem of overfitting of the findings and early halting algorithms were applied.

With the help of OCT images, Subramanian et al.[10] proposed a method for identifying retinal disorders. The approach employs transfer learning in conjunction with two fine-tuning stages and Bayesian optimization. It gathered an OCT image dataset from eight different categories: normal, glaucoma (GL), age-related macular degeneration (AMD), diabetic retinopathy (DR), diabetic macular edema (DME), myopic choroidal neovascularization (CNV), central serous chorioretinopathy (CSR) and optic disc edema (ODE). As feature extractors, four pre-trained CNN models were used by the authors. On the dataset based on APTOS-2019 it was found that VGG16, DenseNet201, InceptionV3 and Xception has an average accuracy of 97.2%. Moreover, on the dataset based on IDRiD, the models got on average 96.9% accuracy.

Alqudah (2019)[11] have suggested a new model based on the CNN architecture called AOCT-NET. The authors have built it for classifying retinal diseases that may contain multiple classes automatically. For the experiment, SD-OCT images were used. The dataset consists of 5 categories. These include AMD, CNV, DME, Drusen, and normal cases. Different metrics like optimizers were used for training. The model was trained on 80% of the dataset and tested on the remaining 20%. The results showed accuracy by the AOCT-NET model is 95.30%. This is a significant improvement over previous methods, which have reported accuracy of around 90%.

LGCNN(Layer Guided Convolutional Neural Network) is a layer-guided neural network proposed by Huang et al.[12]. Three separate sets of OCT pictures from various categories Normal, DME, and CNV used in the study. Three max pooling layers, five convolutional layers and two fully linked layers make up the LGCNN's architecture and the network was trained by Adam optimizer and CNV. Dataset was trained and tested with ration 80:20. The LGCNN achieved 95.6% accuracy on the test set compared to previous methods that reported accuracies around 90%.

Qomariah et al.[13] used both CNN and SVM to predict diabetic retinopathy with the help of retinal images. They collected a dataset of 147 images of DR and 147 images of normal retinas from the Messidor database. Their CNN architecture was used as a modified VGG16 model, which was pre-trained by the ImageNet dataset. DR and Normal were the retinal pictures that used the final few layers of the CNN. The characteristics taken out of the CNN were used to train the SVM classifier. The test set accuracy for the proposed method was 95.83%, which represents a significant rise in the rate over earlier methods, according to the data.

Kim and Tran [14] proposed automated ways to divide images into (DME), (CNV), Drusen and Normal categories for classification. They have conducted a study where a number of (CNNs) are modified and they also have to prepare the picture as inputs for CNN base classifiers. FCN removed noise then it clipped retina layers from the images. For their experiments three different(CNNs) were trained and put in an ensemble learning models InceptionV3, VGG16 and ResNet152 it performed well against all other CNNs and achieved 98.9% accuracy, 98.0% sensitivity, and 99.6% specificity. They intend to continue researching for new features and ensemble models in work to get better results.

Hajabdollahi et al.[15] used the STARE dataset in this study which helped to achieve better accuracy and lower complexity. They showed retinal vessels segmentation in portable retinal diagnostic devices with both punning and quantization by CNN. Simple CNN structure also made it easier for hardware execution in onsite and portable diagnostic devices. After enhancing the original picture the fully connected layer quantization occurred and then convolutional layer pruning implemented for the simplification of CNN. In addition, 60% of convolutional layer weights were removed and after quantization AUC was 97%.

Asif et al.[16] described the use of a deep residual network as a classifier, it can analyze various types of Diabetic Retinopathy images. The authors worked with ResNet50 architecture. It was reformed to gain performance and prevent overfitting including a completely connected block and effectively addresses the challenge of vanishing gradients. After testing and training the suggested network attained classification accuracy of 99.48% in OCT images. Also this passes through training on an available OCT image dataset, pre-training on a substantial dataset like ImageNet.

Li et al.[17] demonstrated a classification system that can categorize optical coherence tomography(OCT) pictures into DME, CNV, DRUSEN, and NORMAL and it was mostly based on an enhanced neural network called ResNet50. The authors have taken data from publicly accessible datasets, DHU and UCSD as well as an independent testing dataset have been used to execute the diagnostic performance. Performance was also evaluated using kappa values and AUC. This ensemble technique is also useful for limited images. Qualitative evolution and occlusion testing were also used for predicting models and understanding the decision-making process respectively. Occlusion testing contributed in the identification of pathological areas and misclassifications, and the approach led to classification, sensitivity, and specificity accuracy measurements of 0.973, 0.963 and 0.985 respectively at the B-scan level.

Najeeb et al.[18] showed retinal disorders were automatically identified and categorized and a convolutional neural network was employed for data classification process and an algorithm was created to recognize an area using OCT scan pictures of different individuals. It was a single layer with a low computational cost. An open source dataset of 83,484 retinal OCT pictures was used to train this model. Using this model, a 95.66% overall accuracy was reached. In comparison to other networks, the network itself is exceedingly thin and computationally effective.

Long and Huang[19] discovered an algorithm which works to detect hard exudates (HE) in colorful images of retina. There are few stages like image processing, determination of candidate HE, localization of optic disc, and extraction and classification of texture features. In the first stage, the image is resized, converted to grayscale and denoised. OD is localized using a combination of morphological operations and thresholding in the second stage. Finally, in the third stage, a dynamic threshold is used to segment the image into candidate HE and non-HE regions. The dynamic threshold is determined according to the worldwide threshold and the local statistics of the image. In the final stage, eight texture characteristics are taken out from the candidate HE regions and fed into an SVM classifier to classify the regions as HE or non-HE. The e-ophtha EX database (47 photos) and the DIARETDB1 database (89 images) were used to test the method. After demonstration, the result is an average sensitivity of 76.5%, a positive predictive value (PPV) of 82.7% and a score of 76.7% on the e-ophtha EX database. On the DIARETDB1 database, it achieved an average of 97.5% sensitive value.

Chapter 3

Methodology and WorkPlan

3.1 Research Methodology

The main motive of our study is to identify retinal disease using OCT images from our dataset. In our dataset, we have four features like CNV, DME, NORMAL, DRUSEN. Then the dataset needs to be pre-processed. In the pre-processing part we have used data-augmentation like centering and normalizing each image etc. After that, data is ready to be split. We have created a custom CNN model for the classification. Moreover, we have used some pre-trained CNN models in our research. The models are VGG-16, ResNet50, Inception V3, DenseNet-121, Xception. We compare the results from our custom model and pre-trained models and determine which one performs better. Finally, after implementing the better model our system will be able to categorize the retinal disease. Moreover, Grad-Cam was used for better analysis.

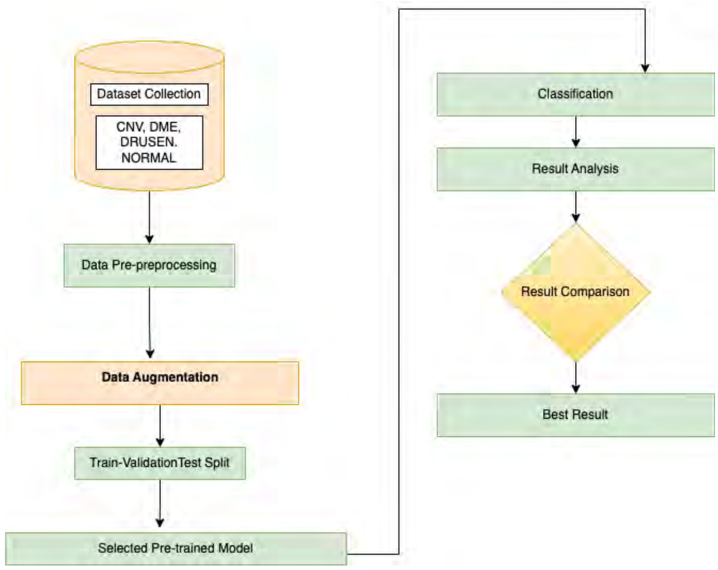

Figure 3.1: Research Methodology

3.2 Dataset

3.2.1 Source

In our research as well as in the proposed model, we have used the publicly available dataset provided by Kermany et al.[20] where four classes are given in the image dataset (CNV, DME, DRUSEN and Normal). Finally, dataset links are cited in this section and referenced properly in the bibliography.

3.2.2 Dataset Description

The main motive is to diagnose and classify eye retinal illnesses using multiple deep learning models. In the beginning, we collected 84,486 OCT photos. Firstly, we distributed 8 images per class (32 images) to validation and 242 images per class (968 images) to the test set. Then the AI system was verified and trained using those 83,486 photos (37,206 with choroidal neovascularization, 8,617 with drusen, 11,349 with diabetic macular edema and 26,315 normal) from 4,686 patients that passed the initial image quality review.

3.2.3 Data Sample

CNV :

This disease has been recognized from histopathologic studies for over 100 years [21]. CNV dynamics have been distinguished by immunohistochemical and specific biologic approaches.

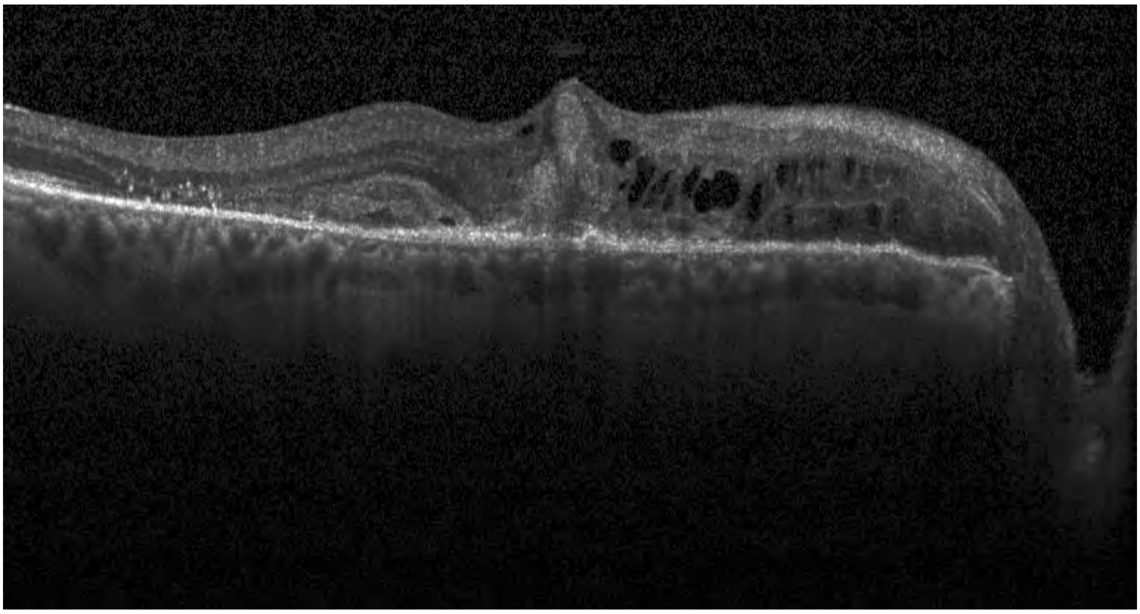

Figure 3.2: CNV

DME :

Diabetic macular edema is one of the major problems of diabetes and it can cause visual deterioration [22].

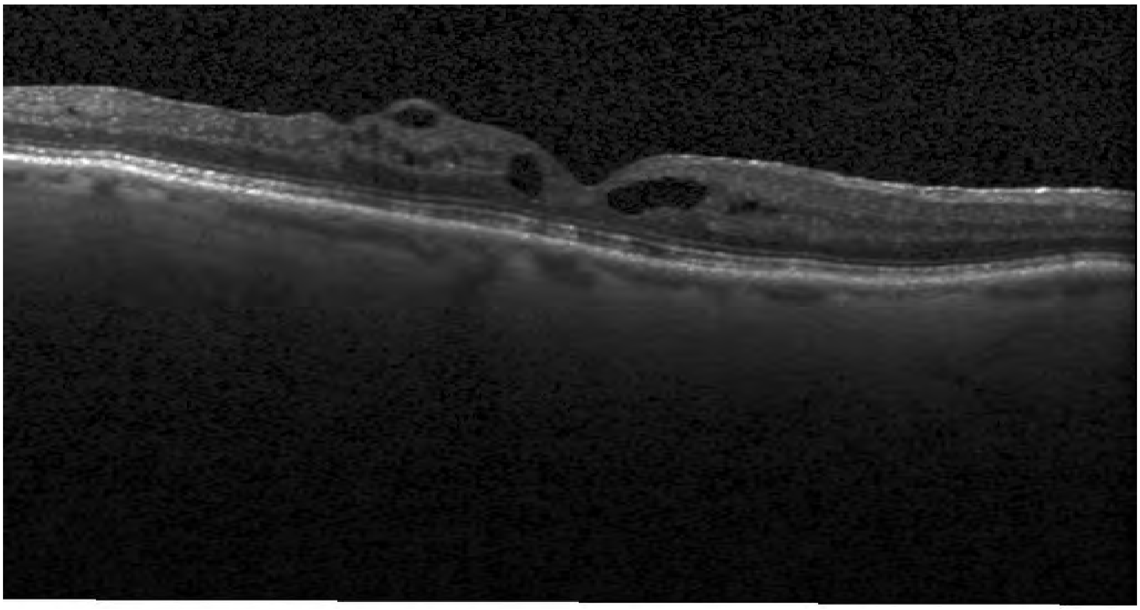

Figure 3.3: DME

DRUSEN :

DRUSEN are yellow layers below the retina. They are invented by lipids and proteins. Drusen can be in many shapes (Small, medium and large).

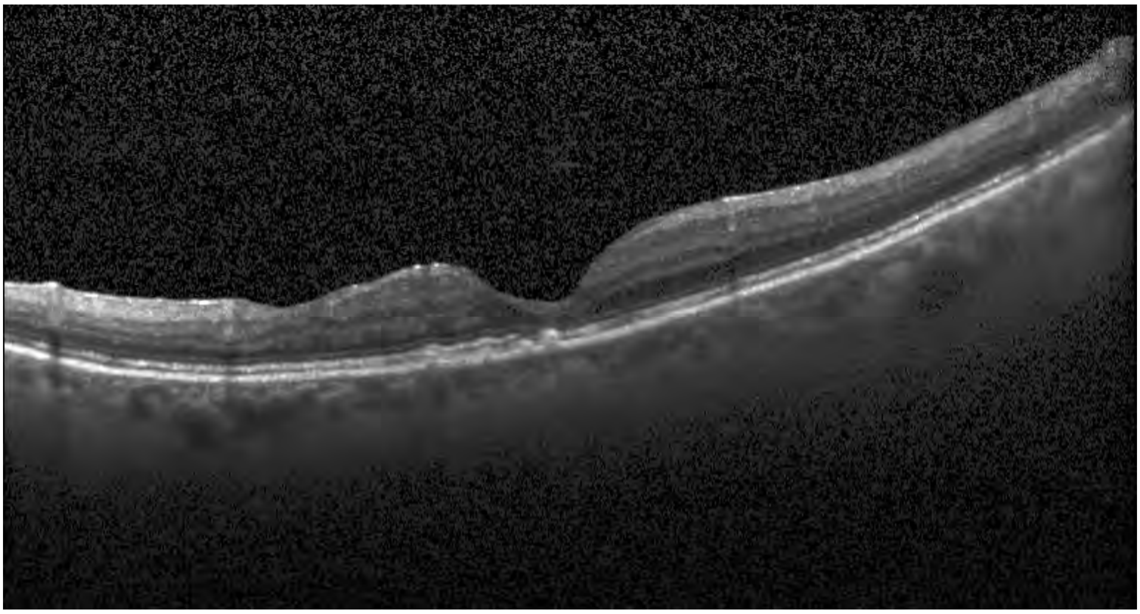

Figure 3.4: DRUSEN

Normal :

Basic normal OCT image without any difficulties or diseases.

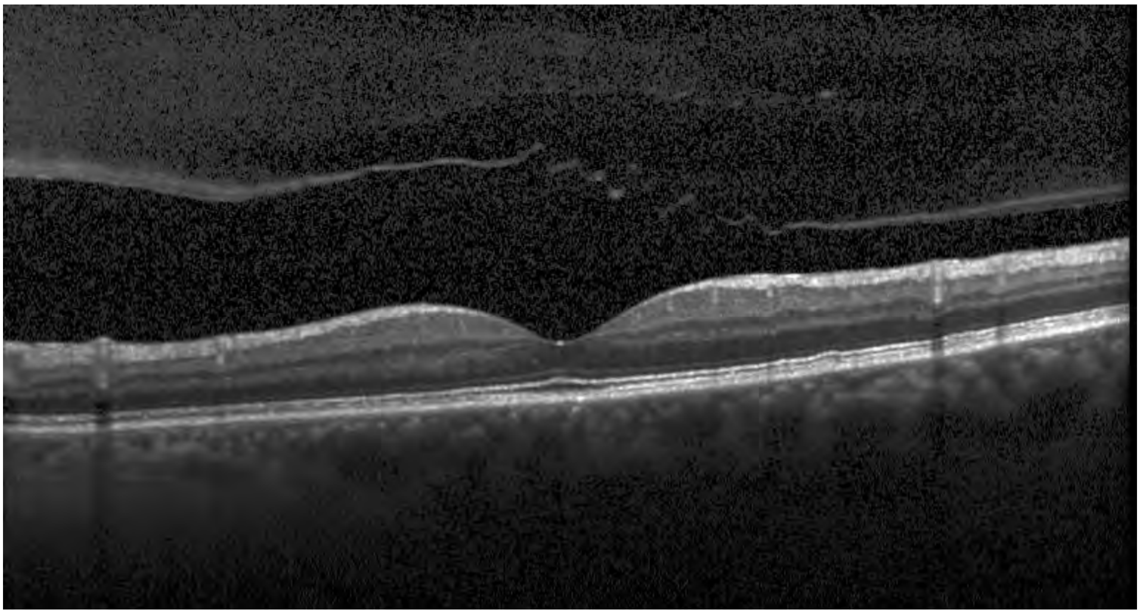

Figure 3.5: Normal

3.3 Data Pre-processing

We have used 20% of training data to validation set and rest are used for training set. In our dataset we have four features, such as CNV, DME, Drusen and Normal. Two directory paths such as "traindir" and "testdir" contains training and testing images. Using "ImageDataGenerator" various augmentation techniques like centering and normalizing each image, horizontal flipping, adjusting height and width shifts, rotation, and zooming. For the pre-processing, image size is set to 224 pixels and the batch size to 64, and then creates two data generators such as traingenerator and validationgenerator. These two generator load images from the "traindir" directory, with the help of validationgenerator which represent a subset (20%) for validation purposes.

3.4 Image Enhancement

The first process is to read and access information or files like images from the path.

3.4.1 Median Blur Filter

The median blur filter is a valuable method for reducing image noise and preserving sharp edges. It is also useful for smoothing an image. This median blur filter helps in image cleaning without missing important details and data. Sometimes it faces difficulties in blurting sharp images and also performs poorly.

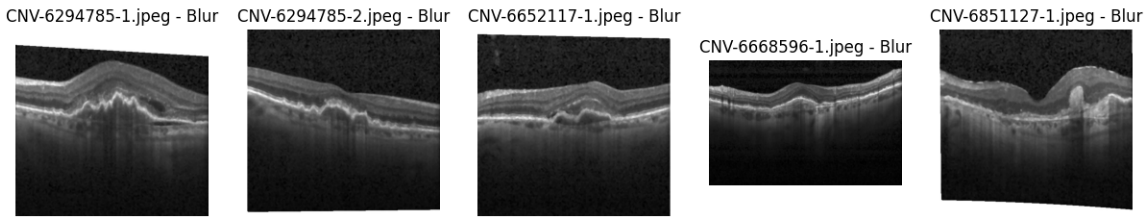

Figure 3.6: Median Blur Filter

3.4.2 Converting to Grayscale

Converting the images to grayscale [23] which is extracted from the RGB image and contains no color information. Moreover, It can help to reduce complexity, and highlights and is also used for improving the image's internal structure edge detection, enhancing pattern and shape. Grayscale conversion is significant for sound and noise reduction and is able to obtain many image processing methods like thresholding, and computational efficiency.

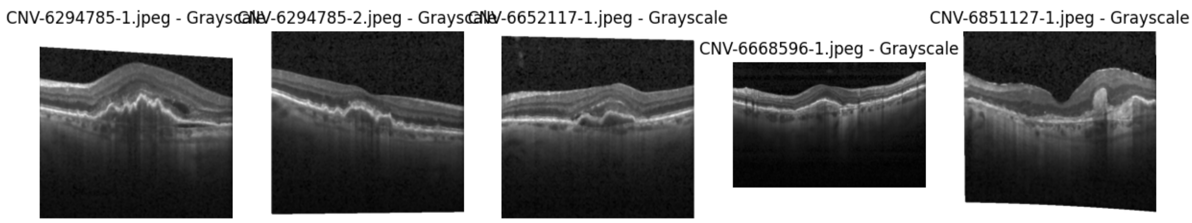

Figure 3.7: GRAY SCALE

3.4.3 CLAHE

CLAHE is an image enhancement method that is applied for image quality upgrades and helps in the visibility of image hidden features by focusing on a picture's different smaller areas known as tiles. It is an advanced version of histogram equalization. It is generally used in various fields of medical imaging, photography, and in computer vision. In OCT images, there are certain parts which are crucial for identifying. CLAHE can help in improving these fine details. Moreover, it can make distinction [24] among layers which can help in better analysis and detection. Using this method helps in better feature extraction and better visibility from the images which is important for identifying in medical images.

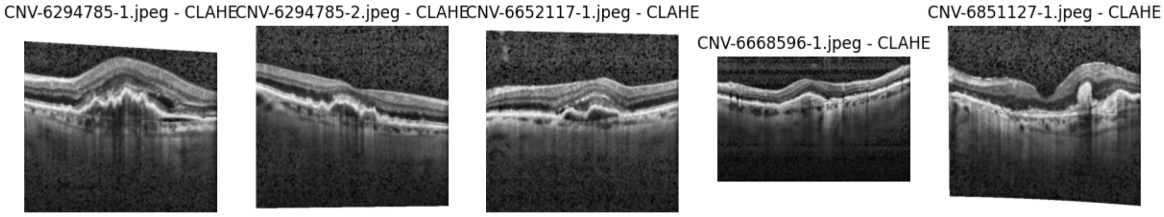

Figure 3.8: Clahe

3.4.4 Image Thresholding

Using image thresholding in practice Images transform grayscale images into binary images in order to distinguish between individual objects. It is a technique where an image's gray values are split into two or more groups. This method is widely used in image segmentation, quality control and pattern recognition.

Figure 3.9: Image Threshold

3.4.5 Morphological Operations

In this step, we have done three operations. The First one is we have created a kernel of 5x5 size. It will be used for the morphological operations we have done in the next two steps. The second step is Morphological Opening which helps in removing noises such as small bright spots from the image. It also detaches objects that are close to each other and break thin connections among them. Following this, we do the next step, which is Morphological Closing. It helps remove the distance among objects in the image. It can also reconnect and detach parts of the objects and thus helps in flattening the contours of the objects. Overall, this process helps in noise removal, connecting, separating, and enhancing the object's shape of the images.

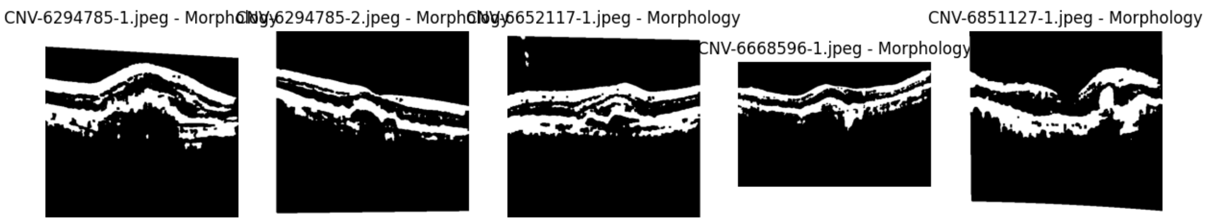

Figure 3.10: Morphology

3.4.6 Extract contours

Contours are curves that connect all continuous points with the same color or intensity along a boundary. In terms of item detection and recognition, and image analysis, contours are a highly helpful tool. In the case of edge detection, it can figure out the boundaries or a region of objects from an image. After the edge detection with the help of contours, we can categorize the shape of an object [25]. Thus it helps in shape analysis. Moreover, it helps in detecting various diseases of medical images and more. We applied contours to measure the thickness of some retinal layers to extract the edges necessary for shape analysis and object detection.

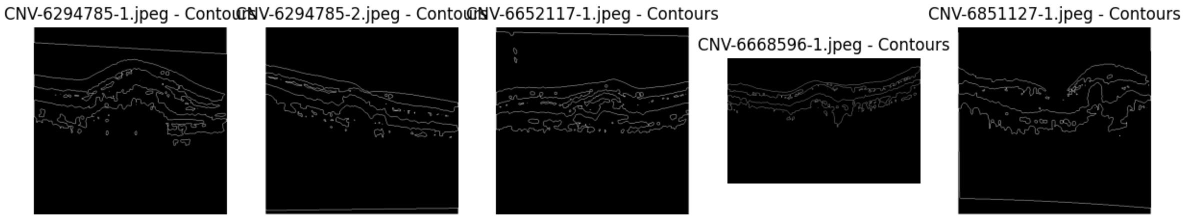

Figure 3.11: Contours

3.4.7 Draw contours

Draw contours is a step after extraction from an unprocessed image and it provides a clear comparison between layers and edges. It helps in visualizing the accuracy of contour exposure.

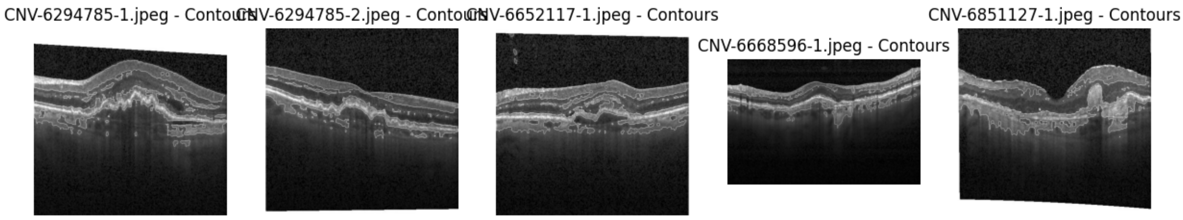

Figure 3.12: Final

3.5 Convolutional Neural Networks

Convolutional Neural Networks (CNN) have taken over the machine learning and computer vision field in the last few years. CNN architecture is formed by an input and output layer also with several hidden layers which are Convolutional, Activation, Pooling, Fully Connected and Normalization Layer. It has been implemented in numerous applications including facial object detection and recognition, image classification, segmentation and superresolution, semantic segmentation, and Natural Language Processing(NLP). Convolutional Neural Networks have been designed with an extremely high computational complication and have been transformed into an effective computer model that has achieved higher level performances and broken all past records for accuracy in every image. CNN [26] uses the sliding windows as filters. In images the filters are very important for identifying the features and patterns. This can include edges, shapes, colors and images which have grid patterns and these filters will figure out these features connected with the image. It detects all the vertical edges that are present in an entire image and the horizontal edge detector filter of it will detect all the horizontal edges. The mathematical operation is called convolution which is a specific linear operation where two functions are multiplied together and then produce a third function. This third function is a modified form of the first function and known as convolution in CNN.

3.5.1 Convolutional Layer

This is the main and first layer and linear operation of CNN. In this layer, the parameters are size and number of filters. Different filters will detect different features, one filter will detect the horizontal edge while one other will detect the vertical edge and another one will detect the circular feature, others are padding and stride. The features from the input images are first extracted and then the convolution mathematical equation with the given (M * M) size between the input image and a filter and the filter's size is calculated by moving over the image. The feature map output presents the image details with its edges, corners, and necessary details. In CNN convolutional layers[27], are very advantageous since they emphasize elements by using a convolution matrix from graphics programming. Moreover, they preserve the pixel spatial relationship. This layer is very important in CNN as they secure and maintain all the pixels spatial connection and significantly reduce the number of parameters required for a convolution layer as all the spatial areas have the same convolution kernel.

Equation, $(n * n * 1) * (f * f * c) = (n - f + 1) * (n - f + 1) * c$ ...(1)

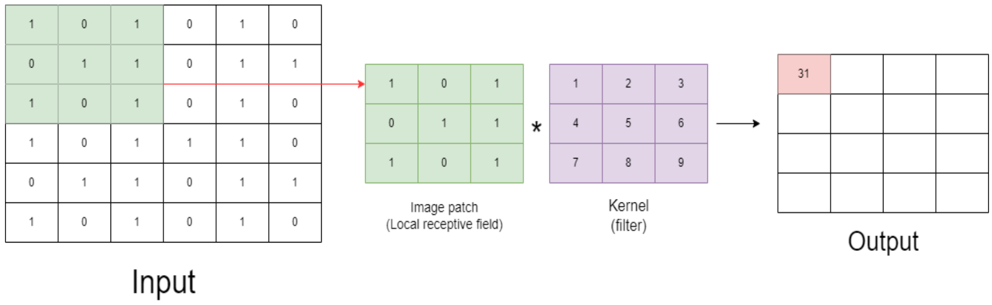

Figure 3.13: Convolutional Layer

3.5.2 Padding

Padding is a process where empty pixels can be added to an image's boundaries because when a convolutional filter is used, padding is used to keep the original size of the image and allow the filter to execute full convolutions on the edge pixels. It refers to the use of extra pixels on input or feature map sides. Through this method, it is ensured that every pixel is considered and captures edge information. For example, if a single layer of padding is added 6×6 the output image will be 8×8 there will be zeros in one pixel added on all the sides.
Feauture Size $= ((Imagesize + 2 * PaddingsizeKernelsize)/Stride) + 1...(2)$

3.5.3 Stride

Strides reduce the sides of the next layer; it also introduces how many numbers are stepping over or convolution filter passing. If stride is equal to 2 then we can directly move by 2 pixels at once and also directly 2 pixels down.

3.6 Max Pooling Layer

The Max pooling layer is most often used in pooling operations. Here, the largest primary information is taken from the feature map and the maximum value is represented within a matrix[28]. It is applied in the last section and it is useful for reducing overfitting and computation, downscaling images, enhancing features and for better generalization. It is operated in images with a dark framework because it will choose pixels that are more enhanced.

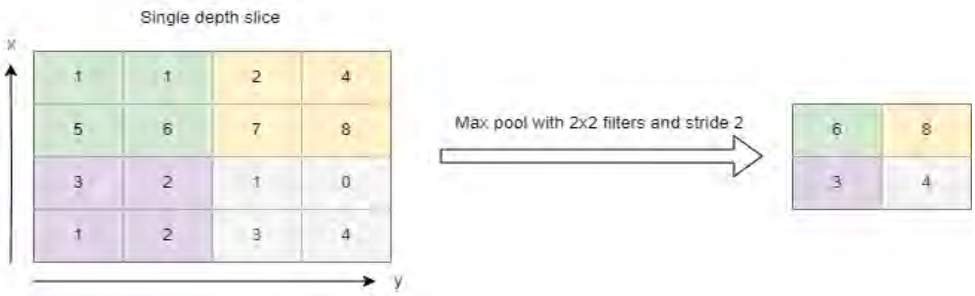

Figure 3.14: Convolutional Layer

MaxPooling, $(X)_{i,j,k} = max(X_{i*s_x:(i+1)*s_x-1,j*s_y:(j+1)*s_y-1,k})....(3)$

3.7 Activation Layer

The activation layer is important to understand how the network is built and to get the nonlinearity in the network and visual diagnostics of CNN. They are not layers and they follow convolution and they are applied in input for learning and predicting complex connections among network parameter types. A main activation function that is popular and used in CNN is called Rectified Linear Unit(ReLU).
Equation (ReLU), $f(x) = max(0, x)$....(4)

3.8 Fully Connected Layer

A fully connected layer can [29] predict given image class from the convolution process output using the features that were obtained in previous steps. The main problem is that it has many parameters that require advanced computing to be used in training examples. For this every potential layer to layer connection is present and each input affects every output in turn.
Fully Connected Layers, $y = Wx + b$...(5)

3.8.1 Flatten

Flatten is the last step and it is linked to the fully connected layer and performed in CNN known as that if any value is greater than 1 dimension then transform it to 1D. To input the data it needs to first be flattened into a 1-dimensional array. The convolution layer's output is flattened because after the process they give a single feature vector which is lengthy.

3.9 Hyperparameter

Hyperparameter is an essential parameter and it consists of the number of layers, neurons needed for layers, learning rate and how many epochs that are required for training.

3.9.1 Optimizer

Optimizer is an algorithm or operation and it is important and used for solving optimization problems and reducing the loss function. This technique is used to modify weights and learning rate in order to minimize the losses of neural networks. The examples of some optimizers which are used generally are Adam, Momentum, Adagrad.

3.9.2 Learning Rate

The learning rate in neural networks decides and controls the updates and learns how much values are adjusted. It is necessary and determined by the amount of the model's weights changing regarding the loss gradient.The learning rate of gradient descent required to set in an ideal value for function. If the learning rate is set high the optimal values will not count and if it is little then it will require many iterations.

3.9.3 Activation Function

Activation Function includes non-linearity to the neural network and converts the network node's input signal into an output signal that is forward to another layer. This method is used to figure the sum of products of inputs and their co weights then produce the layer's output then the layer's output is used as input for the next layer. Types of activation functions are sigmoid function, ReLU and softmax.

3.9.4 Batch Size

The most essential hyperparameter in deep learning is batch size which refers to how many samples are generated in a single forward and backward pass in one iteration through the network. It directly affects the training process's accuracy and computing efficiency.

3.10 Pre-trained model of CNN

3.10.1 VGG16

Convolutional neural network (CNN) model VGG16 has been applied for object and picture categorization purposes. VGG16 is a 16-layer deep neural network, and it uses small 3x3 convolution kernels with a max pooling layer after every two convolution layers. It is feasible to achieve a more precise depiction of data collection attributes during the process of identifying and categorizing images. This system exhibits superior performance when handling challenging background recognition tasks and extensive datasets. The network architecture comprises 13 convolutional layers, complemented by 3 fully connected layers and 5 pooling layers. A medium-sized 3x3 matrix with a moving step of 1 is utilized as the convolution kernel in the 13 convolutional layers that make up the VGG16 network[30]. From 64 in the first layer to 128 to 256 and finally to 512 in the final layer, the number of convolution kernels continuously rose. VGG16 is a powerful tool for image recognition and object detection. It finds applications in diverse fields such as autonomous vehicles, medical imaging, and social media.

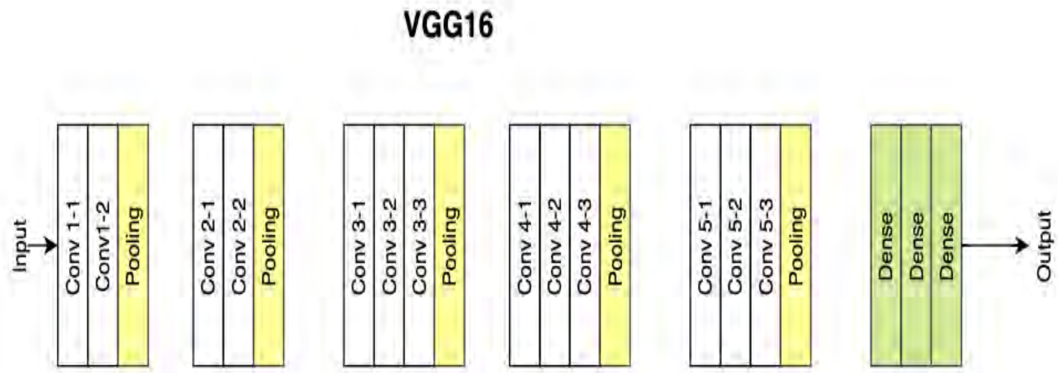

Figure 3.15: VGG 16 architecture

3.10.2 VGG19

The VGG architecture is the foundation for the VGG 19 architecture, which has layers like the SoftMax layer-1, MaxPool layers-5, Fully linked layer-3, and convolution layers-16.The organization that created this network is known as the Visual Geometry group (or VGG for short). With widespread visual recognition in mind, it was developed. The main advantage of this strategy is that everyone can access its source code, which enables us to swiftly deploy transfer learning and modify the network to other designs. Since the system learns complex properties when there are several small-sized kernels present, the approach also collectively learns small-sized kernels rather than learning a single giant kernel[31].

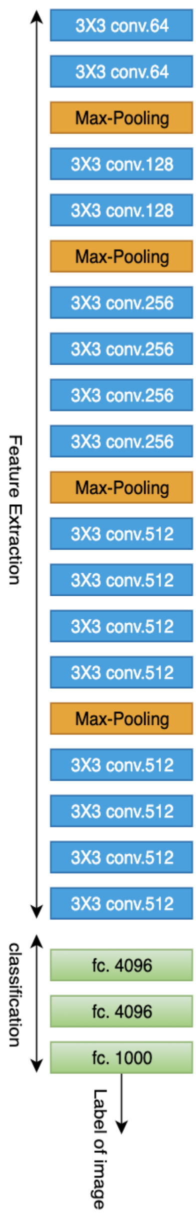

Figure 3.16: VGG 19 architecture

3.10.3 ResNet50

The deep convolutional neural network architecture known as ResNet-50, or Residual Network with 50 layers, is frequently used for a variety of computer vision applications, including image categorization and feature extraction. ResNet-50 stands out for its clever application of residual connections or skip connections[32]. Residual connections make deep networks easier to train by allowing them to learn small changes to existing functions. With the use of identity shortcut connections and many convolutional layers, each residual block allows the network to preserve and propagate gradient information more effectively during training, which mitigates the vanishing gradient problem. ResNet-50, also is a potent feature extractor and transfer learning tool since it includes 50 convolutional layers and was pre-trained on enormous datasets like ImageNet. Its exceptional performance can be attributed to its depth and the inclusion of skip connections, positioning it as a leading choice in the realm of deep learning for computer vision challenges.

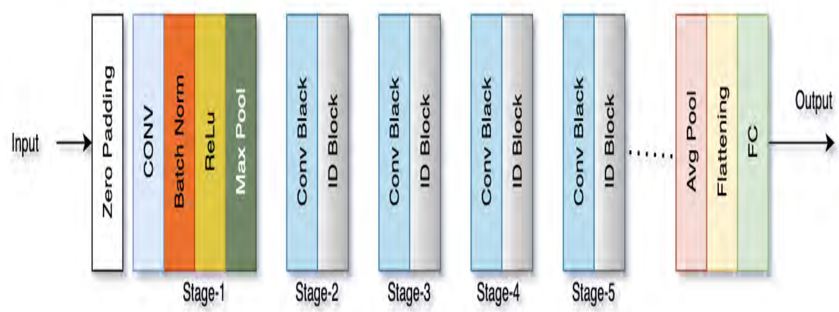

Figure 3.17: ResNet-50 Model architecture

3.10.4 Inception V3

Googlenet designed Inception v3 to help in object detection,enhancing the network with picture review. It is a convolutional neural network by building on the actual architecture of Inception v1, v2. It can consume less computing power from previous Inception architecture versions. Its main components are (299×299). The Inception v3 network [33] structure also introduces a batch normalization layer. Also it divides large volume convolutions into small convolutions using a convolution kernel compared to other structures.

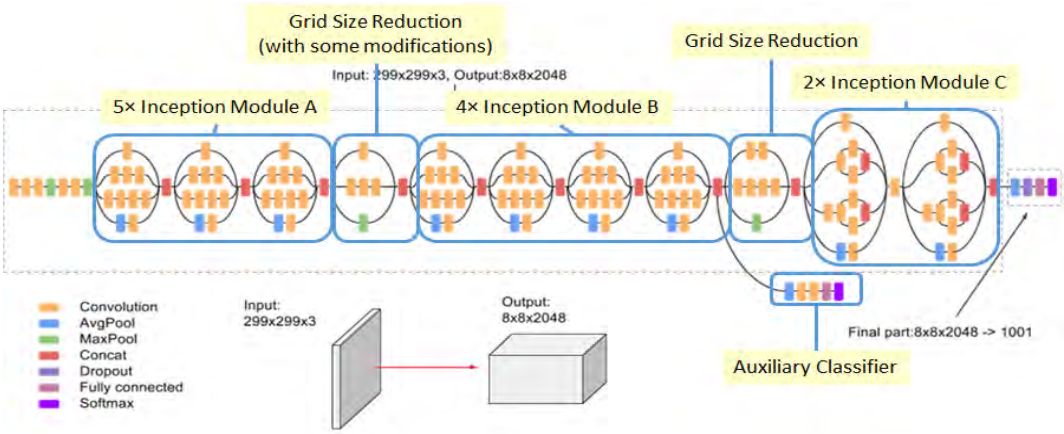

Figure 3.18: Inception V3

3.10.5 DenseNet121

This network design focuses on improving deep learning algorithms in order to increase the productivity of training by using shorter associations between layers. It follows CNN structure and it was built to locate some of the problems of deep learning such as feature reuse and vanishing gradients. In DenseNet121, [34] "121" means, there are in total 121 layers. This architecture uses dense blocks connected with multiple layers. Layers of the blocks reuses the features through the network. This model can be used to detect objects as well as medical field image processing and classification. The bottleneck layers reduce the number of input channels and it also helps in decreasing computational cost. Due to its high efficiency and effectiveness in training deep neural networks, it is really helpful in image classification, object detection and segmentation and nowadays DenseNet121 has become so popular in the deep learning community.

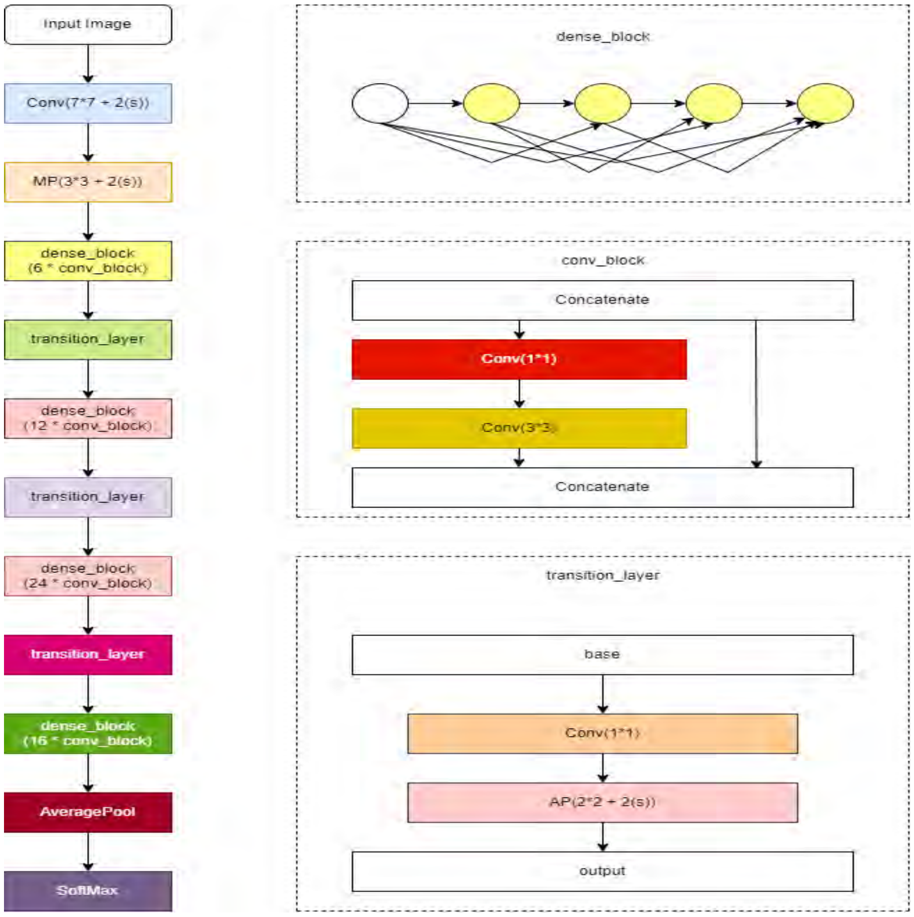

Figure 3.19: DenseNet-121

3.10.6 Xception

Xception is a developed neural network architecture whose main features are depth-wise separable convolutions. Moreover, it outperforms more widely used CNN models like VGG16 in terms of power and overfitting issues[35]. It is a stack containing convolution layers along with residual interactions also an extension of the Inception model [36]. It has advantages like it requires less resources and is good for Image classification performance.

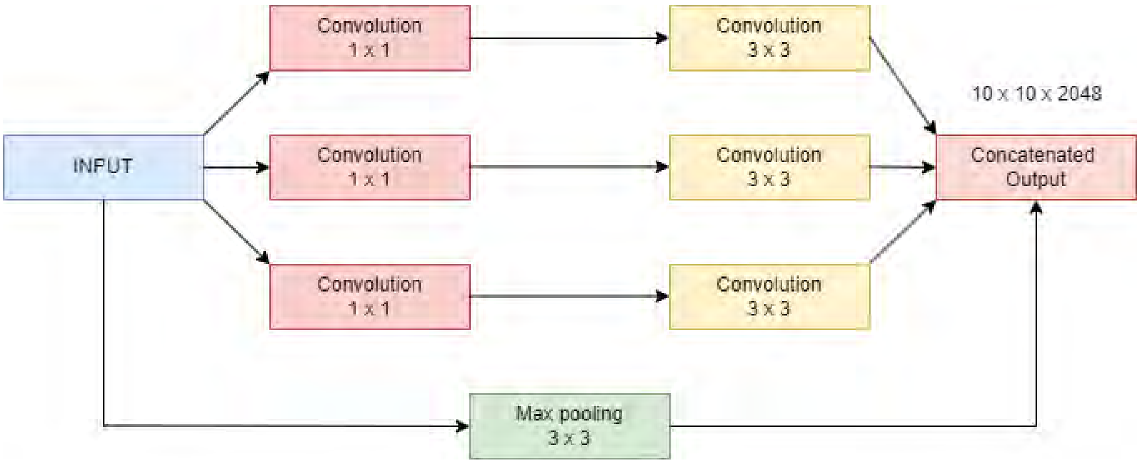

Figure 3.20: Xception

3.11 Workplan

There are three phases in our thesis: Pre-thesis 1, Pre-thesis 2 and defense. In the Pre-thesis 1 part, we read many scholarly articles relevant to our thesis topic. We narrowed down some machine learning models which were common and effective in the articles we read. We used a publicly available dataset [37]. Then we wrote literature review, background, problem statement, abstract and introduction. In the Pre-thesis 2, we applied some models in our dataset which were VGG16, VGG19, ResNet50, InceptionV3, DenseNet121 and MobileNet. We wrote our research objective, methodology and conclusion in that phase. Data pre-processing was also included. Finally, in the defence, we will concentrate to increase the efficiency of the model we used and write our final paper successfully.

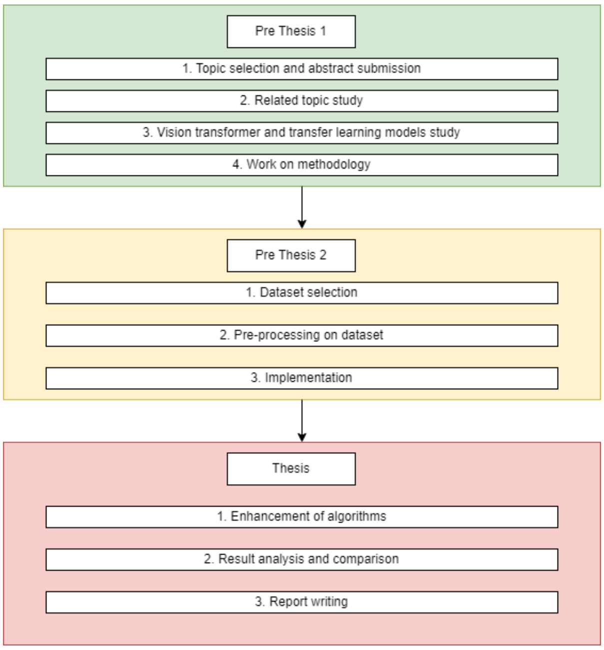

Figure 3.21: Work-flow

Chapter 4

Implementation

4.1 Applied Different DeepLearning Algorithm

Deep Learning is a machine learning approach that learns from trained models how to perform classification. In this research we have used Transfer Learning Models to solve image classification problems. It is mainly important for fine tuning pretrained models,improve the learning process and neural network performance and can train with much less data and time. Transfer learning model is significantly used in ImageNet which is a image database, it helps to use ImageNet's weight precisely and then the weights are applied in newly Dense Layer. Then the newly Dense Layer is applied on the Dataset and after training and extracting complex images there are chances for giving better results. We have used the selected architectures VGG-16, ResNet50, Inception V3, DenseNet-121 and transfer learning techniques were applied. We used this to see how the pre trained models are working on the dataset and for the comparison between custom model and pre-trained model results.
We have built a custom Model and it is trained on the existing dataset. The custom model is designed from scratch for getting better performance and the main motive for creating novelty in our Research. Furthermore, we can get better accuracy, less parameters will beneed and it can be lightweight in a custom model.

4.2 Train Test Split

We have used 83,484 images in total during the test from the dataset. We are able to split these images into 4 distinct classes and then each class contains 1000 images. So, these four classes have been able to create a test set of 4000 images which we used for our model. We utilized the remaining 79,484 images and divided them in 80:20 ratio for the purpose of train and validate our model. We used 80% data for the training set and the remaining 20 % data are known as the validation set.

4.3 Data Augmentation

Deep learning neural networks depend on large datasets to abstain from overfitting. Many medical images can not collaborate with large datasets sometimes. In that case, data augmentation helps to work on limited datasets. It improves the size

and feature of the training dataset to get finer results[38]. Deep learning neural networks need big training data to achieve high performance. Over the past years, these networks were frequently used for image detection and classification. These networks have shown better results in object detection and image recognition and classification. If CNN models train on a small dataset, it may not provide good outputs in test and validation and that is how overfitting occurs. Data augmentation is an effective technique for avoiding this problem[39].
Besides the small dataset problem, there is another problem named uneven class balance. It can also be solved by data augmentation techniques. There are many augmentation techniques such as cropping, zooming, rotating, histogram based methods, style transfers, generative adversarial networks etc. Style transfer method is really helpful for medical data analysis like histopathological images, breast magnetic resonance imaging (MRI) scans analysis and skin melanoma diagnosis[40]. In the augmentation part, we have used the ImageDataGenerator class from Keras which is a high-level neural networks API. We have set the rotation range to 10 which basically rotates the image (degrees, 0 to 10). Also the shift range of width and height help to translate the image vertically and horizontally. Finally shear range and zoom range are adjusted to 0.1 for both cases. There is another term called fill mode = 'nearest' refers to nearest-neighbour filling. We have reserved 20% images for validation.

4.4 Image Input Size

We are working and using the weight of ImageNet. On the ImageNet dataset after training the models the standard default image size is 224x224 without using transfer learning model. We have also used the same image size 224x224 in our custom model.

4.5 Proposed Model

Our proposed model is a 10 layer Convolutional Neural Network (CNN). It starts with an input for 224x224 pixels with three input channels. The network architecture includes four convolutional blocks. Here in each block, convolutional layers are added with the ReLU activation. Moreover, 1 pooling layer is added after each Conv block. These convolutional layers, with varying numbers of filters and a 3x3 kernel size, are essential for feature extraction from images. The max pooling layers following them reduce the spatial dimensions, aiding in computational efficiency. The network then feeds to dense layers by a flattening layer which actually reshapes the 2D feature maps into a 1 dimension vector. This vector feeds into a series of fully connected layers, each with 512 neurons and ReLU activation. Finally, the output has 4 neurons with softmax activation for the classification.

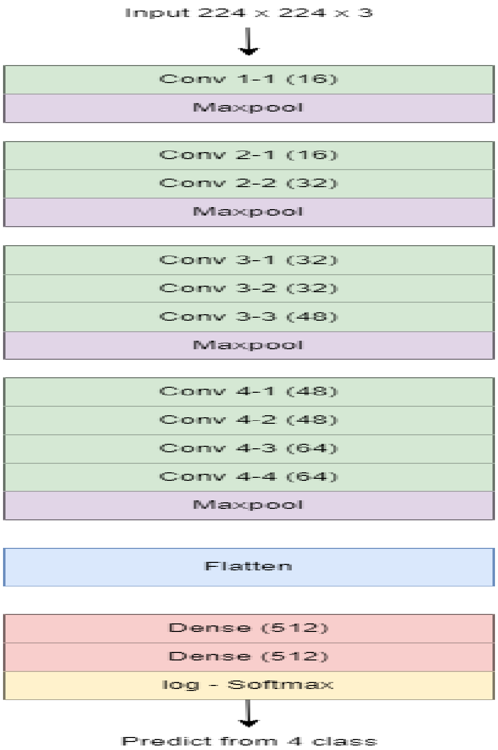

Figure 4.1: 10 Layer CNN Architecture

4.6 Proposed Model with Different Parameters

- **Learning Rate :** We have tried various learning rates like 0.01, 0.1, 0.0001, 0.001 and we have found that our custom model gives a better result with 0.001 learning rate.

- **Batch Size :** We have tried with 3 batch sizes that are 32, 64 and 96. Got better results with batch size 64.

- **Number of Epochs:** We have selected 50 epochs for better analysis. We didn't set the step per epoch number. Rather it was auto calculated by the model.

- **Optimizer Type:** Adam was selected as the optimizer for our custom model.

- **Loss Function:**'Categorical_Crossentropy' was used as a loss function.

- **Activation Functions :** We have used ReLU as our activation function.

- **Number of Layers and Neurons:**There are 4 Conv blocks each with 1,2,3,4 layers respectively. So the total number of Conv layers = (1+2+3+4) = 10 layers. Again, after each Conv block, 1 pooling layer is added. Moreover there are 2 Dense layers with neurons 512 and 512 respectively and 1 output layer with 4 neutrons. Furthermore there is an input layer and output layer. And total number of neurons = 512 + 512 + 4 = 1028 neurons.

- **Callback Function :** We have used ReduceLROnPlateau for monitoring the validation loss and check if there is no improvement, then adjust the learning rate according to it. The new learning rate will be 20% of the previous one and its minimum will be 1eˆ-6. We have set patience to 5, that is if for 5 consecutive epochs, the validation epoch does not improve, the rate will change and it will be reduced.

Chapter 5

Result Analysis

5.1 Experimental Setup

In this Findings, we experiment on a few models such as VGG16, DenseNet121, ResNet50, InceptionV3, Xception and our proposed model. We were able to take advantage of the power of transfer learning. These models offer a solid framework for our study into retinal image classification and have been often used for image classification applications. The NVIDIA RTX 3060 Ti and GeForce GTX 1650 are GPUs that were used in the experimental setup on two separate PCs. These GPUs allow models to learn the features efficiently. Because of its shown efficiency in feature extraction, the DenseNet121 architecture was chosen as an illustration. The model was altered to meet the demands of our mission for classifying retinal images. The final layer is responsible for the prediction. A new dense layer was added in place of the top level. Due to this adaption, the model was able to pick up on discriminative traits unique to our dataset. Already trained on ImageNet training weights used to set up the model. A number of carefully chosen params were used for all the models. There were 50 training epochs for each of the models, a 0.001 learning rate. To get the best results, these tune were iteratively adjusted throughout the model training process. Validation accuracy and loss were combined to monitor the model's performance throughout training. The best-performing model was preserved for later evaluation because model checkpoints were saved depending on the greatest validation accuracy attained. We were able to compare the capacities of several pre-trained models with our proposed model for classifying retinal diseases.

5.2 Evaluation Matrices

5.2.1 Confusion Matrix

An overview of predictions made for a classification task is called a confusion matrix, it helps in analyzing and is required to diagnose details of a model's performance and also helps us to figure out the correct answer of a model for different classes including the errors. This can be used to determine the True positive (TP), True negative (TN), false positive (FP) and false negative (FN) values. The column in the confusion matrix represents a particular of that predicted class and the row represents the details of the actual class. Machine learning classifiers like SVM, Decision Trees, Naive Bayes etc generally computes a confusion matrix to generate a cross tabulation of the measured (true) and predicted (model)classes. It determines how these algorithms are performing. For fine-tuning a multiclass classifier the confusion matrix is necessary because usually it clears any confusion whether the classifier is working properly or not as expected.

5.2.2 Precision

The precision known as metric in a model describes how many items are identified and truly relevant. With Precision we can understand how well the model's classes are precisely guessed and predicted.
Precision $= \frac{TP}{TP+FP}$ (1)

5.2.3 Accuracy

Accuracy essentially tells how many answers are correct out of all the assumptions and notably this is without any regard for whether the predictions were positive or negative. In a method accuracy is given by the number of true positives and true negatives over the entire prediction set.
Accuracy $= \frac{TN+TP}{TP+FP+TN+FN}$ (2)

5.2.4 F1-Score

F1-Score considers into account both recall and precision performance metrics and it is an average between them. A model will have a high F1 score if it performs well in predicting both.
F1-Score $= \frac{2*TP}{FP+FN+(2*TP)}$ (3)

5.2.5 Recall

Recall is a measure of how many applicable elements were detected and classified in a model.
Recall $= \frac{TP}{TP+FN}$ (4)

5.3 10 layer model (learning rate = 0.0001, batch size = 64)

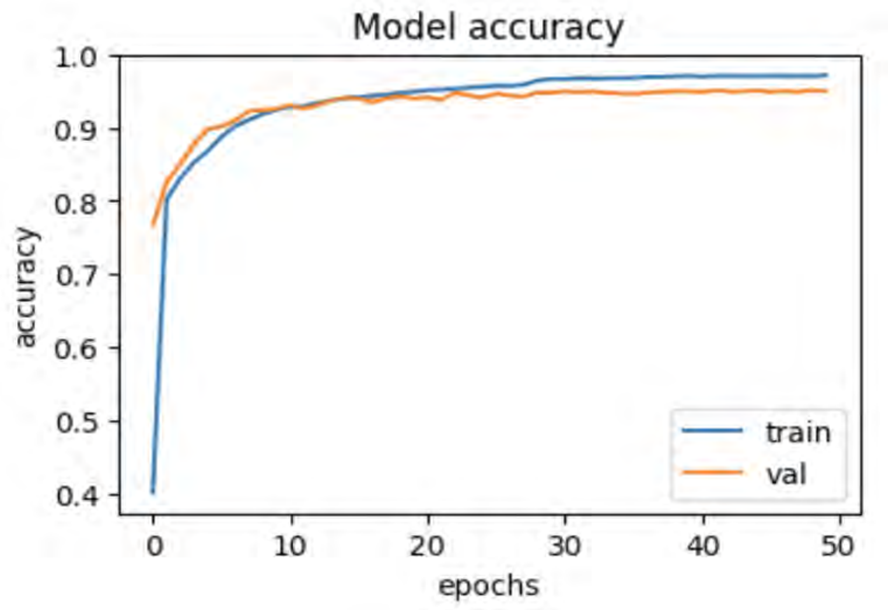

Figure 5.1: 10 layer training and validation accuracy

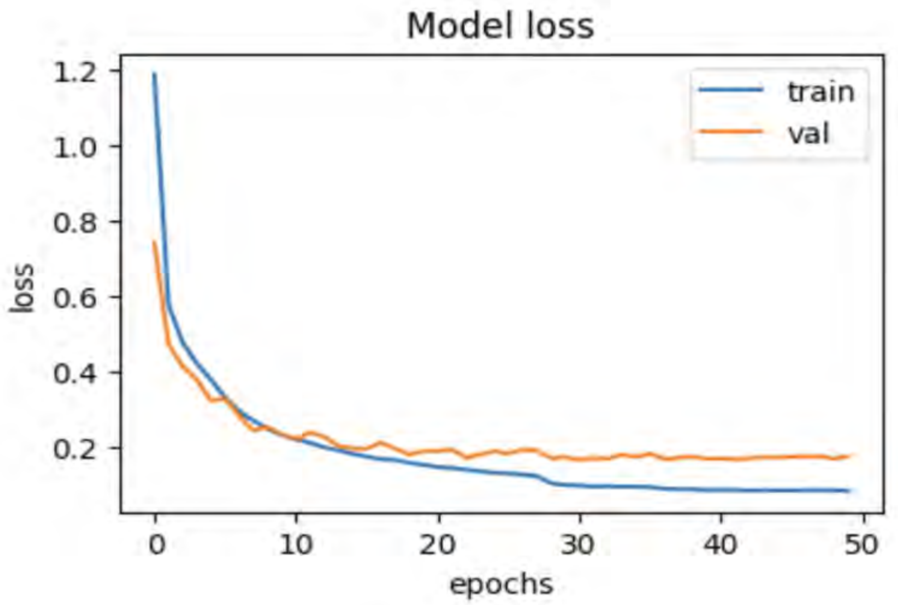

Figure 5.2: 10 layer model Training and validation loss

	precision	recall	f1-score	support
CNV	0.88	0.98	0.92	1000
DME	0.99	0.85	0.92	1000
Drusen	0.95	0.85	0.90	1000
Normal	0.86	0.98	0.92	1000
accuracy			0.91	4000
macro avg	0.92	0.91	0.91	4000
weighted avg	0.92	0.91	0.91	4000

Table 5.1: Classification Report

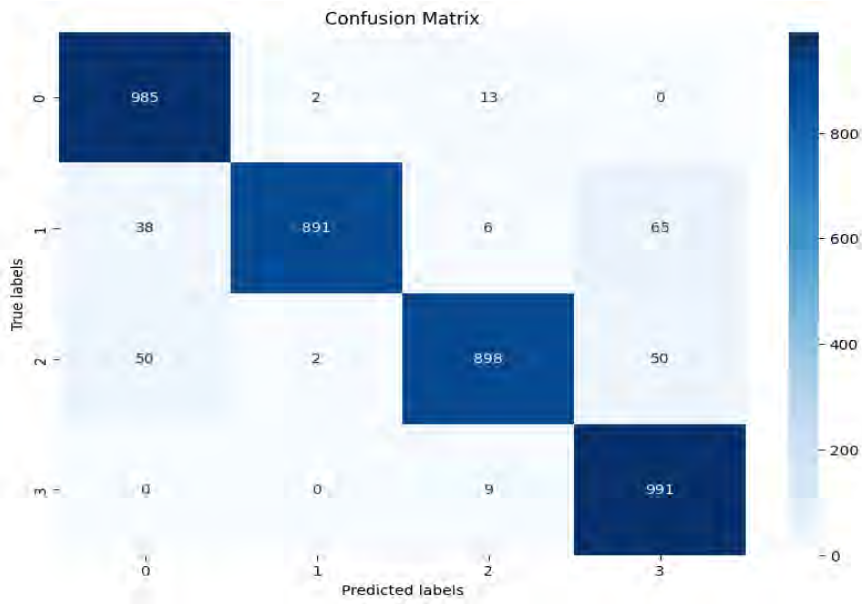

Figure 5.3: Confusion Matrix

5.4 10 layer model (learning rate = 0.001, batch size = 32)

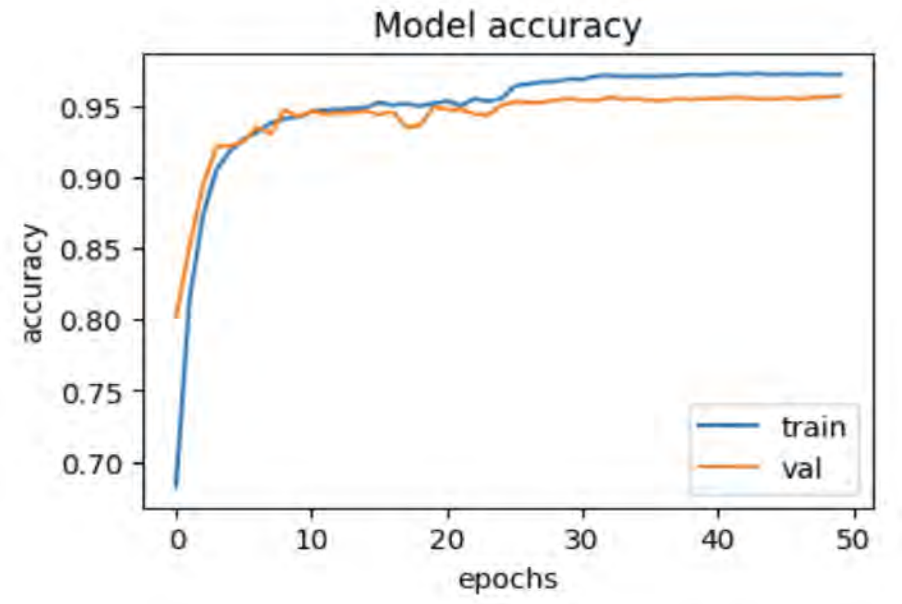

Figure 5.4: 10 layer training and validation accuracy

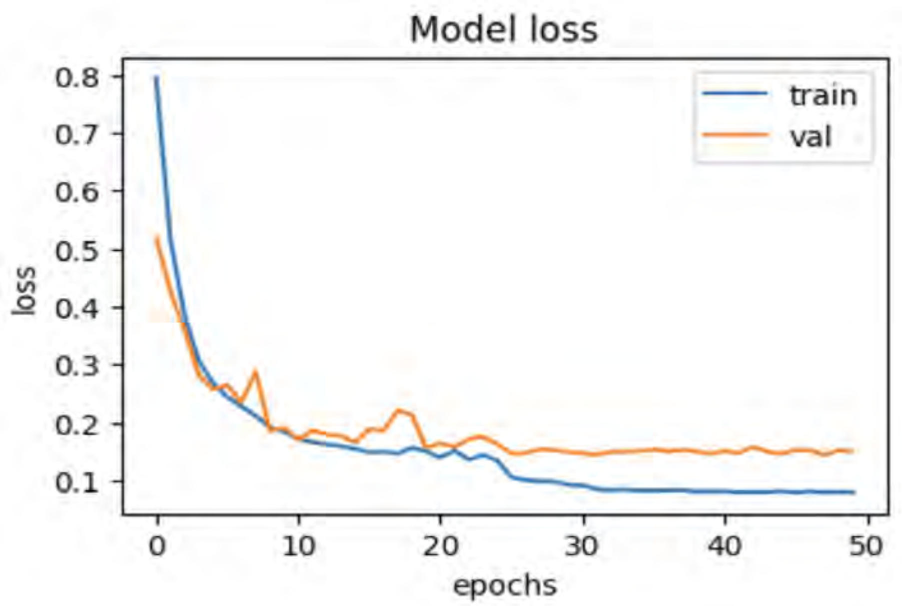

Figure 5.5: 10 layer model Training and validation loss

	precision	recall	f1-score	support
CNV	0.91	0.98	0.94	1000
DME	0.99	0.91	0.94	1000
Drusen	0.97	0.88	0.92	1000
Normal	0.90	0.98	0.94	1000
accuracy			0.94	4000
macro avg	0.94	0.94	0.94	4000
weighted avg	0.94	0.94	0.94	4000

Table 5.2: Classification Report

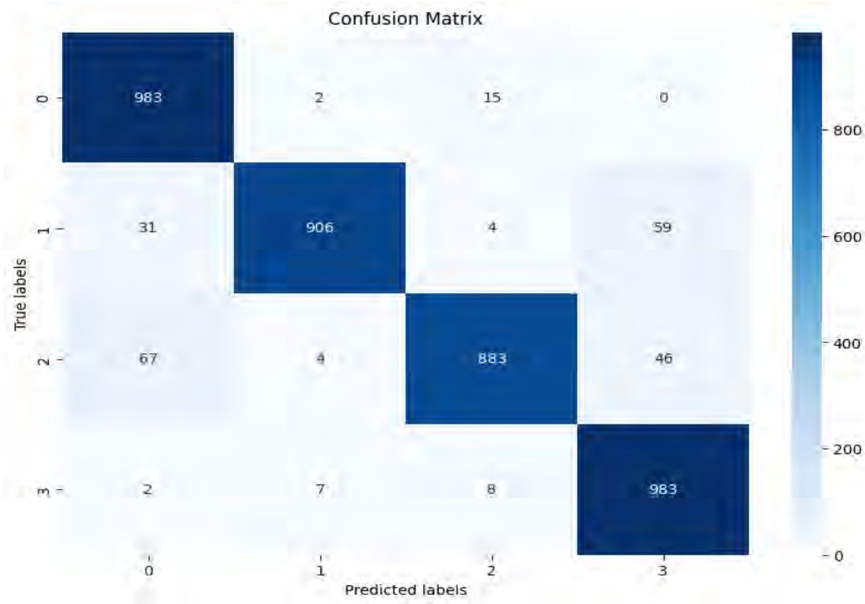

Figure 5.6: Confusion Matrix

5.5 10 layer model (learning rate = 0.001, batch size = 64)

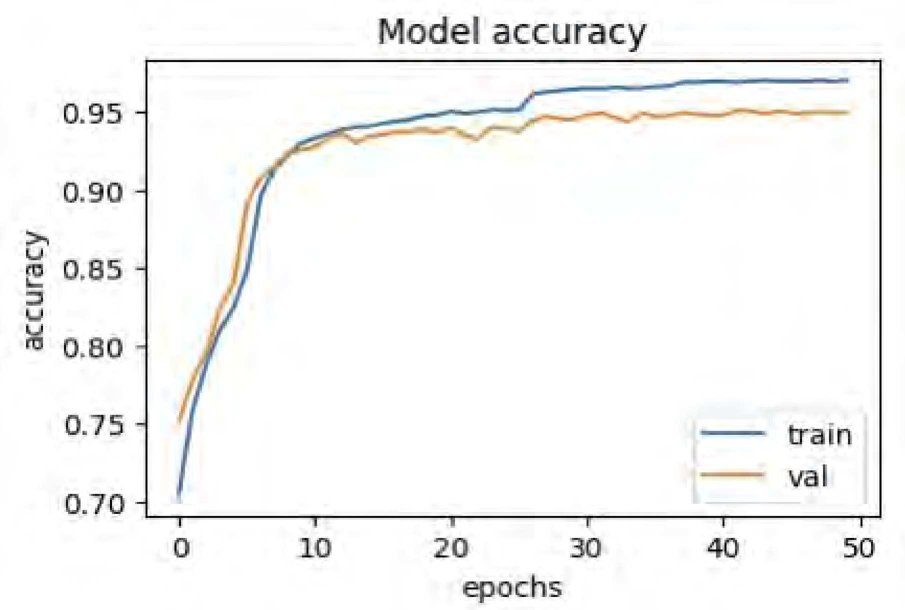

Figure 5.7: 10 layer model Training and validation accuracy

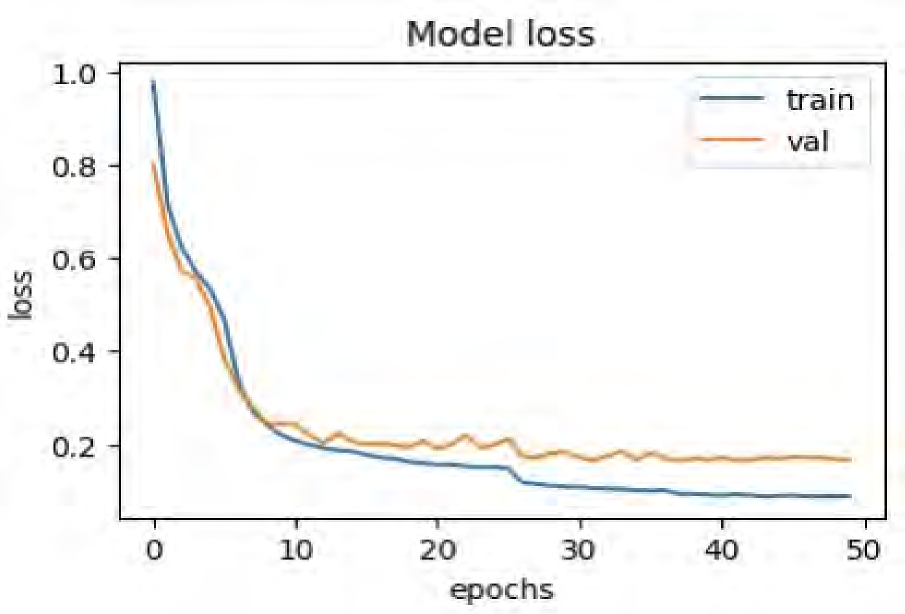

Figure 5.8: 10 layer model Training and validation loss

	precision	recall	f1-score	support
CNV	0.91	0.99	0.95	1000
DME	0.99	0.96	0.98	1000
Drusen	0.99	0.85	0.92	1000
Normal	0.92	0.99	0.96	1000
accuracy			0.95	4000
macro avg	0.95	0.95	0.95	4000
weighted avg	0.95	0.95	0.95	4000

Table 5.3: Classification Report

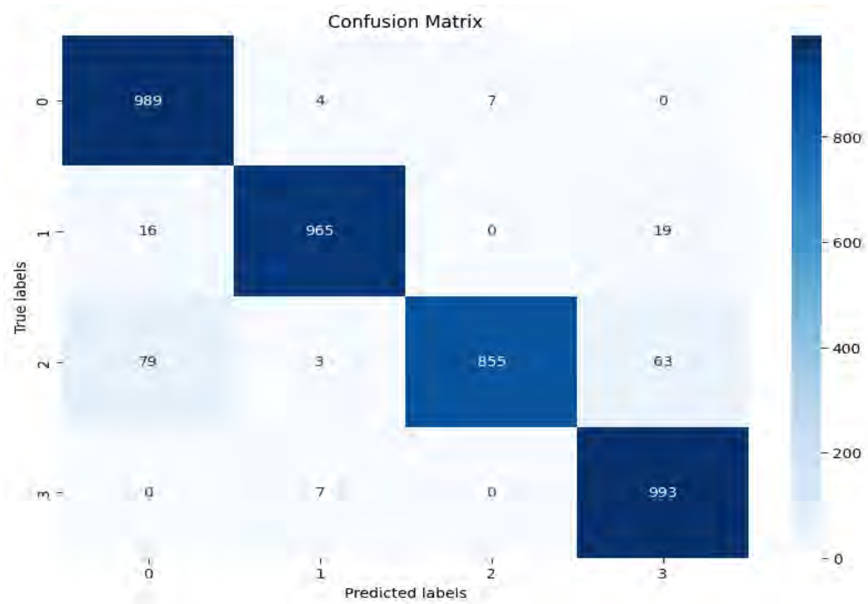

Figure 5.9: Confusion Matrix

Both training and validation accuracy shows an upward trend here.Training accuracy is a bit higher than validation accuracy. There is a steady growth in training accuracy till 26 epochs, after that it reaches a significant value and increases consistently. On the other hand, validation accuracy increases constantly. Since, model accuracy is getting higher in upcoming epochs. So, for both training and validation loss follows a downward trend in respect of time. CNV and Normal both class have greater recall value than DME and Drusen class, so it sympathize that CNV and Normal class are well predicted. Custom 10 layer model successfully identified each class CNV, DME, Drusen and Normal though true positive value of Drusen is lower than all other classes.

5.6 10 layer model (learning rate = 0.001, batch size = 64, image enhancement)

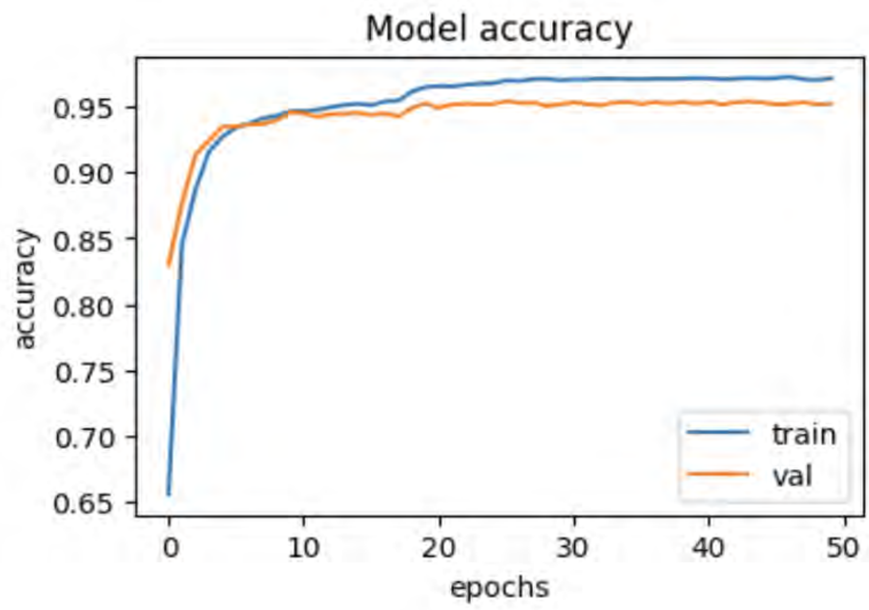

Figure 5.10: 10 layer training and validation accuracy

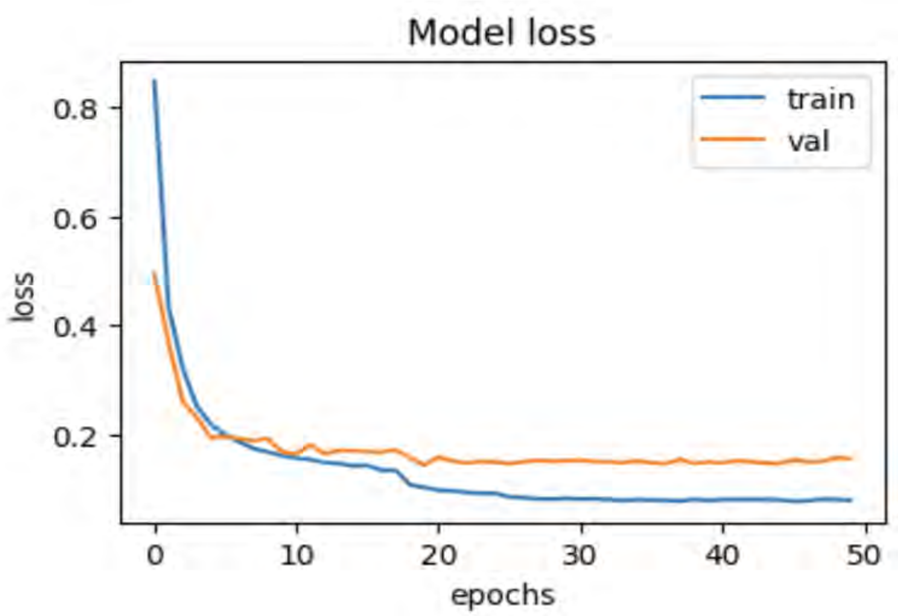

Figure 5.11: 10 layer model Training and validation loss

	precision	recall	f1-score	support
CNV	0.93	0.99	0.96	1000
DME	0.99	0.91	0.95	1000
Drusen	0.98	0.92	0.95	1000
Normal	0.91	0.99	0.95	1000
accuracy			0.95	4000
macro avg	0.95	0.95	0.95	4000
weighted avg	0.95	0.95	0.95	4000

Table 5.4: Classification Report

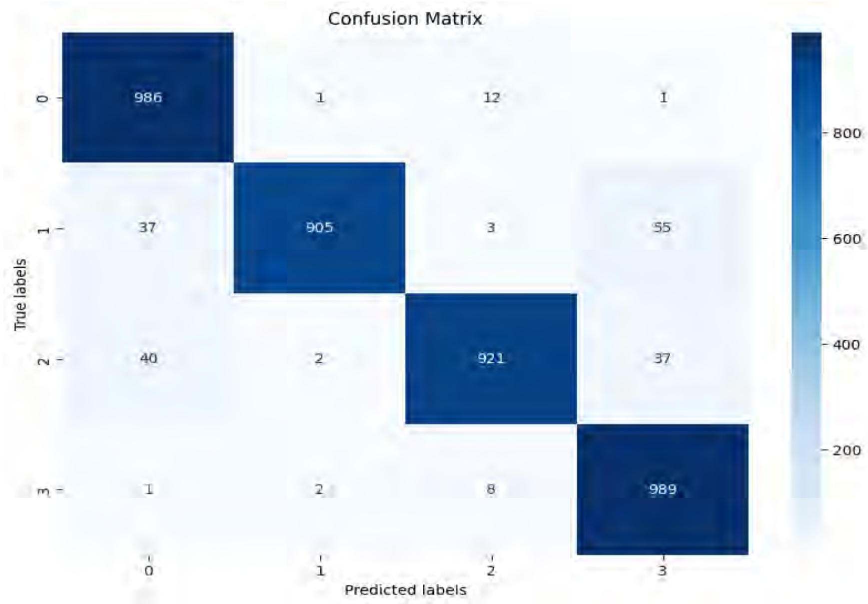

Figure 5.12: Confusion Matrix

Training accuracy initially started increasing after epochs reached at 5 then it became accuracy of 0.9326 and whereas validation accuracy is 0.9350. Then both accuracy ensure an upward trend till last epochs. Training accuracy is higher than validation accuracy and at the end training accuracy gets 0.9758 and validation is 0.9534. For the higher training accuracy, training loss decreases significantly and validation loss falls till 4 epochs. After that validation loss is maintained with consistency with some fluctuations. DME, Drusen and CNV, Normal has precision of 0.97 and 0.92. CNV and Normal ensures a higher recall so that it has a good number of instances. All the four classes such as CNV, DME, Drusen and Normal have classified well while making predictions.
The optimal parameters we choose are lr 0.001 with ReduceLROnPlateau, batch size 64, two dense layers with 512 neurons and 50 epochs for the rest of the comparison. As of this moment, we were not able to get an acknowledgement by the doctor about using this Image enhancement data. That's why we are just comparing the results of it with the regular dataset using the same parameters. Our main focus is not on using enhanced dataset at the moment, rather visualizing the performance of our custom model on both the dataset. Our further works were done on the regular dataset.

5.7 VGG-16

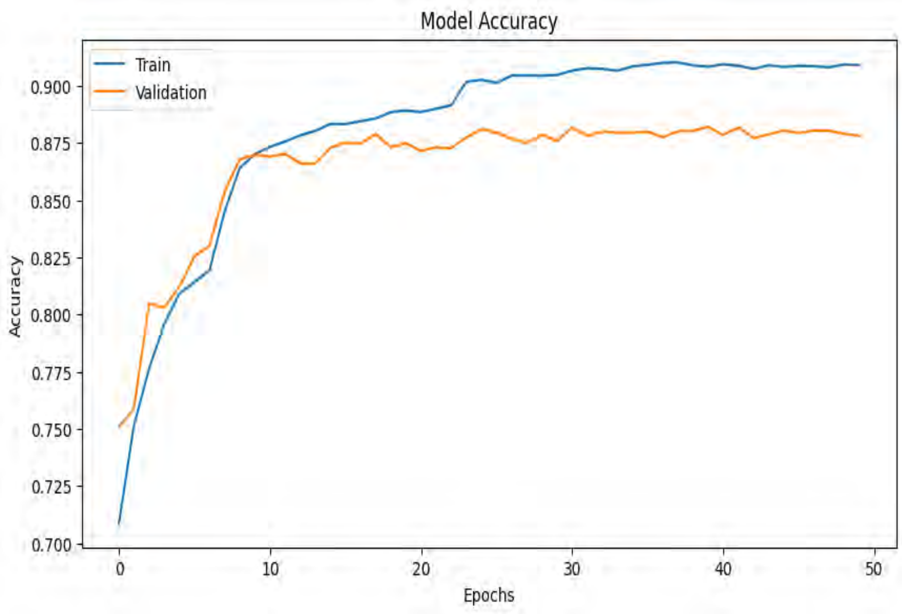

Figure 5.13: Training and validation accuracy

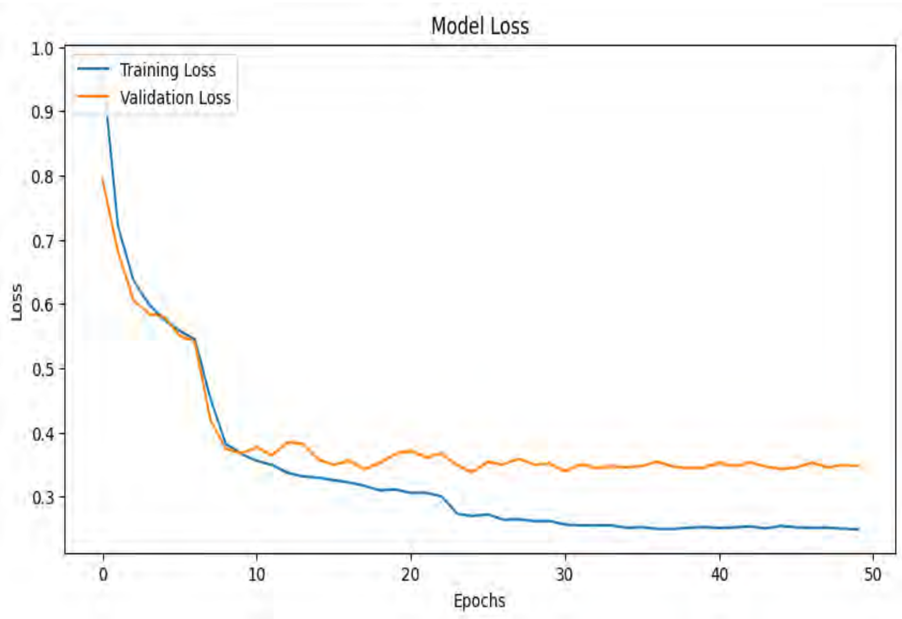

Figure 5.14: Training and validation loss

	precision	recall	f1-score	support
CNV	0.81	0.91	0.86	1000
DME	0.87	0.87	0.87	1000
Drusen	0.90	0.51	0.65	1000
Normal	0.73	0.95	0.83	1000
accuracy			0.81	4000
macro avg	0.83	0.81	0.80	4000
weighted avg	0.83	0.81	0.80	4000

Table 5.5: Classification Report

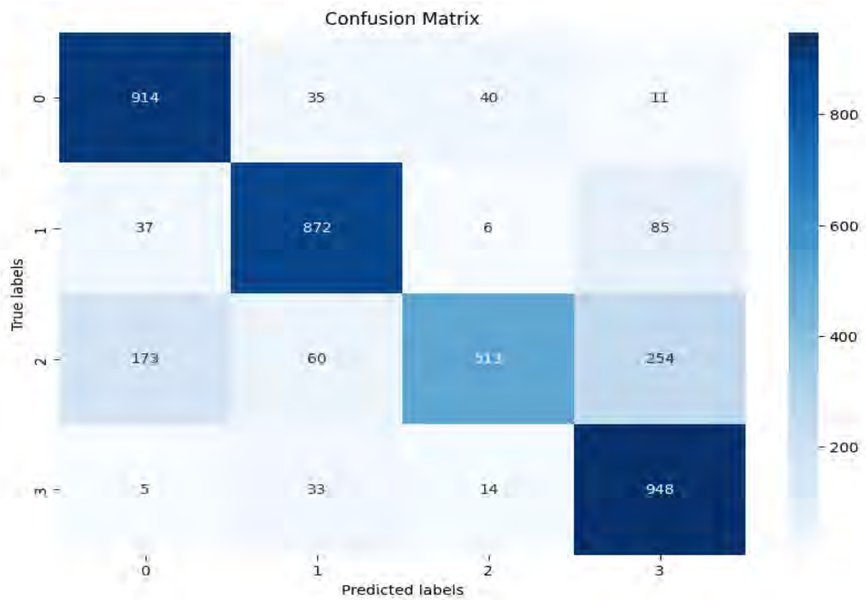

Figure 5.15: Confusion Matrix

In the accuracy graph, we can see that the training accuracy is gradually increasing up to around 0.90. Whereas, the validation accuracy was also increasing in the first 10 to 15 epochs but it started to drop in the following epochs and in the later epochs, the validation graph became constant. Now, there is a gap visible between the training and validation graph. It suggests that there is overfitting, that is the model is not performing well on the validation set. In the loss graph, there is also a gap between the training loss and validation loss. After 10 epochs, the loss graph of validation did not decrease rather became constant whereas the training loss decreased gradually over time. The recall value of CNV and Normal is above 0.90 but the recall value of DRUSEN is 0.51 which is very poor compared to others. It can be said that the model could not handle the imbalanced dataset.

5.8 Inception v3

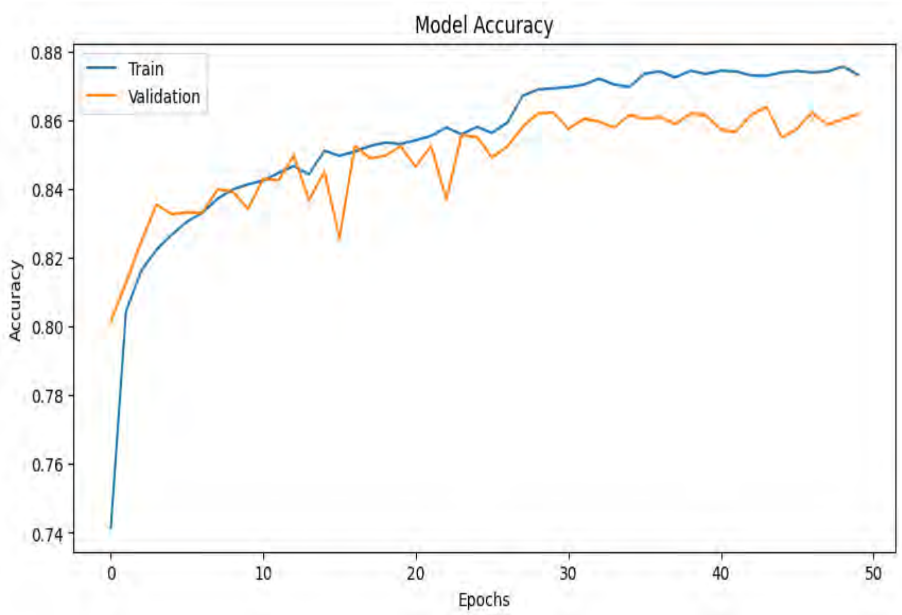

Figure 5.16: Training and validation accuracy

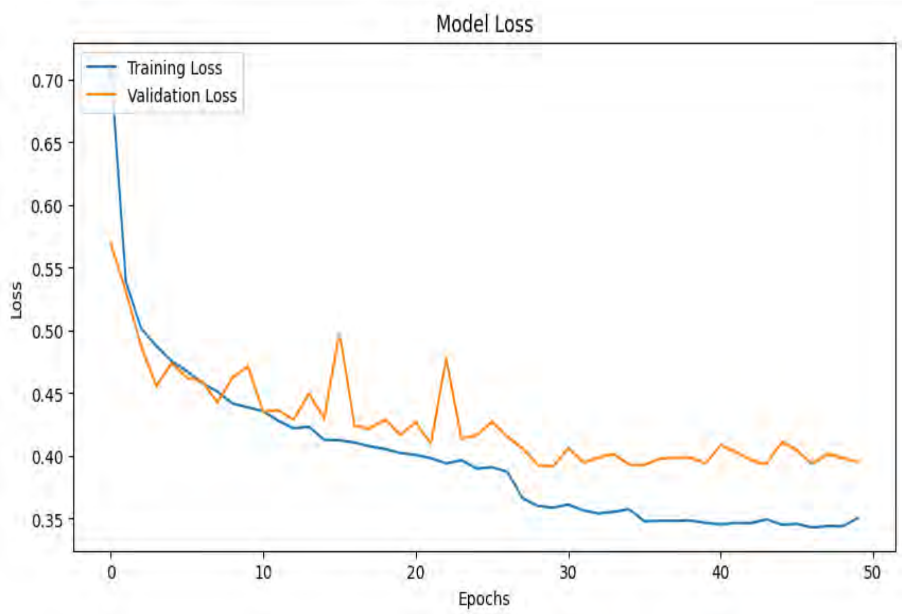

Figure 5.17: Training and validation loss

	precision	recall	f1-score	support
CNV	0.70	0.94	0.80	1000
DME	0.87	0.77	0.82	1000
Drusen	0.89	0.34	0.50	1000
Normal	0.69	0.96	0.80	1000
accuracy			0.75	4000
macro avg	0.79	0.75	0.73	4000
weighted avg	0.79	0.75	0.73	4000

Table 5.6: Classification Report

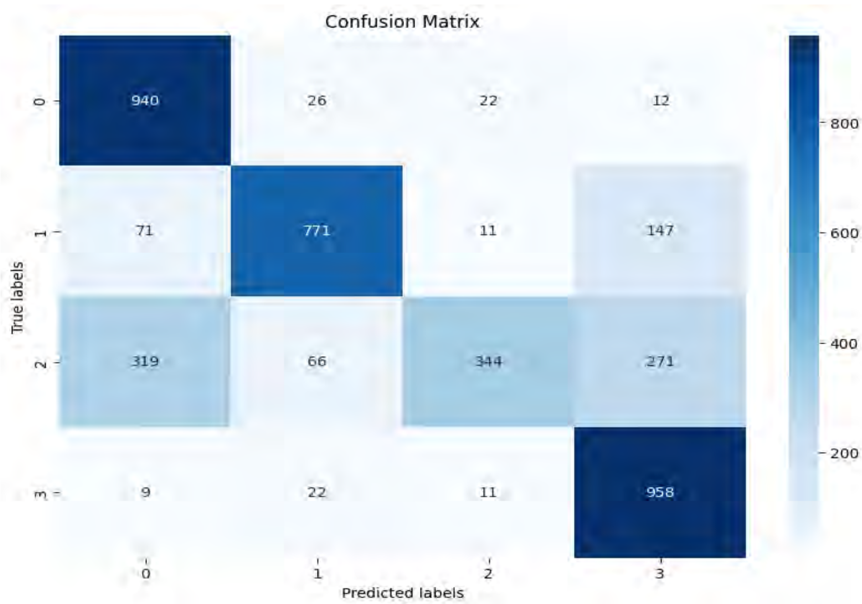

Figure 5.18: Confusion Matrix

Training accuracy starts with just above 0.74 and it gradually increases in later epochs. While training, reducing learning rate reduces learning rate as a result model gets rid of overfitting. Validation accuracy continues to rise up to 3 epochs and within 22 epochs it shows inconsistency. From epochs 30 to 50 both training and validation accuracy shows similar kinds of trends and therefore the model is well trained and generalized. The loss of training shows a sharp decrease at early epochs, indicating that the model learns rapidly. As the epochs progress, the training loss curve still exhibits a gradual decrease, and ends with a downward trend and ends at a training loss of 0.3471. Validation loss curve shows a sharp decrease, characterizing basic learning of the model generalizes well over the unseen validation data set. The use of the ReduceLROnPlateau callback, which reduces the number of learnings as an optimization when verification loss improves, seems to help stabilize the loss curves at later epochs. CNV and Normal has high recall value and struggles with Drusen that has a recall of 0.34. The F1-Score for Drusen is consequently lower at 0.50 due to the poor recall. CNVs and normal classes with high true positive rates of 940 and 958, respectively, indicating robust detection. However, the Drusen class faces challenges as proved by a significant number of misclassifications, notably not identified as CNV.

5.9 DenseNet121

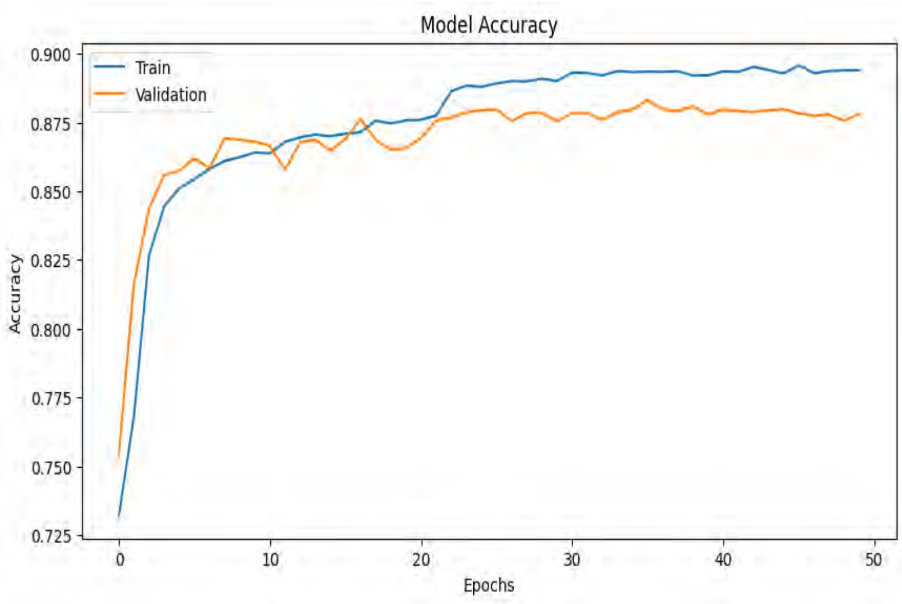

Figure 5.19: Training and validation accuracy

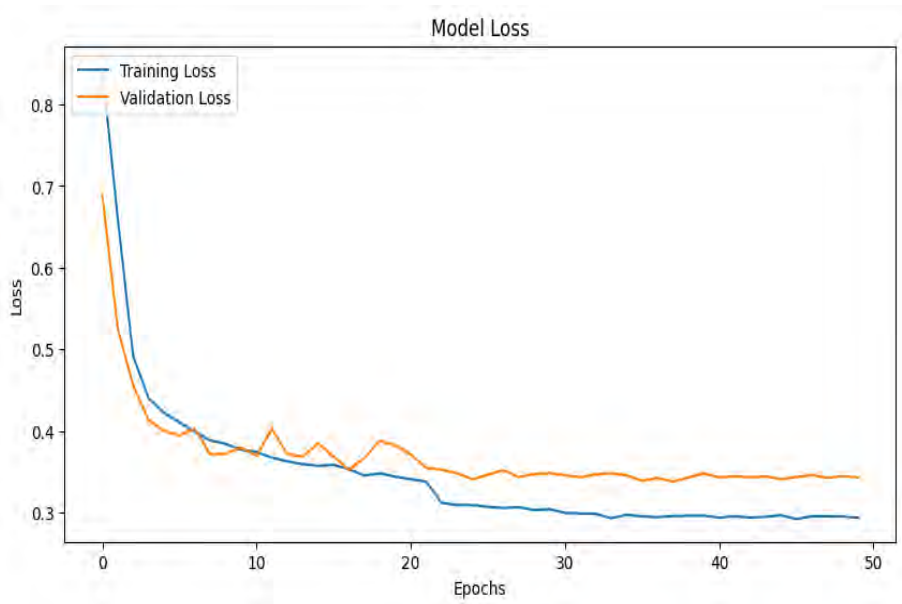

Figure 5.20: Training and validation loss

	precision	recall	f1-score	support
CNV	0.85	0.69	0.76	1000
DME	0.57	0.94	0.71	1000
Drusen	0.93	0.14	0.24	1000
Normal	0.64	0.90	0.75	1000
accuracy			0.67	4000
macro avg	0.75	0.67	0.61	4000
weighted avg	0.75	0.67	0.61	4000

Table 5.7: Classification Report

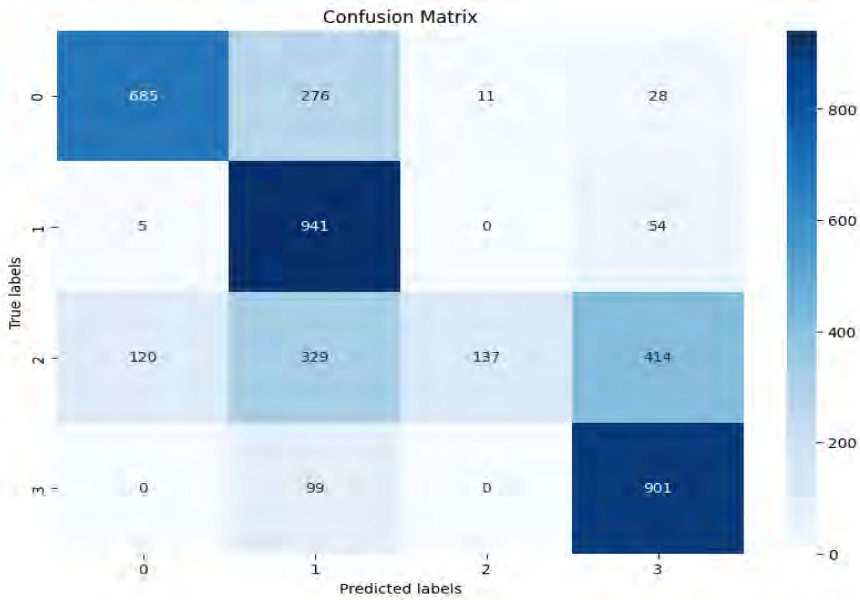

Figure 5.21: Confusion Matrix

Training accuracy begins with 0.7313 and it is an upward trend. There is steady rise in accuracy till 20 epochs and then a significant amount of rise is noticed in epochs 21 to 22. Rest of the epochs training accuracy increases gradually. Validation accuracy flows the same as train accuracy, it shows a steady approach from epochs 25 to 50 epochs though there are too many fluctuations in the beginning till 20 epochs. Both training and validation loss decreases consistently, though there is noticeable fluctuation in validation loss. Normal has a higher recall of 0.90 but a lesser accuracy of 0.64, whereas CNV has a comparatively high recall of 0.69 and a precision of 0.85. However, the recall is quite low at 0.14, showing that the model misses a large number of true Drusen instances. In contrast, Drusen has a very high accuracy of 0.93, demonstrating that when the model predicts Drusen, it is highly likely to be true. Drusen is further demonstrated by the confusion matrix, which shows that only 329 instances are accurately detected, while a significant number of cases are misclassified as CNV(276 misclassifications) and Normal(28 misclassifications).

5.10 ResNet50

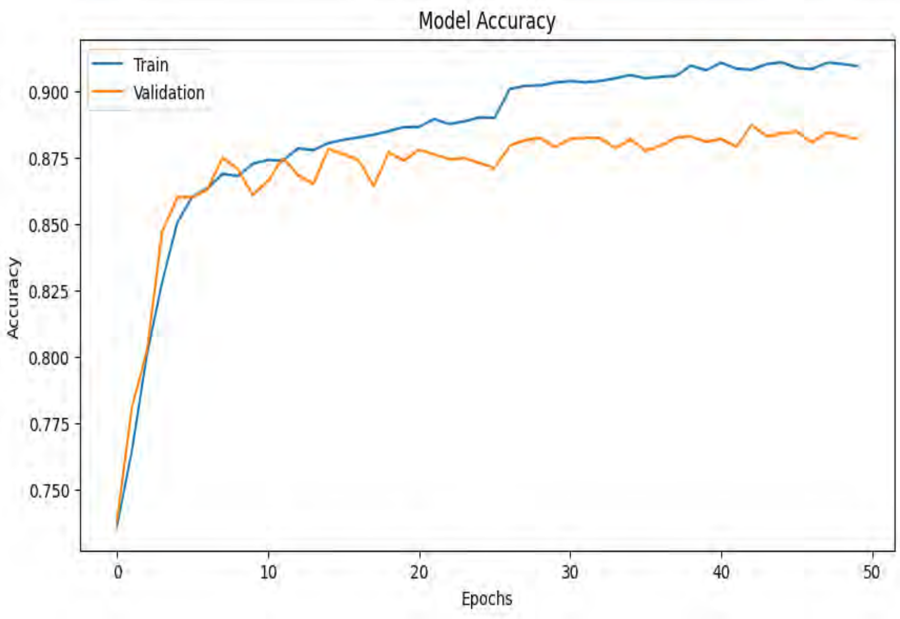

Figure 5.22: Training and validation accuracy

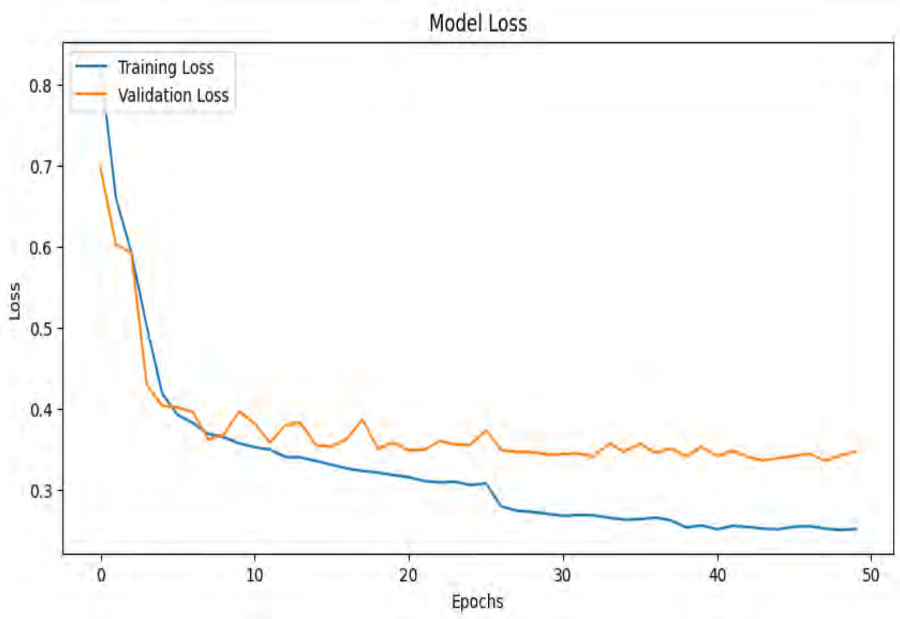

Figure 5.23: Training and validation loss

	precision	recall	f1-score	support
CNV	0.80	0.76	0.78	1000
DME	0.57	0.95	0.71	1000
Drusen	0.94	0.13	0.23	1000
Normal	0.70	0.85	0.77	1000
accuracy			0.67	4000
macro avg	0.75	0.67	0.62	4000
weighted avg	0.75	0.67	0.62	4000

Table 5.8: Classification Report

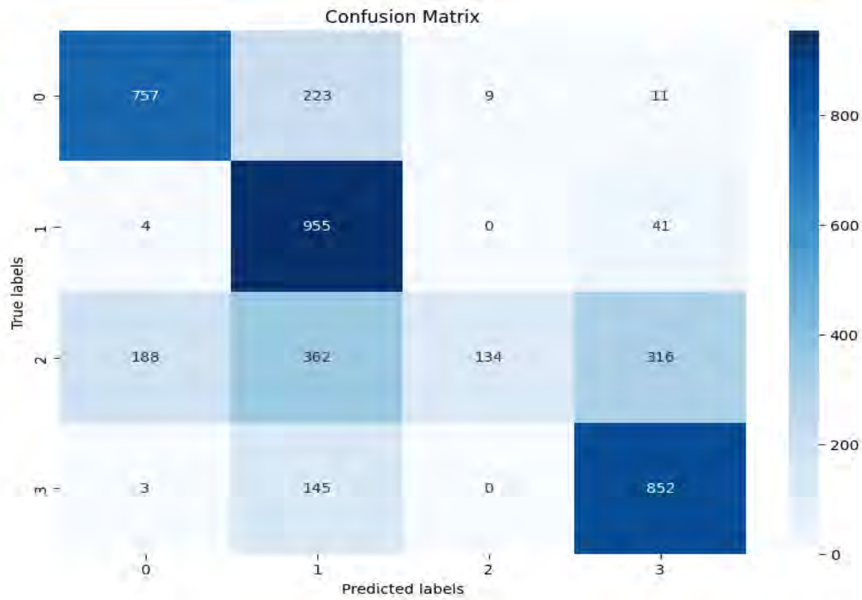

Figure 5.24: Confusion Matrix

Training accuracy and validation accuracy begins with almost similar accuracy values which are 0.7358 and 0.7375. Both accuracy increased rapidly till 8 epochs. After 8 epochs, validation accuracy fluctuates throughout the last epochs whereas training accuracy ensures a consistent rise. Since, the model is well trained, as a result training loss starts with 0.82 and ends up to 0.2508. Validation loss decreases till 10 epochs and it remains constant. Drusen has the highest precision value and on the other hand it struggles in recall. In the confusion matrix, DME makes the accurate prediction compared to CNV, Drusen and Normal.

5.11 Xception

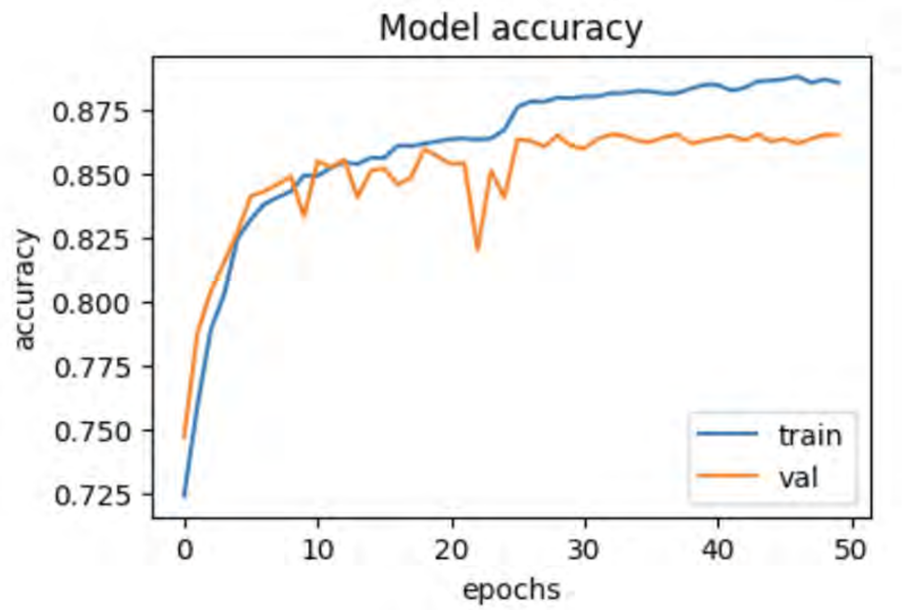

Figure 5.25: Training and validation accuracy

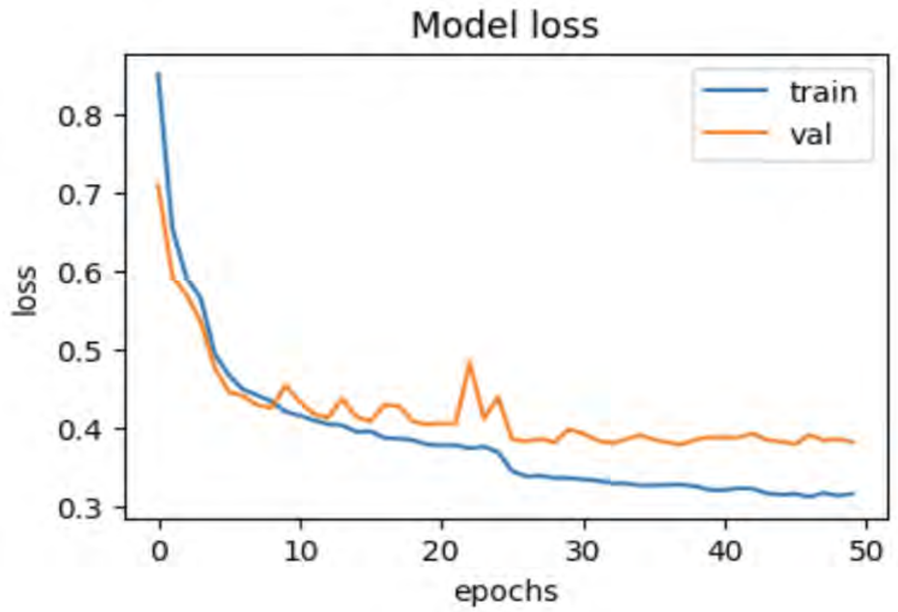

Figure 5.26: Training and validation loss

	precision	recall	f1-score	support
CNV	0.72	0.95	0.82	1000
DME	0.91	0.78	0.84	1000
Drusen	0.91	0.47	0.62	1000
Normal	0.74	0.96	0.83	1000
accuracy			0.79	4000
macro avg	0.82	0.79	0.78	4000
weighted avg	0.82	0.79	0.78	4000

Table 5.9: Classification Report

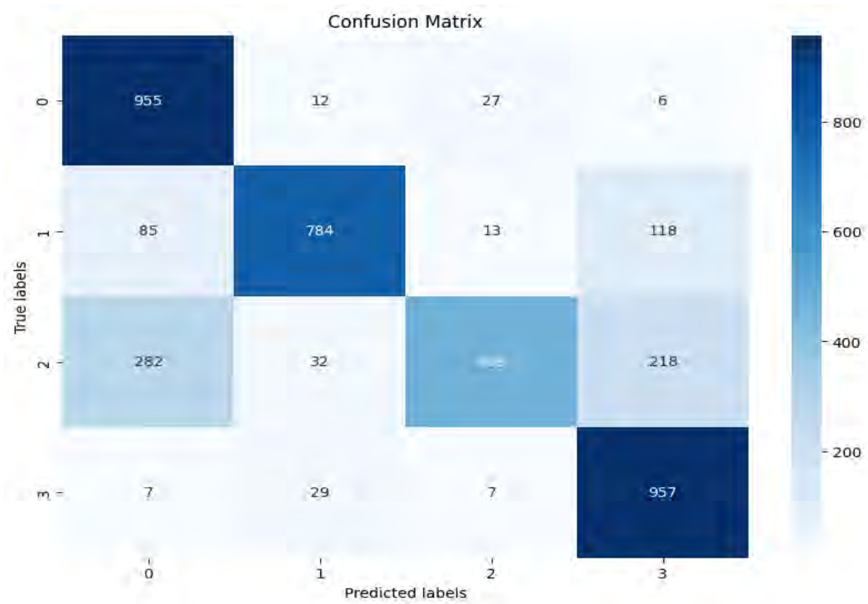

Figure 5.27: Confusion Matrix

5.12 Comparison & Verdict

Our proposed model has a significantly high score compared to other models in accuracy, precision, recall, f1 score. The 10 layer model can correctly predict 95% of the time on the chosen parameters whereas other pretrained models failed to achieve this. On the other hand, pre-trained models showed poor results identifying the minority class. The recall value of the DRUSEN class has 0.34, 0.13, 0.24, 0.53 in InceptionV3, ResNet50, DenseNet121, Xception respectively. It means there is a high number of false negatives. On the other hand, our proposed model has 0.85 recall value in the minority class (DRUSEN). Among all the models, our proposed model has 99% precision value in DME and DRUSEN class and 99% recall value in CNV and NORMAL in class. To add one, our custom model needs much lower parameters to train as well compared to the pretrained models.

Alqudah (2019)[11] used a different dataset combining SD-OCT images to predict five class diseases and got 95.3% accuracy like our proposed model. Satkeyara et al.[2] experimented on the same dataset and found out that VGG16 performed well in comparison with ALexNet and LeNet. VGG16 was able to get an accuracy of

93.01%. Our proposed model beat their result in terms of both accuracy and f1 score per class. Besides, our proposed model did not show any significant spike on the training process. Ma et al.[7] proposed HCTNet got 91.56%, 88.11%, 88.56% accuracy, precision and recall value respectively. Compared to that, our model was able to outperform the HCTNet model with accuracy, precision and recall value of 95.05%, 95.18% and 94.90%, respectively.

Comparison parameters	**10 layer**	**Inceptionv3**	**ResNet50**	**DenseNet121**	**Xception**	**VGG16**
Epochs	50	50	50	50	50	50
Batch Size	64	64	64	64	64	64
Learning Rate	0.001	0.001	0.001	0.001	0.001	0.001
Test Accuracy	0.95	0.75	0.68	0.66	0.79	0.81
Max validation accuracy	0.9528	0.8614	0.87	0.88	0.8621	0.878
CNV Precision	0.91	0.70	0.80	0.85	0.72	0.81
DME Precision	0.99	0.87	0.57	0.57	0.91	0.87
Drusen Precision	0.99	0.89	0.94	0.93	0.91	0.90
Normal Precision	0.92	0.69	0.70	0.64	0.73	0.73
CNV Recall	0.99	0.95	0.76	0.69	0.95	0.91
DME Recall	0.96	0.78	0.95	0.71	0.75	0.87
Drusen Recall	0.85	0.47	0.13	0.24	0.53	0.65
Normal Recall	0.99	0.96	0.85	0.75	0.96	0.95
Total trainable parameter	6,833,752	264,796,40	514,469,36	25,955,352	51,645,464	13110296

Table 5.10: Comparison among models

5.13 Grad-Cam

Grad-CAM makes CNN based models more clear. This approach helps better to understand CNN-based models[41]. Nowadays, many medical technologies help to diagnose many diseases which are not visible with human eyes directly. AI based CNN models have shown tremendous results in terms of classifying or recognising any medical images. In recent years, connection-based deep learning models have shown better performance than algorithm-based models. Grad-Cam is an AI based system which produces heat maps and it helps to explain the classification results. Day by day, this AI based technique is getting more popular than objective metrics. It helps physicians and radiologists by AI generated heat-maps and classification techniques[42]. Grad-CAM can not only classify diabetic retinopathy (DR) fundus images but also indicate the regions of different lesions[43].

By detecting diabetic retinopathy (DR) early with the help of Grad-CAM, blindness can be prevented easily. Grad-CAM can be used for highlighting important regions of images which are basically used for prediction[44]. AI has shown great impact in ophthalmology by detecting many diseases early and classifying them perfectly. Applying explainable AI like SCIM (SHAP-CAM Interpretable Mapping) to different CNN architectures helps to classify many retinal diseases like glaucoma[45]. As diabetic retinopathy (DR) causes early blindness, detecting this disease early with the help of AI can be beneficial for the doctor. Explainable AI also provides various explanations to justify their result[46].

We have applied Grad-CAM on our regular dataset for better analysis and important features and regions.

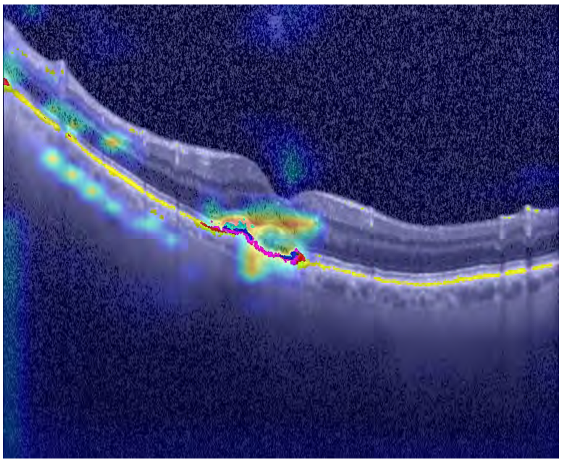

Figure 5.28: Grad-Cam on CNV Image

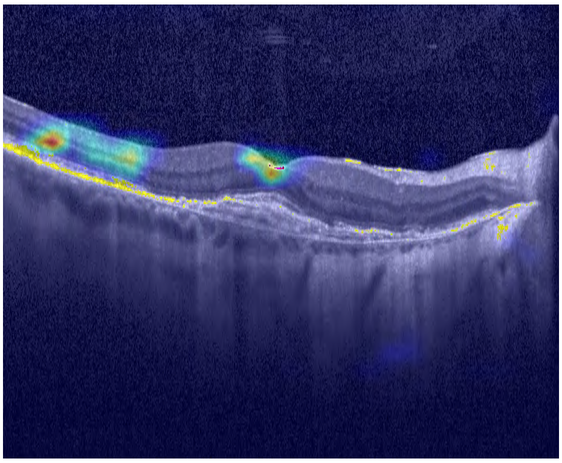

Figure 5.29: Grad-Cam on CNV Image

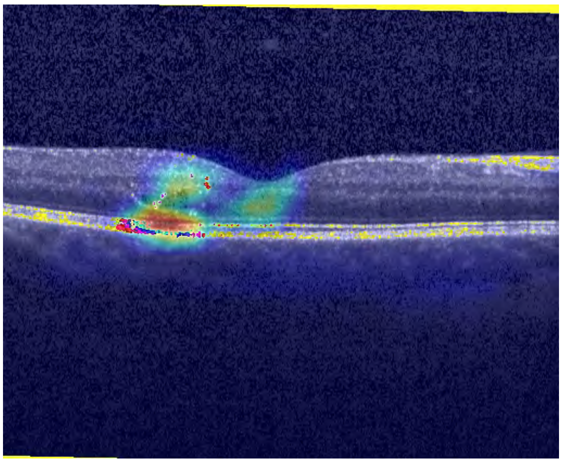

Figure 5.30: Grad-Cam on DME Image

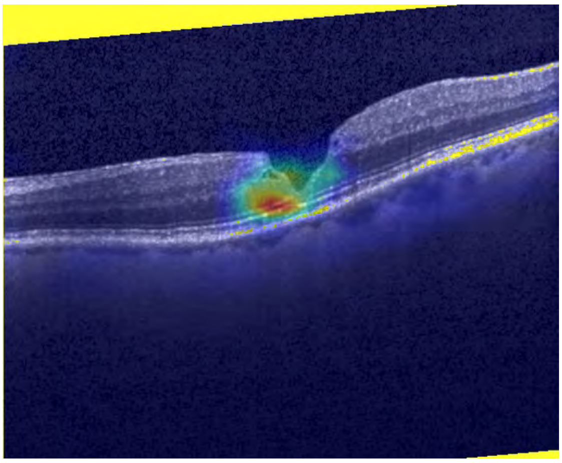

Figure 5.31: Grad-Cam on DME Image

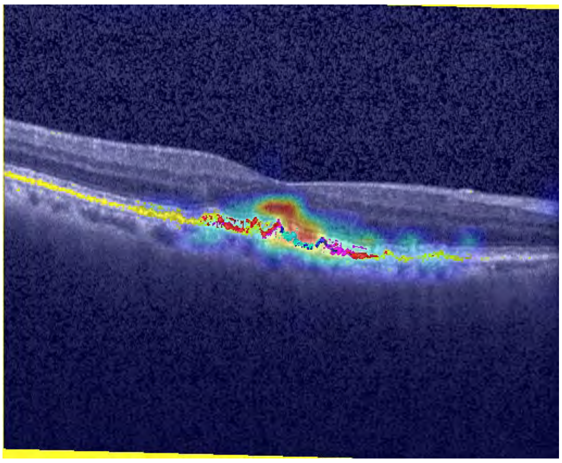

Figure 5.32: Grad-Cam on DRUSEN Image

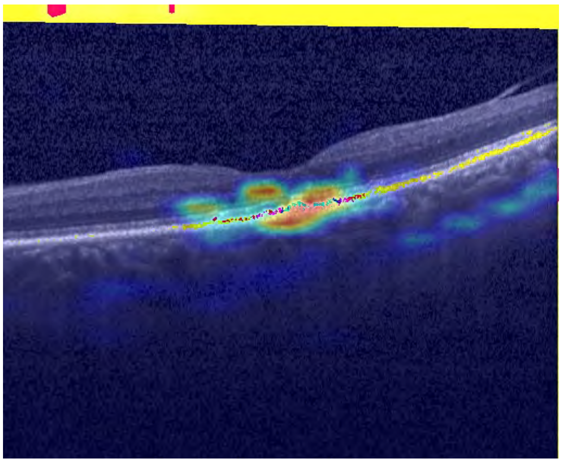

Figure 5.33: Grad-Cam on DRUSEN Image

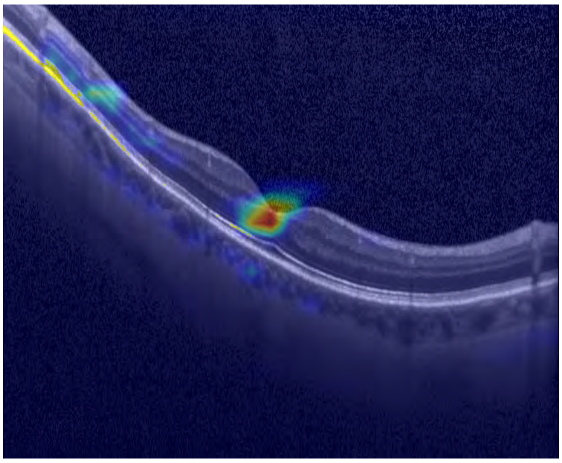

Figure 5.34: Grad-Cam on Normal Image

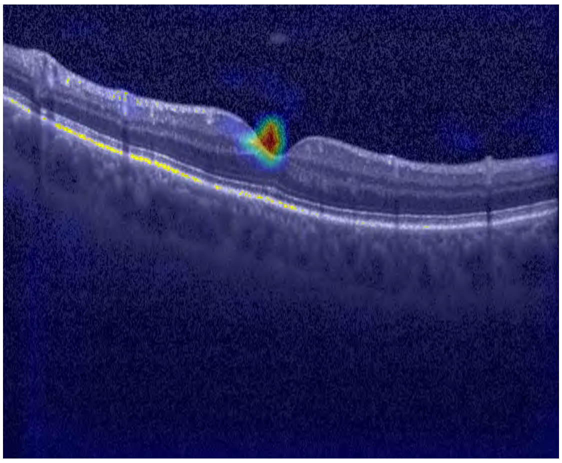

Figure 5.35: Grad-Cam on Normal Image

Chapter 6

Web Application

6.1 Description

We have used 'Flask' which is a python framework for setting up a web application. For the interface part, we have used html, css and javascript. The purpose of this application is to classify retinal diseases images using our model. Firstly, we install flask in the app.py file. It worked as the backend python file where we load our proposed model. index.html file has the frontend part of this website. When the predict button is clicked, the selected image is stored in a desired folder which is then used to predict in the backend. After prediction of the selected image, we return the result. The Frontend part fetches the result and shows it to the user. Finally, user can view the actual selected image with the predicted result as well as the contributors the this project in the screen.

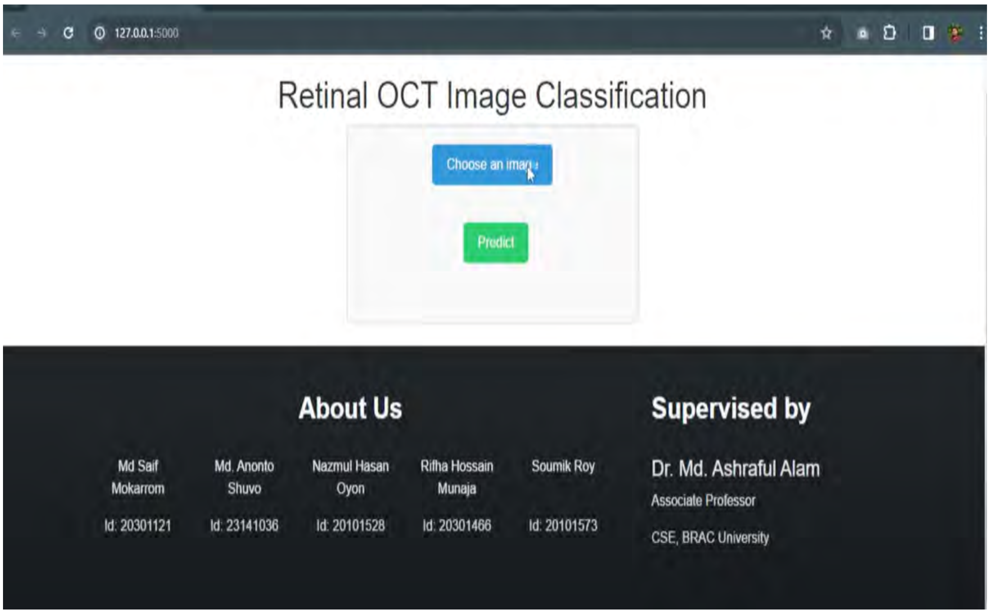

Figure 6.1: Interface of Web Application

6.1.1 CNV

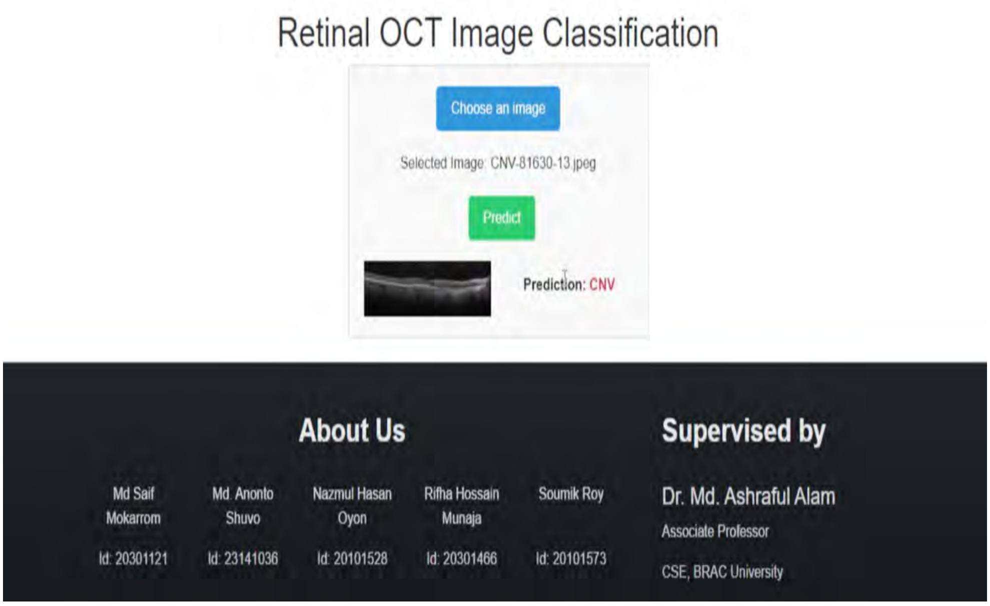

Figure 6.2: CNV Correct Prediction

6.1.2 DME

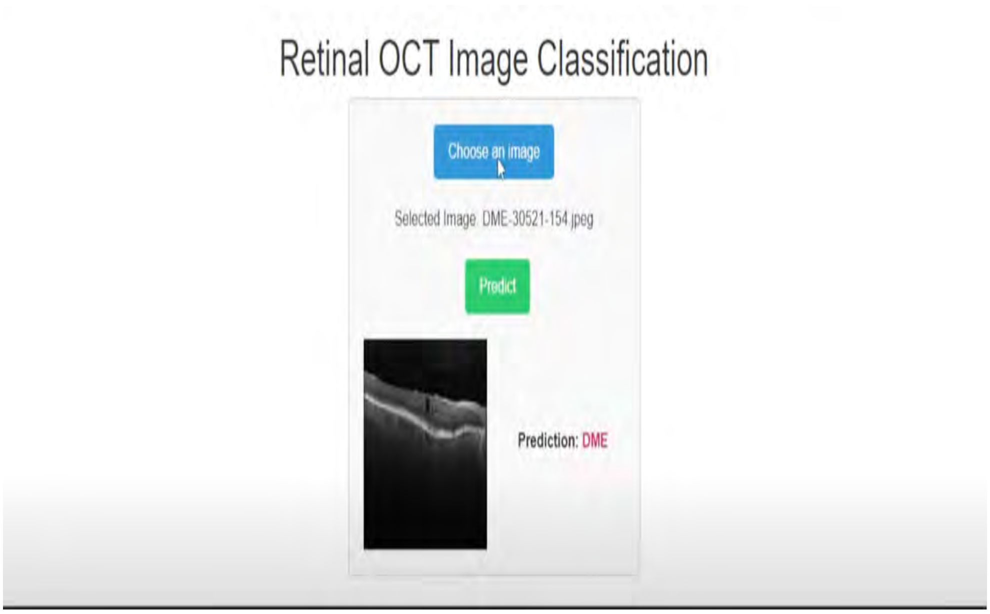

Figure 6.3: DME Correct Prediction

6.1.3 DRUSEN

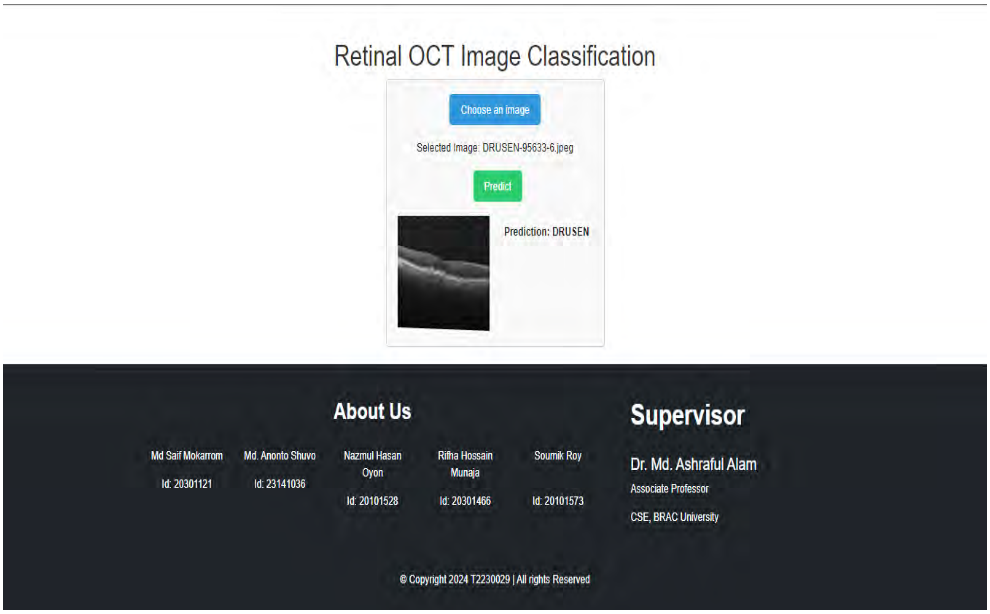

Figure 6.4: DRUSEN Correct Prediction

6.1.4 Normal

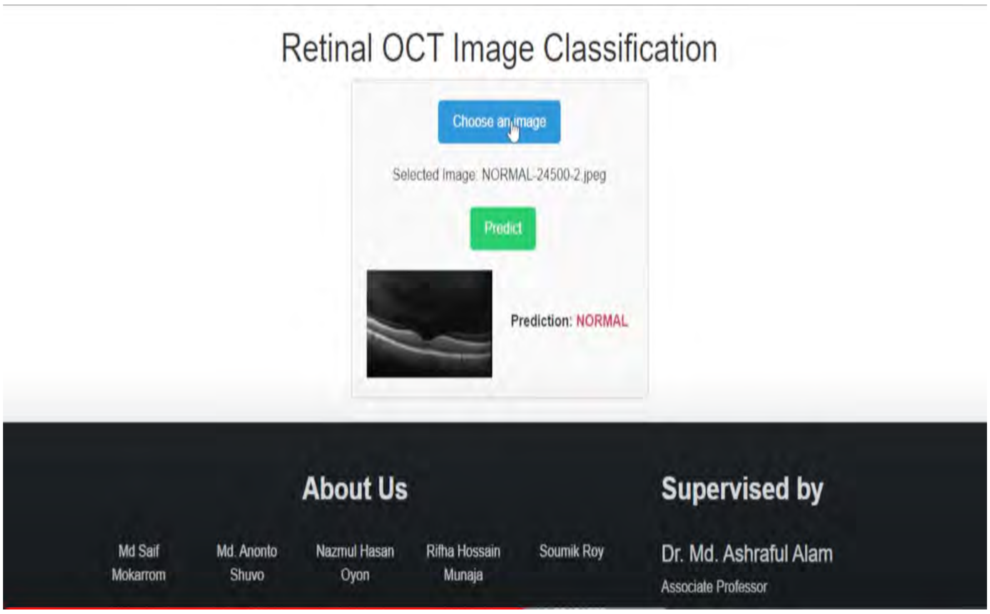

Figure 6.5: Normal Correct Prediction

Chapter 7

Limitation and Future Work

In our experiment, we had to face some limitations due to imbalanced data, lack of retina expert opinion. Firstly, the dataset we used has a different number of images across the class distribution. To compare with this dataset we had created the undersampled balanced dataset but we were not able to train with the oversampled dataset due to resource limitation. Also, as we lack a retinal image expert, we could not re verify the dataset. In the future, we plan to take acknowledgement from the doctor about our Image Enhanced dataset. Moreover, we plan to upgrade our dataset by adding more images to the minority classes. Besides this, we are planning to apply more efficient methods for data balancing such as oversample to the majority class. We also plan to train the dataset on vision transformer architecture as big datasets often work well with the vision transformer architecture and compare its results with our custom model.

Chapter 8

Conclusion

In conclusion, this paper introduces a deep learning approach, using CNNs as key classifiers, classifying OCT images into specific classes (NORMAL, CNV, DME, DRUSEN). Methods to investigate the ability of OCT images to early detect and diagnose retinal layer disorders are also presented using a robust method using our custom CNN model and pre-trained models VGG-16, ResNet50, Inception V3, DenseNet-121 and Xception. The findings shows that our proposed CNN model outperformed other pre-trained models with an accuracy, precision and recall value of 95.05%, 95.18% and 94.90%, respectively. Our ultimate goal is to provide practical and fast diagnostic aids to patients and optometrists, delivering increased productivity, accuracy and efficiency in the examination process, and ultimately it will help patients in need.

Bibliography

[1] J. He, J. Wang, Z. Han, J. Ma, C. Wang, and M. Qi, "An interpretable transformer network for the retinal disease classification using optical coherence tomography," *Scientific Reports*, vol. 13, no. 1, p. 3637, 2023.

[2] M. E. Sertkaya, B. Ergen, and M. Togacar, "Diagnosis of eye retinal diseases based on convolutional neural networks using optical coherence images," in *2019 23rd International conference electronics*, IEEE, 2019, pp. 1–5.

[3] S. A. Kamran, K. F. Hossain, A. Tavakkoli, S. L. Zuckerbrod, and S. A. Baker, "Vtgan: Semi-supervised retinal image synthesis and disease prediction using vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 3235–3245.

[4] N. Eladawi, M. Elmogy, M. Ghazal, *et al.*, "Classifcation of retinal diseases based on oct images," 2018.

[5] M. N. Islam, M. Hasan, M. K. Hossain, M. G. R. Alam, M. Z. Uddin, and A. Soylu, "Vision transformer and explainable transfer learning models for auto detection of kidney cyst, stone and tumor from ct-radiography," *Scientific Reports*, vol. 12, no. 1, p. 11 440, 2022.

[6] A. S. Hosain, M. Islam, M. H. K. Mehedi, I. E. Kabir, and Z. T. Khan, "Gastrointestinal disorder detection with a transformer based approach," in *2022 IEEE 13th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, IEEE, 2022, pp. 0280–0285.

[7] Z. Ma, Q. Xie, P. Xie, F. Fan, X. Gao, and J. Zhu, "Hctnet: A hybrid convnet-transformer network for retinal optical coherence tomography image classification," *Biosensors*, vol. 12, no. 7, p. 542, 2022.

[8] R. Rasti, H. Rabbani, A. Mehridehnavi, and F. Hajizadeh, "Macular oct classification using a multi-scale convolutional neural network ensemble," *IEEE transactions on medical imaging*, vol. 37, no. 4, pp. 1024–1034, 2017.

[9] A. Tayal, J. Gupta, A. Solanki, K. Bisht, A. Nayyar, and M. Masud, "Dl-cnn-based approach with image processing techniques for diagnosis of retinal diseases," *Multimedia Systems*, pp. 1–22, 2021.

[10] M. Subramanian, M. S. Kumar, V. Sathishkumar, *et al.*, "Diagnosis of retinal diseases based on bayesian optimization deep learning network using optical coherence tomography images," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.

[11] A. M. Alqudah, "Aoct-net: A convolutional network automated classification of multiclass retinal diseases using spectral-domain optical coherence tomography images," *Medical & biological engineering & computing*, vol. 58, pp. 41–53, 2020.

[12] L. Huang, X. He, L. Fang, H. Rabbani, and X. Chen, "Automatic classification of retinal optical coherence tomography images with layer guided convolutional neural network," *IEEE Signal Processing Letters*, vol. 26, no. 7, pp. 1026–1030, 2019.

[13] D. U. N. Qomariah, H. Tjandrasa, and C. Fatichah, "Classification of diabetic retinopathy and normal retinal images using cnn and svm," in *2019 12th International Conference on Information & Communication Technology and System (ICTS)*, IEEE, 2019, pp. 152–157.

[14] J. Kim and L. Tran, "Ensemble learning based on convolutional neural networks for the classification of retinal diseases from optical coherence tomography images," in *2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS)*, IEEE, 2020, pp. 532–537.

[15] M. Hajabdollahi, R. Esfandiarpoor, K. Najarian, N. Karimi, S. Samavi, and S. Reza-Soroushmeh, "Low complexity convolutional neural network for vessel segmentation in portable retinal diagnostic devices," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, IEEE, 2018, pp. 2785–2789.

[16] S. Asif and K. Amjad, "Deep residual network for diagnosis of retinal diseases using optical coherence tomography images," *Interdisciplinary Sciences: Computational Life Sciences*, vol. 14, no. 4, pp. 906–916, 2022.

[17] F. Li, H. Chen, Z. Liu, *et al.*, "Deep learning-based automated detection of retinal diseases using optical coherence tomography images," *Biomedical optics express*, vol. 10, no. 12, pp. 6204–6226, 2019.

[18] S. Najeeb, N. Sharmile, M. S. Khan, I. Sahin, M. T. Islam, and M. I. H. Bhuiyan, "Classification of retinal diseases from oct scans using convolutional neural networks," in *2018 10th International conference on electrical and computer engineering (ICECE)*, IEEE, 2018, pp. 465–468.

[19] S. Long, X. Huang, Z. Chen, S. Pardhan, D. Zheng, *et al.*, "Automatic detection of hard exudates in color retinal images using dynamic threshold and svm classification: Algorithm development and evaluation," *BioMed research international*, vol. 2019, 2019.

[20] D. Kermany, K. Zhang, and M. Goldbaum, "Labeled optical coherence tomography (oct) and chest x-ray images for classification (2018)," *Mendeley Data, v2 https://doi. org/10.17632/rscbjbr9sj https://nihcc. app. box. com/v/ChestXray-NIHCC*, 2018.

[21] H. E. Grossniklaus and W. R. Green, "Choroidal neovascularization," *American journal of ophthalmology*, vol. 137, no. 3, pp. 496–503, 2004.

[22] G. E. Lang, "Diabetic macular edema," *Ophthalmologica*, vol. 227, no. Suppl. 1, pp. 21–29, 2012.

[23] J. Fasola and M. Veloso, "Real-time object detection using segmented and grayscale images," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, IEEE, 2006, pp. 4088–4093.

[24] A. W. Setiawan, T. R. Mengko, O. S. Santoso, and A. B. Suksmono, "Color retinal image enhancement using clahe," in *International conference on ICT for smart society*, IEEE, 2013, pp. 1–3.

[25] A. Mishra, A. Wong, K. Bizheva, and D. A. Clausi, "Intra-retinal layer segmentation in optical coherence tomography images," *Optics express*, vol. 17, no. 26, pp. 23 719–23 728, 2009.

[26] M. Shaha and M. Pawar, "Transfer learning for image classification," in *2018 second international conference on electronics, communication and aerospace technology (ICECA)*, IEEE, 2018, pp. 656–660.

[27] S. Mujawar, D. Kiran, and H. Ramasangu, "An efficient cnn architecture for image classification on fpga accelerator," in *2018 Second International Conference on Advances in Electronics, Computers and Communications (ICAECC)*, IEEE, 2018, pp. 1–4.

[28] M. Sun, Z. Song, X. Jiang, J. Pan, and Y. Pang, "Learning pooling for convolutional neural network," *Neurocomputing*, vol. 224, pp. 96–104, 2017.

[29] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 international conference on engineering and technology (ICET)*, Ieee, 2017, pp. 1–6.

[30] S. Mascarenhas and M. Agarwal, "A comparison between vgg16, vgg19 and resnet50 architecture frameworks for image classification," in *2021 International conference on disruptive technologies for multi-disciplinary research and applications (CENTCON)*, IEEE, vol. 1, 2021, pp. 96–99.

[31] A. V. Ikechukwu, S. Murali, R. Deepu, and R. Shivamurthy, "Resnet-50 vs vgg-19 vs training from scratch: A comparative analysis of the segmentation and classification of pneumonia from chest x-ray images," *Global Transitions Proceedings*, vol. 2, no. 2, pp. 375–381, 2021.

[32] B. Kumar, A. K. Singh, and P. Banerjee, "A deep learning approach for product recommendation using resnet-50 cnn model," in *2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS)*, IEEE, 2023, pp. 604–610.

[33] N. Dong, L. Zhao, C.-H. Wu, and J.-F. Chang, "Inception v3 based cervical cell classification combined with artificially extracted features," *Applied Soft Computing*, vol. 93, p. 106 311, 2020.

[34] 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*, 2017, pp. 4700–4708.

[35] W. W. Lo, X. Yang, and Y. Wang, "An xception convolutional neural network for malware classification with transfer learning," in *2019 10th IFIP international conference on new technologies, mobility and security (NTMS)*, IEEE, 2019, pp. 1–5.

[36] 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.

[37] D. S. Kermany, M. Goldbaum, W. Cai, *et al.*, "Identifying medical diagnoses and treatable diseases by image-based deep learning," *cell*, vol. 172, no. 5, pp. 1122–1131, 2018.

[38] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[39] R. Poojary, R. Raina, and A. K. Mondal, "Effect of data-augmentation on fine-tuned cnn model performance," *IAES International Journal of Artificial Intelligence*, vol. 10, no. 1, p. 84, 2021.

[40] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," in *2018 international interdisciplinary PhD workshop (IIPhDW)*, IEEE, 2018, pp. 117–122.

[41] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra, "Grad-cam: Why did you say that?" *arXiv preprint arXiv:1611.07450*, 2016.

[42] J.-C. Chien, J.-D. Lee, C.-S. Hu, and C.-T. Wu, "The usefulness of gradient-weighted cam in assisting medical diagnoses," *Applied Sciences*, vol. 12, no. 15, p. 7748, 2022.

[43] H. Jiang, J. Xu, R. Shi, *et al.*, "A multi-label deep learning model with interpretable grad-cam for diabetic retinopathy classification," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, IEEE, 2020, pp. 1560–1563.

[44] O. Daanouni, B. Cherradi, and A. Tmiri, "Automatic detection of diabetic retinopathy using custom cnn and grad-cam," in *Advances on Smart and Soft Computing: Proceedings of ICACIn 2020*, Springer, 2021, pp. 15–26.

[45] C. M. Vieira, M. V. D. C. Oliveira, M. D. P. Guimarães, L. Rocha, and D. R. C. Dias, "Applied explainable artificial intelligence (xai) in the classification of retinal images for support in the diagnosis of glaucoma," in *Proceedings of the 29th Brazilian Symposium on Multimedia and the Web*, 2023, pp. 82–90.

[46] K. Duvvuri, S. Chethana, S. S. Charan, V. Srihitha, T. Ramesh, and K. Srikanth, "Grad-cam for visualizing diabetic retinopathy," in *2022 3rd International Conference for Emerging Technology (INCET)*, IEEE, 2022, pp. 1–4.